User's Manual

32

RH850/C1x

User's Manual: Hardware

Renesas microcontroller
RH850 Family

Renesas Electronics
www.renesas.com

Rev.1.60 May, 2018

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user's manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.4.0-1 November 2017)

Notes for CMOS devices

(1) Voltage application waveform at input pin:

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between VIL (MAX) and VIH (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between VIL (MAX) and VIH (MIN).

(2) Handling of unused input pins:

Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to power supply or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

(3) Precaution against ESD:

A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

(4) Status before initialization:

Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

(5) Power ON/OFF sequence:

In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current. The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

(6) Input of signal during power off state:

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

How to Use This Manual

Readers This manual is intended for users who wish to understand the functions of the RH850/C1x and design application systems using the following RH850/C1x microcontrollers:

Purpose This manual is intended to give users an understanding of the hardware functions of the RH850/C1x shown in the *Organization* below.

Organization This manual is divided into two parts: Hardware (this manual) and Architecture (RH850G3M User's Manual: Software (R01US0123E)).

Hardware	Software
Pin functions	Overview
CPU function	Processor Model
On-chip peripheral functions	Register Reference
Flash memory programming	Exceptions and Interrupts
	Memory Management
	Instruction Reference
	Reset
	Appendix

How to read this manual It is assumed that the readers of this manual have general knowledge in the fields of electrical engineering, logic circuits, and microcontrollers.

To understand the overall functions of the RH850/C1x.

→ Read this manual according to the Contents.

To understand the details of an instruction function

→ See RH850G3M Family User's Manual: Software (R01US0123E) available separately.

Conventions

Data significance: Higher digits on the left and lower digits on the right

Active low representation: xxx (overscore over pin or signal name)

Memory map address: Higher addresses on the top and lower addresses on the bottom

Note: Footnote for item marked with Note in the text

Caution: Information requiring particular attention

Remark: Supplementary information

Numeric representation: Binary ... xxxx or $xxxx_B$

Decimal ... xxxx

Hexadecimal ... $xxxx_H$

Prefix indicating power of 2 (address space, memory capacity):

K (kilo): $2^{10} = 1,024$

M (mega): $2^{20} = 1,024^2$

G (giga): $2^{30} = 1,024^3$

Description of Registers

Each register description includes register access, register address, and register value after a reset, a bit chart, illustrating the arrangement of bits, and a table of bits, describing the meaning of the bit settings.

The standard format for bit charts and tables are described below.

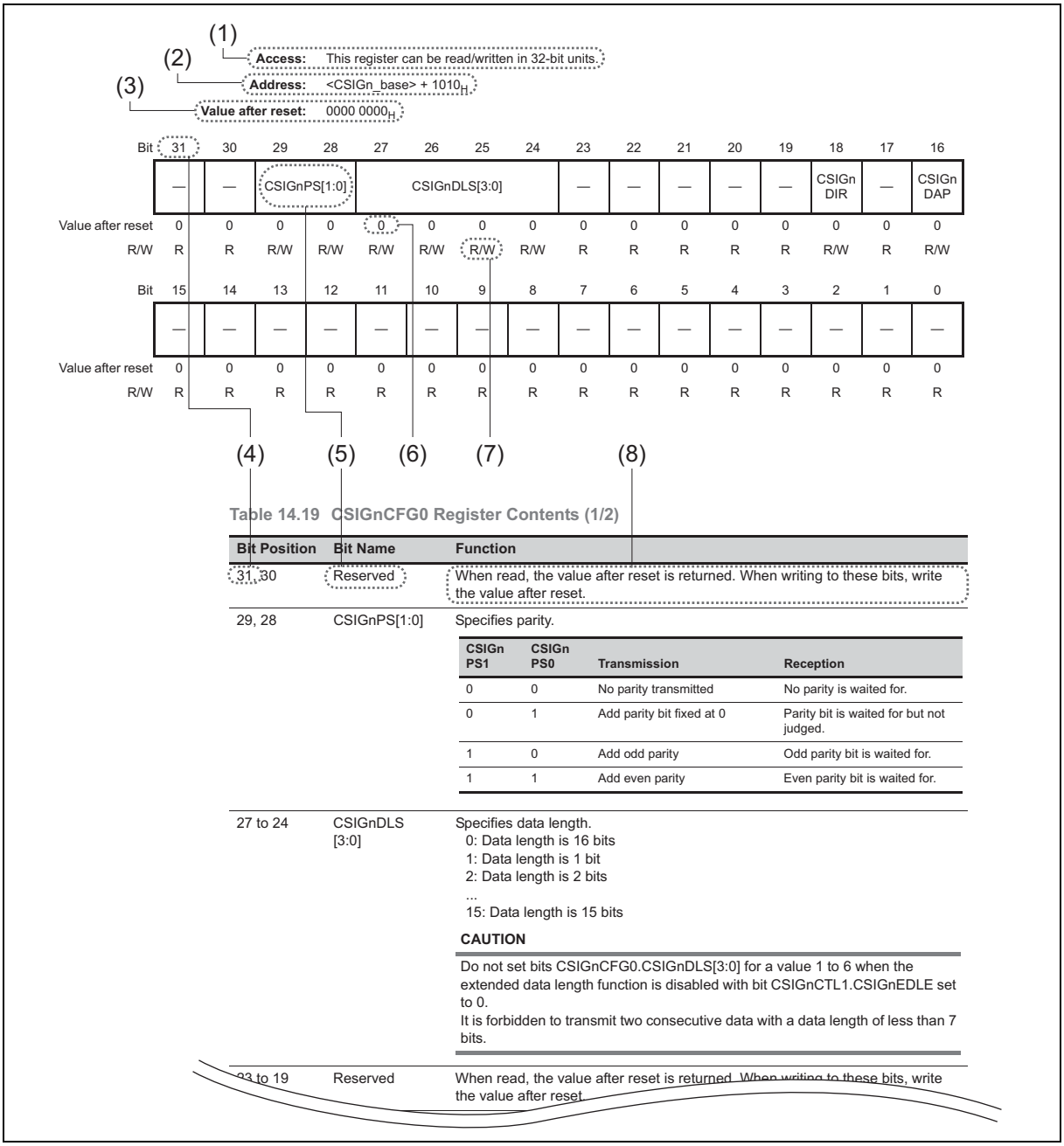


Table 14.19 CSIGnCFG0 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
29, 28	CSIGnPS[1:0]	Specifies parity.
27 to 24	CSIGnDLS [3:0]	Specifies data length. 0: Data length is 16 bits 1: Data length is 1 bit 2: Data length is 2 bits ... 15: Data length is 15 bits
23 to 19	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

CSIGn PS1	CSIGn PS0	Transmission	Reception
0	0	No parity transmitted	No parity is waited for.
0	1	Add parity bit fixed at 0	Parity bit is waited for but not judged.
1	0	Add odd parity	Odd parity bit is waited for.
1	1	Add even parity	Even parity bit is waited for.

CAUTION

Do not set bits CSIGnCFG0.CSIGnDLS[3:0] for a value 1 to 6 when the extended data length function is disabled with bit CSIGnCTL1.CSIGnEDLE set to 0.
It is forbidden to transmit two consecutive data with a data length of less than 7 bits.

(1) Access

The register can be accessed in the bit unit indicated here.

(2) Address

This is the register address.
For base address, see description of base address in each section.

(3) Value after a reset (in hexadecimal notation)

This is the value of all bits of the register after a reset. Values for bytes are given as numbers in the range from 0 to 9 and letters from A to F or as X where they are undefined.

(4) Bit position

This is the bit number.
The bits are numbered from 31 to 0 for 32-bit registers, 15 to 0 for 16-bit registers, and 7 to 0 for 8-bit registers.

(5) Bit name

Bit name or field name is indicated.
When clearly identifying the digits of a bit field is required, do so by using a form such as CSIGnDLS[3:0] above.
Indicate reserved bits by using a dash (—).

(6) Value after a reset (in binary notation)

This is the bit values after a reset.

0	:	The value after a reset is 0.
1	:	The value after a reset is 1.
—	:	The value after a reset is undefined.

(7) R/W

This is the bit attribute of all bits of the register.

R/W	:	The bit or field is readable and writable.
R	:	The bit or field is readable. Note that all reserved bits are indicated as R. When written, the value specified in the bit chart or the value after a reset should be written.
W	:	This bit or field is writable. When read, the value is undefined. If a value is indicated in the bit chart, the value is returned.

(8) Function

This is function of the bit.

Table of Contents

Section 1 Overview

The RH850/C1x is a series of single-chip microcomputers in the RH850 Family from Renesas Electronics. This section covers the features of the RH850/C1x.

1.1 Features of RH850/C1x Products

The CPU of these products is the G3M of the RH850 Family running at 240 MHz for high-speed processing.

The set of peripheral functions in these products is optimized for HEV and EV motor control and includes ROM, RAM, a DMAC, various timers including timers for motor control, various serial interfaces including a CAN interface, a 12-bit A/D converter (ADCC), an R/D converter (RDC2) that converts the signal output by a resolver into digital values representing angles, and a motor control unit (EMU2) that is capable of operating in parallel with the CPU.

Two variants, the C1H for controlling two motors and the C1M for controlling one motor, are available. The products are provided in 252-pin BGA (C1H) and 144-pin QFP (C1M) packages.

Applications

Automotive (motor control for HEVs and EVs)

1.1.1 Functions of the RH850/C1x

Table 1.1 Overview of Products (1/2)

Item		RH850/C1H	RH850/C1M
CPU	CPU system	G3M (LSDC) + G3M	G3M (LSDC)
	CPU frequency	240 MHz	
	PE internal peripheral device protection function (IPG)	Provided	
	System error notification control function (SEG)	Provided	
	Memory protection unit (MPU)	Provided	
	Floating-point unit (FPU)	Provided	
	Mutual Exclusion Control Registers (MEV)	Provided	Not provided
On-chip Memory	Code flash	2 MB × 2	2 MB
	Instruction cache (Icache)	8 KB × 2	8 KB
	Local RAM	64 KB × 2	64 KB
	Data flash	32 KB	
	Global RAM	112 KB	64 KB
External interrupt	Maskable interrupt (IRQ)	8	
	Non-Maskable Interrupt (NMI)	1	
DMA, DTS		16 channels, 128 channels	
Clock	Main oscillator (main OSC)	20 MHz	
	PLL	Provided	
I/O ports		87	66

Table 1.1 Overview of Products (2/2)

Item			RH850/C1H	RH850/C1M
Timers	Timer array unit D (TAUD)		2 units (16 timers/unit)	
	Timer array unit J (TAUJ)		1 unit (4 timers/unit)	
	Motor control timer (TSG3)		2 units	
	Timer option (TAPA)		4 units	
	Timer pattern buffer (TPBA)		2 units	1 unit
	OS timer (OSTM)		3 units	2 units
	Encoder timer (ENCA)		2 units	
	Watchdog timer (WDTA)		2 units	1 unit
Serial interface	Clocked Serial Interface H (CSIH)		2 channels	
	CAN interface (RS-CAN)		4 channels (256 message buffers in total)	
	LIN master interface (RLIN2)		3 channels	Not provided
	Serial Communication Interface (SCI3)		3 channels	
A/D converter	12-bit A/D core		2 units	
	ADCC0	Number of input pins	16	10
		Number of T&H	6	6
	ADCC1	Number of input pins	21	15
		Number of T&H	6	6
Motor control	R/D converter (RDC2)		2 units	1 unit
	Enhanced motor control unit (EMU2)		1unit (2 channels)	
Other functions	Error control module (ECM)		Provided	
	Clock Monitor (CLMA)		Provided	
	Data CRC (DCRA)		2 units	
	Error correction coding (ECC)		Provided	
	On-chip debug (OCD)		Provided	
	Boundary scan		Provided	Not provided
	Peripheral interconnection 1 (PIC1A)		1 unit	
	Peripheral interconnection 2 (PIC2B)		1 unit	
Power supply voltage	Internal power supply*2		1.25V ± 0.1V, 3.3V ± 0.3V	
	I/O power supply		5.0V ± 0.5V	
	R/D converter power supply		5.0V ± 0.5V	
	A/D converter power supply		5.0V ± 0.5V	
Temperature	Junction temperature (Tj)		-40°C to 150°C	
Package			252-pin BGA	144-pin QFP

Note 1. LSDC (Lock Step Dual Core)

Note 2. The core has two internal power supply voltages. Thus, including the I/O power supply, three power supply voltages are provided to the chip.

Table 1.2 List of Products

Group Name	Part Number	Wafer Process Manufacturing Site	Package
RH850/C1H	R7F701270EABG #**0	RSMC Naka Factory	252-pin plastic BGA (0.8-mm ball pitch) (17 × 17 mm)
	R7F701270EABG-C #**4	TSMC	
RH850/C1M	R7F701271EAFP #**0	RSMC Naka Factory	144-pin plastic QFP (0.5-mm pin pitch) (20 × 20 mm)
	R7F701271EAFP #**4	TSMC	

Note: "**" are replaced by letters of the alphabet.

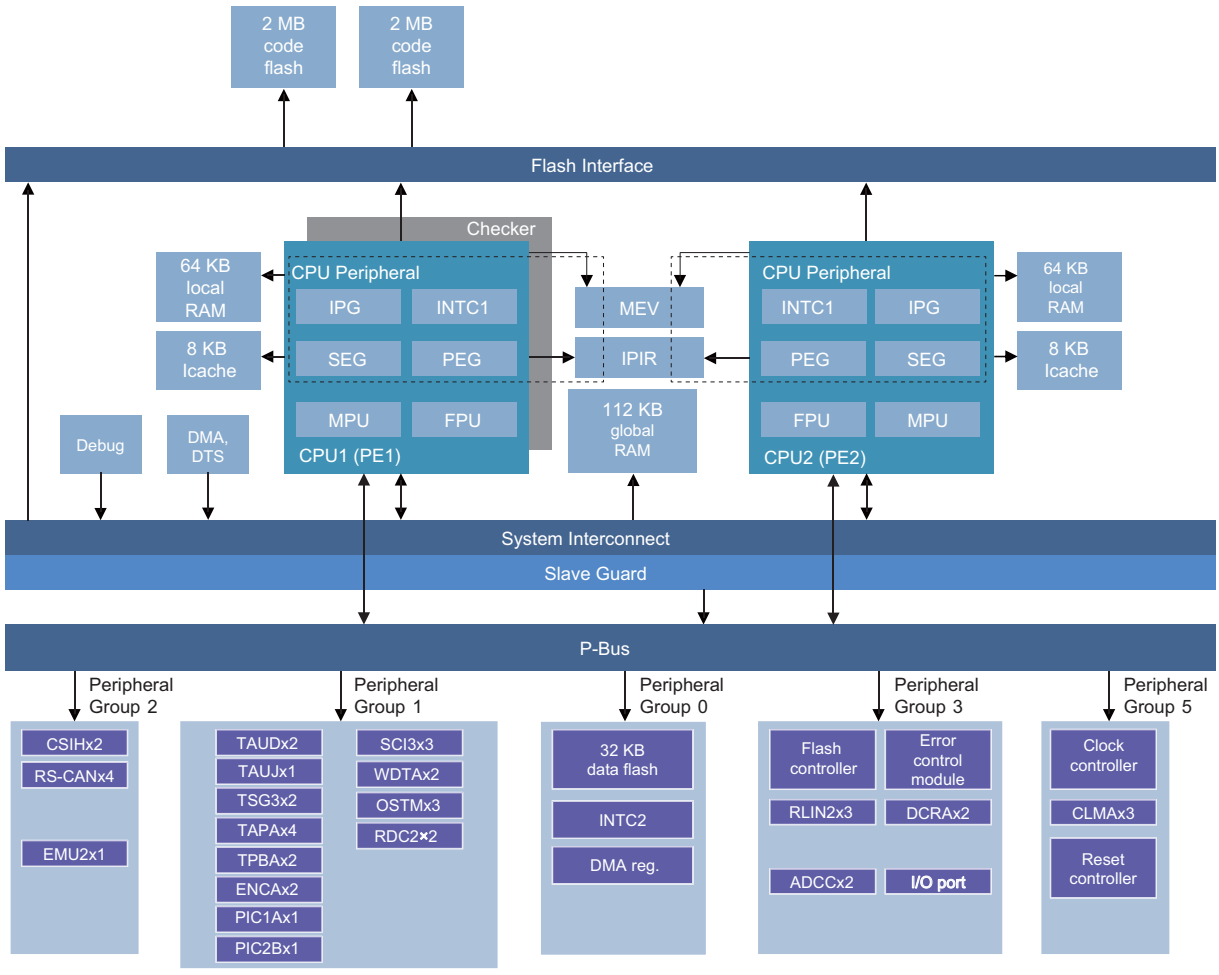
1.1.2 Development and Debugging

Table 1.3 List of Development Tools

Function	Outline
On-chip debug (OCD)	• On-chip debug interfaces – IEEE 1149.1 standard JTAG interface for NEXUS class 3 compliant debugging – Low pin debug (LPD) interface: 4 pins
On-chip debug emulator	• E1 emulator
RAM monitor	• Advanced user debugger II RAM monitor function (252-pin BGA only) • NEXUS RAM monitor function (in all products)
Compiler/Debugger	• CubeSuite+ • MULTI environment from Green Hills
Application support hardware	• RH850 evaluation platform
Flash programming	• PG-FP5 flash programmer • RFP (Renesas flash programmer) + E1 emulator • Self programming library
Software tool (optional)	• AUTOSAR MCAL

1.1.3 Internal Block Diagram

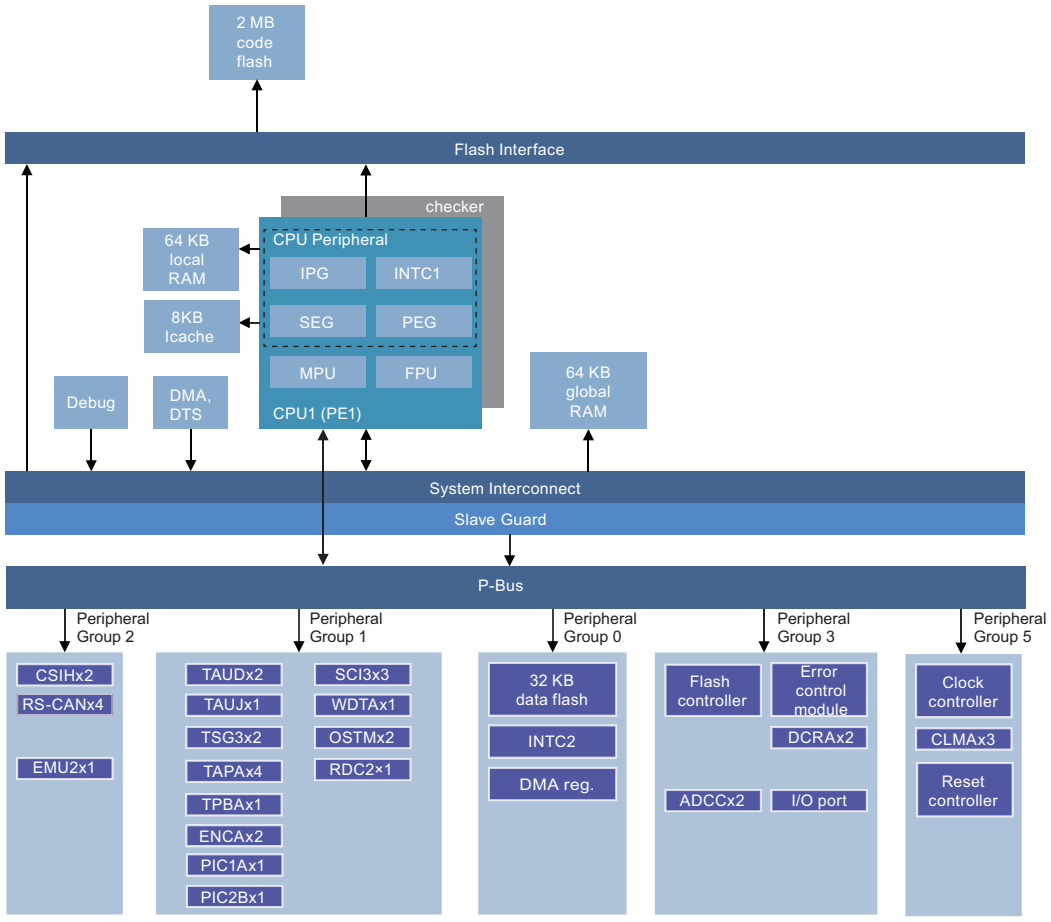
CPU1 and CPU2 include their own CPU peripherals. The CPU peripherals can be accessed only by CPU1 and CPU2. The same address is assigned to both the CPU peripherals of CPU1 and CPU2, but CPU1 always accesses the CPU peripheral of CPU1, and CPU2 always accesses the CPU peripheral of CPU2.



Note: For the details on the correspondence of peripheral modules and peripheral group, see **Section 3.1.2, Periphral Group Configuration**.

Figure 1.1 Internal Block Diagram of RH850/C1H

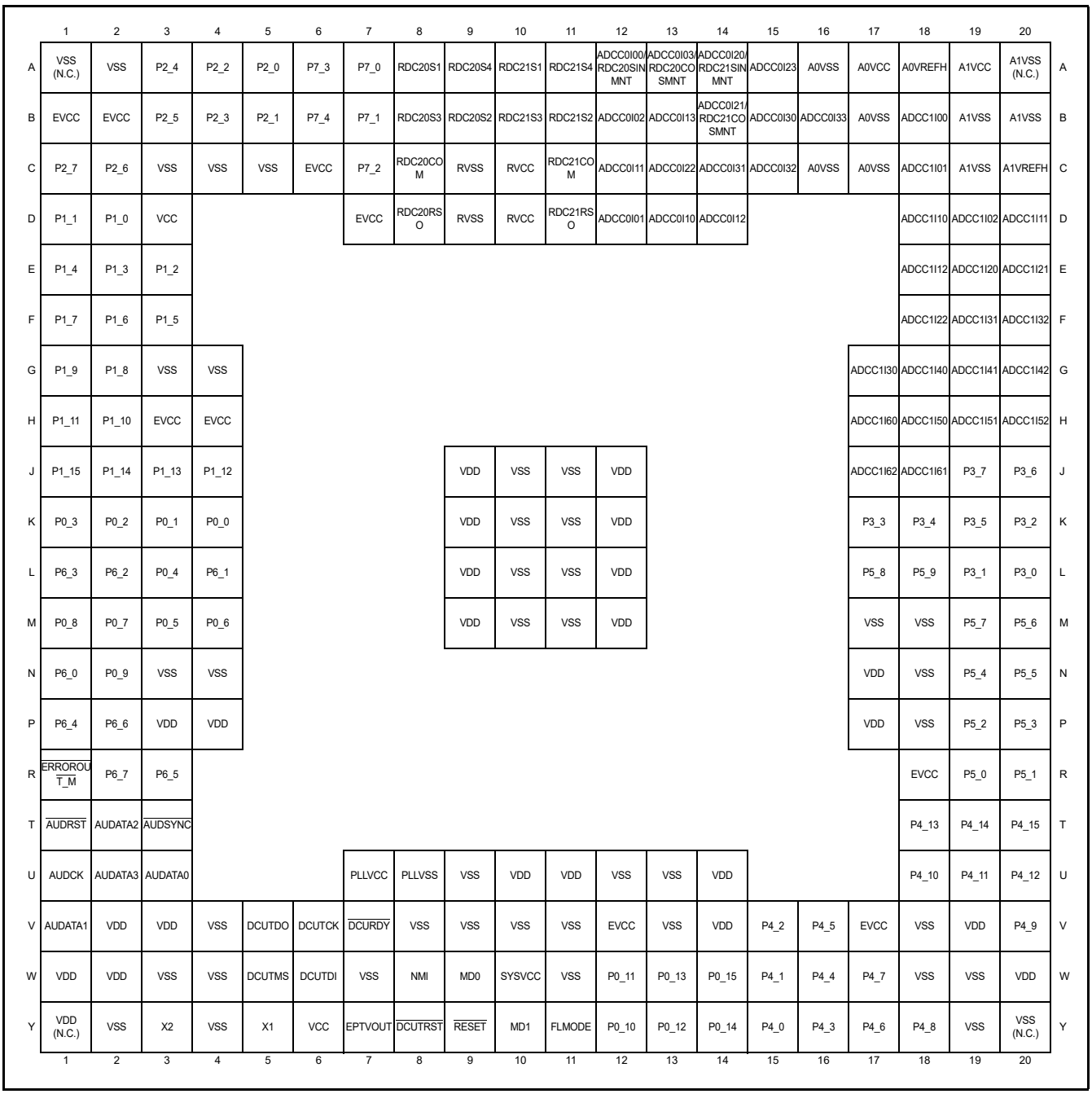
CPU1 includes the CPU peripheral. The CPU peripheral can be accessed only by CPU1.



Note: For the details on the correspondence of peripheral modules and peripheral group, see **Section 3.1.2, Periphral Group Configuration**.

Figure 1.2 Internal Block Diagram of RH850/C1M

1.2 Pin Connection Diagram (Top View)



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
A	VSS (N.C.)	VSS	P2_4	P2_2	P2_0	P7_3	P7_0	RDC20S1	RDC20S4	RDC21S1	RDC21S4	ADCC0I00/ RDC20SIN MNT	ADCC0I03/ RDC20COS MNT	ADCC0I20/ RDC21SIN MNT	ADCC0I23	A0VSS	A0VCC	A0VREFH	A1VCC	A1VSS (N.C.)	A
B	EVCC	EVCC	P2_5	P2_3	P2_1	P7_4	P7_1	RDC20S3	RDC20S2	RDC21S3	RDC21S2	ADCC0I02	ADCC0I13	ADCC0I21/ RDC21COS MNT	ADCC0I30	ADCC0I33	A0VSS	ADCC1I00	A1VSS	A1VSS	B
C	P2_7	P2_6	VSS	VSS	VSS	EVCC	P7_2	RDC20COM	RVSS	RVCC	RDC21COM	ADCC0I11	ADCC0I22	ADCC0I31	ADCC0I32	A0VSS	A0VSS	ADCC1I01	A1VSS	A1VREFH	C
D	P1_1	P1_0	VCC				EVCC	RDC20RSO	RVSS	RVCC	RDC21RSO	ADCC0I01	ADCC0I10	ADCC0I12				ADCC1I10	ADCC1I02	ADCC1I11	D
E	P1_4	P1_3	P1_2															ADCC1I12	ADCC1I20	ADCC1I21	E
F	P1_7	P1_6	P1_5															ADCC1I22	ADCC1I31	ADCC1I32	F
G	P1_9	P1_8	VSS	VSS													ADCC1I30	ADCC1I40	ADCC1I41	ADCC1I42	G
H	P1_11	P1_10	EVCC	EVCC													ADCC1I60	ADCC1I50	ADCC1I51	ADCC1I52	H
J	P1_15	P1_14	P1_13	P1_12					VDD	VSS	VSS	VDD					ADCC1I62	ADCC1I61	P3_7	P3_6	J
K	P0_3	P0_2	P0_1	P0_0					VDD	VSS	VSS	VDD					P3_3	P3_4	P3_5	P3_2	K
L	P6_3	P6_2	P0_4	P6_1					VDD	VSS	VSS	VDD					P5_8	P5_9	P3_1	P3_0	L
M	P0_8	P0_7	P0_5	P0_6					VDD	VSS	VSS	VDD					VSS	VSS	P5_7	P5_6	M
N	P6_0	P0_9	VSS	VSS													VDD	VSS	P5_4	P5_5	N
P	P6_4	P6_6	VDD	VDD													VDD	VSS	P5_2	P5_3	P
R	ERROROUT_M	P6_7	P6_5															EVCC	P5_0	P5_1	R
T	AUDRST	AUDATA2	AUDSYNC															P4_13	P4_14	P4_15	T
U	AUDCK	AUDATA3	AUDATA0				PLLVCC	PLLVSS	VSS	VDD	VDD	VSS	VSS	VDD				P4_10	P4_11	P4_12	U
V	AUDATA1	VDD	VDD	VSS	DCUTDO	DCUTCK	DCURDY	VSS	VSS	VSS	VSS	EVCC	VSS	VDD	P4_2	P4_5	EVCC	VSS	VDD	P4_9	V
W	VDD	VDD	VSS	VSS	DCUTMS	DCUTDI	VSS	NMI	MD0	SYSVCC	VSS	P0_11	P0_13	P0_15	P4_1	P4_4	P4_7	VSS	VSS	VDD	W
Y	VDD (N.C.)	VSS	X2	VSS	X1	VCC	EPTVOUT	DCUTRST	RESET	MD1	FLMODE	P0_10	P0_12	P0_14	P4_0	P4_3	P4_6	P4_8	VSS	VSS (N.C.)	Y
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	

Figure 1.3 Pin Connections of C1H

CAUTION

Pins labelled “power-supply name (N.C.)” do not affect the operation of the microcomputer even if they are left open-circuit. However, from the viewpoint of stability for the power supply, we recommend connecting them to the same wiring as pins with the same power supply name but with “(N.C.)” not included in the label. Moreover, be sure to mount solder balls on the board for the pins with “(N.C.)” in the label. These pins are internally connected to the pins with the same power-supply name but not including “(N.C.)” in the label.

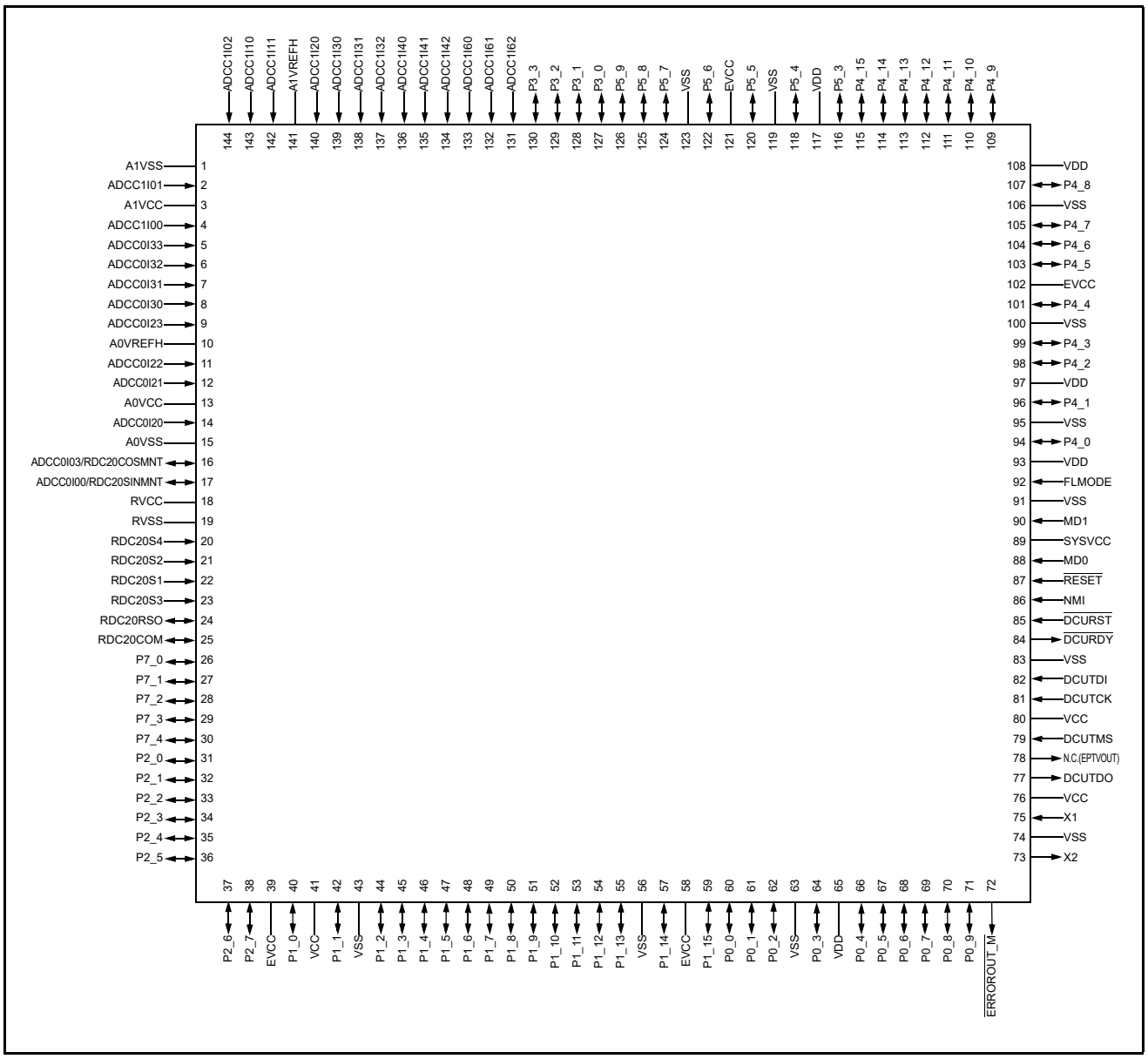


Figure 1.4 Pin Connections of C1M

Table 1.4 Pin Assignments of C1H (1/6)

Pin No.	Pin Name
1A	VSS
1B	EVCC
1C	P2_7 / TAUD0I15 / TAUD0O15 / TAPA0WN / INTP7
1D	P1_1 / TAUD1I1 / TAUD1O1 / ENCA0TIN1 / TSG30O7
1E	P1_4 / TAUD1I4 / TAUD1O4 / TAUD1O5 / TSG30O5
1F	P1_7 / TAUD1I7 / TAUD1O7 / TSG30O6
1G	P1_9 / TAUD1I9 / TAUD1O9 / TSG31O3
1H	P1_11 / TAUD1I11 / TAUD1O11 / TSG31O2
1J	P1_15 / TAUD1I15 / TAUD1O15 / TSG31O7
1K	P0_3 / TAUD0I3 / TAUD0O3 / TAUJ0I3 / TAUJ0O3 / CAN2RX / INTP0
1L	P6_3
1M	P0_8 / TAUD0I8 / TAUD0O8 / TAUJ0I3 / TAUJ0O3 / ENCA0EC / INTP5
1N	P6_0
1P	P6_4 / TAPA0ESO
1R	$\overline{\text{ERROROUT\_M}}$
1T	$\overline{\text{AUDRST}}$
1U	AUDCK
1V	AUDATA1
1W	VDD
1Y	VDD
2A	VSS
2B	EVCC
2C	P2_6 / TAUD0I14 / TAUD0O14 / TAPA0WP / INTP6
2D	P1_0 / TAUD1I0 / TAUD1O0 / TAUD1O1 / ENCA0TIN0 / TSG30O0 / TAPA2ESO
2E	P1_3 / TAUD1I3 / TAUD1O3 / TSG30O3
2F	P1_6 / TAUD1I6 / TAUD1O6 / TAUD1O7 / TSG30O4
2G	P1_8 / TAUD1I8 / TAUD1O8 / TAUD1O9 / TSG31O1
2H	P1_10 / TAUD1I10 / TAUD1O10 / TAUD1O11 / TSG31O5
2J	P1_14 / TAUD1I14 / TAUD1O14 / TAUD1O15 / TSG31O0 / TAPA3ESO
2K	P0_2 / TAUD0I2 / TAUD0O2 / TAUJ0I2 / TAUJ0O2
2L	P6_2
2M	P0_7 / TAUD0I7 / TAUD0O7 / TAUJ0I2 / TAUJ0O2 / ENCA0E1 / INTP4
2N	P0_9 / TAUD0I9 / TAUD0O9 / INTP6
2P	P6_6 / TAUD1I5 / TAUD1O5 / $\overline{\text{ADCC1TRG}}$
2R	P6_7 / TAUD1I7 / TAUD1O7 / $\overline{\text{ADCC0TRG}}$
2T	AUDATA2
2U	AUDATA3
2V	VDD
2W	VDD
2Y	VSS
3A	P2_4 / TAUD0I12 / TAUD0O12 / TAPA0VP / INTP4
3B	P2_5 / TAUD0I13 / TAUD0O13 / TAPA0VN / INTP5
3C	VSS
3D	VCC

Table 1.4 Pin Assignments of C1H (2/6)

Pin No.	Pin Name
3E	P1_2 / TAUD1I2 / TAUD1O2 / TAUD1O3 / TSG30O1
3F	P1_5 / TAUD1I5 / TAUD1O5 / TSG30O2
3G	VSS
3H	EVCC
3J	P1_13 / TAUD1I13 / TAUD1O13 / TSG31O6
3K	P0_1 / TAUD0I1 / TAUD0O1 / TAUJ0I1 / TAUJ0O1
3L	P0_4 / TAUD0I4 / TAUD0O4 / CAN2TX / INTP1
3M	P0_5 / TAUD0I5 / TAUD0O5 / TAUJ0I0 / TAUJ0O0 / INTP2
3N	VSS
3P	VDD
3R	P6_5 / TAPA1ESO
3T	$\overline{\text{AUDSYNC}}$
3U	AUDATA0
3V	VDD
3W	VSS
3Y	X2
4A	P2_2 / TAUD0I10 / TAUD0O10 / TAPA0UP / INTP2
4B	P2_3 / TAUD0I11 / TAUD0O11 / TAPA0UN / INTP3
4C	VSS
4G	VSS
4H	EVCC
4J	P1_12 / TAUD1I12 / TAUD1O12 / TAUD1O13 / TSG31O4
4K	P0_0 / TAUD0I0 / TAUD0O0 / TAUJ0I0 / TAUJ0O0
4L	P6_1
4M	P0_6 / TAUD0I6 / TAUD0O6 / TAUJ0I1 / TAUJ0O1 / ENCA0E0 / INTP3
4N	VSS
4P	VDD
4V	VSS
4W	VSS
4Y	VSS
5A	P2_0 / TAUD0I3 / TAUD0O3 / INTP0
5B	P2_1 / TAUD0I4 / TAUD0O4 / INTP1
5C	VSS
5V	DCUTDO / LPDO / FLSCI3TX
5W	DCUTMS
5Y	X1
6A	P7_3 / SCI0RXD
6B	P7_4 / SCI0TXD
6C	EVCC
6V	DCUTCK / LPDCLK / FLSCI3SCK
6W	DCUTDI / LPDI / FLSCI3RX
6Y	VCC
7A	P7_0 / ENCA1TIN0
7B	P7_1 / ENCA1TIN1

Table 1.4 Pin Assignments of C1H (3/6)

Pin No.	Pin Name
7C	P7_2
7D	EVCC
7U	PLLVCC
7V	$\overline{\text{DCURDY}}$ / LPDCLKOUT
7W	VSS
7Y	EPTVOUT
8A	RDC20S1
8B	RDC20S3
8C	RDC20COM
8D	RDC20RSO
8U	PLLVSS
8V	VSS
8W	NMI
8Y	$\overline{\text{DCUTRST}}$ / $\overline{\text{LPDRST}}$
9A	RDC20S4
9B	RDC20S2
9C	RVSS
9D	RVSS
9J	VDD
9K	VDD
9L	VDD
9M	VDD
9U	VSS
9V	VSS
9W	MD0
9Y	$\overline{\text{RESET}}$
10A	RDC21S1
10B	RDC21S3
10C	RVCC
10D	RVCC
10J	VSS
10K	VSS
10L	VSS
10M	VSS
10U	VDD
10V	VSS
10W	SYSVCC
10Y	MD1
11A	RDC21S4
11B	RDC21S2
11C	RDC21COM
11D	RDC21RSO
11J	VSS
11K	VSS

Table 1.4 Pin Assignments of C1H (4/6)

Pin No.	Pin Name
11L	VSS
11M	VSS
11U	VDD
11V	VSS
11W	VSS
11Y	FLMODE
12A	ADCC0I00 / RDC20SINMNT
12B	ADCC0I02
12C	ADCC0I11
12D	ADCC0I01
12J	VDD
12K	VDD
12L	VDD
12M	VDD
12U	VSS
12V	EVCC
12W	P0_11 / TAUD0I11 / TAUD0O11 / TAPA1UN
12Y	P0_10 / TAUD0I10 / TAUD0O10 / TAPA1UP / INTP7
13A	ADCC0I03 / RDC20COSMNT
13B	ADCC0I13
13C	ADCC0I22
13D	ADCC0I10
13U	VSS
13V	VSS
13W	P0_13 / TAUD0I13 / TAUD0O13 / TAPA1VN
13Y	P0_12 / TAUD0I12 / TAUD0O12 / TAPA1VP
14A	ADCC0I20 / RDC21SINMNT
14B	ADCC0I21 / RDC21COSMNT
14C	ADCC0I31
14D	ADCC0I12
14U	VDD
14V	VDD
14W	P0_15 / TAUD0I15 / TAUD0O15 / TAPA1WN
14Y	P0_14 / TAUD0I14 / TAUD0O14 / TAPA1WP
15A	ADCC0I23
15B	ADCC0I30
15C	ADCC0I32
15V	P4_2 / CSIH1SC
15W	P4_1 / CSIH1SO
15Y	P4_0 / CSIH1SI
16A	A0VSS
16B	ADCC0I33
16C	A0VSS
16V	P4_5 / CAN0RX / CSIH1CSS2

Table 1.4 Pin Assignments of C1H (5/6)

Pin No.	Pin Name
16W	P4_4 / CAN3TX / CSIH0RYI / CSIH0RYO / CSIH1CSS1
16Y	P4_3 / CAN3RX / $\overline{\text{CSIH0SSI}}$ / CSIH1CSS0
17A	A0VCC
17B	A0VSS
17C	A0VSS
17G	ADCC1I30
17H	ADCC1I60
17J	ADCC1I62
17K	P3_3 / TAUD1I7 / TAUD1O7 / TAPA2ESO
17L	P5_8 / SCI2RXD
17M	VSS
17N	VDD
17P	VDD
17V	EVCC
17W	P4_7 / CSIH0SI / CSIH1SSI
17Y	P4_6 / CAN0TX / CSIH1CSS3
18A	A0VREFH
18B	ADCC1I00
18C	ADCC1I01
18D	ADCC1I10
18E	ADCC1I12
18F	ADCC1I22
18G	ADCC1I40
18H	ADCC1I50
18J	ADCC1I61
18K	P3_4 / $\overline{\text{ADCC0TRG}}$ / TAPA3ESO
18L	P5_9 / SCI2TXD
18M	VSS
18N	VSS
18P	VSS
18R	EVCC
18T	P4_13 / ENCA1EC / CSIH0CSS3
18U	P4_10 / TPBA0O / CSIH0CSS0
18V	VSS
18W	VSS
18Y	P4_8 / CSIH0SO / CSIH1RYI / CSIH1RYO
19A	A1VCC
19B	A1VSS
19C	A1VSS
19D	ADCC1I02
19E	ADCC1I20
19F	ADCC1I31
19G	ADCC1I41
19H	ADCC1I51

Table 1.4 Pin Assignments of C1H (6/6)

Pin No.	Pin Name
19J	P3_7
19K	P3_5 / TAPA0ESO
19L	P3_1 / TAUD0I7 / TAUD0O7 / $\overline{\text{ADCC0TRG}}$
19M	P5_7 / SCI2SCK
19N	P5_4 / RLIN20RX / SCI1TXD
19P	P5_2 / RLIN21RX / SCI0SCK
19R	P5_0 / RLIN22RX / SCI0RXD
19T	P4_14 / CAN1RX
19U	P4_11 / ENCA1E0 / CSIH0CSS1
19V	VDD
19W	VSS
19Y	VSS
20A	A1VSS
20B	A1VSS
20C	A1VREFH
20D	ADCC1I11
20E	ADCC1I21
20F	ADCC1I32
20G	ADCC1I42
20H	ADCC1I52
20J	P3_6 / TAPA1ESO
20K	P3_2 / TAUD1I5 / TAUD1O5 / $\overline{\text{ADCC1TRG}}$
20L	P3_0 / TAUD0I5 / TAUD0O5
20M	P5_6 / TPBA1O / TAPA0ESO
20N	P5_5 / RLIN20TX / SCI1SCK / $\overline{\text{ERROROUT\_C}}$
20P	P5_3 / RLIN21TX / SCI1RXD
20R	P5_1 / RLIN22TX / SCI0TXD
20T	P4_15 / CAN1TX / $\overline{\text{ERROROUT\_C}}$
20U	P4_12 / ENCA1E1 / CSIH0CSS2
20V	P4_9 / CSIH0SC
20W	VDD
20Y	VSS

Table 1.5 Pin Assignments of C1M (1/4)

Pin No.	Pin Name
1	A1VSS
2	ADCC1I01
3	A1VCC
4	ADCC1I00
5	ADCC0I33
6	ADCC0I32
7	ADCC0I31
8	ADCC0I30
9	ADCC0I23
10	A0VREFH
11	ADCC0I22
12	ADCC0I21
13	A0VCC
14	ADCC0I20
15	A0VSS
16	ADCC0I03 / RDC20COSMNT
17	ADCC0I00 / RDC20SINMNT
18	RVCC
19	RVSS
20	RDC20S4
21	RDC20S2
22	RDC20S1
23	RDC20S3
24	RDC20RSO
25	RDC20COM
26	P7_0 / ENCA1TIN0
27	P7_1 / ENCA1TIN1
28	P7_2
29	P7_3 / SCI0RXD
30	P7_4 / SCI0TXD
31	P2_0 / TAUD0I3 / TAUD0O3 / INTP0
32	P2_1 / TAUD0I4 / TAUD0O4 / INTP1
33	P2_2 / TAUD0I10 / TAUD0O10 / TAPA0UP / INTP2
34	P2_3 / TAUD0I11 / TAUD0O11 / TAPA0UN / INTP3
35	P2_4 / TAUD0I12 / TAUD0O12 / TAPA0VP / INTP4
36	P2_5 / TAUD0I13 / TAUD0O13 / TAPA0VN / INTP5
37	P2_6 / TAUD0I14 / TAUD0O14 / TAPA0WP / INTP6
38	P2_7 / TAUD0I15 / TAUD0O15 / TAPA0WN / INTP7
39	EVCC
40	P1_0 / TAUD1I0 / TAUD1O0 / TAUD1O1 / ENCA0TIN0 / TSG30O0 / TAPA2ESO
41	VCC
42	P1_1 / TAUD1I1 / TAUD1O1 / ENCA0TIN1 / TSG30O7
43	VSS
44	P1_2 / TAUD1I2 / TAUD1O2 / TAUD1O3 / TSG30O1

Table 1.5 Pin Assignments of C1M (2/4)

Pin No.	Pin Name
45	P1_3 / TAUD1I3 / TAUD1O3 / TSG30O3
46	P1_4 / TAUD1I4 / TAUD1O4 / TAUD1O5 / TSG30O5
47	P1_5 / TAUD1I5 / TAUD1O5 / TSG30O2
48	P1_6 / TAUD1I6 / TAUD1O6 / TAUD1O7 / TSG30O4
49	P1_7 / TAUD1I7 / TAUD1O7 / TSG30O6
50	P1_8 / TAUD1I8 / TAUD1O8 / TAUD1O9 / TSG31O1
51	P1_9 / TAUD1I9 / TAUD1O9 / TSG31O3
52	P1_10 / TAUD1I10 / TAUD1O10 / TAUD1O11 / TSG31O5
53	P1_11 / TAUD1I11 / TAUD1O11 / TSG31O2
54	P1_12 / TAUD1I12 / TAUD1O12 / TAUD1O13 / TSG31O4
55	P1_13 / TAUD1I13 / TAUD1O13 / TSG31O6
56	VSS
57	P1_14 / TAUD1I14 / TAUD1O14 / TAUD1O15 / TSG31O0 / TAPA3ESO
58	EVCC
59	P1_15 / TAUD1I15 / TAUD1O15 / TSG31O7
60	P0_0 / TAUD0I0 / TAUD0O0 / TAUJ0I0 / TAUJ0O0
61	P0_1 / TAUD0I1 / TAUD0O1 / TAUJ0I1 / TAUJ0O1
62	P0_2 / TAUD0I2 / TAUD0O2 / TAUJ0I2 / TAUJ0O2
63	VSS
64	P0_3 / TAUD0I3 / TAUD0O3 / TAUJ0I3 / TAUJ0O3 / CAN2RX / INTP0
65	VDD
66	P0_4 / TAUD0I4 / TAUD0O4 / CAN2TX / INTP1
67	P0_5 / TAUD0I5 / TAUD0O5 / TAUJ0I0 / TAUJ0O0 / INTP2
68	P0_6 / TAUD0I6 / TAUD0O6 / TAUJ0I1 / TAUJ0O1 / ENCA0E0 / INTP3
69	P0_7 / TAUD0I7 / TAUD0O7 / TAUJ0I2 / TAUJ0O2 / ENCA0E1 / INTP4
70	P0_8 / TAUD0I8 / TAUD0O8 / TAUJ0I3 / TAUJ0O3 / ENCA0EC / INTP5
71	P0_9 / TAUD0I9 / TAUD0O9 / INTP6
72	$\overline{\text{ERROROUT\_M}}$
73	X2
74	VSS
75	X1
76	VCC
77	DCUTDO / LPDO / FLSCI3TX
78	N.C.(EPTVOUT)
79	DCUTMS
80	VCC
81	DCUTCK / LPDCLK / FLSCI3SCK
82	DCUTDI / LPDI / FLSCI3RX
83	VSS
84	$\overline{\text{DCURDY}}$ / LPDCLKOUT
85	$\overline{\text{DCUTRST}}$ / $\overline{\text{LPDRST}}$
86	NMI
87	$\overline{\text{RESET}}$
88	MD0

Table 1.5 Pin Assignments of C1M (3/4)

Pin No.	Pin Name
89	SYSVCC
90	MD1
91	VSS
92	FLMODE
93	VDD
94	P4_0 / CSIH1SI
95	VSS
96	P4_1 / CSIH1SO
97	VDD
98	P4_2 / CSIH1SC
99	P4_3 / CAN3RX / $\overline{\text{CSIH0SSI}}$ / CSIH1CSS0
100	VSS
101	P4_4 / CAN3TX / CSIH0RYI / CSIH0RYO / CSIH1CSS1
102	EVCC
103	P4_5 / CAN0RX / CSIH1CSS2
104	P4_6 / CAN0TX / CSIH1CSS3
105	P4_7 / CSIH0SI / $\overline{\text{CSIH1SSI}}$
106	VSS
107	P4_8 / CSIH0SO / CSIH1RYI / CSIH1RYO
108	VDD
109	P4_9 / CSIH0SC
110	P4_10 / TPBA0O / CSIH0CSS0
111	P4_11 / ENCA1E0 / CSIH0CSS1
112	P4_12 / ENCA1E1 / CSIH0CSS2
113	P4_13 / ENCA1EC / CSIH0CSS3
114	P4_14 / CAN1RX
115	P4_15 / CAN1TX / $\overline{\text{ERROROUT\_C}}$
116	P5_3 / SCI1RXD
117	VDD
118	P5_4 / SCI1TXD
119	VSS
120	P5_5 / SCI1SCK / $\overline{\text{ERROROUT\_C}}$
121	EVCC
122	P5_6 / TAPA0ESO
123	VSS
124	P5_7 / SCI2SCK
125	P5_8 / SCI2RXD
126	P5_9 / SCI2TXD
127	P3_0 / TAUD0I5 / TAUD0O5
128	P3_1 / TAUD0I7 / TAUD0O7 / $\overline{\text{ADCC0TRG}}$
129	P3_2 / TAUD1I5 / TAUD1O5 / $\overline{\text{ADCC1TRG}}$
130	P3_3 / TAUD1I7 / TAUD1O7 / TAPA2ESO
131	ADCC1I62
132	ADCC1I61

Table 1.5 Pin Assignments of C1M (4/4)

Pin No.	Pin Name
133	ADCC1I60
134	ADCC1I42
135	ADCC1I41
136	ADCC1I40
137	ADCC1I32
138	ADCC1I31
139	ADCC1I30
140	ADCC1I20
141	A1VREFH
142	ADCC1I11
143	ADCC1I10
144	ADCC1I02

Section 2 Pins

2.1 Port Functions

2.1.1 Features

Port Group

The RH850/C1x provides the following port groups, indicated by the numbers in the table below.

Table 2.1 Port Groups in RH850/C1x

Product		Number of Groups	Name of Group
C1H	BGA252	8	P0 to P7
C1M	QFP144	7	P0 to P5, P7

Port Group Index n

Each port group is identified by its own index “n” (n = 0 to 7) throughout this section; e.g. PMCn for the port mode control register of the Pn pin.

Register Base Address

Port base addresses are listed in the following table.

Port register addresses are given as offsets from the base addresses in general.

Table 2.2 Register Base Address

Base Address Name	Base Address
<PORT_base>	FFC1 0000_H

2.1.2 Overview

This product has various pins for input/output (I/O) functions, known as ports. The ports are organized in port groups.

This product also has several control registers to allocate the functions other than general-purpose I/O to the corresponding pins.

For definitions of pin, port, and port group, see **Section 2.1.2.1, Terms**.

2.1.2.1 Terms

The terms descried in this section are defined as follows.

- Port group
  A port group consists of a maximum of 16 pins. The number of pins differs depending on the port group. All the pins of a specific port group are controlled by the same port control register.
- Port mode/Port
  A pin in port mode functions as a general-purpose I/O pin. This general-purpose I/O pin is called “port”, and is named as Pn_m. For example, P0_7 indicates port 7 of port group 0.
- Alternative mode
  A pin in alternative mode functions as an I/O pin for peripheral functions. A pin has multiple peripheral functions, and the functions to be used can be selected by control registers.

2.1.2.2 Overview of Pin Functions

Pins can operate in three modes.

- Port mode (PMCn.PMCn_m = 0)
  A pin in port mode operates as a general-purpose input/output pin. The I/O mode is selected by setting the PMn.PMn_m bit.
- Software I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 0)
  In this mode, the pins operate as alternative functions. The I/O mode is selected by setting the PMn.PMn_m control bit by using software.
- Direct I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 1)
  In this mode, the pins operate as alternative functions. Unlike the software I/O alternative mode, however, the I/O mode is selected by the alternative function.

Table 2.3 shows the outline of the register settings.

Table 2.3 Pin Function Configuration (Outline)

Mode	Bit			I/O
	PMCn_m	PMn_m	PIPCn_m	
Port mode	0	0	0/1	Output mode
		1	0/1	Input mode*1
Software I/O control alternative mode	1	0	0	Output mode
		1	0	Input mode*2
Direct I/O control alternative mode		0/1	1	Controlled by the alternative function*2

Note 1. The input buffer should be enabled (PIBCn_m = 1).
Note 2. When the pin is used as an input pin in alternative mode, set PIBCn_m to 0.

If a pin is in alternative mode (PMCn.PMCn_m = 1), one of up to seven alternative functions can be selected for that pin by using the PFCn, PFCEn, and PFCAEn registers.

- Software I/O control alternative mode (PIPCn.PIPCn_m = 0):
  - Output (PMn_m = 0): ALT-OUT1 to ALT-OUT7
  - Input (PMn_m = 1): ALT-IN1 to ALT-IN7
- Direct I/O control alternative mode (PIPCn.PIPCn_m = 1):
  - The I/O mode for ALT-OUT1 to ALT-OUT7 and ALT-IN1 to ALT-IN7 is directly selected by the alternative function.

Table 2.4 Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)

Function	Register				I/O
	PFCAE	PFCE	PFC	PM*1	
Alternative output mode 1 (ALT-OUT1)	0	0	0	0	O
Alternative input mode 1 (ALT-IN1)	0	0	0	1	I
Alternative output mode 2 (ALT-OUT2)	0	0	1	0	O
Alternative input mode 2 (ALT-IN2)	0	0	1	1	I
Alternative output mode 3 (ALT-OUT3)	0	1	0	0	O
Alternative input mode 3 (ALT-IN3)	0	1	0	1	I
Alternative output mode 4 (ALT-OUT4)	0	1	1	0	O
Alternative input mode 4 (ALT-IN4)	0	1	1	1	I
Alternative output mode 5 (ALT-OUT5)	1	0	0	0	O
Alternative input mode 5 (ALT-IN5)	1	0	0	1	I
Alternative output mode 6 (ALT-OUT6)	1	0	1	0	O
Alternative input mode 6 (ALT-IN6)	1	0	1	1	I
Alternative output mode 7 (ALT-OUT7)	1	1	0	0	O
Alternative input mode 7 (ALT-IN7)	1	1	0	1	I

Note 1. When PIPCn.PIPCn_m = 1, the I/O direction is directly controlled by the peripheral (alternative) function and PM is ignored.

If a pin is in alternative mode (PMCn.PMCn_m = 1), one of several alternative functions can be selected for that pin by using the PFCn, PFCEn, and PFCAEn registers.

2.1.2.3 Pin Data Input/Output

The registers used for data input and output are described below.

The source of the data to be read via the PPRn register depends on pin mode.

Output data

In port mode (PMCn.PMCn_m = 0), the value of Pn.Pn_m is output from the Pn_m pin.

Input data

A read operation of the PPRn register returns either the value of the Pn_m pin, the associated bit of the port register Pn.Pn_m, or the data output by an alternative function.

The source of the data read via PPRn depends on pin mode and setting of several control bits.

Table 2.5 summarizes the differences of PPRn read modes.

Table 2.5 PPRn_m Read Values

PMC n_m	PM n_m	PIBC n_m	PIPC n_m	Mode	PPRn_m Read Value
0	1	0	X	Port input, input buffer disabled	Pn.Pn_m bit
		1		Port input, input buffer enabled	Pn_m pin
	0	X		Port push-pull output	Pn.Pn_m bit*1
1	1	X	0	Software I/O control alternative input	Pn_m pin
	0			Software I/O control alternative output	Alternative function internal output signal*1
	X		1	Direct I/O control alternative mode	I/O port in alternative mode: • Input: Pn_m pin • Output: Output signal from the alternative function*1

Note 1. When PBDCn_m = 1, the level of the Pn_m pin is returned by the PPRn_m bit.

The control registers in **Table 2.5** have the following effects:

- PMCn.PMCn_m
  This bit selects port mode (PMCn_m = 0) or alternative mode (PMCn_m = 1).
- PMn.PMn_m
  This bit selects input (PMn_m = 1) or output (PMn_m = 0) in port mode (PMCn_m = 0) and software I/O control alternative mode (PMCn_m = 1, PIPCn_m = 0).
- PIBCn.PIBCn_m
  This bit disables (PIBCn_m = 0) or enables (PIBCn_m = 1) the input buffer in input port mode (PMCn_m = 0 and PMn_m = 1). When the input buffer is disabled, PPRn_m reads the Pn.Pn_m bit, otherwise the Pn_m pin level is returned.
- PIPCn.PIPCn_m
  This bit selects the software or direct I/O control alternative mode.
- PBDCn.PBDCn_m
  In output mode, when this bit is set to 1, the pin enters the bidirectional mode. In bidirectional mode, the level of the signal on a Pn_m pin can be read from PPPn.PPRn_m.

Writing to the Pn Register

The data to be output via port Pn_m in port mode (PMCn.PMCn_m = 0) is held in the port register Pn.

Pn data can be overwritten in two ways:

- By writing data directly to the Pn register.
  In this case, new data can be written directly to the Pn register.
- By performing an indirect bitwise operation (a “set”, “reset”, or “not” operation) on the Pn register.
  An indirect bitwise operation (“set”, “reset”, or “not”), can be performed on the Pn register by using the following two registers:
  - Port set/reset register: PSRn
    If the PSRn.PSRn_(m + 16) bit = 1, the value of the Pn.Pn_m bit is determined by the value of the PSRn.PSRn_m bit.
    In other words, the Pn_m bit can be set or reset without writing directly to the Pn register.
  - Port NOT register: PNOTn
    By setting PNOTn.PNOTn_m to 1, the Pn.Pn_m bit can be inverted without writing directly to the Pn register.

An indirect bitwise operation on the Pn register (“set”, “reset”, or “not”), has no effect on the bits that do not need to be updated, allowing you to overwrite only the bit or bits that need to be overwritten.

2.1.3 Port Type

Figure 2.1 shows the overall configuration of the pins. For the details of port blocks, see **Figure 2.2**.

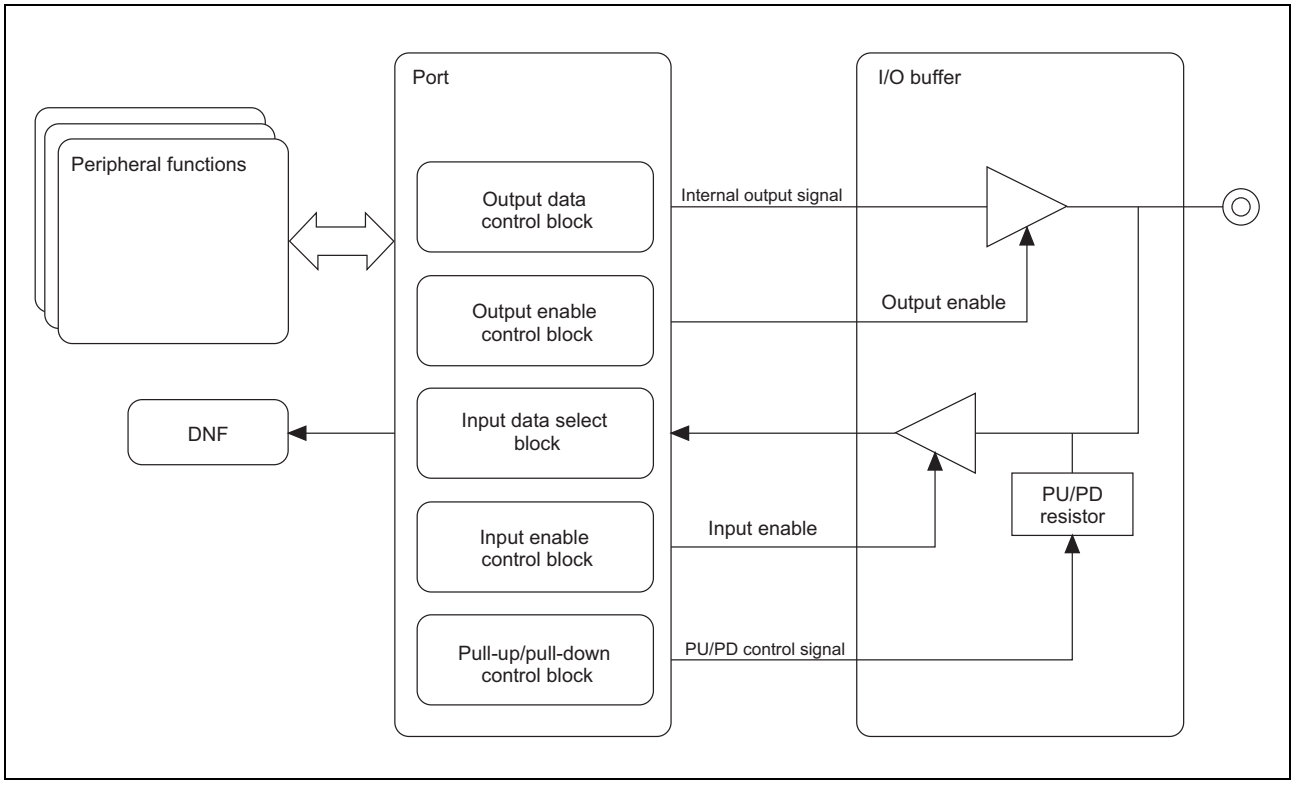
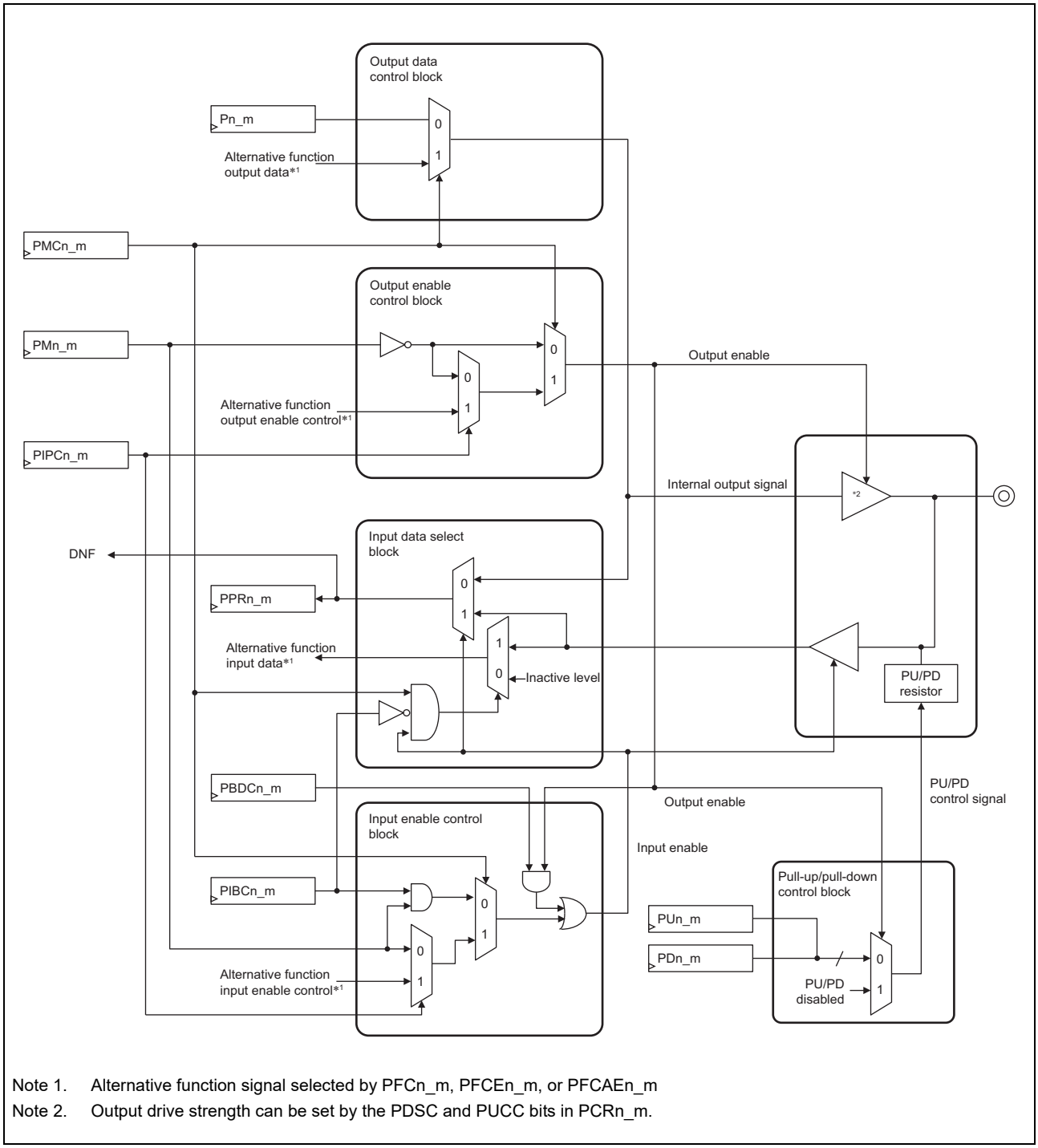


Figure 2.1 Block Diagram of Pin Configuration

Figure 2.2 shows the logical circuitry of the port control functions. The diagram is only a logical reference and does not show the real circuitry.



Note 1. Alternative function signal selected by PFCn_m, PFCEn_m, or PFCAEn_m
Note 2. Output drive strength can be set by the PDSC and PUCC bits in PCRn_m.

Figure 2.2 Port Control Logic Diagram

2.1.4 Port Group Configuration Register

This section starts with an overview of all configuration registers and then presents all registers in detail. The configuration registers are classified as follows:

- **2.1.4.2, Configuration of Pin Function**
- **2.1.4.3, Pin Data Input/Output**
- **2.1.4.4, Configuration of Electrical Characteristics Registers**
- **2.1.4.5, Pin-Unit Register**

2.1.4.1 Outline

The following registers are used for the configuration of the individual pins of the port groups:

Table 2.6 Registers for Port Group Configuration

Register Name	Symbol	Register Configuration Unit	Address
Port register	Pn	Port group	<PORT_base> + 0000_H + n × 40_H
Port set/reset register	PSRn	Port group	<PORT_base> + 0004_H + n × 40_H
Port NOT register	PNOTn	Port group	<PORT_base> + 0008_H + n × 40_H
Port pin read register	PPRn	Port group	<PORT_base> + $000C_H$ + n × 40_H
Port mode register	PMn	Port group	<PORT_base> + 0010_H + n × 40_H
Port mode control register	PMCn	Port group	<PORT_base> + 0014_H + n × 40_H
Port function control register	PFCn	Port group	<PORT_base> + 0018_H + n × 40_H
Port function control expansion register	PFCEn	Port group	<PORT_base> + $001C_H$+ n × 40_H
Port mode set/reset register	PMSRn	Port group	<PORT_base> + 0020_H + n × 40_H
Port mode control set/reset register	PMCSRn	Port group	<PORT_base> + 0024_H + n × 40_H
Port function control additional expansion register	PFCAEn	Port group	<PORT_base> + 0028_H + n × 40_H
Port input buffer control register	PIBCn	Port group	<PORT_base> + 4000_H+ n × 40_H
Port bidirectional control register	PBDCn	Port group	<PORT_base> + 4004_H + n × 40_H
Port IP control register	PIPCn	Port group	<PORT_base> + 4008_H + n × 40_H
Pull-up option register	PUn	Port group	<PORT_base> + $400C_H$ + n × 40_H
Pull-down option register	PDn	Port group	<PORT_base> + 4010_H+ n × 40_H
Port control register	PCRn_m	Pin	<PORT_base> + 2000_H + n × 40_H + m × 4_H

Note: n: Port group number
m: Bit number in a port group

Base address

The base address of PORTn <PORT_base> is defined in Register Base Address in **Section 2.1.1, Features**.

Register value after reset

The value after resets of the registers after reset release depend on the port, and are not described in the following register descriptions, but are given in **Section 2.2.1.1, List of the C1H Port Registers and Section 2.2.2.1, List of the C1M Port Registers**.

2.1.4.2 Configuration of Pin Function

(1) PMCn — Port Mode Control Register

This register specifies whether the individual pins of port group n are in port mode or in alternative mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCn_15	PMCn_14	PMCn_13	PMCn_12	PMCn_11	PMCn_10	PMCn_9	PMCn_8	PMCn_7	PMCn_6	PMCn_5	PMCn_4	PMCn_3	PMCn_2	PMCn_1	PMCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.7 PMCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMCn_[15:0]	Specify the operation mode of the corresponding pin: 0: Port mode 1: Alternative mode

(2) PMCSRn — Port Mode Control Set/Reset Register

This register provides an alternative method to write data to a bit in the PMCn register.

The 16 higher-order bits of PMCSRn (PMCSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PMCSRn_[15:0]) to the corresponding PMCn.PMCn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PMCSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMCn register.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers.**

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMCSRn_31	PMCSRn_30	PMCSRn_29	PMCSRn_28	PMCSRn_27	PMCSRn_26	PMCSRn_25	PMCSRn_24	PMCSRn_23	PMCSRn_22	PMCSRn_21	PMCSRn_20	PMCSRn_19	PMCSRn_18	PMCSRn_17	PMCSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCSRn_15	PMCSRn_14	PMCSRn_13	PMCSRn_12	PMCSRn_11	PMCSRn_10	PMCSRn_9	PMCSRn_8	PMCSRn_7	PMCSRn_6	PMCSRn_5	PMCSRn_4	PMCSRn_3	PMCSRn_2	PMCSRn_1	PMCSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.8 PMCSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMCSRn_ [31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PMCSRn_m is to be written to PMCn_m: 0: PMCn_m does not depend on PMCSRn_m. 1: The value of PMCn_m is the same as that of PMCSRn_m. Example: If PMCSRn.PMCSRn_31 = 1, the value of bit PMCSRn.PMCSRn_15 is written to bit PMCn.PMCn_15.
15 to 0	PMCSRn_ [15:0]	Data bits that specify the PMCn_m value when the corresponding higher-order bit PMCSRn_(m+16) is 1: 0: PMCn_m = 0 1: PMCn_m = 1

(3) PIPCn — Port IP Control Register

This register specifies whether the I/O direction of pin Pn_m is controlled by the port mode register PMn.PMn_m or by an alternative function.

When the Pn_m pin is operated in alternative mode (PMCn.PMCn_m = 1) and the alternative function directly controls the I/O direction of Pn_m, PIPCn.PIPCn_m must be set to 1 as well. This hands over I/O control to the alternative function and overrules the PMn.PMn_m setting.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers.**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIPCn_15	PIPCn_14	PIPCn_13	PIPCn_12	PIPCn_11	PIPCn_10	PIPCn_9	PIPCn_8	PIPCn_7	PIPCn_6	PIPCn_5	PIPCn_4	PIPCn_3	PIPCn_2	PIPCn_1	PIPCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.9 PIPCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIPCn_[15:0]	Specify the I/O mode: 0: I/O mode is selected by PMn.PMn_m (software I/O control). 1: I/O mode is selected by the peripheral function (direct I/O control).

(4) PMn — Port Mode Register

The PMn register specifies whether the individual pins of port group n are in input mode or in output mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers.**

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMn_15	PMn_14	PMn_13	PMn_12	PMn_11	PMn_10	PMn_9	PMn_8	PMn_7	PMn_6	PMn_5	PMn_4	PMn_3	PMn_2	PMn_1	PMn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.10 PMn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMn_[15:0]	Specify input/output mode of the corresponding pin: 0: Output mode (output enabled) 1: Input mode (output disabled)

NOTES

1. To use a pin in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1), the input buffer must be enabled (PIBCn.PIBCn_m = 1).
2. PMn_m specifies the I/O direction in port mode (PMCn.PMCn_m = 0) and alternative mode (PMCn.PMCn_m = 1) because PIPCn.PIPCn_m = 0 after reset.

(5) PMSRn — Port Mode Set/Reset Register

This register provides an alternative method to write data to a bit in the PMn register.

The 16 higher-order bits of PMSRn (PMSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PMSRn_[15:0]) to the corresponding PMn.PMn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PMSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMn register.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMSRn_31	PMSRn_30	PMSRn_29	PMSRn_28	PMSRn_27	PMSRn_26	PMSRn_25	PMSRn_24	PMSRn_23	PMSRn_22	PMSRn_21	PMSRn_20	PMSRn_19	PMSRn_18	PMSRn_17	PMSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMSRn_15	PMSRn_14	PMSRn_13	PMSRn_12	PMSRn_11	PMSRn_10	PMSRn_9	PMSRn_8	PMSRn_7	PMSRn_6	PMSRn_5	PMSRn_4	PMSRn_3	PMSRn_2	PMSRn_1	PMSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.11 PMSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMSRn_[31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PMSRn_m is written to PMn_m: 0: PMn_m does not depend on PMSRn_m. 1: The value of PMn_m is the same as that of PMSRn_m. Example: If PMSRn.PMSRn_31 = 1, the value of bit PMSRn.PMSRn_15 is written to bit PMn.PMn_15.
15 to 0	PMSRn_[15:0]	Data bits that specify the PMn_m value when the corresponding higher-order bit PMSRn_(m+16) is 1: 0: PMn_m = 0 1: PMn_m = 1

(6) PIBCn — Port Input Buffer Control Register

This register enables and disables the input buffer when a pin is used in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1). However, when the pin is used as an input pin in software I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 0) or in direct I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 1), set PIBCn.PIBCn_m to 0.

And when pins are in bidirectional mode (PBDCn.PBDCn_m = 1), alternative output level loopback function and pin output level read function can be selected by the setting of PIBCn.PIBCn_m.

Refer to **(1), PBDCn — Port Bidirectional Control Register** in **Section 2.1.4.3, Pin Data Input/Output**.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIBCn_15	PIBCn_14	PIBCn_13	PIBCn_12	PIBCn_11	PIBCn_10	PIBCn_9	PIBCn_8	PIBCn_7	PIBCn_6	PIBCn_5	PIBCn_4	PIBCn_3	PIBCn_2	PIBCn_1	PIBCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.12 PIBCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIBCn_[15:0]	Enable and disable the input buffer in input port mode. 0: Input buffer is disabled. 1: Input buffer is enabled.

NOTE

When the input buffer is disabled, through current does not flow even when the pin level is Hi-Z. Thus the pin does not need to be fixed to a high or low level externally.

(7) PFCn — Port Function Control Register

This register, together with the PFCEn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCn_15	PFCn_14	PFCn_13	PFCn_12	PFCn_11	PFCn_10	PFCn_9	PFCn_8	PFCn_7	PFCn_6	PFCn_5	PFCn_4	PFCn_3	PFCn_2	PFCn_1	PFCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.13 PFCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(8) PFCEn — Port Function Control Expansion Register

This register, together with the PFCn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCEn_15	PFCEn_14	PFCEn_13	PFCEn_12	PFCEn_11	PFCEn_10	PFCEn_9	PFCEn_8	PFCEn_7	PFCEn_6	PFCEn_5	PFCEn_4	PFCEn_3	PFCEn_2	PFCEn_1	PFCEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.14 PFCEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(9) PFCAEn — Port Function Control Additional Expansion Register

This register, together with the PFCn and PFCEn registers, specifies an alternative function of the pins.

Some alternative functions directly control input/output of pin Pn_m. For such alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCAEn_15	PFCAEn_14	PFCAEn_13	PFCAEn_12	PFCAEn_11	PFCAEn_10	PFCAEn_9	PFCAEn_8	PFCAEn_7	PFCAEn_6	PFCAEn_5	PFCAEn_4	PFCAEn_3	PFCAEn_2	PFCAEn_1	PFCAEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.15 PFCAEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCAEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

2.1.4.3 Pin Data Input/Output

(1) PBDCn — Port Bidirectional Control Register

When a pin is used in output mode, this register enables the input buffer and sets the port to bidirectional mode. The Pn_m pin level is read via PPRn.PPRn_m in bidirectional mode.

- Alternative output level loopback function
  When the Pn_m pin is used as the alternative output function, the actual pin output level based on the alternative output function can be looped back to the alternative input side by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m = 0. For example, the pin output level based on the first alternative function can be looped back to the same alternative input side. Also the pin output level can be read via PPRn.PPRn_m.
- Pin output level read function
  When the Pn_m pin is used as the general-purpose output port function or the alternative output function, the actual pin output level can be read via PPRn.PPRn_m by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m = 1. Under this setting, the pin output level will never be looped back to the alternative input side even in alternative output mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PBDCn_15	PBDCn_14	PBDCn_13	PBDCn_12	PBDCn_11	PBDCn_10	PBDCn_9	PBDCn_8	PBDCn_7	PBDCn_6	PBDCn_5	PBDCn_4	PBDCn_3	PBDCn_2	PBDCn_1	PBDCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.16 PBDCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PBDCn_[15:0]	Enable or disable bidirectional mode of the corresponding pin: 0: Bidirectional mode is disabled. 1: Bidirectional mode is enabled.

(2) PPRn — Port Pin Read Register

This register reflects an actual Pn_m pin level, a Pn.Pn_m bit value, or an output level of the alternative function. The value to be read depends on various control settings as described in **Table 2.5, PPRn_m Read Values**.

Access: This register is only readable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPRn_15	PPRn_14	PPRn_13	PPRn_12	PPRn_11	PPRn_10	PPRn_9	PPRn_8	PPRn_7	PPRn_6	PPRn_5	PPRn_4	PPRn_3	PPRn_2	PPRn_1	PPRn_0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2.17 PPRn Register Contents

Bit Position	Bit Name	Function
15 to 0	PPRn_[15:0]	Indicate a Pn_m pin level, a Pn.Pn_m value, or alternative function output level.

(3) Pn — Port Register

This register sets and holds the Pn.Pn_m data to be output via the related Pn_m port in output port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 0).

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Pn_15	Pn_14	Pn_13	Pn_12	Pn_11	Pn_10	Pn_9	Pn_8	Pn_7	Pn_6	Pn_5	Pn_4	Pn_3	Pn_2	Pn_1	Pn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.18 Pn Register Contents

Bit Position	Bit Name	Function
15 to 0	Pn_[15:0]	Set the output level of pin m (m = 0 to 15): 0: Low level output 1: High level output

NOTE

The bits of this register can be manipulated by various means; refer to the subsection, Write to the Pn Register in **Section 2.1.2.3, Pin Data Input/Output**.

(4) PNOTn — Port NOT Register

This register allows bit Pn_m of the port register Pn to be inverted without directly writing to Pn.

Access: This register is writable in 16-bit units. The read value is always 0000_H.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNOTn_15	PNOTn_14	PNOTn_13	PNOTn_12	PNOTn_11	PNOTn_10	PNOTn_9	PNOTn_8	PNOTn_7	PNOTn_6	PNOTn_5	PNOTn_4	PNOTn_3	PNOTn_2	PNOTn_1	PNOTn_0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 2.19 PNOTn Register Contents

Bit Position	Bit Name	Function
15 to 0	PNOTn_[15:0]	Specify if Pn.Pn_m is inverted or not: 0: Pn.Pn_m is not inverted (Pn_m→Pn_m). 1: Pn.Pn_m is inverted ($\overline{Pn\_m}$→Pn_m)

(5) PSRn — Port Set/Reset Register

This register provides an alternative method to write data to a bit in the Pn register.

The 16 higher-order bits of PSRn (PSRn_[31:16]) select or deselect writing of values from the 16 lower-order bits (PSRn_[15:0]) to the corresponding Pn.Pn_m bits.

Even when pins being used by multiple programs belong to the same port group, the PSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000H. Reading bits 15 to 0 returns the value of the PMCn register.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PSRn_31	PSRn_30	PSRn_29	PSRn_28	PSRn_27	PSRn_26	PSRn_25	PSRn_24	PSRn_23	PSRn_22	PSRn_21	PSRn_20	PSRn_19	PSRn_18	PSRn_17	PSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSRn_15	PSRn_14	PSRn_13	PSRn_12	PSRn_11	PSRn_10	PSRn_9	PSRn_8	PSRn_7	PSRn_6	PSRn_5	PSRn_4	PSRn_3	PSRn_2	PSRn_1	PSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.20 PSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PSRn_[31:16]	Enable bits that specify whether the value of the corresponding lower-order bit of PSRn_m is written to Pn_m: 0: Pn_m does not depend on PSRn_m. 1: The value of Pn_m is the same as that of PSRn_m. Example: When PSRn.PSRn_31 = 1, the value of bit PSRn.PSRn_15 is written to bit Pn.Pn_15 and output.
15 to 0	PSRn_[15:0]	Specify the Pn_m value when the corresponding higher-order bit PSRn_(m+16) is 1: 0: Pn_m = 0 1: Pn_m = 1

2.1.4.4 Configuration of Electrical Characteristics Registers

(1) PUn — Pull-Up Option Register

This register specifies whether an on-chip pull-up resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PUn_15	PUn_14	PUn_13	PUn_12	PUn_11	PUn_10	PUn_9	PUn_8	PUn_7	PUn_6	PUn_5	PUn_4	PUn_3	PUn_2	PUn_1	PUn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.21 PUn Register Contents

Bit Position	Bit Name	Function
15 to 0	PUn_[15:0]	Specify whether an on-chip pull-up resistor is connected to the corresponding pin: 0: No on-chip pull-up resistor is connected. 1: On-chip pull-up resistor is connected.

NOTES

1. Do not set PUn.PUn_m = 1 and PDn.PDn_m = 1 to a single pin.
2. The on-chip pull-up resistor has no effect when the pin is operated in output mode.

(2) PDn — Pull-Down Option Register

This register specifies whether an on-chip pull-down resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDn_15	PDn_14	PDn_13	PDn_12	PDn_11	PDn_10	PDn_9	PDn_8	PDn_7	PDn_6	PDn_5	PDn_4	PDn_3	PDn_2	PDn_1	PDn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.22 PDn Register Contents

Bit Position	Bit Name	Function
15 to 0	PDn_[15:0]	Specify whether an on-chip pull-down resistor is connected to the corresponding pin: 0: No on-chip pull-down resistor is connected. 1: On-chip pull-down resistor is connected.

NOTES

1. Do not set PUn.PUn_m = 1 and PDn.PDn_m = 1 to a single pin.
2. The on-chip pull-down resistor has no effect when the pin is operated in output mode.

2.1.4.5 Pin-Unit Register

(1) PCRn_m — Port Control Register

Each register of a port group can be accessed via this register and a PCRn_m register can set all functions of a single pin. For example, setting bit 6 of the PCRn_m register to 1 sets bit m of the PMCn register to 1 also.

Access: This register is readable/writable in 32-bit units.

Value after reset: See **Section 2.2.1.1, List of the C1H Port Registers** and **Section 2.2.2.1, List of the C1M Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PUCC	PDSC	—	—	—	—	PU	PD	PBDC	PIBC
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	P	—	—	—	PPR	—	PMC	PIPC	PM	—	PFCAE	PFCE	PFC
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 2.23 PCRn_m Register Contents

Bit Position	Bit Name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25	PUCC	PUCC PDSC Selects the output drive strength. 0 0: Low 0 1: High 1 0: Middle Settings other than the above are prohibited.
24	PDSC	
23 to 20	Reserved	When read, the value after reset is read. When writing, write the value after reset.
19	PU	Same function as bit m of the PUn register
18	PD	Same function as bit m of the PDn register
17	PBDC	Same function as bit m of the PBDCn register
16	PIBC	Same function as bit m of the PIBCn register
15 to 13	Reserved	When read, the value after reset is read. When writing, write the value after reset.
12	P	Same function as bit m of the Pn register
11 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	PPR	Same function as bit m of the PPRn register
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6	PMC	Same function as bit m of the PMCn register
5	PIPC	Same function as bit m of the PIPCn register
4	PM	Same function as bit m of the PMn register
3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2	PFCAE	Same function as bit m of the PFCAEn register
1	PFCE	Same function as bit m of the PFCEn register
0	PFC	Same function as bit m of the PFCn register

2.1.4.6 Example Port Settings

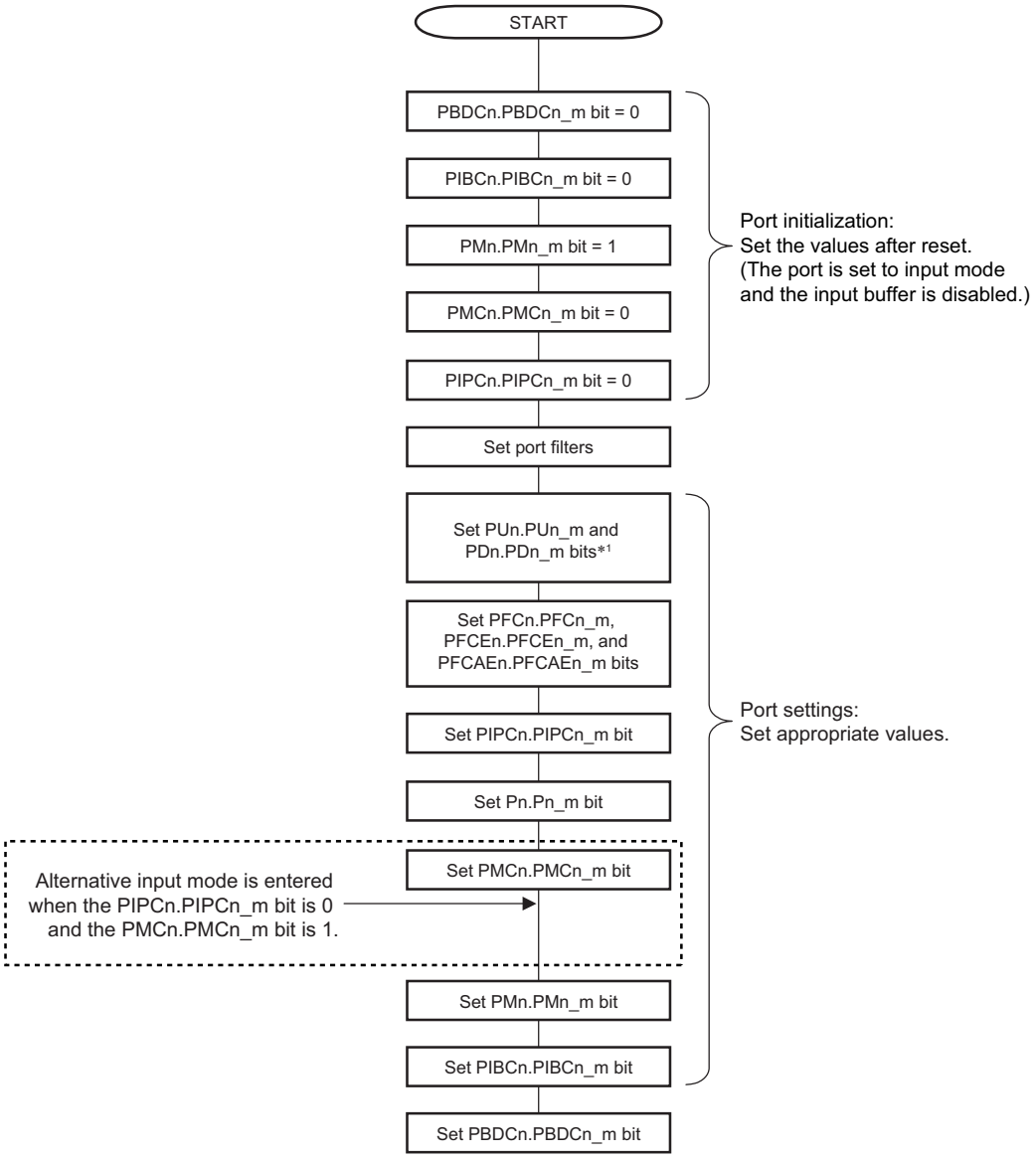
Examples of the port settings are shown in the flowchart below. For port filter setting of each flow chart, see **Section 2.3.2.4, Setting Procedures of Peripheral Function DNF**.

CAUTION

If a port is used in software I/O control alternative mode, it may be temporarily switched to alternative input mode in the following example. This may occur between when PMCn_m is set to 1 and when PMn_m is set to 0.

(1) Batch Setting

An example of specifying batch port settings is shown in the flowchart below.



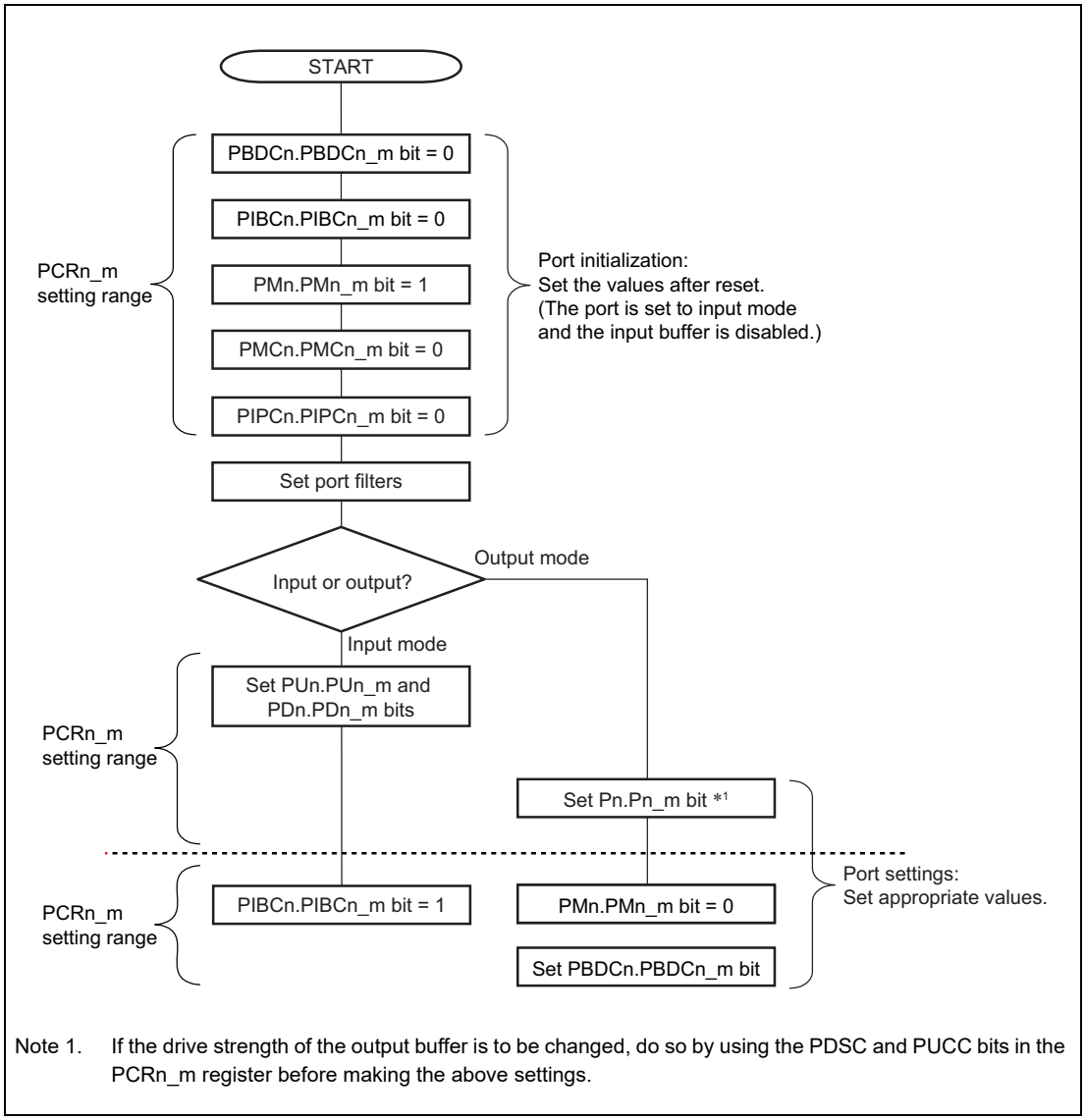
Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.3 Example of Port Settings (When Specified in Batch)

(2) Individual Settings

An example of specifying individual port settings is shown in the flowchart below.

Furthermore, setting of multiple bits within the range for setting of the PCRn_m register shown in the figure below is possible.



Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.4 Example of Port Settings (in Port Mode)

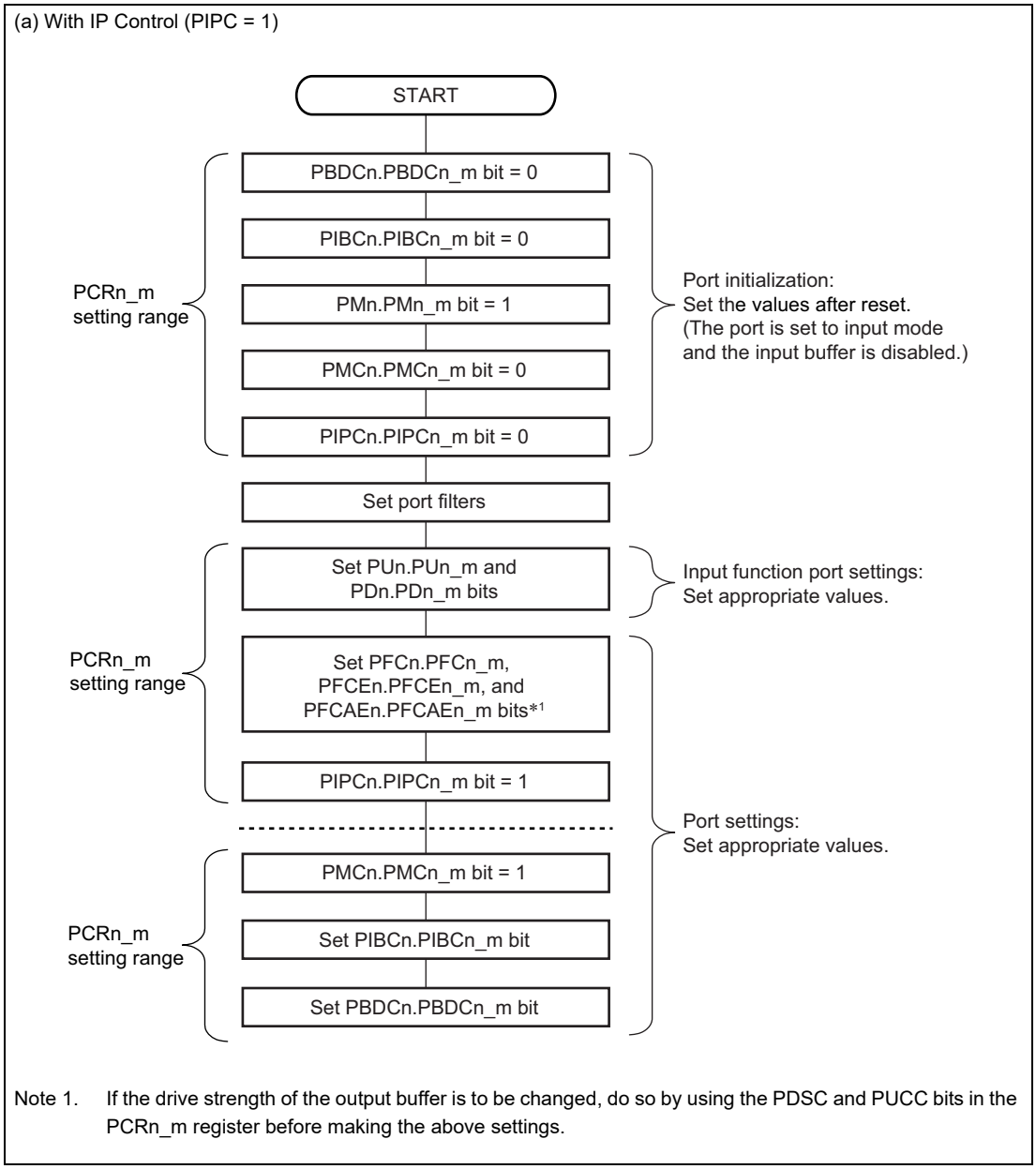
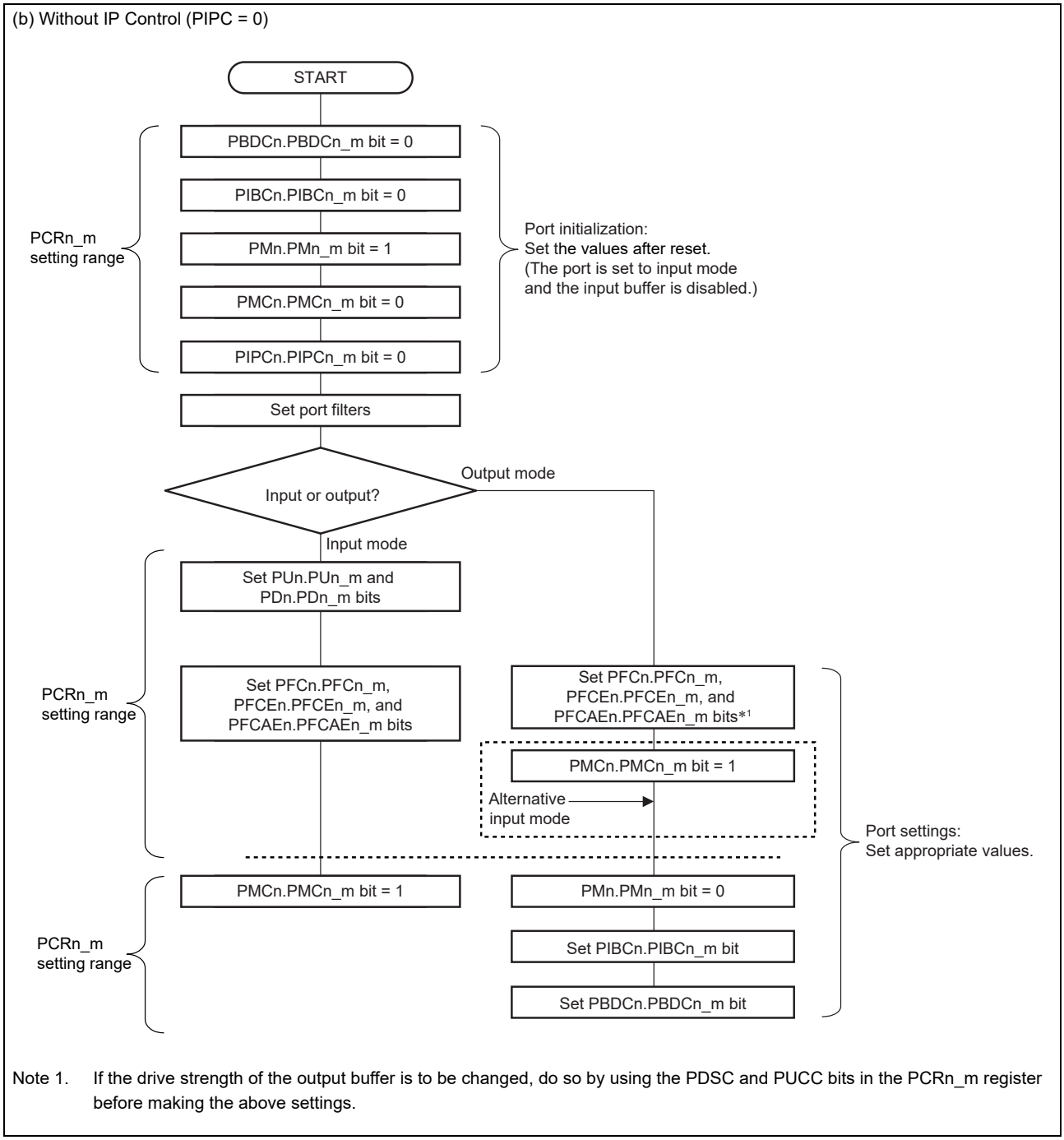


Figure 2.5 Example of Port Settings (in Alternative Mode) (1/2)



Note 1. If the drive strength of the output buffer is to be changed, do so by using the PDSC and PUCC bits in the PCRn_m register before making the above settings.

Figure 2.6 Example of Port Settings (in Alternative Mode) (2/2)

2.1.5 Functional Selection

2.1.5.1 Register Configuration in Use of the Alternative Function

When the pin alternative function is used, set PMCn_m = 1 and select the alternative numbers of PFCn_m, PFCEn_m, and PFCAEn_m. In several peripheral functions, a single alternative I/O function is allocated to multiple pins. However, such an alternative function should not be enabled in multiple pins at the same time.

2.1.5.2 Alternative Function to be Used in Direct I/O Control Alternative Mode

When the alternative functions described in **Table 2.24** are used, switch to direct I/O control alternative mode. When setting PIPCn_m = 1, the PMn_m value which has been set is ignored because the peripheral function enables or disables inputs and outputs of the buffer. If you are using an alternative function not listed in **Table 2.24**, set PIPCn_m = 0.

Table 2.24 List of the Pins which Require PIPC Register Setting

Category	Pin Name	I/O	Function
SCIn (n = 0 to 2)	SCInRXD	I	Receive data input
	SCInTXD	O	Transmit data output
	SCInSCK*1	I/O	Serial clock input/output
CSIHn (n = 0, 1)	CSIHnSO	O	Transmit data output
	CSIHnSC	I/O	Serial clock input/output
	CSIHnRYI/ CSIHnRYO	I/O	Handshake signal input/output
TSG3n (n = 0, 1)	TSG3nO1	O	Three-phase PWM output (Hi-Z control)
	TSG3nO2	O	
	TSG3nO3	O	
	TSG3nO4	O	
	TSG3nO5	O	
	TSG3nO6	O	
TAPAn (n = 0, 1)*2	TAPAn UN	O	Three-phase PWM output (Hi-Z control)
	TAPAn UP	O	
	TAPAn VN	O	
	TAPAn VP	O	
	TAPAn WN	O	
	TAPAn WP	O	

Note 1. RH850/C1M does not have SCI0SCK pin.

Note 2. RH850/C1M does not have TAPA1 pins.

2.1.5.3 Setting of the $\overline{\text{ERROROUT\_C}}$ Pin

When an error output function of the $\overline{\text{ERROROUT\_C}}$ pin is used, enable the alternative output level loopback function for fault diagnosis.

2.1.5.4 Selecting Function for JTAG Port

For the connection of multiple tools to the JTAG port, the interface is multiplexed on multiple sets of pins. The interface is selected by the combination of the settings of mode pins and an option byte. When user boot mode is selected as the operating mode, the interface is selected by the setting of option byte OPBT2.

When any other operating mode is selected, the setting of OPBT2 has no effect and the I/F that corresponds to the operating mode is selected. For details, refer to **Section 5.2, Operating Mode**.

2.2 Organization of Port Groups

2.2.1 C1H Port Function

2.2.1.1 List of the C1H Port Registers

Table 2.25 to **Table 2.32** show detailed bitmaps of control registers in each port group. In the bitmap field, "√" means an effective bit and "—" means a reserved one. Reserved bits are always read as value after resets. The write value also should be an value after reset.

Table 2.25 List of Registers in C1H Port Group 0

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	0	P0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR0	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR0	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM0	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE0	R/W	0000_H	16	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE0	R/W	0000_H	16	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT0	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR0	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR0	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC0	R/W	0000_H	16	√	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	
PORT		PU0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD0	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap														
						25	24	19	18	17	16	12	8	6	5	4	2	1	0	
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC	
PORT	0	PCR0_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR0_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR0_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	—	—	√	
PORT		PCR0_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	—	—	√	
PORT		PCR0_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	—	—	√	
PORT		PCR0_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	—	—	√	
PORT		PCR0_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	—	—	√	

Table 2.26 List of Registers in C1H Port Group 1

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	1	P1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR1	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM1	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC1	R/W	0000_H	16	—	√	—	√	—	√	—	√	√	√	√	√	√	√	√	√	
PORT		PFCE1	R/W	0000_H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
PORT		PFCAE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT1	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR1	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC1	R/W	0000_H	16	—	—	√	√	√	√	√	√	√	√	√	√	√	√	—	—	
PORT		PU1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															
						25	24	19	18	17	16	12	8	6	5	4	2	1	0		
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC		
PORT	1	PCR1_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR1_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR1_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR1_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—		
PORT		PCR1_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√		
PORT		PCR1_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—		
PORT		PCR1_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√		
PORT		PCR1_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—		
PORT		PCR1_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√		
PORT		PCR1_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	—		

Table 2.27 List of Registers in C1H Port Group 2

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	2	P2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR2	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM2	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT2	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR2	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	—	
PORT		PU2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															
						25	24	19	18	17	16	12	8	6	5	4	2	1	0		
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC		
PORT	2	PCR2_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR2_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR2_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR2_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR2_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR2_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR2_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR2_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		

Table 2.28 List of Registers in C1H Port Group 3

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	3	P3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR3	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM3	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	
PORT		PFC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	
PORT		PFCE3	R/W	0000H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√	√	
PORT		PFCAE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	
PORT		PNOT3	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR3	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	
PORT		PU3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	3	PCR3_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR3_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR3_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR3_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR3_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	—	—	√	—	—	—

Table 2.29 List of Registers in C1H Port Group 4

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	4	P4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR4	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM4	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	—	—	√	√	—	—	—	
PORT		PFCE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT4	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR4	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC4	R/W	0000_H	16	—	—	—	—	—	—	√	√	—	—	—	√	—	√	√	—	
PORT		PU4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															
						25	24	19	18	17	16	12	8	6	5	4	2	1	0		
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC		
PORT	4	PCR4_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—		
PORT		PCR4_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—		
PORT		PCR4_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—		
PORT		PCR4_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR4_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—		
PORT		PCR4_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—		
PORT		PCR4_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR4_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR4_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR4_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		

Table 2.30 List of Registers in C1H Port Group 5

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	5	P5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR5	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PM5	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT5	W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR5	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	—	√	√	√	√	√	√	
PORT		PU5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PD5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															
						25	24	19	18	17	16	12	8	6	5	4	2	1	0		
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC		
PORT	5	PCR5_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	—	√	√	√	√	√		
PORT		PCR5_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		

Table 2.31 List of Registers in C1H Port Group 6

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	6	P6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR6	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM6	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	—	
PORT		PFC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	—	
PORT		PFCE6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	—	—	√	√	√	—	
PORT		PFCAE6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	
PORT		PNOT6	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR6	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	—	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	—	Upper 16 bits
PORT		PIBC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC6	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
PORT		PU6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD6	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	6	PCR6_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	—	—	√	—	—	—
PORT		PCR6_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR6_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√
PORT		PCR6_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√
PORT		PCR6_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	—	√	√

Table 2.32 List of Registers in C1H Port Group 7

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	7	P7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PPR7	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PM7	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PMC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFCE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFCAE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PNOT7	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PMSR7	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PMCSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PIBC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PBDC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PIPC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	
PORT		PU7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PD7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap														
						25	24	19	18	17	16	12	8	6	5	4	2	1	0	
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC	
PORT	7	PCR7_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR7_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

2.2.1.2 Alternative Function List of the C1H Pins

Table 2.33 to **Table 2.40** show the list of alternative functions of each port pin. In the tables, "—" means a reserved bit which cannot be selected.

Table 2.33 List of Pin Alternative Functions in C1H Port Group 0

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P0_0	IO	TAUD0I0	TAUD0O0	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	—	—
P0_1	IO	TAUD0I1	TAUD0O1	TAUJ0I1	TAUJ0O1	—	—	—	—	—	—	—	—	—	—
P0_2	IO	TAUD0I2	TAUD0O2	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	—	—	—	—
P0_3	IO	TAUD0I3	TAUD0O3	TAUJ0I3	TAUJ0O3	—	—	—	—	CAN2RX	—	—	—	INTP0	—
P0_4	IO	TAUD0I4	TAUD0O4	—	—	—	—	—	—	—	CAN2TX	—	—	INTP1	—
P0_5	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	INTP2	—
P0_6	IO	TAUD0I6	TAUD0O6	TAUJ0I1	TAUJ0O1	ENCA0E0	—	—	—	—	—	—	—	INTP3	—
P0_7	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	ENCA0E1	—	—	—	—	—	—	—	INTP4	—
P0_8	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	ENCA0EC	—	—	—	—	—	—	—	INTP5	—
P0_9	IO	TAUD0I9	TAUD0O9	—	—	—	—	—	—	—	—	—	—	INTP6	—
P0_10	IO	TAUD0I10	TAUD0O10	—	TAPA1UP	—	—	—	—	—	—	—	—	INTP7	—
P0_11	IO	TAUD0I11	TAUD0O11	—	TAPA1UN	—	—	—	—	—	—	—	—	—	—
P0_12	IO	TAUD0I12	TAUD0O12	—	TAPA1VP	—	—	—	—	—	—	—	—	—	—
P0_13	IO	TAUD0I13	TAUD0O13	—	TAPA1VN	—	—	—	—	—	—	—	—	—	—
P0_14	IO	TAUD0I14	TAUD0O14	—	TAPA1WP	—	—	—	—	—	—	—	—	—	—
P0_15	IO	TAUD0I15	TAUD0O15	—	TAPA1WN	—	—	—	—	—	—	—	—	—	—

Table 2.34 List of Pin Alternative Functions in C1H Port Group 1

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P1_0	IO	TAUD1I0	TAUD1O0	—	TAUD1O1	ENCA0TIN0	—	—	—	—	TSG30O0	TAPA2ESO	—	—	—
P1_1	IO	TAUD1I1	TAUD1O1	—	—	ENCA0TIN1	—	—	—	—	TSG30O7	—	—	—	—
P1_2	IO	TAUD1I2	TAUD1O2	—	TAUD1O3	—	—	—	—	—	TSG30O1	—	—	—	—
P1_3	IO	TAUD1I3	TAUD1O3	—	—	—	—	—	—	—	TSG30O3	—	—	—	—
P1_4	IO	TAUD1I4	TAUD1O4	—	TAUD1O5	—	—	—	—	—	TSG30O5	—	—	—	—
P1_5	IO	TAUD1I5	TAUD1O5	—	—	—	—	—	—	—	TSG30O2	—	—	—	—
P1_6	IO	TAUD1I6	TAUD1O6	—	TAUD1O7	—	—	—	—	—	TSG30O4	—	—	—	—
P1_7	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	TSG30O6	—	—	—	—
P1_8	IO	TAUD1I8	TAUD1O8	—	TAUD1O9	—	—	—	—	—	TSG31O1	—	—	—	—
P1_9	IO	TAUD1I9	TAUD1O9	—	—	—	—	—	—	—	TSG31O3	—	—	—	—
P1_10	IO	TAUD1I10	TAUD1O10	—	TAUD1O11	—	—	—	—	—	TSG31O5	—	—	—	—
P1_11	IO	TAUD1I11	TAUD1O11	—	—	—	—	—	—	—	TSG31O2	—	—	—	—
P1_12	IO	TAUD1I12	TAUD1O12	—	TAUD1O13	—	—	—	—	—	TSG31O4	—	—	—	—
P1_13	IO	TAUD1I13	TAUD1O13	—	—	—	—	—	—	—	TSG31O6	—	—	—	—
P1_14	IO	TAUD1I14	TAUD1O14	—	TAUD1O15	—	—	—	—	—	TSG31O0	TAPA3ESO	—	—	—
P1_15	IO	TAUD1I15	TAUD1O15	—	—	—	—	—	—	—	TSG31O7	—	—	—	—

Table 2.35 List of Pin Alternative Functions in C1H Port Group 2

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P2_0	IO	TAUD0I3	TAUD0O3	—	—	—	—	—	—	—	—	—	—	INTP0	—
P2_1	IO	TAUD0I4	TAUD0O4	—	—	—	—	—	—	—	—	—	—	INTP1	—
P2_2	IO	TAUD0I10	TAUD0O10	—	TAPA0UP	—	—	—	—	—	—	—	—	INTP2	—
P2_3	IO	TAUD0I11	TAUD0O11	—	TAPA0UN	—	—	—	—	—	—	—	—	INTP3	—
P2_4	IO	TAUD0I12	TAUD0O12	—	TAPA0VP	—	—	—	—	—	—	—	—	INTP4	—
P2_5	IO	TAUD0I13	TAUD0O13	—	TAPA0VN	—	—	—	—	—	—	—	—	INTP5	—
P2_6	IO	TAUD0I14	TAUD0O14	—	TAPA0WP	—	—	—	—	—	—	—	—	INTP6	—
P2_7	IO	TAUD0I15	TAUD0O15	—	TAPA0WN	—	—	—	—	—	—	—	—	INTP7	—

Table 2.36 List of Pin Alternative Functions in C1H Port Group 3

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P3_0	IO	TAUD0I5	TAUD0O5	—	—	—	—	—	—	—	—	—	—	—	—
P3_1	IO	TAUD0I7	TAUD0O7	—	—	—	—	$\overline{\text{ADCC0TRG}}$	—	—	—	—	—	—	—
P3_2	IO	TAUD1I5	TAUD1O5	—	—	—	—	$\overline{\text{ADCC1TRG}}$	—	—	—	—	—	—	—
P3_3	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	—	TAPA2ESO	—	—	—
P3_4	IO	—	—	—	—	—	—	$\overline{\text{ADCC0TRG}}$	—	—	—	TAPA3ESO	—	—	—
P3_5	IO	—	—	—	—	—	—	—	—	—	—	TAPA0ESO	—	—	—
P3_6	IO	—	—	—	—	—	—	—	—	—	—	TAPA1ESO	—	—	—
P3_7	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—

Table 2.37 List of Pin Alternative Functions in C1H Port Group 4

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P4_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SI	—
P4_1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SO
P4_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SC	CSIH1SC
P4_3	IO	—	—	—	—	—	—	—	—	CAN3RX	—	$\overline{\text{CSIH0SSI}}$	—	—	CSIH1CSS0
P4_4	IO	—	—	—	—	—	—	—	—	—	CAN3TX	CSIH0RYI	CSIH0RYO	—	CSIH1CSS1
P4_5	IO	—	—	—	—	—	—	—	—	CAN0RX	—	—	—	—	CSIH1CSS2
P4_6	IO	—	—	—	—	—	—	—	—	—	CAN0TX	—	—	—	CSIH1CSS3
P4_7	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SI	—	$\overline{\text{CSIH1SSI}}$	—
P4_8	IO	—	—	—	—	—	—	—	—	—	—	—	CSIH0SO	CSIH1RYI	CSIH1RYO
P4_9	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SC	CSIH0SC	—	—
P4_10	IO	—	—	—	—	—	TPBA0O	—	—	—	—	—	CSIH0CSS0	—	—
P4_11	IO	—	—	—	—	ENCA1E0	—	—	—	—	—	—	CSIH0CSS1	—	—
P4_12	IO	—	—	—	—	ENCA1E1	—	—	—	—	—	—	CSIH0CSS2	—	—
P4_13	IO	—	—	—	—	ENCA1EC	—	—	—	—	—	—	CSIH0CSS3	—	—
P4_14	IO	—	—	—	—	—	—	—	—	CAN1RX	—	—	—	—	—
P4_15	IO	—	—	—	—	—	—	—	—	—	CAN1TX	—	$\overline{\text{ERROROUT\_IC}}$	—	—

Table 2.38 List of Pin Alternative Functions in C1H Port Group 5

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P5_0	IO	—	—	—	—	—	—	RLIN22RX	—	—	—	SCI0RXD	—	—	—
P5_1	IO	—	—	—	—	—	—	—	RLIN22TX	—	—	—	SCI0TXD	—	—
P5_2	IO	—	—	—	—	—	—	RLIN21RX	—	—	—	SCI0SCK	SCI0SCK	—	—
P5_3	IO	—	—	—	—	—	—	—	RLIN21TX	—	—	SCI1RXD	—	—	—
P5_4	IO	—	—	—	—	—	—	RLIN20RX	—	—	—	—	SCI1TXD	—	—
P5_5	IO	—	—	—	—	—	—	—	RLIN20TX	—	—	SCI1SCK	SCI1SCK	—	$\overline{\text{ERROROUT\_IC}}$
P5_6	IO	—	—	—	—	—	TPBA1O	—	—	—	—	TAPA0ESO	—	—	—
P5_7	IO	—	—	—	—	—	—	—	—	—	—	SCI2SCK	SCI2SCK	—	—
P5_8	IO	—	—	—	—	—	—	—	—	—	—	SCI2RXD	—	—	—
P5_9	IO	—	—	—	—	—	—	—	—	—	—	—	SCI2TXD	—	—

Table 2.39 List of Pin Alternative Functions in C1H Port Group 6

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P6_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_3	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P6_4	IO	—	—	—	—	—	—	—	—	—	—	TAPA0ESO	—	—	—
P6_5	IO	—	—	—	—	—	—	—	—	—	—	TAPA1ESO	—	—	—
P6_6	IO	TAUD1I5	TAUD1O5	—	—	—	—	$\overline{\text{ADCC1TRG}}$	—	—	—	—	—	—	—
P6_7	IO	TAUD1I7	TAUD1O7	—	—	—	—	$\overline{\text{ADCC0TRG}}$	—	—	—	—	—	—	—

Table 2.40 List of Pin Alternative Functions in C1H Port Group 7

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P7_0	IO	—	—	—	—	ENCA1TIN0	—	—	—	—	—	—	—	—	—
P7_1	IO	—	—	—	—	ENCA1TIN1	—	—	—	—	—	—	—	—	—
P7_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P7_3	IO	—	—	—	—	—	—	—	—	—	—	SCI0RXD	—	—	—
P7_4	IO	—	—	—	—	—	—	—	—	—	—	—	SCI0TXD	—	—

2.2.2 C1M Port Function

2.2.2.1 List of the C1M Port Registers

Table 2.41 to **Table 2.47** show detailed bitmaps of control registers in each port group. In the bitmap field, "√" means an effective bit and "—" means a reserved one. Reserved bits are always read as value after resets. The write value also should be an value after reset.

Table 2.41 List of Registers in C1M Port Group 0

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	0	P0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PSR0	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR0	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PM0	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMC0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFC0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCE0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT0	W	0000H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR0	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR0	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC0	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
PORT		PU0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
PORT		PD0	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																					
						25	24	19	18	17	16	12	8	6	5	4	2	1	0								
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC								
PORT	0	PCR0_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								
PORT		PCR0_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√								

Table 2.42 List of Registers in C1M Port Group 1

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	1	P1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR1	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM1	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC1	R/W	0000_H	16	—	√	—	√	—	√	—	√	√	√	√	√	√	√	√	√	
PORT		PFCE1	R/W	0000_H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
PORT		PFCAE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT1	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR1	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC1	R/W	0000_H	16	—	—	√	√	√	√	√	√	√	√	√	√	√	√	—	—	
PORT		PU1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap														
						25	24	19	18	17	16	12	8	6	5	4	2	1	0	
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC	
PORT	1	PCR1_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR1_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR1_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR1_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—	
PORT		PCR1_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	
PORT		PCR1_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—	
PORT		PCR1_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	
PORT		PCR1_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	—	
PORT		PCR1_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√	
PORT		PCR1_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	—	

Table 2.43 List of Registers in C1M Port Group 2

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	2	P2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR2	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PM2	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PFCAE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PNOT2	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PMSR2	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PBDC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PIPC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	—	—	
PORT		PU2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
PORT		PD2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	2	PCR2_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR2_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR2_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.44 List of Registers in C1M Port Group 3

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	3	P3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Upper 16 bits
PORT		PPR3	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PM3	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PMC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PFC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PFCE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	
PORT		PFCAE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PNOT3	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PMSR3	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Upper 16 bits
PORT		PMCSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Upper 16 bits
PORT		PIBC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PBDC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PIPC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	
PORT		PU3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
PORT		PD3	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	3	PCR3_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR3_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	—	√

Table 2.45 List of Registers in C1M Port Group 4

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
PORT	4	P4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PPR4	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PM4	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	—	—	√	√	—	—	—	
PORT		PFCE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PFCAE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PNOT4	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PMSR4	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PMCSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
						√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
PORT		PIBC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PBDC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PIPC4	R/W	0000_H	16	—	—	—	—	—	—	√	√	—	—	—	√	—	√	√	—	
PORT		PU4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PD4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
PORT	4	PCR4_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—
PORT		PCR4_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—
PORT		PCR4_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—
PORT		PCR4_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	—
PORT		PCR4_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√
PORT		PCR4_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√
PORT		PCR4_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√

Table 2.46 List of Registers in C1M Port Group 5

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	5	P5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Upper 16 bits
PORT		PPR5	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PM5	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PMC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PFC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PFCE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PFCAE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PNOT5	W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PMSR5	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Upper 16 bits
PORT		PMCSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Lower 16 bits
						—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	Upper 16 bits
PORT		PIBC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PBDC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PIPC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	—	√	√	√	—	—	—	
PORT		PU5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	
PORT		PD5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	—	—	—	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap															
						25	24	19	18	17	16	12	8	6	5	4	2	1	0		
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC		
PORT	5	PCR5_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√		
PORT		PCR5_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		
PORT		PCR5_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√		

Table 2.47 List of Registers in C1M Port Group 7

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap																Remarks
						15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
PORT	7	P7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PPR7	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PM7	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PMC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFCE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PFCAE7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PNOT7	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PMSR7	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PMCSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
						—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
PORT		PIBC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PBDC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PIPC7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	
PORT		PU7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
PORT		PD7	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	

Module Name	Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap														
						25	24	19	18	17	16	12	8	6	5	4	2	1	0	
						PUCC	PDSC	PU	PD	PBDC	PIBC	P	PPR	PMC	PIPC	PM	PFCAE	PFCE	PFC	
PORT	7	PCR7_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	√	√	√	√	
PORT		PCR7_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
PORT		PCR7_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

2.2.2.2 Alternative Function List of the C1M pins

Table 2.48 to **Table 2.54** show the list of alternative functions of each port pin. In the tables, "—" means a reserved bit which cannot be selected.

Table 2.48 List of Pin Alternative Functions in C1M Port Group 0

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P0_0	IO	TAUD0I0	TAUD0O0	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	—	—
P0_1	IO	TAUD0I1	TAUD0O1	TAUJ0I1	TAUJ0O1	—	—	—	—	—	—	—	—	—	—
P0_2	IO	TAUD0I2	TAUD0O2	TAUJ0I2	TAUJ0O2	—	—	—	—	—	—	—	—	—	—
P0_3	IO	TAUD0I3	TAUD0O3	TAUJ0I3	TAUJ0O3	—	—	—	—	CAN2RX	—	—	—	INTP0	—
P0_4	IO	TAUD0I4	TAUD0O4	—	—	—	—	—	—	—	CAN2TX	—	—	INTP1	—
P0_5	IO	TAUD0I5	TAUD0O5	TAUJ0I0	TAUJ0O0	—	—	—	—	—	—	—	—	INTP2	—
P0_6	IO	TAUD0I6	TAUD0O6	TAUJ0I1	TAUJ0O1	ENCA0E0	—	—	—	—	—	—	—	INTP3	—
P0_7	IO	TAUD0I7	TAUD0O7	TAUJ0I2	TAUJ0O2	ENCA0E1	—	—	—	—	—	—	—	INTP4	—
P0_8	IO	TAUD0I8	TAUD0O8	TAUJ0I3	TAUJ0O3	ENCA0EC	—	—	—	—	—	—	—	INTP5	—
P0_9	IO	TAUD0I9	TAUD0O9	—	—	—	—	—	—	—	—	—	—	INTP6	—

Table 2.49 List of Pin Alternative Functions in C1M Port Group 1

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P1_0	IO	TAUD1I0	TAUD1O0	—	TAUD1O1	ENCA0TIN0	—	—	—	—	TSG30O0	TAPA2ESO	—	—	—
P1_1	IO	TAUD1I1	TAUD1O1	—	—	ENCA0TIN1	—	—	—	—	TSG30O7	—	—	—	—
P1_2	IO	TAUD1I2	TAUD1O2	—	TAUD1O3	—	—	—	—	—	TSG30O1	—	—	—	—
P1_3	IO	TAUD1I3	TAUD1O3	—	—	—	—	—	—	—	TSG30O3	—	—	—	—
P1_4	IO	TAUD1I4	TAUD1O4	—	TAUD1O5	—	—	—	—	—	TSG30O5	—	—	—	—
P1_5	IO	TAUD1I5	TAUD1O5	—	—	—	—	—	—	—	TSG30O2	—	—	—	—
P1_6	IO	TAUD1I6	TAUD1O6	—	TAUD1O7	—	—	—	—	—	TSG30O4	—	—	—	—
P1_7	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	TSG30O6	—	—	—	—
P1_8	IO	TAUD1I8	TAUD1O8	—	TAUD1O9	—	—	—	—	—	TSG31O1	—	—	—	—
P1_9	IO	TAUD1I9	TAUD1O9	—	—	—	—	—	—	—	TSG31O3	—	—	—	—
P1_10	IO	TAUD1I10	TAUD1O10	—	TAUD1O11	—	—	—	—	—	TSG31O5	—	—	—	—
P1_11	IO	TAUD1I11	TAUD1O11	—	—	—	—	—	—	—	TSG31O2	—	—	—	—
P1_12	IO	TAUD1I12	TAUD1O12	—	TAUD1O13	—	—	—	—	—	TSG31O4	—	—	—	—
P1_13	IO	TAUD1I13	TAUD1O13	—	—	—	—	—	—	—	TSG31O6	—	—	—	—
P1_14	IO	TAUD1I14	TAUD1O14	—	TAUD1O15	—	—	—	—	—	TSG31O0	TAPA3ESO	—	—	—
P1_15	IO	TAUD1I15	TAUD1O15	—	—	—	—	—	—	—	TSG31O7	—	—	—	—

Table 2.50 List of Pin Alternative Functions in C1M Port Group 2

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P2_0	IO	TAUD0I3	TAUD0O3	—	—	—	—	—	—	—	—	—	—	INTP0	—
P2_1	IO	TAUD0I4	TAUD0O4	—	—	—	—	—	—	—	—	—	—	INTP1	—
P2_2	IO	TAUD0I10	TAUD0O10	—	TAPA0UP	—	—	—	—	—	—	—	—	INTP2	—
P2_3	IO	TAUD0I11	TAUD0O11	—	TAPA0UN	—	—	—	—	—	—	—	—	INTP3	—
P2_4	IO	TAUD0I12	TAUD0O12	—	TAPA0VP	—	—	—	—	—	—	—	—	INTP4	—
P2_5	IO	TAUD0I13	TAUD0O13	—	TAPA0VN	—	—	—	—	—	—	—	—	INTP5	—
P2_6	IO	TAUD0I14	TAUD0O14	—	TAPA0WP	—	—	—	—	—	—	—	—	INTP6	—
P2_7	IO	TAUD0I15	TAUD0O15	—	TAPA0WN	—	—	—	—	—	—	—	—	INTP7	—

Table 2.51 List of Pin Alternative Functions in C1M Port Group 3

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P3_0	IO	TAUD0I5	TAUD0O5	—	—	—	—	—	—	—	—	—	—	—	—
P3_1	IO	TAUD0I7	TAUD0O7	—	—	—	—	$\overline{\text{ADCC0TRG}}$	—	—	—	—	—	—	—
P3_2	IO	TAUD1I5	TAUD1O5	—	—	—	—	$\overline{\text{ADCC1TRG}}$	—	—	—	—	—	—	—
P3_3	IO	TAUD1I7	TAUD1O7	—	—	—	—	—	—	—	—	TAPA2ESO	—	—	—

Table 2.52 List of Pin Alternative Functions in C1M Port Group 4

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P4_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SI	—
P4_1	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SO
P4_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	CSIH1SC	CSIH1SC
P4_3	IO	—	—	—	—	—	—	—	—	CAN3RX	—	$\overline{\text{CSIH0SSI}}$	—	—	CSIH1CSS0
P4_4	IO	—	—	—	—	—	—	—	—	—	CAN3TX	CSIH0RYI	CSIH0RYO	—	CSIH1CSS1
P4_5	IO	—	—	—	—	—	—	—	—	CAN0RX	—	—	—	—	CSIH1CSS2
P4_6	IO	—	—	—	—	—	—	—	—	—	CAN0TX	—	—	—	CSIH1CSS3
P4_7	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SI	—	$\overline{\text{CSIH1SSI}}$	—
P4_8	IO	—	—	—	—	—	—	—	—	—	—	—	CSIH0SO	CSIH1RYI	CSIH1RYO
P4_9	IO	—	—	—	—	—	—	—	—	—	—	CSIH0SC	CSIH0SC	—	—
P4_10	IO	—	—	—	—	—	TPBA0O	—	—	—	—	—	CSIH0CSS0	—	—
P4_11	IO	—	—	—	—	ENCA1E0	—	—	—	—	—	—	CSIH0CSS1	—	—
P4_12	IO	—	—	—	—	ENCA1E1	—	—	—	—	—	—	CSIH0CSS2	—	—
P4_13	IO	—	—	—	—	ENCA1EC	—	—	—	—	—	—	CSIH0CSS3	—	—
P4_14	IO	—	—	—	—	—	—	—	—	CAN1RX	—	—	—	—	—
P4_15	IO	—	—	—	—	—	—	—	—	—	CAN1TX	—	$\overline{\text{ERROROUT\_C}}$	—	—

Table 2.53 List of Pin Alternative Functions in C1M Port Group 5

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P5_3	IO	—	—	—	—	—	—	—	—	—	—	SCI1RXD	—	—	—
P5_4	IO	—	—	—	—	—	—	—	—	—	—	—	SCI1TXD	—	—
P5_5	IO	—	—	—	—	—	—	—	—	—	—	SCI1SCK	SCI1SCK	—	$\overline{\text{ERROROUT\_C}}$
P5_6	IO	—	—	—	—	—	—	—	—	—	—	TAPA0ESO	—	—	—
P5_7	IO	—	—	—	—	—	—	—	—	—	—	SCI2SCK	SCI2SCK	—	—
P5_8	IO	—	—	—	—	—	—	—	—	—	—	SCI2RXD	—	—	—
P5_9	IO	—	—	—	—	—	—	—	—	—	—	—	SCI2TXD	—	—

Table 2.54 List of Pin Alternative Functions in C1M Port Group 7

Port	General-Purpose I/O Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7
P7_0	IO	—	—	—	—	ENCA1TIN0	—	—	—	—	—	—	—	—	—
P7_1	IO	—	—	—	—	ENCA1TIN1	—	—	—	—	—	—	—	—	—
P7_2	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P7_3	IO	—	—	—	—	—	—	—	—	—	—	SCI0RXD	—	—	—
P7_4	IO	—	—	—	—	—	—	—	—	—	—	—	SCI0TXD	—	—

2.3 DNF

Digital Noise Filter (DNF) eliminates digital noise from external input signals. This product incorporates peripheral function DNF.

2.3.1 Example of Noise Elimination

Figure 2.7 shows an example of noise elimination in peripheral function DNF. In this example, the sampling clock, the sampling count, and the current output level are set to 1/2 of the DNF input clock, two (twice), and low, respectively. "○" in the figure means that high level is detected.

In input examples 1, 2, and 3, the output level changes from low to high because the same level is detected twice in a row through sampling. In input examples 4, 5, and 6, on the other hand, the same level is not detected twice consecutively. Therefore, these inputs are regarded as noise and the input signal state is eliminated.

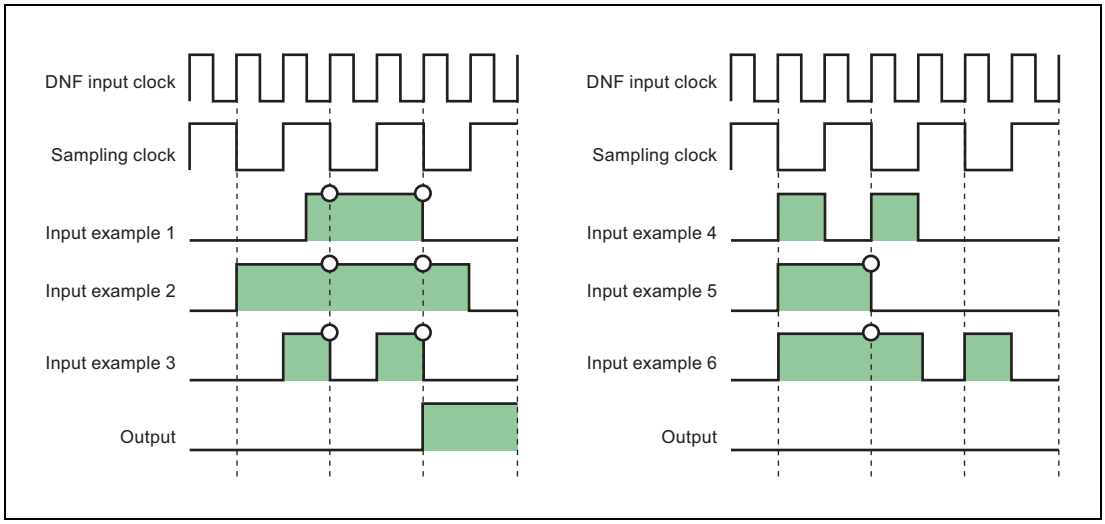


Figure 2.7 Timing Chart of Digital Noise Elimination

2.3.2 Peripheral Function DNF

2.3.2.1 Overview of Peripheral Function DNF

This DNF eliminates noise from the input function pins for the peripheral function.

Peripheral function DNF has the following functions:

- Eliminates digital noise from input signals and outputs noiseless signals.
- Selects whether to output signals from which digital noise is eliminated or signals containing digital noise.
- Selects the digital noise elimination width from 2, 3, 4, and 5 counts of the sampling clock.
- Selects five types of sampling frequency shown below:
  1/1, 1/2, 1/4, 1/8, and 1/16 of the DNF input clock.
- The conditions for noise elimination can be set by each channel via the registers.
- The DNF input clock of DNF group number 0 is a low-speed peripheral clock.
- The DNF input clock of DNF group number 1, 2, 3, 4, and 5 is an unmodulated high-speed peripheral clock.

2.3.2.2 Details of the Control Registers

Base addresses of peripheral function DNF are listed in the following table.
Register addresses of peripheral function DNF are given as offsets from the base addresses in general.

Table 2.55 Register Base Address

Base Address Name	Base address
<DNF0_base>	FFC3 0000_H
<DNF1_base>	FFC3 0400_H
<DNF2_base>	FFC3 0800_H
<DNF3_base>	FFC3 $0C00_H$
<DNF4_base>	FFC3 1000_H
<DNF5_base>	FFC3 1400_H

2.3.2.3 DNFP01nCTLm — Digital Noise Elimination Control Register

This register sets conditions for noise elimination of channel number m in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 4_H × m (m: channel number)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NFEN	SLST[1:0]		—	—	PRS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Table 2.56 DNFP01nCTLm Register Contents

Bit Position	Bit Name	Function
7	NFEN	Enables/disables digital noise elimination. 0: Digital noise is not eliminated. 1: Digital noise is eliminated.
6, 5	SLST[1:0]	Specifies the sampling count for digital noise elimination. 00: Sampling count = 2 01: Sampling count = 3 10: Sampling count = 4 11: Sampling count = 5
4, 3	Reserved	When read, 0 is read. When writing, write 0.
2 to 0	PRS[2:0]	Specifies the sampling clock division ratio for digital noise elimination. 000: DNF input clock/1 001: DNF input clock/2 010: DNF input clock/4 011: DNF input clock/8 100: DNF input clock/16 Other than the above: Setting is prohibited.

2.3.2.4 Setting Procedures of Peripheral Function DNF

The following shows the procedures to set peripheral function DNF. If the input level to a pin varies during (4) and (5), an unexpected signal may be input to the peripheral function. Therefore, the corresponding flag at the peripheral function side or the like should be cleared in (6).

(1) Set the PRS2 to PRS0 bits (bits 2 to 0) and the SLST1 and SLST0 bits (bits 6 and 5) in the DNFP01nCTLm register.

(2) Set the NFEN bit (bit 7) in the DNFP01nCTLm register. This setting can be made at the same time as (1).

(3) Set a port register to select an alternative function.

(4) Wait for the following time: Sampling clock period × Number of samples + DNF input clock period × 2.

(5) Enable an operation of a peripheral function in the destination to which DNF is connected.

(6) Clear the flag of peripheral function, and the like.

To change the setting while the DNF is in operation, clear the NFEN bit (bit 7) in the DNFP01nCTLm register first, and then execute the procedure in steps (1) to (6) above to remake the settings.

2.3.2.5 Peripheral Function Pin for DNF Insertion

The signals shown in the table below are the targets of DNF insertion.

Table 2.57 DNF Insertion Targets (1/2)

DNF Group Number n	DNF Channel Number m	Target Pins for DNF Insertion		
		Pin Name	Pin Function	Peripheral IP
0	0	INTP0	External interrupt input	INTC
	1	INTP1		
	2	INTP2		
	3	INTP3		
	4	INTP4		
	5	INTP5		
	6	INTP6		
	7	INTP7		
	8	TAPA0ESO	Emergency Hi-Z request input (for TAUD0 PWM)	TAPA (via PIC1A)
	9	TAPA1ESO*1	Emergency Hi-Z request input (for TAUD1 PWM)	
	10	TAPA2ESO	Emergency Hi-Z request input (for TSG30 PWM)	
	11	TAPA3ESO	Emergency Hi-Z request input (for TSG31 PWM)	
1	0	$\overline{\text{ADCC0TRG}}$	SAR-AD conversion startup trigger input	SAR-AD0
	1	$\overline{\text{ADCC1TRG}}$		SAR-AD1
2	0	ENCA0TIN0	Capture trigger input	ENCA0
	1	ENCA0TIN1		
	2	ENCA0E0	Encoder input	
	3	ENCA0E1		
	4	ENCA0EC		
	5	ENCA1TIN0	Capture trigger input	ENCA1
	6	ENCA1TIN1		
	7	ENCA1E0	Encoder input	
	8	ENCA1E1		
	9	ENCA1EC		

Table 2.57 DNF Insertion Targets (2/2)

DNF Group Number n	DNF Channel Number m	Target Pins for DNF Insertion		
		Pin Name	Pin Function	Peripheral IP
3	0	TAUD0I0	TAUD0 channel input	TAUD0
	1	TAUD0I1		
	2	TAUD0I2		
	3	TAUD0I3		
	4	TAUD0I4		
	5	TAUD0I5		
	6	TAUD0I6		
	7	TAUD0I7		
	8	TAUD0I8		
	9	TAUD0I9		
	10	TAUD0I10		
	11	TAUD0I11		
	12	TAUD0I12		
	13	TAUD0I13		
	14	TAUD0I14		
	15	TAUD0I15		
4	0	TAUD1I0	TAUD1 channel input	TAUD1
	1	TAUD1I1		
	2	TAUD1I2		
	3	TAUD1I3		
	4	TAUD1I4		
	5	TAUD1I5		
	6	TAUD1I6		
	7	TAUD1I7		
	8	TAUD1I8		
	9	TAUD1I9		
	10	TAUD1I10		
	11	TAUD1I11		
	12	TAUD1I12		
	13	TAUD1I13		
	14	TAUD1I14		
	15	TAUD1I15		
5	0	TAUJ0I0	TAUJ0 channel input	TAUJ0
	1	TAUJ0I1		
	2	TAUJ0I2		
	3	TAUJ0I3		

Note 1. RH850/C1M does not have TAPA1ESO pin.

2.4 Pin Description

2.4.1 Overview

This subsection describes pin functions, external pin lists, and external pin states at a reset and in other each operating status.

2.4.2 List of Pin Functions

Table 2.58 and **Table 2.59** list functions of each pin.

Table 2.58 C1H Pin Function (1/3)

Pin Name	I/O	Function
AnVREF (n = 0, 1)	—	ADCCn voltage supply and reference voltage
AnVSS (n = 0, 1)	—	ADCCn ground
$\overline{\text{ADCCnTRG}}$ (n = 0, 1)	I	ADCCn trigger
ADCC0Ipq (p = 0 to 3, q = 0 to 3)	I	ADCC0 input channel pq
ADCC1Ipq (p = 0 to 6, q = 0 to 2)	I	ADCC1 input channel pq
AUDATAm (m = 0 to 3)	IO	AUDR command / address / data / flag m
AUDCK	I	AUDR clock
$\overline{\text{AUDRST}}$	I	AUDR reset
$\overline{\text{AUDSYNC}}$	I	AUDR timing control
AnVCC (n = 0, 1)	—	ADCCn voltage supply
CANmRX (m = 0 to 3)	I	CANm receive data input
CANmTX (m = 0 to 3)	O	CANm transmit data output
CSIHnCSS0 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 0
CSIHnCSS1 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 1
CSIHnCSS2 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 2
CSIHnCSS3 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 3
CSIHnRYI (n = 0, 1)	I	CSIHn ready (1) / busy (0) input signal
CSIHnRYO (n = 0, 1)	O	CSIHn ready (1) / busy (0) output signal
CSIHnSC (n = 0, 1)	IO	CSIHn serial clock signal
CSIHnSI (n = 0, 1)	I	CSIHn serial data input
CSIHnSO (n = 0, 1)	O	CSIHn serial data output
$\overline{\text{CSIHnSSI}}$ (n = 0, 1)	I	CSIHn slave select input signal
$\overline{\text{DCURDY}}$	O	Debug ready
DCUTCK	I	Debug clock
DCUTDI	I	Debug data input
DCUTDO	O	Debug data output
DCUTMS	I	Debug mode select
$\overline{\text{DCUTRST}}$	I	Debug reset
FLSCI3TX (FPDT)	O	Transmit data output
FLSCI3RX (FPDR)	I	Receive data input
FLSCI3SCK (FPCK)	I	Serial clock input
ENCAnE0 (n = 0, 1)	I	ENCAn encoder input (count pulse 0)
ENCAnE1 (n = 0, 1)	I	ENCAn encoder input (count pulse 1)
ENCAnTINm (n = 0, 1, m = 0, 1)	I	ENCAn capture trigger input nm
ENCAnEC (n = 0, 1)	I	ENCAn encoder input (clear pulse)

Table 2.58 C1H Pin Function (2/3)

Pin Name	I/O	Function
EPTVOUT	O	EPT control for VDD
EVCC	—	Port buffer voltage supply
$\overline{\text{ERROROUT\_M}}$	O	ECM error output (MAIN)
$\overline{\text{ERROROUT\_C}}$	O	ECM error output (Checker)
FLMODE	I	Operating mode select pin
MD0	I	Operating mode select pin 0
MD1	I	Operating mode select pin 1
INTPm (m = 0 to 7)	I	External interrupt input m
LPDCLK	I	LPD clock input (4-pin mode)
LPDCLKOUT	O	LPD clock output (4-pin mode)
LPDI	I	LPD data input (4-pin mode)
LPDO	O	LPD data output (4-pin mode)
$\overline{\text{LPDRST}}$	I	LPD Reset (4-pin mode)
NMI	I	External non-maskable interrupt
P0_m (m = 0 to 15)	IO	Port 0_m
P1_m (m = 0 to 15)	IO	Port 1_m
P2_m (m = 0 to 7)	IO	Port 2_m
P3_m (m = 0 to 7)	IO	Port 3_m
P4_m (m = 0 to 15)	IO	Port 4_m
P5_m (m = 0 to 9)	IO	Port 5_m
P6_m (m = 0 to 7)	IO	Port 6_m
P7_m (m = 0 to 4)	IO	Port 7_m
PLLVCC	—	Voltage supply for PLL
PLLVSS	—	Ground for PLL
SYSVCC	—	Voltage supply for system
VCC	—	Voltage supply for oscillator and Flash
VDD	—	Voltage regulators voltage supply
VSS	—	Ground
RVCC	—	Voltage supply for RDC
RVSS	—	Ground for RDC
$\overline{\text{RESET}}$	I	External reset input
RDC2nCOM (n = 0, 1)	IO	Excitation common signal input/output
RDC2nCOSMNT (n = 0, 1)	O	COS-side monitoring signal output
RDC2nRSO (n = 0, 1)	IO	Excitation signal input/output
RDC2nS1 (n = 0, 1)	I	Resolver signal input
RDC2nS2 (n = 0, 1)	I	Resolver signal input
RDC2nS3 (n = 0, 1)	I	Resolver signal input
RDC2nS4 (n = 0, 1)	I	Resolver signal input
RDC2nSINMNT (n = 0, 1)	O	SIN-side monitoring signal output
RLIN2mRX (m = 0 to 2)	I	RLIN2m receive data input
RLIN2mTX (m = 0 to 2)	O	RLIN2m transmit data output
SCInRXD (n = 0 to 2)	I	SCIn receive data
SCInSCK (n = 0 to 2)	IO	SCIn clock
SCInTXD (n = 0 to 2)	O	SCIn transmit data

Table 2.58 C1H Pin Function (3/3)

Pin Name	I/O	Function
TAPAnESO (n = 0 to 3)	I	Hi-Z control
TAPAnUN (n = 0, 1)	O	Motor control output U phase (negative)
TAPAnUP (n = 0, 1)	O	Motor control output U phase (positive)
TAPAnVN (n = 0, 1)	O	Motor control output V phase (negative)
TAPAnVP (n = 0, 1)	O	Motor control output V phase (positive)
TAPAnWN (n = 0, 1)	O	Motor control output W phase (negative)
TAPAnWP (n = 0, 1)	O	Motor control output W phase (positive)
TPBAnO (n = 0, 1)	O	TPBAn channel output
TAUDnIm (n = 0, 1, m = 0 to 15)	I	TAUDn channel input m
TAUDnOm (n = 0, 1, m = 0 to 15)	O	TAUDn channel output m
TAUJ0Im (m = 0 to 3)	I	TAUJ0 channel input m
TAUJ0Om (m = 0 to 3)	O	TAUJ0 channel output m
TSG3nOm (n = 0, 1, m = 0 to 7)	O	TSG3n channel output m
X1, X2	—	Crystal oscillator connections

CAUTION

When SCI30 is used in the C1H, use pin functions multiplexed with pins of the same GPIO port.

- **If the serial clock I/O signal is to be used:**
  - **P5_0 (SCI0RXD), P5_1 (SCI0TXD), and P5_2 (SCI0SCK)**
- **If the serial clock I/O signal is not to be used:**
  - **P5_0 (SCI0RXD), P5_1 (SCI0TXD) or**
  - **P7_3 (SCI0RXD), P7_4 (SCI0TXD)**

Table 2.59 C1M Pin Function (1/2)

Pin Name	I/O	Function
AnVREF (n = 0, 1)	—	ADCCn voltage supply and reference voltage
AnVSS (n = 0, 1)	—	ADCCn ground
$\overline{\text{ADCCnTRG}}$ (n = 0, 1)	I	ADCCn trigger
ADCC0Ipq (p = 0, 2, 3, q = 0 to 3) Combinations of p = 0 and q = 1 or 2 are excluded.	I	ADCC0 input channel pq
ADCC1Ipq (p = 0 to 4, 6, q = 0 to 2) Combinations of p = 1 and q = 2 and of p = 2 and q = 1 or 2 are excluded.	I	ADCC1 input channel pq
AnVCC (n = 0, 1)	—	ADCCn voltage supply
CANmRX (m = 0 to 3)	I	CANm receive data input
CANmTX (m = 0 to 3)	O	CANm transmit data output
CSIHnCSS0 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 0
CSIHnCSS1 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 1
CSIHnCSS2 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 2
CSIHnCSS3 (n = 0, 1)	O	CSIHn serial peripheral chip select signal 3
CSIHnRYI (n = 0, 1)	I	CSIHn ready (1) / busy (0) input signal
CSIHnRYO (n = 0, 1)	O	CSIHn ready (1) / busy (0) output signal
CSIHnSC (n = 0, 1)	IO	CSIHn serial clock signal
CSIHnSI (n = 0, 1)	I	CSIHn serial data input
CSIHnSO (n = 0, 1)	O	CSIHn serial data output
$\overline{\text{CSIHnSSI}}$ (n = 0, 1)	I	CSIHn slave select input signal
$\overline{\text{DCURDY}}$	O	Debug ready
DCUTCK	I	Debug clock
DCUTDI	I	Debug data input
DCUTDO	O	Debug data output
DCUTMS	I	Debug mode select
$\overline{\text{DCUTRST}}$	I	Debug reset
FLSCI3TX (FPDT)	O	Transmit data output
FLSCI3RX (FPDR)	I	Receive data input
FLSCI3SCK (FPCK)	I	Serial clock input
ENCAnE0 (n = 0, 1)	I	ENCAn encoder input (count pulse 0)
ENCAnE1 (n = 0, 1)	I	ENCAn encoder input (count pulse 1)
ENCAnTINm (n = 0, 1, m = 0, 1)	I	ENCAn capture trigger input nm
ENCAnEC (n = 0, 1)	I	ENCAn encoder input (clear pulse)
EVCC	—	Port buffer voltage supply
$\overline{\text{ERROROUT\_M}}$	O	ECM error output (MAIN)
$\overline{\text{ERROROUT\_C}}$	O	ECM error output (Checker)
FLMODE	I	Operating mode select pin
MD0	I	Operating mode select pin 0
MD1	I	Operating mode select pin 1
INTPm (m = 0 to 7)	I	External interrupt input m
LPDCLK	I	LPD clock input (4-pin mode)
LPDCLKOUT	O	LPD clock output (4-pin mode)
LPDI	I	LPD data input (4-pin mode)
LPDO	O	LPD data input/output (4-pin mode)

Table 2.59 C1M Pin Function (2/2)

Pin Name	I/O	Function
$\overline{\text{LPDRST}}$	I	LPD reset (4-pin mode)
NMI	I	External non-maskable interrupt
P0_m (m = 0 to 9)	IO	Port 0_m
P1_m (m = 0 to 15)	IO	Port 1_m
P2_m (m = 0 to 7)	IO	Port 2_m
P3_m (m = 0 to 3)	IO	Port 3_m
P4_m (m = 0 to 15)	IO	Port 4_m
P5_m (m = 3 to 9)	IO	Port 5_m
P7_m (m = 0 to 4)	IO	Port 7_m
SYSVCC	—	Voltage supply for system
VCC	—	Voltage supply for oscillator and Flash
VDD	—	Voltage regulators voltage supply
VSS	—	Ground
RVCC	—	Voltage supply for RDC
RVSS	—	Ground for RDC
$\overline{\text{RESET}}$	I	External reset input
RDC20COM	IO	Excitation common signal input/output
RDC20COSMNT	O	COS-side monitoring signal output
RDC20RSO	IO	Excitation signal input/output
RDC20S1	I	Resolver signal input
RDC20S2	I	Resolver signal input
RDC20S3	I	Resolver signal input
RDC20S4	I	Resolver signal input
RDC20SINMNT	O	SIN-side monitoring signal output
SCInRXD (n = 0 to 2)	I	SCIn receive data
SCInSCK (n = 1, 2)	IO	SCIn clock
SCInTXD (n = 0 to 2)	O	SCIn transmit data
TAPAnESO (n = 0, 2, 3)	I	Hi-Z control
TAPA0UN	O	Motor control output U phase (negative)
TAPA0UP	O	Motor control output U phase (positive)
TAPA0VN	O	Motor control output V phase (negative)
TAPA0VP	O	Motor control output V phase (positive)
TAPA0WN	O	Motor control output W phase (negative)
TAPA0WP	O	Motor control output W phase (positive)
TPBA0O	O	TPBAn channel output
TAUDnIm (n = 0, 1, m = 0 to 15)	I	TAUDn channel input m
TAUDnOm (n = 0, 1, m = 0 to 15)	O	TAUDn channel output m
TAUJ0Im (m = 0 to 3)	I	TAUJ0 channel input m
TAUJ0Om (m = 0 to 3)	O	TAUJ0 channel output m
TSG3nOm (n = 0, 1, m = 0 to 7)	O	TSG3n channel output m
X1, X2	—	Crystal oscillator connections

2.4.3 Pin State

Definition of Reset State

In description of pin state, each reset state is defined as shown in **Table 2.60**.

Table 2.60 Definition of Reset State

Reset State	Definition
External reset	Reset state from an external pin ($\overline{\text{RESET}}$ = L)
Internal reset	Between external reset release to internal reset release
After internal reset release	State where internal reset is released

Table 2.61 shows detailed pin states. Some pins may be excluded depending on grades and packages of this product. For the pins included in each product, see **Section 1.2, Pin Connection Diagram (Top View).**

Table 2.61 Pin State (1/2)

Pin Function		Pin State		
		$\overline{\text{RESET}}$ = L	$\overline{\text{RESET}}$ = H	
Classification	Pin Name	External Reset State	Before Internal Reset Release	After Internal Reset Release
Clock	X1	I	I	I
	X2	O	O	O
System control	$\overline{\text{RESET}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD0	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD1	I (Pull-down)	I (Pull-down)	I (Pull-down)
	FLMODE	I (Pull-down)	I (Pull-down)	I (Pull-down)
Interrupt	NMI	I (Pull-down)	I (Pull-down)	I (Pull-down)
ECM	$\overline{\text{ERROROUT\_M}}$	O	O	O
General-purpose I/O ports	P0_x	Z	Z	Z
	P1_x	Z	Z	Z
	P2_x	Z	Z	Z
	P3_x	Z	Z	Z
	P4_x	Z	Z	Z
	P5_x	Z	Z	Z
	P6_x	Z	Z	Z
	P7_x	Z	Z	Z
SAR A/D	ADCC0Ixx	Z	Z	Z
	ADCC1Ixx	Z	Z	Z
Resolver signal input	RDC20Sx	Z	Z	Z
	RDC21Sx	Z	Z	Z
Excitation signal output	RDC20RSO, RDC20COM	Z	Z	Z
	RDC21RSO, RDC21COM	Z	Z	Z
AUD RAM monitoring	$\overline{\text{AUDRST}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
	AUDCK	I (Pull-up)	I (Pull-up)	I (Pull-up)
	$\overline{\text{AUDSYNC}}$	I (Pull-up)	I (Pull-up)	I (Pull-up)
	AUDATA0 to AUDATA3	I (Pull-up)	I (Pull-up)	I (Pull-up)

Table 2.61 Pin State (2/2)

Pin Function			Pin State		
			$\overline{\text{RESET}}$ = L	$\overline{\text{RESET}}$ = H	
Classification	Pin Name		External Reset State	Before Internal Reset Release	After Internal Reset Release
Debug system	DCUTDI/LPDI/FLSCI3RX	Nexus: DCUTDI	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: LPDI	Z	I (Pull-up)	I (Pull-up)
		Writer I/F: FLSCI3RX	Z	Z	Z
		BSCAN: DCUTDI	I (Pull-up)	I (Pull-up)	I (Pull-up)
	DCUTDO/LPDO/ FLSCI3TX	Nexus: DCUTDO	Z	Z	Z
		LPD-4pin: LPDO	Z	O	O
		Writer I/F: FLSCI3TX	Z	Z	Z
		BSCAN: DCUTDO	Z	Z	Z
	DCUTCK/LPDCLK/ FLSCI3SCK	Nexus: DCUTCK	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: LPDCLK	Z	I (Pull-up)	I (Pull-up)
		Writer I/F: FLSCI3SCK	Z	Z	Z
		BSCAN: DCUTCK	I (Pull-up)	I (Pull-up)	I (Pull-up)
	DCUTMS	Nexus: DCUTMS	Z	I (Pull-up)	I (Pull-up)
		LPD-4pin: $\overline{\text{EVTO}}$*1	Z	O	O
		Writer I/F: (w/o function)	Z	Z	Z
		BSCAN: DCUTMS	I (Pull-up)	I (Pull-up)	I (Pull-up)
	$\overline{\text{DCUTRST}}$/$\overline{\text{LPDRST}}$	Nexus: $\overline{\text{DCUTRST}}$	I (Pull-down)	I (Pull-down)	I (Pull-down)
		LPD-4pin: $\overline{\text{LPDRST}}$			
		Writer I/F: (w/o function)			
		BSCAN: $\overline{\text{DCUTRST}}$			
	$\overline{\text{DCURDY}}$/LPDCLKOUT	Nexus: $\overline{\text{DCURDY}}$	Z	O	O
		LPD-4pin: LPDCLKOUT	Z	O	O
		Writer I/F: (w/o function)	Z	Z	Z
		Writer I/F: (w/o function)	Z	Z	Z

Note: I: Input
O: Output
Z: High impedance
Pull-up: On-chip pull-up resistor
Pull-down: On-chip pull-down resistor

Note 1. $\overline{\text{EVTO}}$ is not used in this device.

2.4.4 Handling of Unused Pins

Table 2.62 shows an example handling of unused pins.

Table 2.62 Example Handling of Unused Pins (1/2)

Category	Pins	IO	Example handling of unused pins	Internal pull-up/pull-down resistor
Clock	X1	I	(Required)	None
	X2	O	(Required)	None
System control	$\overline{\text{RESET}}$	I	(Required)	An internal pull-down resistor is included.
	MD0, FLMODE	I	(Required. When they are used in user boot mode, separately connect them with VSS via resistors.)	An internal pull-down resistor is included.
	MD1	I	• Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
Interrupt	NMI	I	• Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
ECM	$\overline{\text{ERROROUT\_M}}$	O	• Leave the pin open.	None
General-purpose I/O port	P0_m, P1_m, P2_m, P3_m, P4_m, P5_m, P6_m, P7_m	IO	[Input mode] • Leave the pins open, and disable "input enable" (PMCn_m = 0, PMn_m = 1, and PIBCn_m = 0 (values after reset)). • Leave the pins open, and enable the internal pull-up/pull-down resistors (use PUn_m and PDn_m). • Separately connect each pin with the power supply/GND via a resistor. [Output mode] • Leave the pins open.	Internal pull-up/pull-down resistors that can be set by the registers are included.
ADCC	ADCC0Ipq ADCC1Ipq (Analog input-only)	I	• Leave the pin open.	None
RDC2	RDC2nCOM, RDC2nRSO	I/O	• Leave the pins open and stop the analog circuit.	None
	RDC2nS1, RDC2nS2, RDC2nS3, RDC2nS4	I	• Leave the pins open and stop the analog circuit.	None
Debug system (AUDRAM)	$\overline{\text{AUDRST}}$	I	• Leave the pin open. • Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
	AUDCK, $\overline{\text{AUDSYNC}}$	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor.	An internal pull-up resistor is included.
	AUDATAn	IO	• Leave the pin open.	An internal pull-up resistor is included.

Table 2.62 Example Handling of Unused Pins (2/2)

Category	Pins	IO	Example handling of unused pins	Internal pull-up/pull-down resistor
Debug system (NEXUS/LPD)	DCUTDI	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor. (Serial programming mode is disabled.)	An internal pull-up resistor is included.
	DCUTDO	O	• Leave the pin open. (Serial programming mode is disabled.)	None
	DCUTCK	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor. (Serial programming mode is disabled.)	An internal pull-up resistor is included.
	DCUTMS	I	• Leave the pin open. • Separately connect the pin with VCC via a resistor.	An internal pull-up resistor is included.
	$\overline{\text{DCUTRST}}$	I	• Separately connect the pin with VSS via a resistor.	An internal pull-down resistor is included.
	$\overline{\text{DCURDY}}$	O	• Leave the pin open.	None
Power system	EVCC	—	(Required)	
	SYSVCC	—	(Required)	
	VCC	—	(Required)	
	PLLVCC	—	(Required)	
	PLLVSS	—	(Required)	
	VDD	—	(Required)	
	VSS	—	(Required)	
	EPTVOUT	—	• Leave the pin open.	
	AnVCC	—	(Required)	
	AnVSS	—	(Required)	
	AnVREFH	—	(Required)	
	RVCC	—	(Required)	
	RVSS	—	(Required)	

Note 1. When a pull-up/pull-down is performed via a resistor outside the product, set the resistance value to 1 kΩ or more.

Note 2. Note 2. When a pull-up/pull-down is also performed via a resistor outside the product for a pin that includes internal pull-up/pull-down resistors, be careful so that the resistance voltage is not divided.

Section 3 CPU System

3.1 Overview

3.1.1 Block Configuration

Figure 3.1 is a block diagram of the C1H configuration.

The C1M does not incorporate CPU2 (PE2).

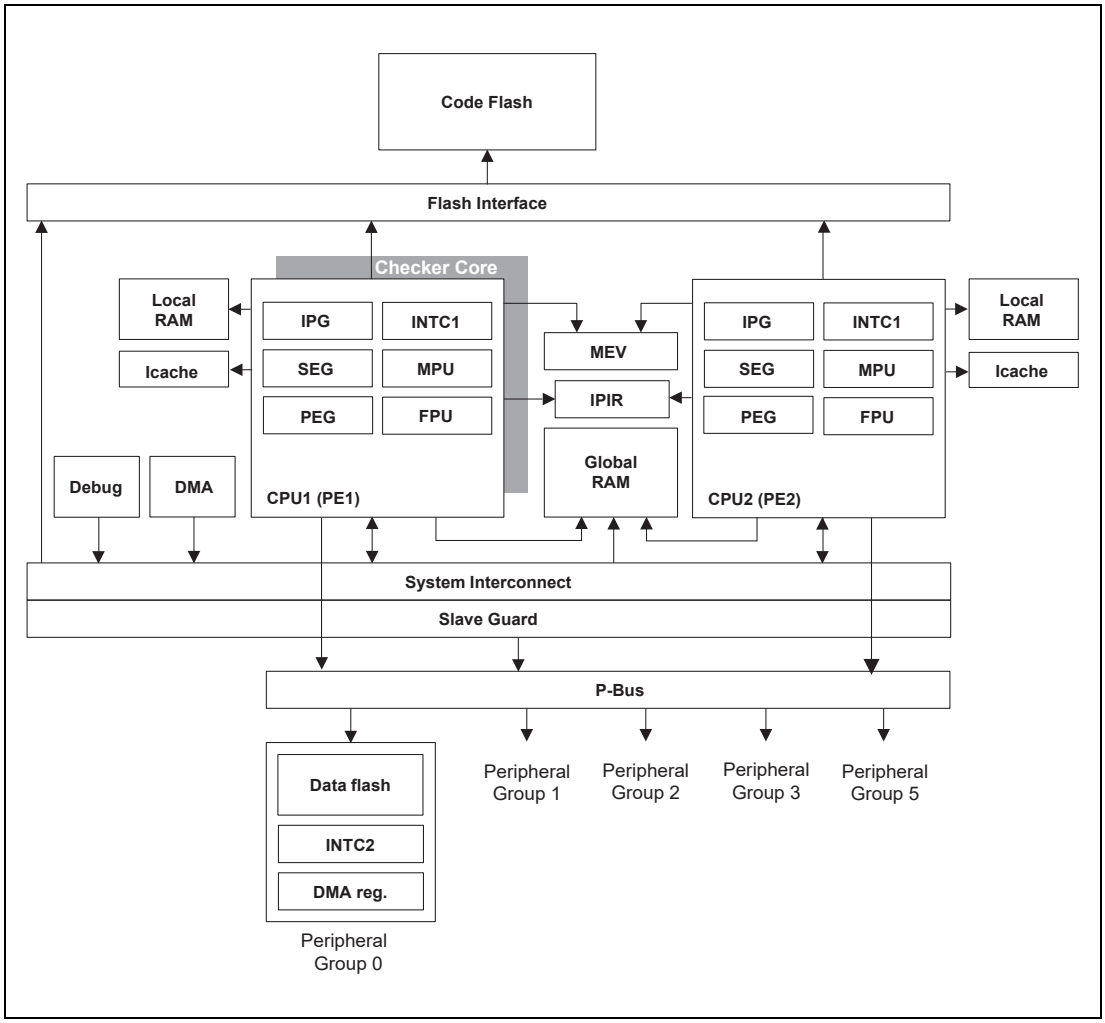


Figure 3.1 Block Diagram of the C1H Configuration

CPU1 (PE1)

The RH850 G3M Core is included as a main CPU. CPU1 also includes the checker core for safety assurance.

CPU2 (PE2)

The RH850 G3M is included for enhanced performance CPU (only in the C1H).

Local RAM

Each PE has a high-speed accessible RAM.

Global RAM

The global RAM is a sharable large-capacity RAM.

Code Flash

A large-capacity flash memory is included for program storage. CPU1 and CPU2 share the code flash and they are connected with each other via the flash interface.

Data Flash

It is a flash memory being rewritable by the CPUs.

P-Bus

The P-Bus connects the peripheral IPs. The P-Bus is divided into five peripheral Groups 0 to 3, and 5. For the details, see **Section 3.1.2, Periphral Group Configuration**.

INTC1, INTC2

INTC1 is an interrupt controller exclusive to each PE. INTC2 is a common interrupt controller that all PEs share, being able to set the binding destination PE of an interrupt request by the registers.

DMA

Two DMA transfer modules, DMAC and DTS, are included.

Slave Guard

The slave guard is a function to prevent unauthorized access from the specific bus master, consisting of the following guard functions:

(1) PE Guard (PEG)

The PE guard is a function to prevent unauthorized access to the resources in the PE from the bus master other than the PE itself. Access from the PE itself is only enabled but all other accesses are disabled after release from the reset state. For details, see **Section 3.2.4.1, PE Guard Function (PEG)**.

(2) Global RAM Guard (GRG)

The global RAM guard is a function to prevent unauthorized access to the global RAM from the bus master. The global RAM is in the unprotected state (accessible from all bus masters) after release from the reset state. For details, see **Section 27.4.2, GRG (Global RAM Guard)** in the section of Functional Safety.

(3) Peripheral Guard (PBG)

The peripheral guard is a function to prevent unauthorized access to peripheral devices from the bus master. Access from CPU1 is possible after release from the reset state, but for some peripheral devices, accesses from other than CPU1 are disabled. For details, see **Section 27.4.3, PBG** in the section of Functional Safety.

3.1.2 Peripheral Group Configuration

P-Bus is connected to peripheral group 0 to 3, 5 and CPU, each module belongs to either one of the peripheral groups. The following shows the relation of modules and peripheral groups. For the module names, see the list of registers in each section.

Table 3.1 List of Peripheral Group Configuration (1/2)

Peripheral Group	Module Name*1
CPU	INTC1
	IPIRSS*2
	IPG
	MEV*2
	PEG
	SEG
	TESTCOMP
CPU(DEBUG)	AUD
0	APDP [INTC2]
	APDP [PDMA]
	DMASS
	ECCCPU1
	ECCEEP
	ECCEEPC
	ECCFLI
	ECCGRAM
	ECCIC1
	FACI [FCUFAREA]
	FLASH[FRDCYCLD]
	INTC2
	MGDGR
	MISG
	PBG [PBG0, PBG1]
	(Data Flash)
1	OSTMn (n = 0 to 2)*2
	SCI3n (n = 0 to 2)
	WDTAn (n = 0, 1)*2
	RDC2n (n = 0, 1)*2
	PBG [PBG2]
	TAUDn (n = 0, 1)
	TAUJn (n = 0)
	TSG3n (n = 0, 1)
	TAPAn (n = 0 to 3)
	TPBAn (n = 0, 1)*2
	ENCAn (n = 0, 1)
	PBG [PBG4]
	PIC1A
	PIC2B

Table 3.1 List of Peripheral Group Configuration (2/2)

Peripheral Group	Module Name*1
2	APDP [CSIHn (n = 0, 1)]
	CSIHn (n = 0, 1)
	E7RC0M/C
	E7CS0M/C
	E7CS1M/C
	INTIF
	PBG [PBG5]
	RS-CAN0
	EMU2n (n = 0, 1)
3	ADCCn (n = 0, 1)
	APDP [ADCCn (n = 0, 1)]
	APDP [PORT]
	DCRAn (n = 0, 1)
	DNF
	RLN21n (n = 0 to 2)*2
	RLN21nm (n = 0 to 2, m = 0 to 2)*2
	ECM
	ECM [ECMC]
	ECM [ECMM]
	EINT
	FACI [Other than FCUFAREA]
	FLASH [SELFIDn (n = 0 to 3, ST)]
	FLASH [OPBTn (n = 0, 2), PRDNAMEn (n = 1 to 4)]
	PBG [PBG3]
	PORT
5	CLMAC
	CLMAn (n = 0 to 2)
	FLASH [FHVE15]
	FLASH [FHVE3]
	SYS

Note 1. [ ] of module name shows the registers and the applicable modules.

Note 2. The above modules and the unit numbers are for C1H implementation, some modules are not implemented in C1M. The difference in C1M from C1H is shown blow.

- Non provided module: IPIRSS, MEV, RLN21n, RLN21nm
- Module with modified unit number: OSTMn (n = 0, 1), WDTAn (n = 0), RDC2n (n = 0), TPBAn (n = 0)

3.2 CPU

3.2.1 Core Functions

3.2.1.1 Features

Table 3.2 lists features of the RH850G3M core.

Table 3.2 Features of the RH850G3M Core

Item	Feature
CPU	• Advanced 32-bit architecture for embedded control • 32-bit internal data bus • Thirty-two 32-bit general registers • RISC-type instruction sets – Long-/short-format load/store instructions – Three-operand instructions – Instruction sets based on C language • CPU operating modes – User mode and supervisor mode • Address space: 4-Gbyte linear address space for both data and instructions • Instructions: A snooze instruction (SNOOZE) is included for temporary suspension by switching the CPU clock signal (CLK_CPU) off for 32 clock cycles.
Coprocessor	• A floating-point operation coprocessor (FPU) mounted – Supports single precision (32 bits) and double precision (64 bits). – Supports data types and exceptions conforming to IEEE754. – Rounding mode: Neighborhood, 0 direction, +∞ direction, and -∞ direction – Handling denormalized numbers: Rounding down to 0 or exception notification to conform to IEEE754
Exception/ Interrupt	• 16 interrupt priority levels settable for each channel • Vector selection method selectable according to performance request or memory usage – Direct branching exception vectors – Indirect branching exception vectors referring to the address table • Supports the high-speed save/return processing of the context by the dedicated instructions (PUSHSP and POPSP) at the generation of an interrupt
Memory Management	• Memory protection function (MPU): 12 areas settable
Cache	• Instruction cache

3.2.1.2 Register Set

(1) Program Registers

Program registers include the general-purpose registers (r0 to r31) and program counter (PC).

Table 3.3 List of Program Registers

Program Register	Name	Function	Description
General-purpose registers	r0	Zero register	Always retains “0”
	r1	Assembler reserved register	Used as working register for generating addresses
	r2	Register for address and data variables (used when the real-time OS used does not use this register)	
	r3	Stack pointer (SP)	Used for generating a stack frame when a function is called
	r4	Global pointer (GP)	Used for accessing a global variable in the data area
	r5	Text pointer (TP)	Used as a register that indicates the start of the text area (area where program code is placed)
	r6 to r29	Register for addresses and data variables	
	r30	Element pointer (EP)	Used as a base pointer for generating addresses when accessing memory
	r31	Link pointer (LP)	Used when the compiler calls a function
Program counter	PC	Retains instruction addresses during execution of programs	

NOTE

For further descriptions of r1, r3 to r5, and r31 used for an assembler and/or C compiler, see the manual of each software development environment.

(a) General-purpose registers

A total of 32 general-purpose registers (r0 to r31) are provided. All of these registers can be used for either data variables or address variables. The general-purpose register r0 always retains 0, but the values of the registers r1 to r31 after reset are undefined. The general-purpose registers are duplicated set-by-set per hardware thread. Of the general-purpose registers, r0 to r5, r30, and r31 are assumed to be used for special purposes in software development environments, so it is necessary to note the following when using them.

1. r0, r3, and r30

   These registers are implicitly used by instructions.

   r0 is a register that always retains “0”. It is used for operations that use 0 and addressing with base address being 0.

   r3 is implicitly used by the PREPARE, DISPOSE, PUSHSP, and POPSP instructions.

   r30 is used as a base pointer when the SLD or SST instruction accesses memory.

2. r1, r4, r5, and r31

   These registers are implicitly used by the assembler and C compiler.

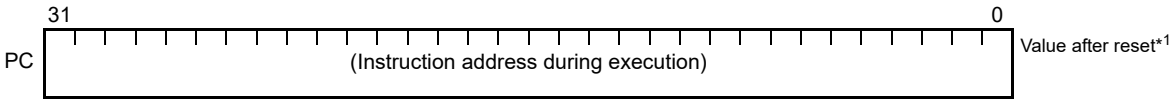
   When using these registers, register contents must first be saved so they are not lost and can be restored after the registers are used.

3. r2

   This register might be used by a real-time OS in some cases. If the real-time OS that is being used is not using r2, r2 can be used as a register for address variables or data variables.

(b) PC – Program counter

The PC retains the address of the instruction being executed. Bit 0 is fixed to 0, and branching to an odd number address is disabled.



Note 1. The value after reset differs depending on the startup area. For details, see **Section 4, Address Space**.

(2) Basic System Registers

The basic system registers are used to control CPU status and to retain exception information.

System registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and a selection ID.

Table 3.4 Basic System Registers

Register No. (regID, selID)	Symbol	Function	Access Permission
SR0, 0	EIPC	Status save registers when acknowledging EI level exception	SV
SR1, 0	EIPSW	Status save registers when acknowledging EI level exception	SV
SR2, 0	FEPC	Status save registers when acknowledging FE level exception	SV
SR3, 0	FEPSW	Status save registers when acknowledging FE level exception	SV
SR5, 0	PSW	Program status word	*1
SR6, 0	FPSR	(Refer to FPU function registers.)	CU0 and SV
SR7, 0	FPEPC	(Refer to FPU function registers.)	CU0 and SV
SR8, 0	FPST	(Refer to FPU function registers.)	CU0
SR9, 0	FPCC	(Refer to FPU function registers.)	CU0
SR10, 0	FPCFG	(Refer to FPU function registers.)	CU0
SR11, 0	FPEC	(Refer to FPU function registers.)	CU0 and SV
SR13, 0	EIIC	EI level exception cause	SV
SR14, 0	FEIC	FE level exception cause	SV
SR16, 0	CTPC	CALLT execution status save register	UM
SR17, 0	CTPSW	CALLT execution status save register	UM
SR20, 0	CTBP	CALLT base pointer	UM
SR28, 0	EIWR	EI level exception working register	SV
SR29, 0	FEWR	FE level exception working register	SV
SR0, 1	MCFG0	Machine configuration	SV
SR2, 1	RBASE	Reset vector base address	SV
SR3, 1	EBASE	Exception handler vector address	SV
SR4, 1	INTBP	Base address of the interrupt handler table	SV
SR5, 1	MCTL	CPU control	SV
SR6, 1	PID	Processor ID	SV
SR11, 1	SCCFG	SYSCALL operation setting	SV
SR12, 1	SCBP	SYSCALL base pointer	SV
SR0, 2	HTCFG0	Thread configuration	SV
SR6, 2	MEA	Memory error address	SV
SR7, 2	ASID	Address space ID	SV
SR8, 2	MEI	Memory error information	SV

Note 1. The access permission differs depending on the bit.

(a) EIPC — Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the address of the instruction that was being executed when the EI level exception occurred, or of the next instruction, is saved to the EIPC register (see ***section 4.1.3, Types of Exceptions* in the *RH850G3M User's Manual: Software***).

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the EIPC register. An odd-numbered address must not be specified.

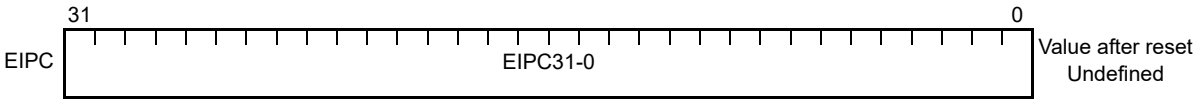


Table 3.5 EIPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 1	EIPC31-1	These bits indicate the return PC when an EI level exception is acknowledged.	R/W	Undefined
0	EIPC0	This bit indicates the return PC when an EI level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the EIRET instruction is executed is 0.	R/W	Undefined

(b) EIPSW — Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the current PSW setting is saved to the EIPSW register.

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

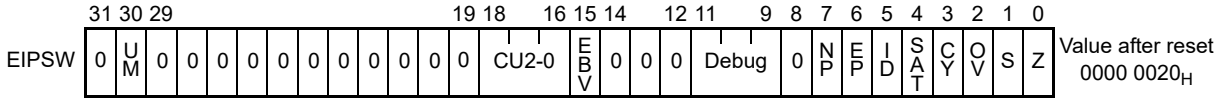


Table 3.6 EIPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
30	UM	This bit saves the PSW.UM bit setting when an EI level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
18 to 16	CU2-0	These bits save the PSW.CU2-0 field setting when an EI level exception is acknowledged. (The CU2-1 bits are reserved for future expansion. These bits should be set to 0.)	R/W	0
15	EBV	This bit saves the PSW.EBV bit setting when an EI level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an EI level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
7	NP	This bit saves the PSW.NP bit setting when an EI level exception is acknowledged.	R/W	0
6	EP	This bit saves the PSW.EP bit setting when an EI level exception is acknowledged.	R/W	0
5	ID	This bit saves the PSW.ID bit setting when an EI level exception is acknowledged.	R/W	1
4	SAT	This bit saves the PSW.SAT bit setting when an EI level exception is acknowledged.	R/W	0
3	CY	This bit saves the PSW.CY bit setting when an EI level exception is acknowledged.	R/W	0
2	OV	This bit saves the PSW.OV bit setting when an EI level exception is acknowledged.	R/W	0
1	S	This bit saves the PSW.S bit setting when an EI level exception is acknowledged.	R/W	0
0	Z	This bit saves the PSW.Z bit setting when an EI level exception is acknowledged.	R/W	0

(c) FEPC — Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the address of the instruction that was being executed when the FE level exception occurred, or of the next instruction, is saved to the FEPC register (see ***section 4.1.3, Types of Exceptions* in the *RH850G3M User's Manual: Software***).

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the FEPC register. An odd-numbered address must not be specified.

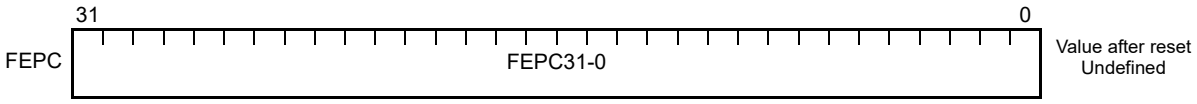


Table 3.7 FEPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 1	FEPC31-1	These bits indicate the return PC when an FE level exception is acknowledged.	R/W	Undefined
0	FEPC0	This bit indicates the return PC when an FE level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the FERET instruction is executed is 0.	R/W	Undefined

(d) FEPSW — Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the current PSW setting is saved to the FEPSW register.

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

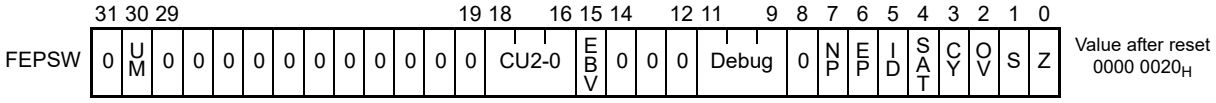


Table 3.8 FEPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
30	UM	This bit saves the PSW.UM bit setting when an FE level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
18 to 16	CU2-0	These bits save the PSW.CU2-0 field setting when an FE level exception is acknowledged. (The CU2-1 bits are reserved for future expansion. These bits should be set to 0.)	R/W	0
15	EBV	This bit saves the PSW.EBV bit setting when an FE level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an FE level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
7	NP	This bit saves the PSW.NP bit setting when an FE level exception is acknowledged.	R/W	0
6	EP	This bit saves the PSW.EP bit setting when an FE level exception is acknowledged.	R/W	0
5	ID	This bit saves the PSW.ID bit setting when an FE level exception is acknowledged.	R/W	1
4	SAT	This bit saves the PSW.SAT bit setting when an FE level exception is acknowledged.	R/W	0
3	CY	This bit saves the PSW.CY bit setting when an FE level exception is acknowledged.	R/W	0
2	OV	This bit saves the PSW.OV bit setting when an FE level exception is acknowledged.	R/W	0
1	S	This bit saves the PSW.S bit setting when an FE level exception is acknowledged.	R/W	0
0	Z	This bit saves the PSW.Z bit setting when an FE level exception is acknowledged.	R/W	0

(e) PSW — Program status word

PSW (program status word) is a set of flags that indicate the program status (instruction execution result) and bits that indicate the operation status of the CPU (flags are bits in the PSW that are referenced by a condition instruction (Bcond, CMOV, etc.)).

When the LDSR instruction is used to change the contents of bits 7 to 0 in this register, the changed contents become valid immediately after completion of the LDSR instruction execution.

The access permission for the PSW register differs depending on the bit. All bits can be read, but some bits can only be written under certain conditions. See **Table 3.9, Access Permission for PSW Register** for the access permission for each bit.

Table 3.9 Access Permission for PSW Register

Bit		Access Permission When Reading	Access Permission When Writing
30	UM	UM	SV*1
18 to 16	CU2-0	UM	SV*1
15	EBV	UM	SV*1
11 to 9	Debug	UM	Special*1
7	NP	UM	SV*1
6	EP	UM	SV*1
5	ID	UM	SV*1
4	SAT	UM	UM
3	CY	UM	UM
2	OV	UM	UM
1	S	UM	UM
0	Z	UM	UM

Note 1. The access permission for the whole PSW register is UM, so the PIE exception does not occur even if the register is written by using an LDSR instruction when PSW.UM is 1. In this case, writing is ignored.

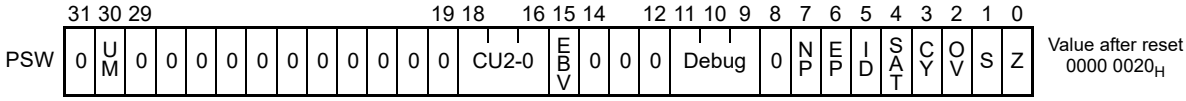


Table 3.10 PSW Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after reset
31	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
30	UM	This bit indicates that the CPU is in user mode (UM mode). 0: Supervisor mode 1: User mode	R/W	0
29 to 19	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
18, 17	CU2-CU1	These bits indicate the coprocessor use permissions. (Reserved for future expansion. These bits should be set to 0.)	R	00
16	CU0	This bit indicates the coprocessor use permissions. When the bit corresponding to the coprocessor is 0, a coprocessor use prohibition exception is generated in response to execution of a coprocessor instruction or access to coprocessor resources (system registers). CU0 bit: FPU	R/W	0
15	EBV	This bit indicates the reset vector and exception vector operation. See the descriptions of the RBASE register in **(q), RBASE — Reset vector base address register** and the EBASE register in **(r), EBASE — Exception handler vector address register**.	R/W	0

Table 3.10 PSW Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after reset
14 to 12	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
11 to 9	Debug	These bits are used in the debugging functions of development tools. In normal operation, clear these bits to 0.	—	0
8	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
7	NP	This bit disables acceptance of FE level exceptions. Once an FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions or FE level exceptions. For the exceptions disabled by the NP bit, see the list of the exception sources in the *RH850G3M User's Manual: Software*. 0: An acceptance of FE level exceptions is enabled. 1: An acceptance of FE level exceptions is disabled.	R/W	0
6	EP	This bit indicates that an exception other than an interrupt is being serviced. It is set to 1 when the corresponding exception occurs. This bit does not affect acknowledging an exception request even when it is set to 1. 0: An exception other than an interrupt is not being serviced. 1: An interrupt is being serviced.	R/W	0
5	ID	This bit disables acceptance of EI level exceptions. Once an EI level exception or FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions. For the exceptions disabled by the ID bit, see the list of the exception sources in the *RH850G3M User's Manual: Software*. This bit is also used to disable EI level exceptions from being acknowledged as a critical section while an ordinary program or interrupt is being serviced. It is set to 1 when the DI instruction is executed, and cleared to 0 when the EI instruction is executed. A change in the ID bit by the EI or DI instruction is enabled from the next instruction. 0: An acceptance of EI level exceptions is enabled. 1: An acceptance of EI level exceptions is disabled.	R/W	1
4	SAT*1	This bit indicates that the operation result is saturated because the result of a saturated operation instruction operation has overflowed. This is a cumulative flag, so when the operation result of the saturated operation instruction becomes saturated, this bit is set to 1, but it is not cleared to 0 when the operation result for a subsequent instruction is not saturated. This bit is cleared to 0 by the LDSR instruction. This bit is neither set to 1 nor cleared to 0 when an arithmetic operation instruction is executed. 0: Not saturated 1: Saturated	R/W	0
3	CY	This bit indicates whether a carry or borrow has occurred in the operation result. 0: Carry and borrow have not occurred. 1: Carry or borrow has occurred.	R/W	0
2	OV*1	This bit indicates whether or not an overflow has occurred during an operation. 0: Overflow has not occurred. 1: Overflow has occurred.	R/W	0
1	S*1	This bit indicates whether or not the result of an operation is negative. 0: Result of operation is positive or 0. 1: Result of operation is negative	R/W	0
0	Z	This bit indicates whether or not the result of an operation is 0. 0: Result of operation is not 0. 1: Result of operation is 0.	R/W	0

Note 1. The operation result of the saturation processing is determined in accordance with the contents of the OV flag and S flag during a saturated operation. When only the OV flag is set to 1 during a saturated operation, the SAT flag is set to 1.

Operation Result Status	Flag Status			Operation Result after Saturation Processing
	SAT	OV	S	
Exceeded positive maximum value	1	1	0	7FFF FFFF$_H$
Exceeded negative maximum value	1	1	1	8000 0000$_H$
Positive (maximum value not exceeded)	Value prior to operation is retained.	0	0	Operation result itself
Negative (maximum value not exceeded)			1	

(f) EIIC — EI level exception source register

The EIIC register retains the source of any EI level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception source.

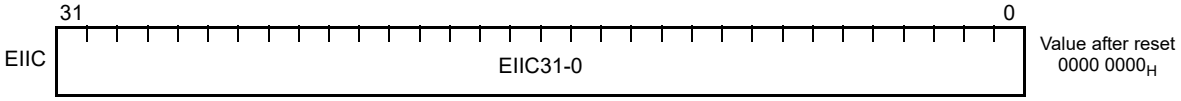


Table 3.11 EIIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	EIIC31-0	These bits store the exception source code when an EI level exception is acknowledged. Regarding the stored exception source code, see **Table 6.11, Interrupt Exception Handlers and Orders of Priority** and the list of the exception sources in the *RH850G3M User's Manual: Software*. The EIIC31 to EIIC16 field stores detailed exception source codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(g) FEIC — FE level exception source register

The FEIC register retains the source of any FE level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception source.

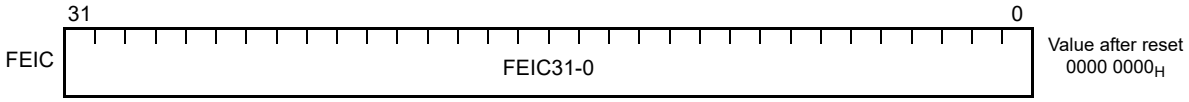


Table 3.12 FEIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	FEIC31-0	These bits store the exception source code when an FE level exception is acknowledged. Regarding the stored exception source code, see **Table 6.11, Interrupt Exception Handlers and Orders of Priority** and the list of the exception sources in the *RH850G3M User's Manual: Software*. The FEIC31 to FEIC16 field stores detailed exception source codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(h) CTPC — Status save register when executing CALLT instruction

When a CALLT instruction is executed, the address of the next instruction after the CALLT instruction is saved to CTPC. Be sure to set an even-numbered address to the CTPC register. An odd-numbered address must not be specified.

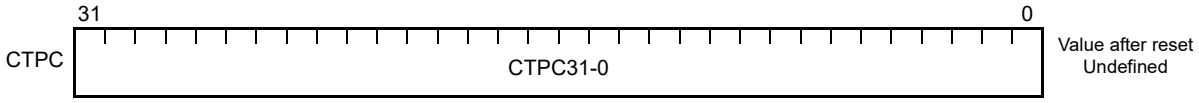


Table 3.13 CTPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 1	CTPC31-1	These bits indicate the PC of the instruction after the CALLT instruction.	R/W	Undefined
0	CTPC0	This bit indicates the PC of the instruction after the CALLT instruction. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the CTRET instruction is executed is 0.	R/W	Undefined

(i) CTPSW — Status save register when executing CALLT instruction

When a CALLT instruction is executed, some of the PSW (program status word) settings are saved to CTPSW.

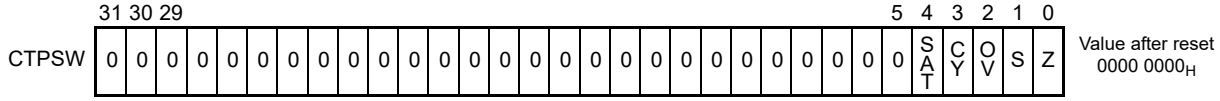


Table 3.14 CTPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 5	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
4	SAT	This bit saves the PSW.SAT bit setting when the CALLT instruction is executed.	R/W	0
3	CY	This bit saves the PSW.CY bit setting when the CALLT instruction is executed.	R/W	0
2	OV	This bit saves the PSW.OV bit setting when the CALLT instruction is executed.	R/W	0
1	S	This bit saves the PSW.S bit setting when the CALLT instruction is executed.	R/W	0
0	Z	This bit saves the PSW.Z bit setting when the CALLT instruction is executed.	R/W	0

(j) CTBP — CALLT base pointer register

The CTBP register is used to specify table addresses of the CALLT instruction and generate target addresses.

Be sure to set the CTBP register to a halfword address.

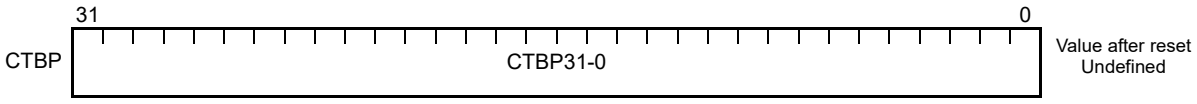


Table 3.15 CTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 1	CTBP31-1	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the first address in the table used by the CALLT instruction.	R/W	Undefined
0	CTBP0	This bit indicates the base pointer address of the CALLT instruction. This bit indicates the first address in the table used by the CALLT instruction. Always set this bit to 0.	R	0

(k) ASID — Address space ID register

This is the address space ID. This is used to identify the address space provided by the memory management function.

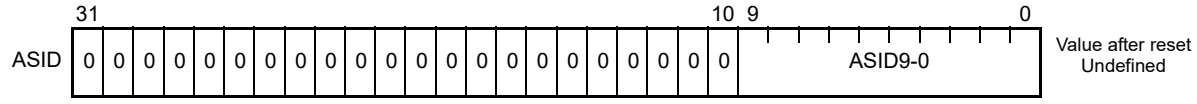


Table 3.16 ASID Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 10	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
9 to 0	ASID9-0	This field is the address space ID.	R/W	Undefined

(l) EIWR — EI level exception working register

The EIWR register is used as a working register when an EI level exception has occurred.

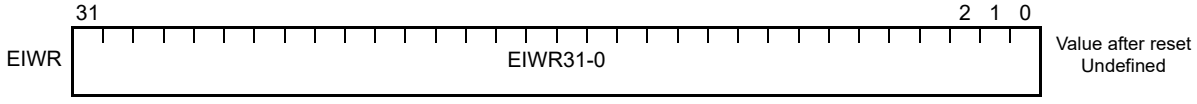


Table 3.17 EIWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	EIWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an EI level exception. Use this register for purposes such as temporal saving the values of general-purpose registers.	R/W	Undefined

(m) FEWR — FE level exception working register

The FEWR register is used as a working register when an FE level exception has occurred.

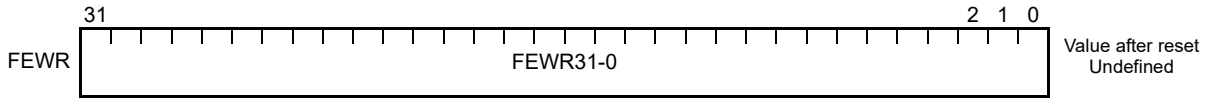


Table 3.18 FEWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	FEWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an FE level exception. Use this register for purposes such as temporal saving the values of general-purpose registers.	R/W	Undefined

(n) HTCFG0 — Thread configuration register

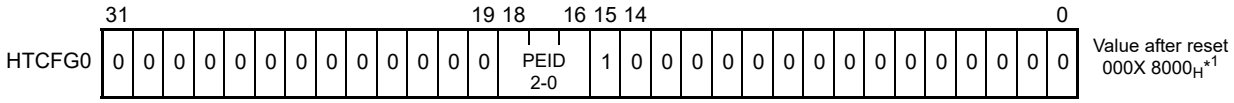


Table 3.19 HTCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 19	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
18 to 16	PEID2-0	These bits indicate the processor element number.	R	CPU1 (PE1): 001_B CPU2 (PE2): 010_B
15	—	(Reserved for future expansion. This bit should be set to 1.)	R	1
14 to 0	—	(Reserved for future expansion. These bits should be set to 0.)	R	0

Note 1. The value depends on the CPU that has access.

(o) MEA — Memory error address register

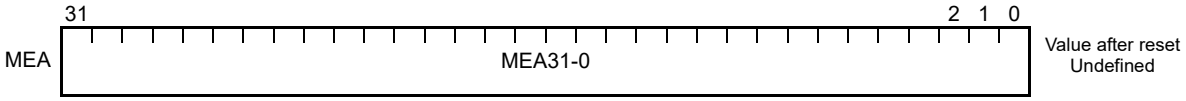


Table 3.20 MEA Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	MEA31-0	These bits store an address when a MAE (misaligning) or MPU violation occurs.	R/W	Undefined

(p) MEI — Memory error information register

This register is used to store information about the instruction that caused the exception when a misaligned (MAE) or memory protection (MDP) exception had occurred.

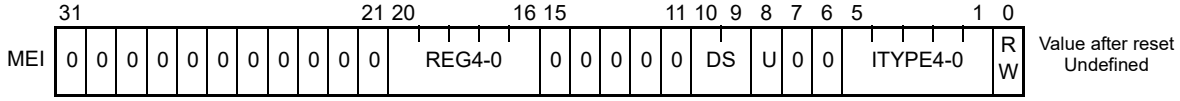


Table 3.21 MEI Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 21	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
20 to 16	REG4-0	These bits indicate the number of the source or destination register accessed by the instruction that caused the exception. For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
15 to 11	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
10, 9	DS	These bits indicate the type of data handled by the instruction that caused the exception.*1 0: Byte (8 bits) 1: Halfword (16 bits) 2: Word (32 bits) 3: Double-word (64 bits) For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
8	U	These bits indicate the sign extension method of the instruction that caused the exception. 0: Signed 1: Unsigned For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
7, 6	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
5 to 1	ITYPE4-0	These bits indicate the instruction that caused the exception. For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
0	RW	This bit indicates whether the operation of the instruction that caused the exception was read (Load-memory) or write (Store-memory). 0: Read (Load-memory) 1: Write (Store-memory) For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined

Note 1. Even if the data is divided and access is made several times due to the specifications of the hardware, the original data type indicated by the instruction is stored.

Table 3.22 Instructions Causing Exceptions and Values of MEI Register (1/2)

Instruction	REG	DS	U	RW	ITYPE
SLD.B	dst	0 (byte)	0 (signed)	0 (read)	00000_B
SLD.BU	dst	0 (byte)	1 (unsigned)	0 (read)	00000_B
SLD.H	dst	1 (halfword)	0 (signed)	0 (read)	00000_B
SLD.HU	dst	1 (halfword)	1 (unsigned)	0 (read)	00000_B
SLD.W	dst	2 (word)	0 (signed)	0 (read)	00000_B
SST.B	src	0 (byte)	0 (signed)	1 (write)	00000_B
SST.H	src	1 (halfword)	0 (signed)	1 (write)	00000_B
SST.W	src	2 (word)	0 (signed)	1 (write)	00000_B
LD.B (disp16)	dst	0 (byte)	0 (signed)	0 (read)	00001_B

Table 3.22 Instructions Causing Exceptions and Values of MEI Register (2/2)

Instruction	REG	DS	U	RW	ITYPE
LD.BU (disp16)	dst	0 (byte)	1 (unsigned)	0 (read)	00001_B
LD.H (disp16)	dst	1 (halfword)	0 (signed)	0 (read)	00001_B
LD.HU (disp16)	dst	1 (halfword)	1 (unsigned)	0 (read)	00001_B
LD.W (disp16)	dst	2 (word)	0 (signed)	0 (read)	00001_B
ST.B (disp16)	src	0 (byte)	0 (signed)	1 (write)	00001_B
ST.H (disp16)	src	1 (halfword)	0 (signed)	1 (write)	00001_B
ST.W (disp16)	src	2 (word)	0 (signed)	1 (write)	00001_B
LD.B (disp23)	dst	0 (byte)	0 (signed)	0 (read)	00010_B
LD.BU (disp23)	dst	0 (byte)	1 (unsigned)	0 (read)	00010_B
LD.H (disp23)	dst	1 (halfword)	0 (signed)	0 (read)	00010_B
LD.HU (disp23)	dst	1 (halfword)	1 (unsigned)	0 (read)	00010_B
LD.W (disp23)	dst	2 (word)	0 (signed)	0 (read)	00010_B
ST.B (disp23)	src	0 (byte)	0 (signed)	1 (write)	00010_B
ST.H (disp23)	src	1 (halfword)	0 (signed)	1 (write)	00010_B
ST.W (disp23)	src	2 (word)	0 (signed)	1 (write)	00010_B
LD.DW (disp23)	dst	3 (double-word)	0 (signed)	0 (read)	00010_B
ST.DW (disp23)	src	3 (double-word)	0 (signed)	1 (write)	00010_B
LDL.W	dst	2 (word)	0 (signed)	0 (read)	00111_B
STC.W	src	2 (word)	0 (signed)	1 (write)	00111_B
CAXI	dst	2 (word)	1 (unsigned)	0 (read)*1	01000_B
SET1	—	0 (byte)	1 (unsigned)	0 (read)*1	01001_B
CLR1	—	0 (byte)	1 (unsigned)	0 (read)*1	01001_B
NOT1	—	0 (byte)	1 (unsigned)	0 (read)*1	01001_B
TST1	—	0 (byte)	1 (unsigned)	0 (read)	01001_B
PREPARE	—	2 (word)	1 (unsigned)	1 (write)	01100_B
DISPOSE	—	2 (word)	1 (unsigned)	0 (read)	01100_B
PUSHSP	—	2 (word)	1 (unsigned)	1 (write)	01101_B
POPSP	—	2 (word)	1 (unsigned)	0 (read)	01101_B
SWITCH	—	1 (halfword)	0 (signed)	0 (read)	10000_B
CALLT	—	1 (halfword)	1 (unsigned)	0 (read)	10001_B
SYSCALL	—	2 (word)	1 (unsigned)	0 (read)	10010_B
CACHE	—	—	—	0 (read)	10100_B
Interrupt (table reference)*2	—	2 (word)	1 (unsigned)	0 (read)	10101_B

Note 1. This exception occurs when the instruction executes a read access.

Note 2. An exception occurs when the table reference interrupt vector is read.

NOTE

dst: destination register number, src: source register number

(q) RBASE — Reset vector base address register

This register indicates the reset vector address when there is a reset. If the PSW.EBV bit is 0, this vector address is also used as the exception vector address.

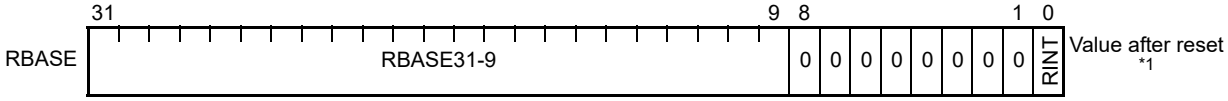


Table 3.23 RBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 9	RBASE31-9	These bits indicate the reset vector when there is a reset. When PSW.EBV = 0, this address is also used as the exception vector. The RBASE8 to RBASE0 bits are not assigned names because these bits are implicitly set to 0.	R	*1
8 to 1	—	(Reserved for future expansion. Always write 0.)	R	0
0	RINT	This bit specifies the setting of the offset address for direct branching method. This bit is valid when PSW.EBV = 0. 0: Offset address ($+100_H$ to $+1F0_H$) is determined according to interrupt priority order. 1: Offset address is fixed to $+100_H$. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.	R	0

Note 1. The reset vector differs depending on the startup area. For details, see **Section 4, Address Space**.

(r) EBASE — Exception handler vector address register

This register indicates the exception handler vector address. This register is valid when the PSW.EBV bit is 1.

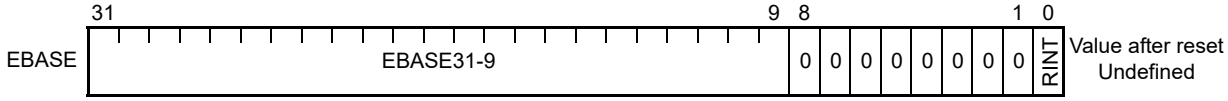


Table 3.24 EBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 9	EBASE31-9	The exception handler routine address is changed to the address resulting from adding the offset address of each exception to the base address specified for this register. The EBASE8 to EBASE0 bits are not assigned names because these bits are implicitly set to 0.	R/W	Undefined
8 to 1	—	(Reserved for future expansion. Always write 0.)	R	0
0	RINT	This bit specifies the setting of the offset address for direct branching method. This bit is valid when PSW.EBV = 0. 0: Offset address ($+100_H$ to $+1F0_H$) is determined according to interrupt priority order. 1: Offset address is fixed to $+100_H$. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.	R/W	Undefined

(s) INTBP — Base address register of the interrupt handler table

This register indicates the base address of the address table when the table reference method is selected as the interrupt handler address selection method.

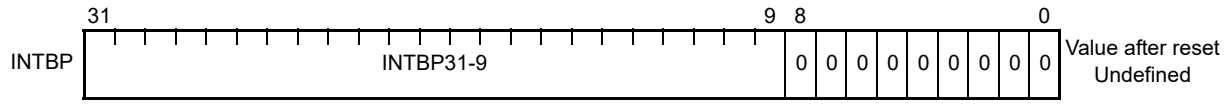


Table 3.25 INTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 9	INTBP31-9	These bits indicate the base pointer address for an interrupt prescribed in the expanded specifications. The value indicated by these bits is the first address in the table used to determine the exception handler when the interrupts (EIINT0 to EIINT255) prescribed by the expanded specifications is acknowledged. The INTBP8 to RBASE0 bits are not assigned names because these bits are implicitly set to 0.	R/W	Undefined
8 to 0	—	(Reserved for future expansion. Always write 0.)	R	0

(t) PID — Processor ID register

The PID register retains a processor identifier that is unique to the CPU. The PID register is a read-only register.

Table 3.26 PID Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 24	PID	Architecture Identifier Indicate the architecture of the processor.	R	05_H
23 to 8		Function Identifier Indicate the functions of the processor. These bits indicate whether or not functions defined per bit are implemented (1: implemented, 0: not implemented). Bits 23 to 11: Reserved Bit 10: Double-precision floating-point operation function Bit 9: Single-precision floating-point operation function Bit 8: Memory protection function (MPU)	R	8007_H
7 to 0		Version Identifier Indicate the version of the processor.	R	02_H

(u) SCCFG — SYSCALL operation setting register

This register is used to specify operations related to the SYSCALL instruction. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

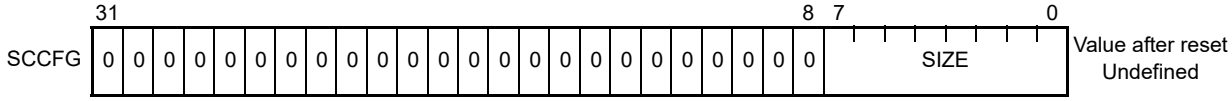


Table 3.27 SCCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 8	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
7 to 0	SIZE	These bits specify the maximum number of entries of a table that the SYSCALL instruction references. The maximum number of entries the SYSCALL instruction references is 1 if SIZE is 0, and 256 if SIZE is 255. By setting the maximum number of entries appropriately in accordance with the number of functions branched by the SYSCALL instruction, the memory area can be effectively used. If a vector exceeding the maximum number of entries is specified by the SYSCALL instruction, the first entry is selected. Place an error processing routine at the first entry.	R/W	Undefined

(v) SCBP — SYSCALL base pointer register

The SCBP register is used to specify a table address of the SYSCALL instruction and generate a target address. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

Be sure to set a word address to the SCBP register.

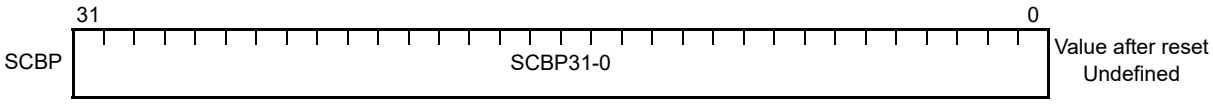


Table 3.28 SCBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	SCBP31-2	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction.	R/W	Undefined
1, 0	SCBP1-0	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction. Always set these bits to 0.	R	0

(w) MCFG0 — Machine configuration register

This register indicates the CPU configuration.

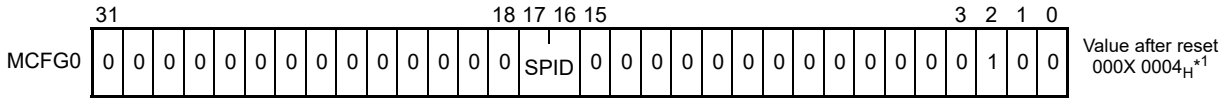


Table 3.29 MCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 18	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
17, 16	SPID	These bits indicate the system protection number.	R/W	CPU1 (PE1): 01_B CPU2 (PE2): 10_B
15 to 3	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
2	—	(Reserved for future expansion. This bit should be set to 1.)	R	1
1, 0	—	(Reserved for future expansion. These bits should be set to 1.)	R	0

Note 1. The value depends on the CPU that has access.

(x) MCTL — Machine control register

This register is used to control the CPU.

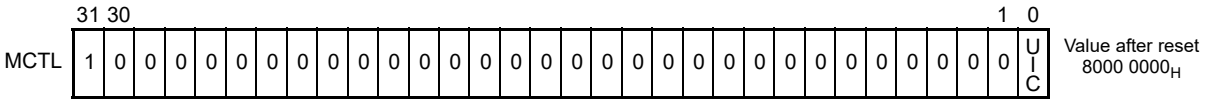


Table 3.30 MCTL Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31	—	(Reserved for future expansion. This bit should be set to 0.)	R	1
30 to 1	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
0	UIC	This bit is used to control the interrupt enable/disable operation in user mode. When this bit is set to 1, executing the EI/DI instruction become possible in user mode.	R/W	0

(3) Interrupt Function Registers

The interrupt function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.31 Interrupt Function System Registers

Register No. (regID, selID)	Symbol	Function	Access Permission
SR7, 1	FPIPR	FPI exception interrupt priority setting	SV
SR10, 2	ISPR	Priority of interrupt being serviced	SV
SR11, 2	PMR	Interrupt priority masking	SV
SR12, 2	ICSR	Interrupt control status	SV
SR13, 2	INTCFG	Interrupt function setting	SV

(a) FPIPR — FPI exception interrupt priority setting register

This register is used to set the interrupt priority of FPI exception.

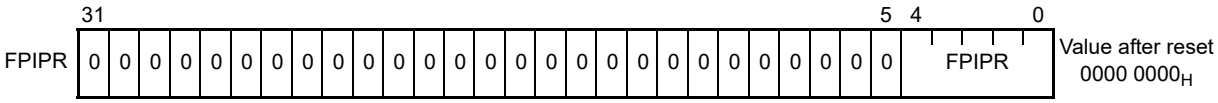


Table 3.32 FPIPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 5	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
4 to 0	FPIPR	These bits specify the priority of the floating-point operation exception interrupt (FPI, indicating imprecision). Values from 0 to 15 should be used; the setting for the values above 16 is prohibited. FPI exceptions are handled according to this interrupt priority, which is specified in advance. When generated at the same time as another interrupt with the same priority level, the FPI takes priority.	R/W	0

(b) ISPR — Priority of interrupt being serviced register

This register retains the priority of the EIINTn interrupt being serviced. This priority value is then used to perform priority ceiling processing when multiple interrupts are generated.

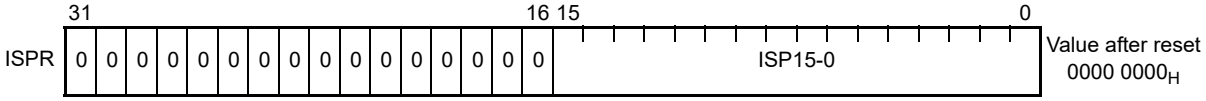


Table 3.33 ISPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 16	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
15 to 0	ISP15-0	These bits indicate the acknowledgment status of an EIINT*n* interrupt with a priority that corresponds to the relevant bit position. 0: An interrupt request for an interrupt whose priority corresponds to the relevant bit position has not been acknowledged. 1: An interrupt request for an interrupt whose priority corresponds to the relevant position is being serviced by the CPU core. The bit positions correspond to the following priority levels: **Bit** / **Priority** 0 / Priority 0 (highest) 1 / Priority 1 : / : 14 / Priority 14 15 / Priority 15 When an interrupt request (EIINT*n*) is acknowledged, the bit corresponding to the acknowledged interrupt request is automatically set to 1. If PSW.EP is 0 when the EIRET instruction is executed, the bit with the highest priority among the ISP15-0 bits that are set (0 is the highest priority) is cleared to 0.*1 While a bit in this register is set to 1, lower priority interrupts (EIINT*n*) and FPI exception*2 are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged. When performing software-based priority control using the PMR register, be sure to clear this register by using the INTCFG.ISPC bit.	R*3	0

Note 1. Interrupt acknowledgment and auto-updating of values when the EIRET instruction is executed are disabled by setting (1) to the INTCFG.ISPC bit. It is recommended to enable auto-updating of values, so in normal cases, the INTCFG.ISPC bit should be cleared to 0.

Note 2. Since FPI exceptions have the same level of priority as this interrupt (EIINT*n*), they are affected by interrupts in the same way as the ISPR. The priority of the FPI exception is set by the FPIPR register.

Note 3. This is R or R/W, depending on the setting of the INTCFG.ISPC bit. It is recommended to use this register as a read-only (R) register.

(c) PMR — Interrupt priority masking register

This register is used to mask the specified interrupt priority.

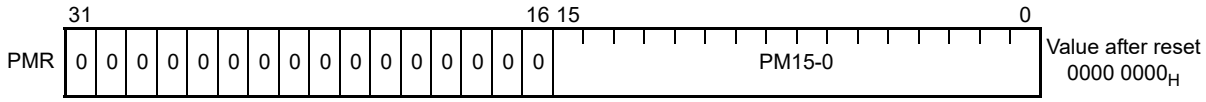


Table 3.34 PMR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 16	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
15 to 0	PM15-0	These bits mask an interrupt request with a priority level that corresponds to the relevant bit position. 0: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is enabled. 1: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is disabled. The bit positions correspond to the following priority levels: **Bit** / **Priority** 0 / Priority 0 (highest) 1 / Priority 1 : / : 14 / Priority 14 15 / Priority 15 (lowest) While a bit in this register is set to 1, interrupts (EIINT*n*) and FPI exceptions*1with the priority corresponding to that bit are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged*2.	R/W	0

Note 1. Since FPI exceptions are specified as the same level of priority as that of interrupts (EIINT*n*), it is affected by the PMR like interrupts. The priority of FPI exceptions is set by the FPIPR register.

Note 2. Specify the masks by setting the bits to 1 in order from the lowest-priority bit. For example, $FF00_H$ can be set, but $F0F0_H$ or $00FF_H$ cannot.

(d) ICSR — Interrupt control status register

This register indicates the interrupt control status in the CPU.

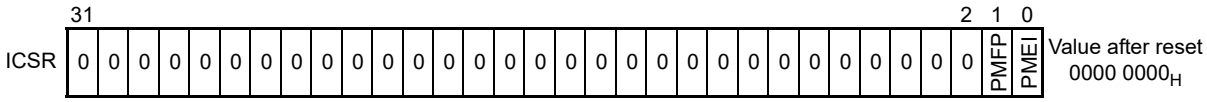


Table 3.35 ICSR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
1	PMFP	This bit indicates that an FPI exception with the priority level masked by the PMR register exists. 0: FPI exception does not exist. 1: FPI exception exists.	R	0
0	PMEI	This bit indicates that an interrupt (EIINT*n*) with the priority level masked by the PMR register exists. 0: Interrupt (EIINT*n*) does not exist. 1: Interrupt (EIINT*n*) exists.	R	0

(e) INTCFG — Interrupt function setting register

This register is used to specify settings related to the CPU's internal interrupt function.

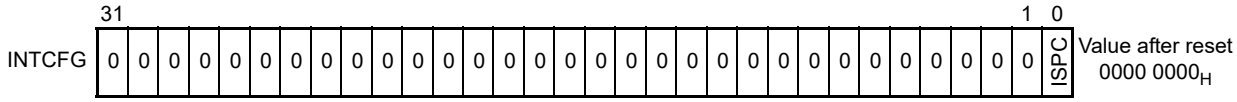


Table 3.36 INTCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 1	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
0	ISPC	This bit changes how the ISPR register is written. 0: The ISPR register is automatically updated. Updates triggered by the program (via execution of an LDSR instruction) are ignored. 1: The ISPR register is not automatically updated. Updates triggered by the program (via execution of an LDSR instruction) are performed. If this bit is cleared to 0, the bits of the ISPR register are automatically set to 1 when an interrupt (EIINT*n*) is acknowledged, and cleared to 0 when the EIRET instruction is executed. In this case, the bits are not updated by an LDSR instruction executed by the program. If this bit is set to 1, the bits of the ISPR register are not updated by the acknowledgement of an interrupt (EIINT*n*) or by execution of the EIRET instruction. In this case, the bits can be updated by an LDSR instruction executed by the program. In normal cases, the ISPC bit should be cleared. When performing software-based priority control, however, set this bit to 1 and perform priority control by using the PMR register.	R/W	0

(4) FPU Function Registers

The FPU can use the following system registers to control floating-point operation. The FPU function registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID. For details of the registers, see *section 3.4.2, Floating-point system registers* in the RH850G3M Family User's Manual: Software.

Table 3.37 FPU System Registers

Register No. (reg ID, sel ID)	Symbol	Function	Access Permission
SR6, 0	FPSR	Floating-point operation setting/status	CU0 and SV
SR7, 0	FPEPC	Floating-point operation exception program counter	CU0 and SV
SR8, 0	FPST	Floating-point status	CU0
SR9, 0	FPCC	Floating-point operation comparison result	CU0
SR10, 0	FPCFG	Floating-point function setting	CU0
SR11, 0	FPEC	Floating-point operation exception control	CU0 and SV

(5) MPU function registers

The MPU function registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.38 MPU Function System Registers (1/2)

Register No. (regID, selID)	Symbol	Function	Access Permission
SR0, 5	MPM	Memory protection operation mode setting	SV
SR1, 5	MPRC	MPU region control	SV
SR4, 5	MPBRGN	MPU base region number	SV
SR5, 5	MPTRGN	MPU end region number	SV
SR8, 5	MCA	Memory protection setting check address	SV
SR9, 5	MCS	Memory protection setting check size	SV
SR10, 5	MCC	Memory protection setting check command	SV
SR11, 5	MCR	Memory protection setting check result	SV
SR0, 6	MPLA0	Protection area lower limit address	SV
SR1, 6	MPUA0	Protection area upper limit address	SV
SR2, 6	MPAT0	Protection area attribute	SV
SR4, 6	MPLA1	Protection area lower limit address	SV
SR5, 6	MPUA1	Protection area upper limit address	SV
SR6, 6	MPAT1	Protection area attribute	SV
SR8, 6	MPLA2	Protection area lower limit address	SV
SR9, 6	MPUA2	Protection area upper limit address	SV
SR10, 6	MPAT2	Protection area attribute	SV
SR12, 6	MPLA3	Protection area lower limit address	SV
SR13, 6	MPUA3	Protection area upper limit address	SV
SR14, 6	MPAT3	Protection area attribute	SV
SR16, 6	MPLA4	Protection area lower limit address	SV
SR17, 6	MPUA4	Protection area upper limit address	SV

Table 3.38 MPU Function System Registers (2/2)

Register No. (regID, selID)	Symbol	Function	Access Permission
SR18, 6	MPAT4	Protection area attribute	SV
SR20, 6	MPLA5	Protection area lower limit address	SV
SR21, 6	MPUA5	Protection area upper limit address	SV
SR22, 6	MPAT5	Protection area attribute	SV
SR24, 6	MPLA6	Protection area lower limit address	SV
SR25, 6	MPUA6	Protection area upper limit address	SV
SR26, 6	MPAT6	Protection area attribute	SV
SR28, 6	MPLA7	Protection area lower limit address	SV
SR29, 6	MPUA7	Protection area upper limit address	SV
SR30, 6	MPAT7	Protection area attribute	SV
SR0, 7	MPLA8	Protection area lower limit address	SV
SR1, 7	MPUA8	Protection area upper limit address	SV
SR2, 7	MPAT8	Protection area attribute	SV
SR4, 7	MPLA9	Protection area lower limit address	SV
SR5, 7	MPUA9	Protection area upper limit address	SV
SR6, 7	MPAT9	Protection area attribute	SV
SR8, 7	MPLA10	Protection area lower limit address	SV
SR9, 7	MPUA10	Protection area upper limit address	SV
SR10, 7	MPAT10	Protection area attribute	SV
SR12, 7	MPLA11	Protection area lower limit address	SV
SR13, 7	MPUA11	Protection area upper limit address	SV
SR14, 7	MPAT11	Protection area attribute	SV

(a) MPM — Memory protection operation mode register

The memory protection mode register is used to define the basic operating state of the memory protection function.

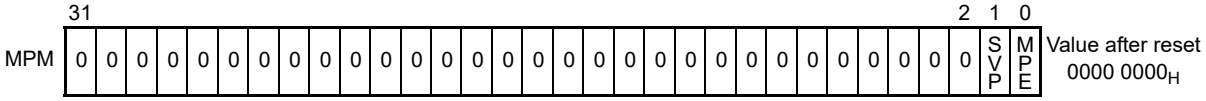


Table 3.39 MPM Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
1	SVP	In SV mode (when PSW.UM = 0), this bit is used to specify whether to restrict access according to the SX, SW, and SR bits of the MPAT register for each protection area.*1 0: As usual, implicitly enable all access in SV mode. 1: Restrict access according to the SX, SW, and SR bits even in SV mode.*2	R/W	0
0	MPE	This bit is used to specify whether to enable or disable the MPU function. 0: Disable 1: Enable	R/W	0

Note 1. When the SVP bit is set to 1, access will become restricted after a certain number of clock cycles, even in SV mode. Therefore, specify the protection area before setting the SVP bit to prevent restriction of access by the program itself.

Note 2. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(b) MPRC — MPU region control register

Bits used to perform special memory protection function operations are located in this register.

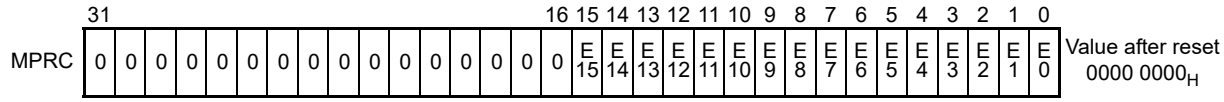


Table 3.40 MPRC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 16	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
15 to 12	E15-E12	Reserved for future expansion. These bits should be set to 0.	R	0
11 to 0	E11-E0	These are the enable bits for each protection area. Bit En is a copy of bit MPATn.E (where n = 11 to 0).	R/W	0

(c) MPBRGN — MPU base region register

This register indicates the minimum usable MPU area number.

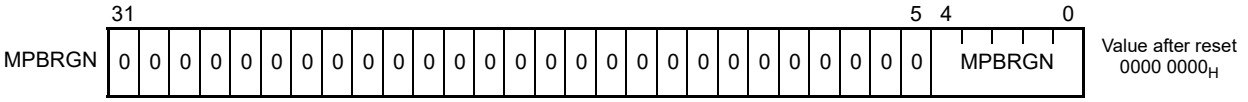


Table 3.41 MPBRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 5	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
4 to 0	MPBRGN	These bits indicate the smallest number of an MPU area. These bits are always read as 0.	R	0

(d) MPTRGN — MPU end region register

This register indicates the maximum usable MPU area number + 1.

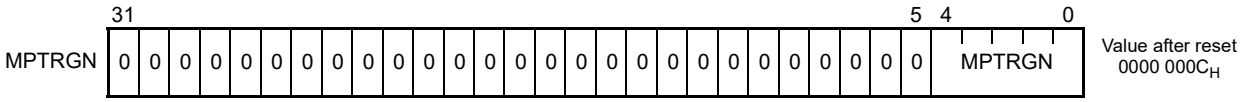


Table 3.42 MPTRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 5	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
4 to 0	MPTRGN	These bits indicate the largest number of an MPU area plus one. These bits always indicate the maximum number of MPU areas that the hardware can support.	R	0C$_H$

(e) MCA — Memory protection setting check address register

This register is used to specify the base address of the area for which a memory protection setting check is to be performed.

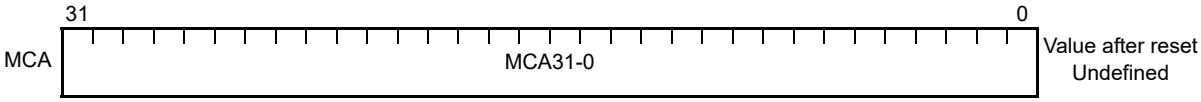


Table 3.43 MCA Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	MCA31-0	These bits are used to specify the start address of the memory area subject to a memory protection setting check in bytes.	R/W	Undefined

(f) MCS — Memory protection setting check size register

This register is used to specify the size of the area for which a memory protection setting check is to be performed.

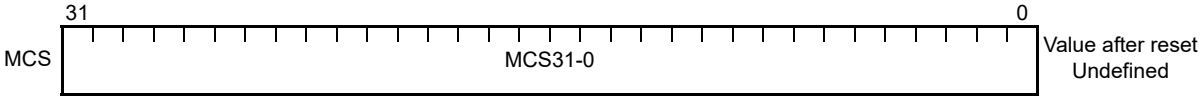


Table 3.44 MCS Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	MCS31-0	These bits specify the size of the target area to specify the size of memory area subject to a memory protection setting check in bytes. Checking in areas below the address value in the MCS register is not possible because the specified size is handled as an unsigned integer. Do not set 0000 0000$_H$ in the MCS register.	R/W	Undefined

(g) MCC — Memory protection setting check command register

This command register is used to start a memory protection setting check.

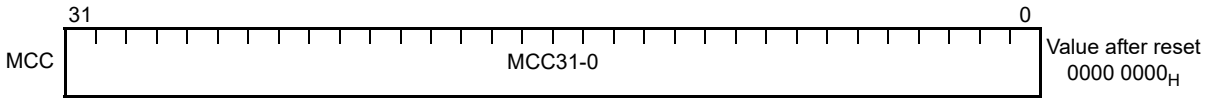


Table 3.45 MCC Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	MCC31-0	When any value is written to the MCC register, a memory protection setting check starts. Setting the MCA and MCS registers and then writing to this register leads to storage of the result of checking in the MCR register. Since writing any value to this register starts the check, doing so does not require any extra registers when r0 is used as a source register. The result of checking is reflected in MCR according to any area setting regardless of the setting of the PSW.UM bit. The value read from the MCC register is always 0000 0000_H.	R/W	0

(h) MCR — Memory protection setting check result register

This register is used to store the results of a memory protection setting check.

Be sure to clear bits 31 to 9, 7 and 6 to 0.

CAUTION

If the area for which checking is specified crosses 0000 0000_H, the area is judged to have been specified incorrectly, and the MCR.OV bit is set to 1. For this reason, in access to the results of checking, check the MCR.OV bit and confirm that the result is valid (OV = 0) before using any results of checking.

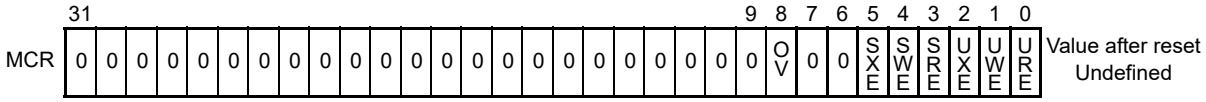


Table 3.46 MCR Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 9	—	(Reserved for future expansion. Always write 0.)	R	0
8	OV	If the specified area includes 0000 0000_H or 7FFF $FFFF_H$, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
7, 6	—	(Reserved for future expansion. Always write 0.)	R	0
5	SXE	If the specified area is contained within one protection area and execution is permitted for that area in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
4	SWE	If the specified area is contained within one protection area and writing to that area is permitted in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
3	SRE	If the specified area is contained within one protection area and reading from that area is permitted in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined

Table 3.46 MCR Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after reset
2	UXE	If the specified area is contained within one protection area and execution is permitted for that area in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
1	UWE	If the specified area is contained within one protection area and writing to that area is permitted in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
0	URE	If the specified area is contained within one protection area and reading from that area is permitted in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined

(i) MPLAn — Protection area lower limit address register

These registers indicate the lower limit address of area n (where n = 0 to 11).

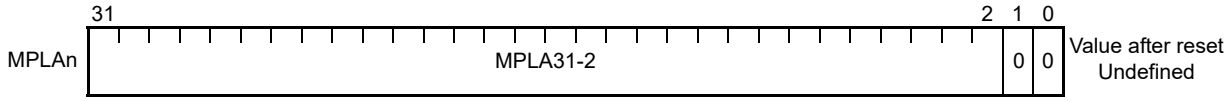


Table 3.47 MPLAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	MPLA31-2	These bits indicate the lower limit address of area n. The MPLA1 and MPLA0 bits are not assigned names because these bits are implicitly set to 0.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

(j) MPUAn — Protection area upper limit address register

These registers indicate the upper limit address of area n (where n = 0 to 11).

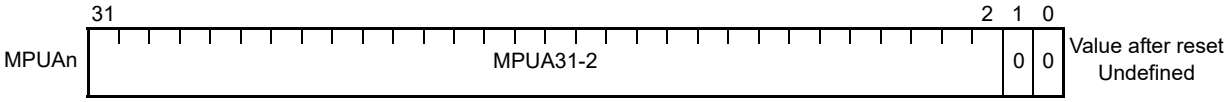


Table 3.48 MPUAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	MPUA31-2	These bits indicate the upper limit address of area n. The MPUA1 and MPUA0 bits are not assigned names because these bits are implicitly set to 1.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

(k) MPATn — Protection area attribute register

These registers indicate the attributes of area n (where n = 0 to 11).

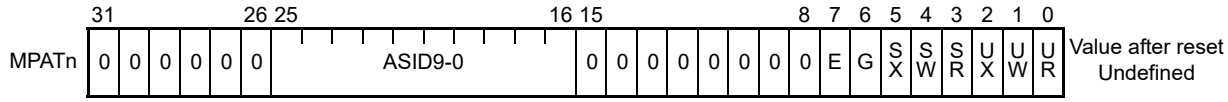


Table 3.49 MPATn Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 26	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
25 to 16	ASID9-0	These bits indicate the ASID value to be used as the area match condition.	R/W	Undefined
15 to 8	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
7	E	This bit indicates whether area n is enabled or disabled. 0: Area n is disabled. 1: Area n is enabled.	R/W	0
6	G	0: ASID match is the condition. 1: ASID match is not the condition. When this bit is 0, the condition of the area match is MPATn.ASID = ASID.ASID. When this bit is 1, the area match of the values of MPATn.ASID and ASID.ASID is not the condition.	R/W	Undefined
5	SX	This bit indicates the execution privilege in supervisor mode.*1 0: Execution is disabled. 1: Execution is enabled.	R/W	Undefined
4	SW	This bit indicates the write permission in supervisor mode.*1 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
3	SR	This bit indicates the read permission in supervisor mode.*1 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined
2	UX	This bit indicates the execution privilege in user mode. 0: Execution is disabled. 1: Execution is enabled.	R/W	Undefined
1	UW	This bit indicates the write permission in user mode. 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
0	UR	This bit indicates the read permission in user mode. 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined

Note 1. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(6) Cache Operation Function Registers

The cache operation function are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.50 Cache Operation Function Registers

Register No. (regID, selID)	Symbol	Function	Access Permission
SR16, 4	ICTAGL	Instruction cache tag Lo access	SV
SR17, 4	ICTAGH	Instruction cache tag Hi access	SV
SR18, 4	ICDATL	Instruction cache data Lo access	SV
SR19, 4	ICDATH	Instruction cache data Hi access	SV
SR24, 4	ICCTRL	Instruction cache control	SV
SR26, 4	ICCFG	Instruction cache configuration	SV
SR28, 4	ICERR	Instruction cache error	SV

(a) ICTAGL — Instruction cache tag Lo access register

This register is used for CIST and CILD instructions for the instruction cache. This register retains values to be stored in the tag RAM of the instruction cache by the execution of CIST instructions and values read from the tag RAM of the instruction cache by the execution of CILD instructions.

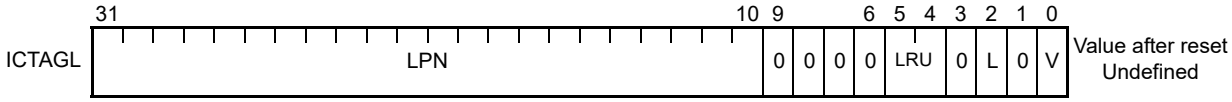


Table 3.51 ICTAGL Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 10	LPN	These bits retain the values of bits 24 to 11, i.e. the physical page numbers. When writing, always write 0 to bits 31 to 25 and 10.	R/W	Undefined
9 to 6	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
5, 4	LRU	These bits indicate the LRU information of the specified cache line. The CIST instruction cannot be used to change the LRU information to desired values.	R/W	Undefined
3	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
2	L	This bit retains the lock information.	R/W	Undefined
1	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
0	V	This bit retains whether the specified cache line is enabled or disabled.	R/W	Undefined

(b) ICTAGH — Instruction cache tag Hi access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache tag RAM by the execution of CIST instruction and the value read from the instruction cache tag RAM by the execution of CILD instruction.

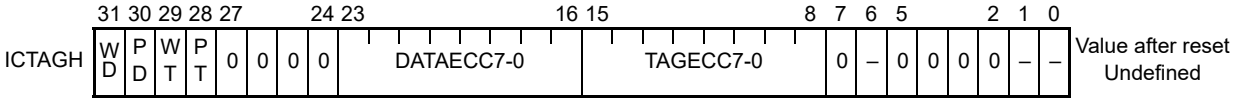


Table 3.52 ICTAGH Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31	WD	When this bit is set to 1, executing a CIST instruction updates the cache data RAM.	R/W	Undefined
30	PD	When this bit is set to 1, the value in the DATAECC field of this register is written to the ECC field of the data RAM on execution of a CIST instruction. When this bit is cleared to 0, the ECC is automatically generated from the written data.	R/W	Undefined
29	WT	When this bit is set to 1, executing a CIST instruction updates the cache data RAM.	R/W	Undefined
28	PT	When this bit is set to 1, the value in the TAGECC field is written to the ECC of the tag RAM on execution of a CIST execution. When this bit is cleared to 0, the ECC is automatically generated from the written data.	R/W	Undefined
27 to 24	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
23 to 16	DATAECC7 to DATAECC0	These bits retain ECC of the data RAM.	R/W	Undefined
15 to 8	TAGECC7 to TAGECC0	These bits retain ECC of the tag RAM. Be sure to clear bit 15 to 0.	R/W	Undefined
7	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
6	—	(Reserved for future expansion. This bit should be set to 0.)	R	Undefined
5 to 2	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
1, 0	—	(Reserved for future expansion. This bit should be set to 0.)	R	Undefined

(c) ICDATL — Instruction cache data Lo access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM by the execution of CIST instruction and the value read from the instruction cache data RAM by the execution of CILD instruction.

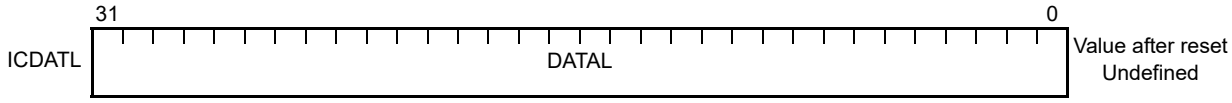


Table 3.53 ICDATL Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	DATAL	These bits retain the values of bits 31 to 0 or of bits 95 to 64 from the instruction data of the block within the specified cache line. The offset of the index specifies the target range of bit numbers. Index offset = 0000: bits 31 to 0 Index offset = 1000: bits 95 to 64	R/W	Undefined

(d) ICDATH — Instruction cache data Hi access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM by the execution of CIST instruction and the value read from the instruction cache data RAM by the execution of CILD instruction.

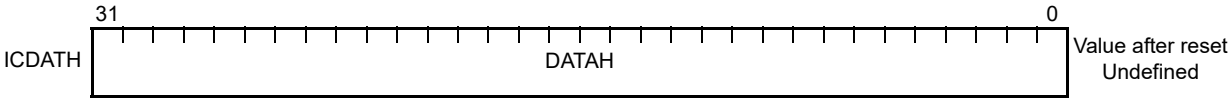


Table 3.54 ICDATH Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 0	DATAH	These bits retain the values of bit 63 to 32 or those of bit 127 to 96 from the instruction data of the block within the specified cache line. The index offset specifies the bit number to be retained. Index offset = 0000: bits 63 to 32 Index offset = 1000: bits 127 to 96	R/W	Undefined

(e) ICCTRL — Instruction cache control register

This register controls the instruction cache.

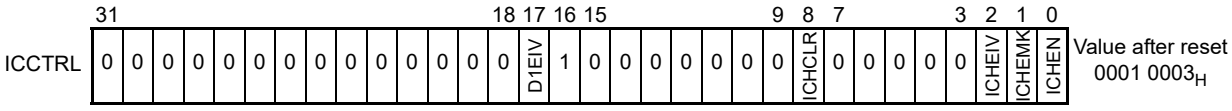


Table 3.55 ICCTRL Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 18	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
17	D1EIV	This bit selects the operation in response to 1-bit errors in the data RAM. 0: The error is corrected and then processing continues, but the address of the entry that had an error is retained. 1: The error is not corrected, the entry is cleared, and fetching is repeated. This bit is read as the previous value until the setting is actually reflected in the instruction cache.	R/W	0
16	—	(Reserved for future expansion. Be sure to set to 1.)	R	1
15 to 9	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
8	ICHCLR	Setting this bit to 1 selects clearing of the whole instruction cache in a single operation. After this bit has been set to 1, it will be read as 1 until clearing is completed. The bit is cleared to 0 once clearing of the cache is completed.	R/W	0
7 to 3	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
2	ICHEIV	Setting this bit to 1 allows the instruction cache to be automatically disabled (ICHEN bit to be cleared to 0) when a cache error occurs.	R/W	0
1	ICHEMK	Setting this bit to 1 selects masking of cache error exception notifications for the CPU when a cache error occurs.	R/W	1
0	ICHEN	This bit disables or enables the instruction cache. 0: Instruction cache is disabled. 1: Instruction cache is enabled. This bit is read as the previous value until the setting is actually reflected in the instruction cache.	R/W	1

(f) ICCFG — Instruction cache configuration register

This register shows the configuration of the instruction cache.

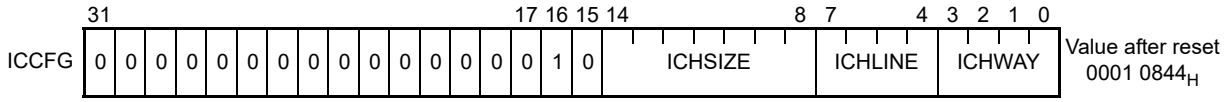


Table 3.56 ICCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 17	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
16	—	(Reserved for future expansion. Be sure to set to 1.)	R	1
15	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
14 to 8	ICHSIZE	These bits indicate the capacity (in Kbytes) of the instruction cache. 000 1000: 8 Kbytes	R	08_H
7 to 4	ICHLINE	These bits indicate the number of lines per 1 way in the instruction cache. 0100: 128 lines	R	4_H
3 to 0	ICHWAY	These bits indicate the number of ways of the instruction cache. 0100: 4 ways	R	4_H

(g) ICERR — Instruction cache error register

This register stores cache error data of the instruction cache.

Once the ICHERR bit is set to 1, subsequent cache error data is not stored in this register until the ICHERR bit is cleared to 0.

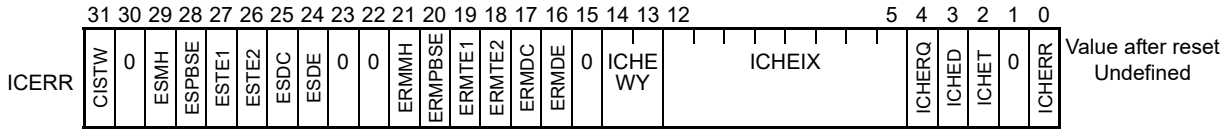


Table 3.57 ICERR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31	CISTW	This bit is set to indicate that the destination way specified for a CISTI instruction was in error. Although the entry information is overwritten so that writing is completed, the V bit will be cleared the next time the cache line is read (i.e. reading will be judged to have missed the cache). However, setting of this bit is not accompanied by an exception for the CPU.	R/W	0
30	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
29	ESMH	Error status: Multi-hit	R/W	Undefined
28	ESPBSE	Error status: WAY error	R/W	Undefined
27	ESTE1	Error status: 1-bit error in the tag RAM	R/W	Undefined
26	ESTE2	Error status: 2-bit error in the tag RAM	R/W	Undefined
25	ESDC	Error status: 1-bit correction in the data RAM	R/W	Undefined
24	ESDE	Error status: 2-bit error in the data RAM	R/W	Undefined
23, 22	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
21	ERMMH	Error exception notification mask: Multi-hit	R/W	0
20	ERMPBSE	Error exception notification mask: WAY error	R/W	0
19	ERMTE1	Error exception notification mask: 1-bit error in the tag RAM	R/W	0
18	ERMTE2	Error exception notification mask: 2-bit error in the tag RAM	R/W	0
17	ERMDC	Error exception notification mask: 1-bit correction in the data RAM	R/W	0
16	ERMDE	Error exception notification mask: 2-bit error in the data RAM	R/W	0
15	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
14, 13	ICHEWY	These bits retain a way number where a cache error occurs.	R/W	Undefined
12 to 5	ICHEIX	These bits retain a cache index where a cache error occurs.	R/W	Undefined
4	ICHERQ	Setting of this bit to 1 indicates that the CPU is being notified of a cache error exception. If cache error exceptions are masked, however, the CPU is not notified of an exception even when this bit is set to 1.	R/W	0
3	ICHED	This bit indicates that an error occurs in the data RAM.	R/W	0
2	ICHET	This bit indicates that an error occurs in the tag RAM.	R/W	0
1	—	(Reserved for future expansion. This bit should be set to 0.)	R	0
0	ICHERR	This bit is set to 1 when a cache error occurs.	R/W	0

(7) Data Buffer Operation Function Registers

The data buffer control registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.58 Data Buffer Operation Function Registers

Register No. (regID, selID)	Symbol	Function	Access Permission
SR24,13	CDBCR	Data buffer control register	SV

(a) CDBCR — Data buffer control register

This register controls the data buffer.

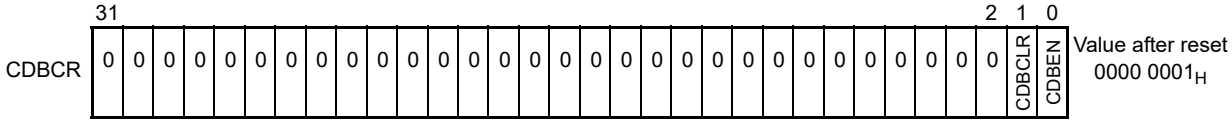


Table 3.59 CDBCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after reset
31 to 2	—	(Reserved for future expansion. These bits should be set to 0.)	R	0
1	CDBCLR	When this bit is set to 1, the data buffer is cleared at a time. This bit is always read as 0.	W	0
0	CDBEN	This bit disables or enables the data buffer. 0: Data buffer is disabled. 1: Data buffer is enabled.	R/W	1

3.2.2 Instruction Cache and Data Buffer

3.2.2.1 Features

An 8-Kbyte and 4-way set-associative instruction cache is mounted between the CPU and the code flash. The instruction cache and the code flash are connected to each other via a 128-bit dedicated bus to minimize penalties caused by a cache mis-hit. Also a data buffer is mounted between the CPU and the code flash to achieve high-speed data access. The 32-MB area from 0000 0000_H to 01FF $FFFF_H$ in the address space is intended for the instruction cache and data buffer.

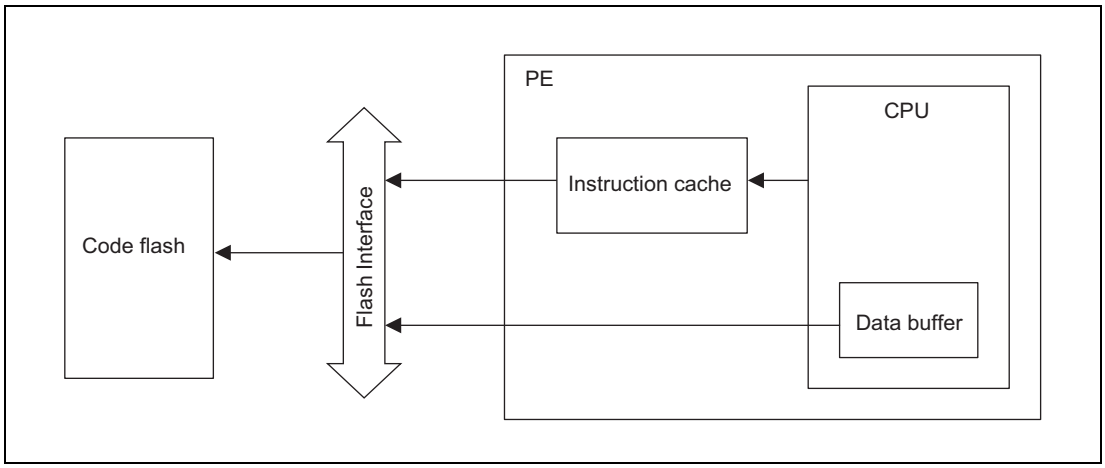


Figure 3.2 Instruction Cache and Data Buffer

3.2.2.2 Instruction Cache Function

The 8-Kbyte and 4-way set-associative cache includes four Ways consisting of 128-entry blocks of four words per line, amounting to 8-Kbyte capacity in total. The Ways are divided into two groups, Way Group 0 consisting of Way0 and Way1 and Way Group 1 consisting of Way2 and Way3. The Way Group can be selected and used by decoding of the address information of the access destination. If a cache error occurs, each line is refilled by a replace algorithm using LRU. The cache destination is the instruction fetch access to the code flash area.

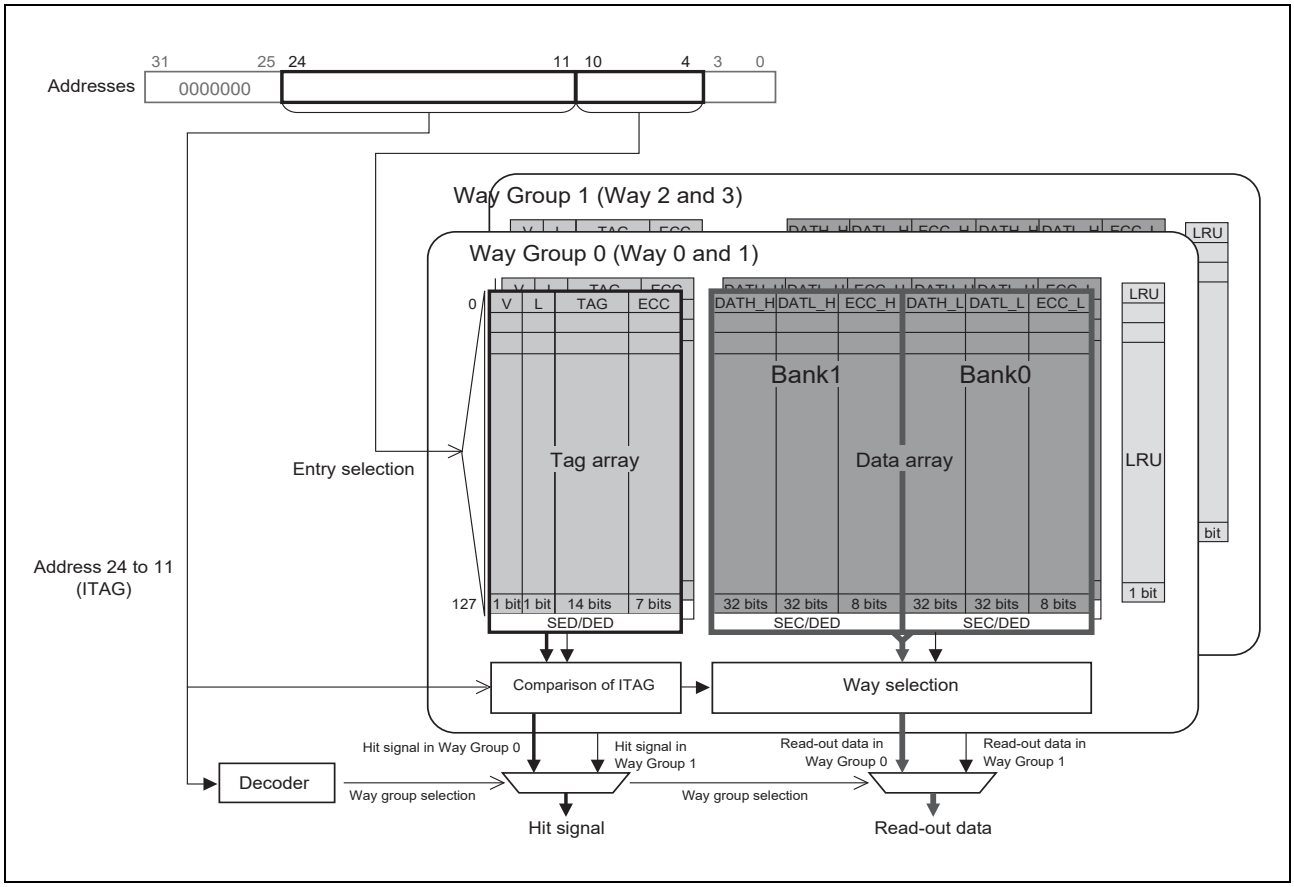


Figure 3.3 Instruction Cache Configuration

Tag Array

V bit	This bit indicates whether valid data is stored in the cache line. Setting of this bit to 1 makes the cache line data valid. The V bit is initialized to 0 by reset.
L bit	This bit indicates whether a cache line is locked or not. Setting of this bit to 1 locks the cache line and it cannot be replaced with new data. The L bit is valid only when the V bit is 1, and it is not initialized by reset.
TAG	Among 32 bits in the operable addresses of the data line to be cached, bits 24 to 11 are stored in this bit. The TAG bit is not initialized by reset.
ECC	The ECC of the tag array is stored in this bit. The ECC bit is not initialized by reset.

Data Array

DATH_H, DATH_L, DATL_H, DATL_L	The 128-bit cache line data is stored per 32 bits as follows: bits [127:96], [95:64], [63:32], and [31:0] are stored in DATH_H, DATL_H, DATH_L, and DATL_L, respectively. In the CIST or CILD operation in response to a cache instruction, the ICDATH register is used for DATH_H and DATH_L and the ICDATL register is used for DATL_H and DATL_L.
ECC_H, ECC_L	The ECC of the data in bits [127:64] and [63:0] are stored in ECC_H and ECC_L, respectively.

LRU

LRU	The LRU information in the same Way Group is stored in this data array. The LRU is initialized by reset.

CAUTIONS

When an instruction is fetched from an applicable line after issuing a CIST instruction for writing test data to the tag array of the instruction cache, the tag information must be written in a group unit. For example, when writing tag information to a line on the Way 0 side of Way group 0, also write tag information for the same line to Way 1, and then execute the instruction fetch.

- **When writing to Way group 0 (Way 0 and Way 1), write a value such that the exclusive OR of the ICTAGL.LPN bits is 0.**
- **When writing to Way Group 1 (Way 2 and Way 3), write a value such that the exclusive OR of the ICTAGL.LPN bits is 1.**

Fetching an instruction after a value that does not follow the above rule has been written to the tag array causes a Way error and setting of the ICERR.ESPBSE bit to 1. Fetching an instruction after the same tag information has been written to the same lines of both Ways in a Way group causes a multi-hit error and setting of the ICERR.ESMH bit to 1.

3.2.2.3 Data Buffer Function

The four-line buffer with 128 bits per line is mounted as a data buffer. The data of 128 bits per line read from the code flash is stored in the data buffer. The data is read out from the data buffer after the next access to the same address, so the code flash is not accessed again.

3.2.3 Inter-Processor Interrupts

Registers (IPIR_CHn) for interrupt communication between PEs are provided for four channels.

IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for specific PEs (including own PE) can be requested by manipulating bits corresponding to respective PEs.

3.2.3.1 Inter-Processor Interrupt Control Registers

These registers are located in the CPU peripheral of each PE. Each PE has the IPIR_CH0 to IPIR_CH3 registers and cannot access the registers in other PEs.

Table 3.60 List of Registers

Module Name	Register Name	Symbol	Address
IPIRSS	Inter-PE interrupt register 0	IPIR_CH0	FFFE EC80$_H$
IPIRSS	Inter-PE interrupt register 1	IPIR_CH1	FFFE EC84$_H$
IPIRSS	Inter-PE interrupt register 2	IPIR_CH2	FFFE EC88$_H$
IPIRSS	Inter-PE interrupt register 3	IPIR_CH3	FFFE EC8C$_H$

(1) IPIR_CHn — Inter-PE interrupt register n (n = 0 to 3)

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PE2	PE1
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 3.61 IPIR_CHn Interrupt Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should always be 0.
1	PE2	Inter-PE Interrupt Request to PE2 Writing 1 to this bit makes an interrupt request to PE2. This bit is automatically cleared to 0 when the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.
0	PE1	Inter-PE Interrupt Request to PE1 Writing 1 to this bit makes an interrupt request to PE1. This bit is automatically cleared to 0 when the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.

3.2.4 Reliability Functions

3.2.4.1 PE Guard Function (PEG)

(1) Overview of the PEG Function

The PEG is a constituent of the Slave Guard function and prevents unauthorized accesses to the resources in the PE from the bus master. This function protects access to the local RAM in the PE. For accesses to the register set in PE, no access restriction is provided independently for the PEG function. Set access protections such as IPG at the user side as necessary. In the initial state after a reset, all access by the bus masters other than the PE itself is disabled. Setting the registers listed in **(3), List of PEG Setting Registers**, enables access by the bus masters other than the PE itself.

(1) Detecting PE guard violation

If the bus master outside the PE makes an unauthorized access to the resource area in the PE for which PE guard is set, the access is detected as a PE guard violation.

(2) Blocking unauthorized accesses

When a PE guard violation is detected, unauthorized accesses to the internal resources of the PE are blocked to prevent the contents of PE resources from being modified illegally.

(3) Notifying occurrence of violation

When a PE guard violation is detected, it is notified to ECM. When the DMAC or DTS makes an unauthorized access, meanwhile, a DMA transfer error is detected.

(2) Protection Made by SPID

- Setting PEG
  - Up to four areas can be set depending on the local RAM address of the own PE.
  - The area range is specified by the base address and the mask bit (4 Kbytes to 4 Gbytes).
  - “Read enable” and “write enable” can be set for each area.
  - “Enable” or “disable” can be selected on each system protection identifier (SPID) basis for each area.
- Access permission by the system protection identifier (SPID) (see **Figure 3.4**)

1. When the local RAM area is to be accessed, go to step 2.
   Otherwise, return an error response.
2. When any of enabled areas 0 to 3 is to be accessed, go to step 3.
   Otherwise, return an error response.
3. Are all the conditions below for the relevant area met?
   - The system protection identifier (SPID) is enabled.
   - Required operations (read/write) are enabled.

   Otherwise, return an error response.

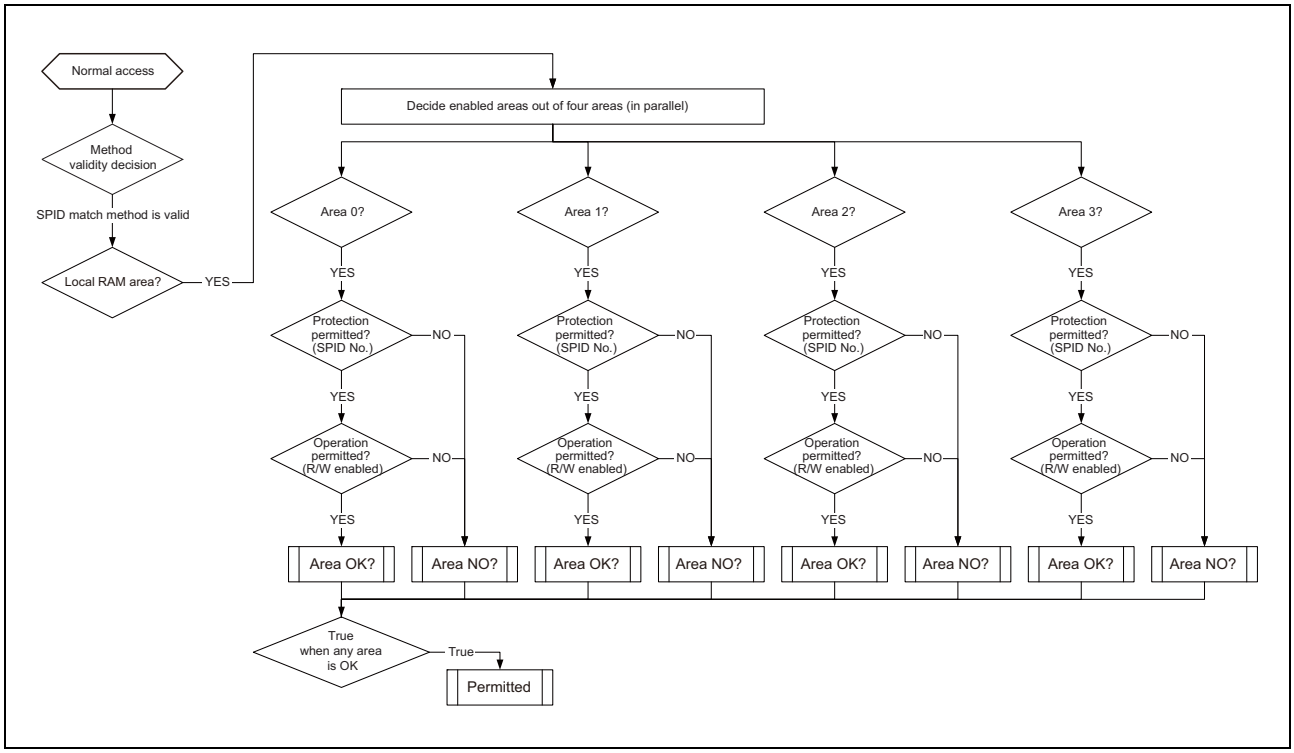


Figure 3.4 Access Permission by the System Protection Identifier (SPID)

(3) List of PEG Setting Registers

Make necessary settings for the following registers to PE resources from unauthorized accesses by the bus master.

- PEG can be set by following the procedure below.

1. Set the PE guard area n mask setting register as PEGGnMK.
2. Set the PE guard area n base setting register (n = 0 to 3) as PEGGnBA.

Table 3.62 List of Registers

Module Name	Register Name	Symbol	Address
PEG	PE guard SPID master decision control register	PEGSP	FFFE E60C$_H$
PEG	PE guard area 0 mask setting register	PEGG0MK	FFFE E680$_H$
PEG	PE guard area 0 base setting register	PEGG0BA	FFFE E684$_H$
PEG	PE guard area 1 mask setting register	PEGG1MK	FFFE E690$_H$
PEG	PE guard area 1 base setting register	PEGG1BA	FFFE E694$_H$
PEG	PE guard area 2 mask setting register	PEGG2MK	FFFE E6A0$_H$
PEG	PE guard area 2 base setting register	PEGG2BA	FFFE E6A4$_H$
PEG	PE guard area 3 mask setting register	PEGG3MK	FFFE E6B0$_H$
PEG	PE guard area 3 base setting register	PEGG3BA	FFFE E6B4$_H$

(4) Register Set

(a) PEGSP — PE guard SPID master decision control register

This register is used to enable or disable access by the bus master to the resources in the PE. The value after reset of the SPEN bit is 0, which disables access to the PE resources by the bus master. Setting the SPEN bit to 1 enables access by the bus master under the conditions set by PEGGnMK and PEGGnBA.

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

Table 3.63 PEGSP Register Contents

Bit Position	Bit Name	Function
15 to 1	—	Reserved. These bits are always read as 0. The write value should always be 0.
0	SPEN	This bit enables or disables access settings by the bus master having SPID. 0: Access settings by the bus master having SPID is disabled and access by SPID is disabled. 1: Access settings by the bus master having SPID is enabled and access by SPID is enabled.

(b) PEGGnMK — PE guard area n mask setting register

In combination with the PEGGnBA register, this register specifies a range or ranges within PE guard area n. Setting a GnMASK bit to 1 masks the corresponding address bit of the PEGGnBA register and places the corresponding area or areas within the range of PE guard area n. Note that the minimum setting unit for PE guard area n is 4 Kbytes.

Expression for access valid address
(PEGGnMK[31:12] | addr[31:12])==(PEGGnMK[31:12] | PEGGnBA[31:12])

Example: When PEGGnBA[31:12] = FEBF6H and PEGGnMK[31:12] = 00008_H are set, PE guard areas n are FEBF 6000_H to FEBF $6FFF_H$ and FEBF $E000_H$ to FEBF$EFFF_H$

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

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

Table 3.64 PEGGnMK Register Contents

Bit Position	Bit Name	Function
31 to 12	GnMASK	These bits determine whether to mask the base address bits PEGGnBA[31:12] that specify the range of PE guard area n. 0: Target address bits are compared when determining the PE guard area. 1: Target address bits are not compared when determining the PE guard area.
11 to 0	—	Reserved. These bits are always read as 0. The write value should always be 0.

(c) PEGGnBA — PE guard area n base setting register (n = 0 to 3)

In combination with the PEGGnMK register, this register specifies a range or ranges within PE guard area n and sets the access enable conditions for the specified area. Setting the GnEN bit to 1 brings the address enable conditions specified by this register and the PEGGnMK register into effect.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GnBASE				—	—	—	—	GnSP3	GnSP2	GnSP1	GnSP0	—	GnWR	GnRD	GnEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 3.65 PEGGnBA Register Contents

Bit Position	Bit Name	Function
31 to 12	GnBASE	These bits set a base address that specifies PE guard area n.
11 to 8	—	Reserved. These bits are always read as 0. The write value should always be 0.
7	GnSP3	Enables accesses to PE guard area n by the bus master. 0: Access by the bus master having SPID = 3 is disabled. 1: Access by the bus master having SPID = 3 is enabled.
6	GnSP2	Enables accesses to PE guard area n by the bus master. 0: Access by the bus master having SPID = 2 is disabled. 1: Access by the bus master having SPID = 2 is enabled.
5	GnSP1	Enables accesses to PE guard area n by the bus master. 0: Access by the bus master having SPID = 1 is disabled. 1: Access by the bus master having SPID = 1 is enabled.
4	GnSP0	Enables accesses to PE guard area n by the bus master. 0: Access by the bus master having SPID = 0 is disabled. 1: Access by the bus master having SPID = 0 is enabled.
3	—	Reserved. This bit is always read as 0. The write value should always be 0.
2	GnWR	Enables write access to PE guard area n. 0: Write access is disabled. 1: Write access is enabled.
1	GnRD	Enables read access to PE guard area n. 0: Read access is disabled. 1: Read access is enabled.
0	GnEN	Enables settings for access enable conditions for PE guard area n. 0: Settings for access enable conditions are disabled. 1: Settings for access enable conditions are enabled.

CAUTION

PEGGnBA.GnEN is cleared by writing to the PEGGnMK register.

3.2.4.2 Internal Peripheral Guard Function (IPG)

(1) Overview of the IPG Function

The IPG is a function to prevent unauthorized accesses to peripheral devices from the CPU core equipped with the IPG. The IPG achieves the following functions. The IPG covers peripheral devices on the P-Bus, the global RAM, and peripheral devices in the PE except the code flash and the local RAM.

(1) Detecting violation of peripheral device protection

If the CPU makes an unauthorized access to an area (peripheral device) for which peripheral device protection is set, the access is detected as “violation of peripheral device protection.”

(2) Storing unauthorized-access information

When a violation of peripheral device protection is detected, the unauthorized-access information is stored in the IPG’s internal register.

(3) Blocking unauthorized accesses

When a violation of peripheral device protection is detected, unauthorized accesses to peripheral devices are blocked to prevent contents of peripheral devices from being modified illegally.

(4) Notifying violation

When a violation of peripheral guard is detected, a request for generating a system error exception (SYSERR exception) is made to ask the CPU to stop the processing.

For the details of the system error exception (SYSERR exception), see **Section 3.2.4.3, System Error Notification Control Function (SEG)**.

(5) Invalidating subsequent accesses

When a violation of peripheral guard is detected, subsequent accesses (regardless of authorized or unauthorized accesses) are blocked until instructions from the CPU are received.

NOTE

Even if a request for generating an exception is immediately sent to the CPU in step (4) above, a subsequent access issued (before receiving a request from the IPG) by the CPU that does not know an occurrence of violation may illegally modify contents of peripheral devices. (Accesses after a violation has occurred result in unauthorized accesses.)

(2) IPG Function

(1) This function invalidates accesses according to their attributes (including address, transfer type, and access right).

(2) After an access right violation is detected until the error flag (described later) is cleared by writing by the software, subsequent accesses are invalidated. However, invalidation is applied only to accesses from the CPU and is not applied to accesses from the bus master other than CPU core. Invalidation is performed independently of addresses.

(3) When a request for accessing different peripheral devices simultaneously is made due to misalignment or double-word access, the access is executed when all such accesses are enabled.

(3) IPG Setting Registers for Illegal Users

To protect peripheral devices from unauthorized accesses by programs in user mode, necessary settings are required for the registers listed below.

Accesses in user mode are to be detected.

- This register set is intended for IPG settings related to user mode and for reading of the IPG settings.

Table 3.66 List of Registers

Module Name	Register Name	Symbol	Address
IPG	Peripheral device protection violation access information register	IPGECRUM	FFFE E002$_H$
IPG	Peripheral device protection violation access address register	IPGADRUM	FFFE E008$_H$
IPG	Peripheral device protection enable register	IPGENUM	FFFE E00D$_H$
IPG	Peripheral device protection setting register 0	IPGPMTUM0	FFFE E020$_H$
IPG	Peripheral device protection setting register 1	IPGPMTUM1	FFFE E021$_H$
IPG	Peripheral device protection setting register 2	IPGPMTUM2	FFFE E022$_H$
IPG	Peripheral device protection setting register 3	IPGPMTUM3	FFFE E023$_H$
IPG	Peripheral device protection setting register 4	IPGPMTUM4	FFFE E024$_H$

(4) Register Set

(a) IPGECRUM — Peripheral device protection violation access information register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DS				EX	WR	RD	VD
Value after reset	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

×: Undefined (retained)

Table 3.67 IPGECRUM Register Contents

Bit Position	Bit Name	Function
15, 14	—	Reserved. These bits are always read as 0. The write value should always be 0.
13 to 8	—	Reserved. These bits are read as undefined. The write value should always be 0.
7 to 4	DS	These bits store the data size of access that made a violation. 1000: Double word (8 bytes) 0100: Word (4 bytes) 0010: Half word (2 bytes) 0001: Byte Other than above: Reserved
3	EX	This bit is set to 1 when a violation occurred in an instruction fetch read access. In other cases, this bit is cleared to 0.
2	WR	This bit is set to 1 when a violation occurred in a write access, bit operation, or execution of the CAXI instruction. In other cases, this bit is cleared to 0.
1	RD	This bit is set to 1 when a violation occurred in a read access, bit operation, or execution of the CAXI instruction. In other cases, this bit is cleared to 0.
0	VD	This bit is set to 1 when a violation of peripheral device protection is detected by a program with the relevant right. If another violation of peripheral device protection is detected, data of this IPGECRUM register and the IPGADRUM register are updated.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device protection by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(b) IPGADRUM — Peripheral device protection violation access address register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EADR															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

×: Undefined (retained)

Table 3.68 IPGADRUM Register Contents

Bit Position	Bit Name	Function
31 to 0	EADR	These bits store the address of the access in which a violation occurred.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device protection by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(c) IPGENUM — Peripheral device protection enable register

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IRE	E
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.69 IPGENUM Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved. These bits are always read as 0. The write value should always be 0.
1	IRE	This bit sets whether to store the access information in the peripheral device protection violation access address register and the peripheral device protection violation access information register when a violation of peripheral device protection occurred in an instruction fetch access. 0: Instruction fetch access information is not stored. (value after reset) 1: Instruction fetch access information is stored. **CAUTION** If you do not want to detect speculative instruction fetches (no instruction is executed in some cases), clear this bit to 0.
0	E	This bit enables or disables the peripheral devices protection function against accesses by the relevant access right. 0: The peripheral device protection function is disabled. 1: The peripheral device protection function is enabled.

(d) IPGPMTUM0 — Peripheral device protection setting register 0

Bit	7	6	5	4	3	2	1	0
	—	X1	W1	R1	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R

Table 3.70 IPGPMTUM0 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved. This bit is always read as 0. The write value should always be 0.
6	X1	This bit sets whether to enable instruction fetch read access to Peripheral Groups 0 to 3 and 5. 0: Instruction fetch read access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Instruction fetch read access to Peripheral Groups 0 to 3 and 5 is not restricted.
5	W1	This bit sets whether to enable write access to Peripheral Groups 0 to 3 and 5. 0: Write access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Write access to Peripheral Groups 0 to 3 and 5 is not restricted.
4	R1	This bit sets whether to enable read access to Peripheral Groups 0 to 3 and 5. 0: Read access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Read access to Peripheral Groups 0 to 3 and 5 is not restricted.
3 to 0	—	Reserved. This bit is always read as 0. The write value should always be 0.

(e) IPGPMTUM1 — Peripheral device protection setting register 1

Bit	7	6	5	4	3	2	1	0
	—	X1	—	—	—	X0	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R	R/W	R	R

Table 3.71 IPGPMTUM1 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved. This bit is always read as 0. The write value should always be 0.
6	X1	This bit sets whether to enable instruction fetch read access to global RAM bank #1. 0: Instruction fetch read access to global RAM bank #1 is treated as violation. 1: Instruction fetch read access to global RAM bank #1 is not restricted.
5 to 3	—	Reserved. These bits are always read as 0. The write value should always be 0.
2	X0	This bit sets whether to enable instruction fetch read access to global RAM bank #0. 0: Instruction fetch read access to global RAM bank #0 is treated as violation. 1: Instruction fetch read access to global RAM bank #0 is not restricted.
1, 0	—	Reserved. These bits are always read as 0. The write value should always be 0.

(f) IPGPMTUM2 — Peripheral device protection setting register 2

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	W0	R0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 3.72 IPGPMTUM2 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved. This bit is always read as 0. The write value should always be 0.
5	W1	This bit sets whether to enable write access to IPIR, MEV, and COMPTEST. 0: Write access to IPIR, MEV and COMPTEST is treated as violation. 1: Write access to IPIR, MEV and COMPTEST is not restricted.
4	R1	This bit sets whether to enable read access to IPIR. 0: Read access to IPIR, MEV and COMPTEST is treated as violation. 1: Read access to IPIR, MEV and COMPTEST is not restricted.
3, 2	—	Reserved. This bit is always read as 0. The write value should always be 0.
1	W0	This bit sets whether to enable write access to INTC1. 0: Write access to INTC1 is treated as violation. 1: Write access to INTC1 is not restricted.
0	R0	This bit sets whether to enable read access to INTC1. 0: Read access to INTC1 is treated as violation. 1: Read access to INTC1 is not restricted.

(g) IPGPMTUM3 — Peripheral device protection setting register 3

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R

Table 3.73 IPGPMTUM3 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved. This bit is always read as 0. The write value should always be 0.
5	W1	This bit sets whether to enable write access to SEG. 0: Write access to SEG is treated as violation. 1: Write access to SEG is not restricted.
4	R1	This bit sets whether to enable read access to SEG. 0: Read access to SEG is treated as violation. 1: Read access to SEG is not restricted
3 to 0	—	Reserved. These bits are always read as 0. The write value should always be 0.

(h) IPGPMTUM4 — Peripheral device protection setting register 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	W0	R0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.74 IPGPMTUM4 Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved. These bits are always read as 0. The write value should always be 0.
1	W0	This bit sets whether to enable write access to PEG 0: Write access to PEG is treated as violation. 1: Write access to PEG is not restricted.
0	R0	This bit sets whether to enable read access to PEG. 0: Read access to PEG is treated as violation. 1: Read access to PEG is not restricted

3.2.4.3 System Error Notification Control Function (SEG)

Errors due to an instruction fetch or data access can be the sources of system error exceptions. A system error exception is an FE level exception from which return or recovery is not possible.

For source codes (FEIC) of the system error exceptions and error handling, see **Table 3.77, Error Factor Codes and Handling of G3M Core System Error Exceptions**. SEG controls the notification and record of the error by the data access.

Multiple error occurrence inputs are categorized according to error factors, and are processed sequentially from the highest-priority error factor, generating an FE-level asynchronous exception (SYSERR).

The bit position of the SEGFLAG register becomes the priority of error factors. Error factors of lower bits take precedence over error factors of upper bits.

Error information is recorded once regardless of error frequency.

The error with the highest priority of error factor is valid in case errors occurred simultaneously. Recorded error information is not overwritten by subsequent errors.

(1) List of SEG Function Control Registers

Table 3.75 List of Registers

Module Name	Register Name	Symbol	Address
SEG	Error notification control register	SEGCONT	FFFE E980$_H$
SEG	Error occurrence retention register	SEGFLAG	FFFE E982$_H$
SEG	Error factor retention register (address)	SEGADDR	FFFE E988$_H$

NOTES

- If an access is made with an address offset or operable bits other than those specified above, an error response is returned.
- Write access is only possible in supervisor mode (UM = 0). Attempting to write, if these conditions do not hold, leads to an error response being returned.
- No restriction is provided for read accesses.
  - Read accesses to ranges permitted by other protection systems are enabled at any time.

(2) Register Set

(a) SEGCONT — Error notification control register

This register is used to enable (= 1) or disable (= 0) notification of SYSERR request in response to error flags that store error occurrence status according to factors.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SS1E	—	—	—	—	—	VPGE	VCRE	—	TCME	ROME	VCIE	—	ICCE	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R

Table 3.76 SEGCONT Register Contents (1/2)

Bit Position	Bit Name	Function
15	SS1E	This bit notifies of an address parity error during accessing data in its own local RAM.
14 to 10	—	Reserved. This bit is always read as 0. The write value should always be 0.
9	VPGE	This bit notifies of a response to an error in P-Bus. The error includes the followings: • P-Bus guard error in writing access (P-Bus guard for each register regarding the INTC2, DMA, or GRG)
8	VCRE	This bit notifies of IPG violation access detection and subsequent access blocking (including the case where an instruction is fetched).*2
7	—	Reserved. This bit is always read as 0. The write value should always be 0.
6	TCME	This bit notifies of an error during data access to its own local RAM. The error includes the following cases: • When an ECC error which cannot be corrected occurs*1 • When an access to the RAM-unimplemented area in the local RAM is detected
5	ROME	This bit notifies of an error in access to the code flash when a table reference is read in response to a table reference interrupt. The error includes the following cases: • When an ECC error which cannot be corrected occurs*1 • When an address parity error occurs

Table 3.76 SEGCONT Register Contents (2/2)

Bit Position	Bit Name	Function
4	VCIE	• This bit notifies of a response to an error in P-Bus (excluding errors in writing to P-Bus). The error includes the following cases: – When an unimplemented area (FFFF 7900_H to FFFF $7EFF_H$) is accessed – P-Bus guard error (P-Bus guard for each register regarding the INTC2, DMA, or GRG) – P-Bus data parity error (for a peripheral device which the data parity is applied to) – An error when DTSRAM is read • When an unimplemented area in the on-chip I/O register self area is accessed. • Notification of a response to an error in the code flash. The error includes the following cases: – When an ECC error which cannot be corrected occurs*1 – When an address parity error occurs • Notification of a response to an error in global RAM. The error includes the following cases: – When an address parity error occurs – When an access protection violation occurs – When an ECC error which cannot be corrected occurs*1 • Notification of a response to an error in the data flash. – When an ECC error which cannot be corrected occurs*1 • Notification of a detection of access to an interconnect reserved area. PE1 – FFFF 0000_H to FFFF $4FFF_H$ – FFFE 0000_H to FFFE $BFFF_H$ – FB00 0000_H to FE9F $FFFF_H$ – F300 0000_H to F8FF $FFFF_H$ PE2 – FFFF 0000_H to FFFF $4FFF_H$ – FFFE 0000_H to FFFE $9FFF_H$ – FB00 0000_H to FE7F $FFFF_H$ – F900 0000_H to F9FF $FFFF_H$ – F300 0000_H to F7FF $FFFF_H$ • This bit notifies of a IPG violation access detection and subsequent access blocking.*2 • This bit notifies of an access privilege violation. – Read or write access to the IPG Protection Setting Registers by user mode (PSW.UM = 1). – Write access to the SEG Function Control Registers by user mode (PSW.UM = 1).
3	—	Reserved. This bit is always read as 0. The write value should always be 0.
2	ICCE	Instruction Cache Error Notification Enable The following error is handled when the instruction cache system register, ICCTRL.ICHEMK, is set to 0 (the value after reset is 1): ECC errors which cannot be corrected*1.
1, 0	—	Reserved. These bits are always read as 0. The write value should always be 0.

Note 1. While ECC error notification is enabled, an uncorrectable ECC error has occurred. Whether or not the error can be corrected depends on the ECC settings. For details, see **Section 27.2, ECC and EDC**.

Note 2. For the error factor address, see **Section 3.2.4.2, Internal Peripheral Guard Function (IPG)**, IPGADRUM register.

Table 3.77 Error Factor Codes and Handling of G3M Core System Error Exceptions

Source Code	Error Description
10	Reserved
11	Instruction fetch errors (from code flash memory)
12	Errors to be indicated and enabled by the SEGCONT2 bit
13	Instruction fetch errors (from other than code flash memory)
14	Errors to be indicated and enabled by the SEGCONT4 bit
15	Errors to be indicated and enabled by the SEGCONT5 bit
16	Errors to be indicated and enabled by the SEGCONT6 bit
17	Reserved
18	Errors to be indicated and enabled by the SEGCONT8 bit
19	Errors to be indicated and enabled by the SEGCONT9 bit
1A	Reserved
1B	Reserved
1C	Reserved
1D	Reserved
1E	Reserved
1F	Errors to be indicated and enabled by the SEGCONT15 bit

(b) SEGFLAG — Error occurrence retention register

This register sets the flags to 1 at error occurrence, that store each error occurrence status by factors. Those are not automatically cleared to 0.

Both writing 0 to clear the flags and writing 1 to retain the status are possible.

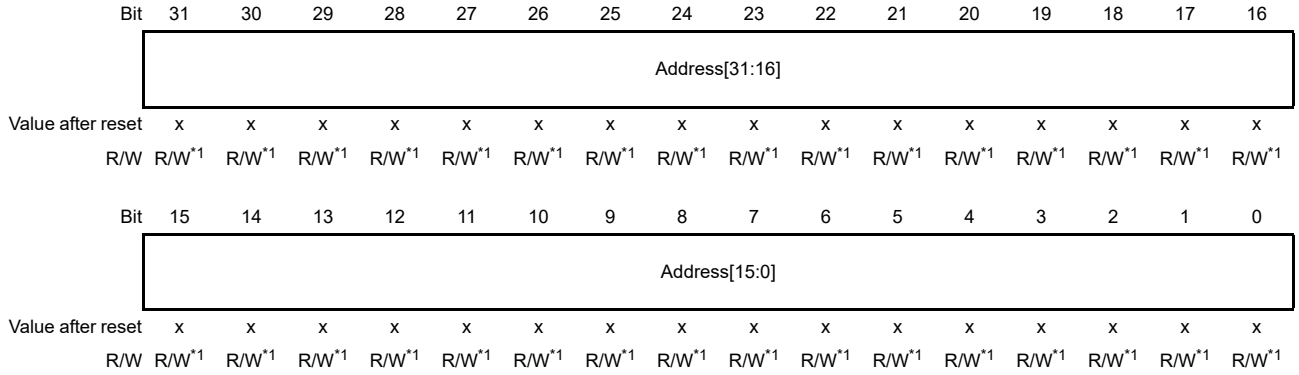
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SS1F	—	—	—	—	—	VPGF	VCRF	—	TCMF	ROMF	VCIF	—	ICCF	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R

Table 3.78 SEGFLAG Register Contents

Bit Position	Bit Name	Function
15	SS1F	Flag corresponding to bit 15 of the SEGCONT register
14 to 10	—	Reserved. These bits are always read as 0. The write value should always be 0.
9	VPGF	Flag corresponding to bit 9 of the SEGCONT register
8	VCRF	Flag corresponding to bit 8 of the SEGCONT register
7	—	Reserved. This bit is always read as 0. The write value should always be 0.
6	TCMF	Flag corresponding to bit 6 of the SEGCONT register
5	ROMF	Flag corresponding to bit 5 of the SEGCONT register
4	VCIF	Flag corresponding to bit 4 of the SEGCONT register
3	—	Reserved. This bit is always read as 0. The write value should always be 0.
2	ICCF	Flag corresponding to bit 2 of the SEGCONT register
1, 0	—	Reserved. These bits are always read as 0. The write value should always be 0.

(c) SEGADDR — Error factor retention register (address)

This register records information of an error factor that notifies a SYSERR request (only one history is recorded). It records the addresses where the error factors for the VCIF, ROMF and TCMF bits of the SEGFLAG register were found. The error factors except VCIF, ROMF and TCMF bits of the SEGFLAG register to be recorded as 0000 0000_H. This cannot be modified when the error occurrence flag to enable notification is set.



×: Undefined (retained)

Note 1. This cannot be modified when the error occurrence flag to enable notification is set.

Table 3.79 SEGSIDE Register Contents

Bit Position	Bit Name	Function
31 to 0	Address[31:0]	These bits hold the address at which the SYSERR source occurred. (When an error occurs during an access to the local RAM area, these bits hold the 19 lower-order bits only, and the 13 higher-order bits are cleared to 0.)

(3) SEG Function

(a) SEG function: Notifying a SYSERR request due to an error flag

- Setting an error flag takes precedence over clearing the same flag.
  - Simultaneous clearing operation is ignored.
- Priority of error factors
  - The bit position of the notification-enabled SEGFLAG register becomes the priority of error factors. Error factors of lower bits take precedence over error factors of upper bits. Notification is made from the highest-priority error factor.
  - The bit position of error factors is notified as a "SYSERR factor code."
- Conditions for starting SYSERR request notification
  - Even if a notification-disabled flag is set to 1, notification is not made.
  - Notification is made immediately after a notification-enabled flag is set to 1.
  - After clearing of a flag, notification is made depending on the flag state (re-arbitration).
- Finishing notification at a SYSERR request response
  - Even after notification is finished, the flag is not cleared automatically.
  - Notification is not made until re-arbitration is performed by setting or clearing the flag.
  - If an error flag that is prioritized higher than the error factor is set prior to a request response, the notification information may be replaced with an upper SYSERR factor code.

(b) SEG function: Recording error factor information

- When notification-enabled error occurrence is input, the error address is retained in the SEGADDR register.
  - No information is retained by setting or clearing an error flag described in "**(a), SEG function: Notifying a SYSERR request due to an error flag**" above.
  - When multiple error occurrence inputs are present simultaneously, information other than the prioritized error factor is not retained.
- While the notification-enabled error flag described in "**(a), SEG function: Notifying a SYSERR request due to an error flag**" above is set to 1, overwrite to the above register is inhibited.
  - If error occurrence input continues, information of subsequent error factors is not retained.
  - To reset the inhibition of overwrite to the register, clear either SEGCONT or SEGFLAG register (or both of them).

(c) Supplementary notes on SYSERR exception

- Even when a SYSERR exception occurs, the value of the PSW.EBV bit is held, and the base address of the exception handler does not switch.
- Error detection in the instruction cache
  Even when an error is detected in the instruction cache, a rerun-typed SYSERR exception caused by instruction fetch does not occur. The instruction cache automatically invalidates the target entry in which an error occurred, and that entry is fetched again from the code flash. Thus execution of CPU instructions continues. Setting the ICCTRL.ICHEMK bit of the system register to 0 notifies the SEG of an error occurred in the instruction cache.

3.2.4.4 Checker Core

CPU1 has the checker core for safety assurance, resulting in a highly reliable system. Monitoring the outputs from CPU1 and the checker core with the comparator all the time enables immediate detection of the CPU1 abnormal operations. The CPU core, FPU, MPU, PEG, IPG, SEG, and INTC1 are duplicated by the checker core. CPU1 can also conduct a fault diagnosis test of its own comparator through a pseudo-error generated by the COMPTEST module. For details of the COMPTEST module, see **Section 27, Functional Safety**.

CAUTION

Reading of any register with an undefined value after a reset in a PE or writing to memory or a register outside the PE may cause a lock step comparison error. The values of some program registers and system registers are undefined after a reset, so pay attention to this when saving register values on the stack in RAM.

3.3 Inter-CPU Functions (only in the C1H)

3.3.1 Processor Element Identifier

The PEID, each processor element ID number, can be read from the PEID field in the HTCFG0 register. Which CPU core performs a specific program can be checked by referring to the PEID. The following shows the PEID of this product.

CPU Core	PEID
CPU1 (PE1)	001_B
CPU2 (PE2)	010_B

3.3.2 Inter-Processor Interrupt Function

The CPU has the IPIR register as its own peripheral device. Setting of the IPIR register enables an EI-level interrupt request from a CPU to another CPU. For details, see **Section 3.2.3, Inter-Processor Interrupts**.

3.3.3 Exclusive Function

The local RAM, global RAM, and exclusive control register (MEV) are available as a resource for exclusive control. Atomic operation instructions of LDL/STC, CAXI, SET1, CLR1, and NOT1 can be performed for the local RAM and the global RAM. For the exclusive control register (MEV), CAXI, SET1, CLR1, and NOT1 can be performed. Note that atomic operations are not performed for the LD and ST instructions even though they can access those resources.

3.3.3.1 Exclusive Control Register (MEV)

This register supports exclusive control for variables shared between PEs (common resources).

(MEV = Mutual Exclusion Variable Register)

- Thirty-two 32-bit MEV registers are included.
- 1-, 8-, 16-, and 32-bit accesses are available for each MEV.
- Accesses from CPU1 (PE1) and CPU2 (PE2) can be made.
- Atomic operation instructions of CAXI, SET1, CLR1, and NOT1 can be performed.

CPU1 (PE1) and CPU2 (PE2) each have an independent access path for the MEV registers. Therefore, when CPU1 (PE1) and CPU2 (PE2) each access different MEV registers, they do not need to wait for access. When they access the same MEV register, however, waiting for access is required.

Table 3.80 List of Registers

Module Name	Register Name	Symbol	Address
MEV	Exclusive control register 0	G0MEV0	FFFE EC00$_H$
MEV	Exclusive control register 1	G0MEV1	FFFE EC04$_H$
MEV	Exclusive control register 2	G0MEV2	FFFE EC08$_H$
MEV	Exclusive control register 3	G0MEV3	FFFE EC0C$_H$
:	:	:	:
MEV	Exclusive control register 31	G0MEV31	FFFE EC7C$_H$

3.3.3.2 Operations of the LDL.W and STC.W Instructions

The LDL.W and STC.W instructions can be used to correctly handle memory update in a multicore system by enabling atomic read-modify-write processing. The operations of the LDL.W and STC.W instructions are as follows. For the operations of the LDL.W and STC.W instructions, see *the RH850G3M User's Manual: Software for RH850G3M*.

- Link generation: The CPU can generate links for both the local RAM and global RAM. If the LDL.W instruction is executed to read the target, the link is generated with the link address registration and the link flag set. The following two types of link flags are provided:
  (1) For the own local RAM: 1
  (2) For the global RAM: 1
  The link (Example: to the own local RAM), that is generated first, would not be lost by LDL instruction execution to (Example: global RAM) the different type, after the generation of the link to (Example: the own local RAM) either type from CPU since the link flags are generated independently from each other.
- Store success: Store processing is executed only when the STC.W instruction corresponding to the generated link is executed after the link was generated.
- Store failure: Store processing is not executed while the link is lost even if the STC.W instruction is executed. Store processing is also not executed when the STC.W instruction that does not correspond to the link is executed.
- Condition of store success: The STC.W instruction is determined that it corresponds to the link if the following condition is satisfied:
  – The address of the STC.W instruction is the same as the address of the LDL.W instruction that generated the link.
- Link lost: When either of the following occurs, the link flag is cleared, and the link is lost:
  – When either of the following events occurs in the CPU that generated the link:
    - The STC.W instruction is executed. The corresponding link ((1) or (2) above) is lost regardless of whether the store succeeded or failed.
    - Some of the various types of exceptions occur, or a restore instruction (FERET or EIRET) from the exception is executed.
      Two types of link flags are all cleared.
    - Multiple LDL.W instructions are executed in succession for the same type of link flags.
      The link generated by the preceding LDL.W instruction is lost. Do not execute such processing.
    - A store operation other than the STC.W instruction is executed for the address*1 for which the link is generated.
      Do not execute such processing.
  – When the following access is executed by another bus master:
    - A store operation (including execution of the STC.W instruction) is executed for the address*1 for which the link is generated.
      The corresponding link is lost.

Note 1. It indicates the matching address with the 27 higher-order bits of link address.

When the STC.W instruction is successful, an atomic read-modify-write processing has been executed by the LDL.W and STC.W instructions.

3.4 Notes

3.4.1 Synchronization of Store Instruction Completion and Subsequent Instruction Generation

When a control register is updated by a store instruction, there is a time lag after CPU implementation of the store instruction and actual updating of the control register. Therefore, adequate synchronization processing is needed to ensure the control register reflects updated contents before generation of a subsequent instruction. How to perform synchronization processing is shown below. For the procedures to synchronize updating system registers by LDSR instruction and the subsequent instruction execution, see *APPENDIX A. Hazard Resolution Procedure for System Registers* in *the RH850G3M User's Manual: Software*.

3.4.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction:

Example 1: This includes the following case: an interrupt is enabled by implementation of an EI instruction after an interrupt request is cleared by access from the control register in the INTC2 and the peripheral circuits.
Proceed as follows in this case.

(1) Store instruction to update a control register (ST.W, etc.)

(2) Dummy reading of the above-mentioned control register (LD.W etc.)*1

(3) SYNCP

(4) Subsequent instruction (EI, etc.)

Example 2: If an access to another control register (control register B) is needed after a control register (control register A) has been completely updated, execute the similar processing. This includes the following cases: different peripheral devices are linked, or the interrupt mask for INTC is cleared after the peripheral device is set. Note that this processing is not required if the control registers A and B belong to the same peripheral group. For the details of Peripheral Group, see **Section 3.1.2, Periphral Group Configuration**.

(1) Store instruction to update the control register A (ST.W, etc.)

(2) Dummy read of the above-mentioned control register (LD.W etc.)*1

(3) SYNCP

(4) Instruction to access the control register B (ST.W, LD.W, etc.)

The similar processing is also required when an access to a memory or control register to be protected is started after a safety function (such as some kind of memory protection and ECC) has been completely set up.

Note 1. Dummy reading of any register of the same peripheral group can be used instead.

3.4.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction:

(a) In case of writing the instructions to the RAM before jumping to the RAM to execute instrunctions from the RAM, take the following procedure.

(1) Execute the store instruction to update a memory (ST.W, etc).

(2) Perform a dummy read of the above memory (LD.W, etc).

(3) Execute SYNCP.

(4) Execute SYNCI.

(5) Execute the subsequent instruction (branch instruction, etc).

(b) In case of updating control registers for memory protection and ECC functions before jumping to the memory to be controlled by the registers, take the following procedure.

(1) Execute the store instruction to update a control register (ST.W, etc).

(2) Perform a dummy read of the control register (LD.W, etc).

(3) Execute SYNCP.

(4) Execute SYNCI.

(5) Execute the subsequent instruction (branch instruction, etc).

3.4.1.3 When switching the code flash memory area:

In this case, see *Section 9, Usage Notes, (7) Update of FCUFAREA register* in *the RH850/C1x Flash Memory User's Manual: Hardware Interface*.

3.4.2 Access to Registers by Bit-Manipulation Instructions

Processing of writing by bit-manipulation instructions consists of atomic read-modify-write processing in 8-bit units. Thus, access by a bit-manipulation instruction is only possible for registers for which reading and writing in 8-bit units is possible. If a register includes multiple flag bits, the read-modify-write operation may lead to the clearing of flags that were not actually targets for clearing.

3.4.3 Ensure Coherency after Rewriting the Code Flash

The CPU is equipped with the valid instruction cache and data buffer for the code flash area.

Therefore, clear the instruction cache and data buffer to ensure coherency after rewriting the code flash by self-programming. The instruction cache is cleared by the ICCTRL register, and the data buffer is cleared by the CDBCR register.

3.4.4 Overwriting Context when Acknowledging Multiple Exceptions

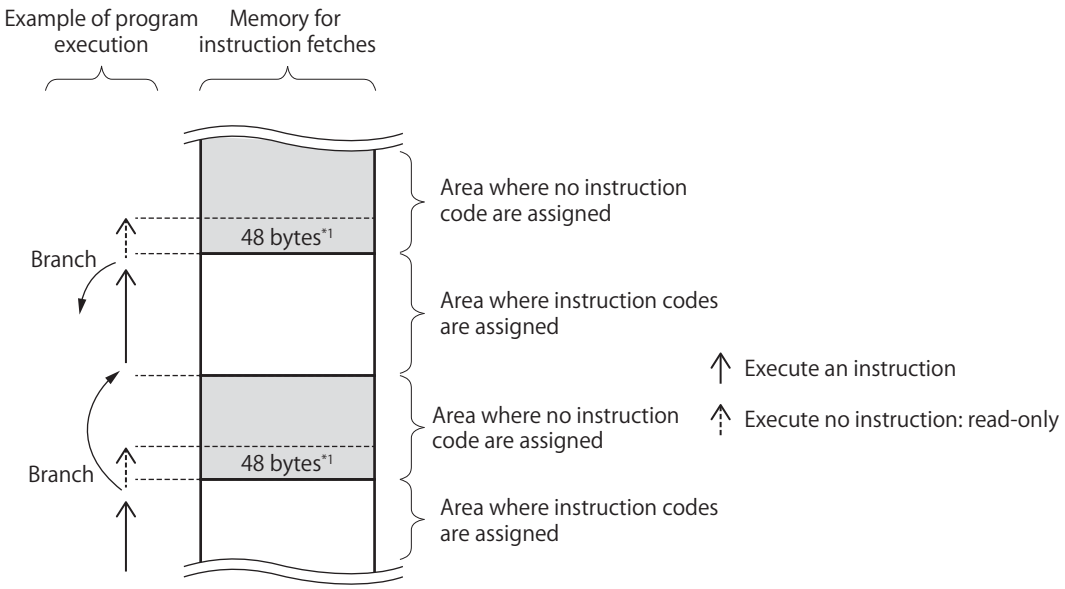
Exceptions may be acknowledged regardless of the states of the ID or NP bits of the PSW register. This depends on the type of exception. When multiple exceptions occur, the contents of the system registers which hold the context information are overwritten. Regarding the conditions for acknowledging exceptions from each source and the possibility of return and recovery, see the list of exception sources in the *RH850G3M User's Manual: Software*.

3.4.5 Usage Notes on Prefetching

CPU executes speculative instruction fetching from locations later than the current value of the program counter to maintain the throughput of instruction fetches. Reading from memory due to such prefetching may proceed even from locations to which instruction codes have not been assigned (note 1 in **Figure 3.5**). Please note the following. The CPU does not execute values read in such cases.

These notes apply to instruction fetching from memory in general.

- Occurrence of ECC errors due to values in memory being undefined
  This prefetching may lead to an ECC error in case of reading from the code flash memory after it has been erased or from the local RAM or global RAM before initialization. When instruction codes are assigned to memory, initialize said area with values as desired (note 1 in **Figure 3.5**).
- Detection of illegal access by the GRG or IPG
  The GRG or IPG may detect such prefetching as illegal access. To prevent prefetching being detected as an illegal access, do not allow any region of overlap area with said areas (note 1 in **Figure 3.5**) and areas to which access is prohibited by the GRG or IPG. Reading from an area protected by the MPU does not cause a memory protection exception.
- Access to Access Prohibited Area
  Assign instruction codes to memory without allowing any overlap between said area (note 1 in **Figure 3.5**) and an access-prohibited area.



Note 1. Area from final address of instruction code to final address + 48 bytes. Even if instruction codes are not placed in this area, requests for prefetching may lead to reading from memory at the time of branching.

Figure 3.5 Area that Requires Attention Regarding Prefetching

Section 4 Address Space

4.1 Address Space (C1H)

4.1.1 Address Space

Table 4.1 shows the address space of the RH850/C1H.

Do not access an address with which no register is mapped in the on-chip I/O register space. Do not access any address that is not specified in **Table 4.1** and any reserved area. If an unspecified address or reserved area is accessed, operation is not guaranteed.

Table 4.1 Address Space (C1H)

Address	Address Space Type	Size
0000 0000_H to 001F $FFFF_H$ (0001 7000_H to 0001 $7FFF_H$)	Code flash (user area read) (FCU firmware area (Map is switched by the FCUFAREA register.)) 4 MB provided in C1H (2-bank configuration)*3	2.0 MB (4 KB)
0020 0000_H to 007F $FFFF_H$	Reserved area	
0080 0000_H to 009F $FFFF_H$	Code flash (user area read) 4 MB provided in C1H (2-bank configuration)	2.0 MB
00A0 0000_H to 00FF $FFFF_H$	Reserved area	
0100 0000_H to 0100 $7FFF_H$	Code flash (user boot area read)	32 KB
0100 8000_H to FE9E $FFFF_H$	Reserved area	
FE9F 0000_H to FE9F $FFFF_H$	Local RAM (CPU2 area)	64 KB
FEA0 0000_H to FEBE $FFFF_H$	Reserved area	
FEBF 0000_H to FEBF $FFFF_H$	Local RAM (CPU1 area)	64 KB
FEC0 0000_H to FEDE $FFFF_H$	Reserved area	
FEDF 0000_H to FEDF $FFFF_H$	Local RAM (self area*1)	64 KB
FEE0 0000_H to FEEE $FFFF_H$	Reserved area	
FEEF 0000_H to FEF0 $BFFF_H$	Global RAM	112 KB
FEF0 $C000_H$ to FEFF $FFFF_H$	Reserved area	
FF00 0000_H to FFFD $FFFF_H$ (FF20 0000_H to FF20 $7FFF_H$) (FFA1 2000_H to FFA1 $2FFF_H$)	On-chip I/O register (Data flash (read/write) (FCU RAM area)	16 MB-128 KB (32 KB) (4 KB)
FFFE 0000_H to FFFE $DFFF_H$	Reserved area	
FFFE $E000_H$ to FFFE $FFFF_H$	On-chip I/O register (self area*2)	8 KB
FFFF 0000_H to FFFF $4FFF_H$	Reserved area	
FFFF 5000_H to FFFF $FFFF_H$	On-chip I/O register	44 KB

Note 1. When a local RAM self area is accessed, access the local RAM (self) included in each CPU (CPU1/CPU2).

Note 2. When an on-chip I/O register self area is accessed, access the on-chip I/O register included in each CPU (CPU1/CPU2).

Note 3. For details, see the *RH850/C1x Flash Memory User's Manual: Hardware Interface.*

4.1.2 Address Space Viewed from Each Bus Master

Figure 4.1 shows address spaces viewed from each bus master.

4.1.2.1 Space in which Instructions can be Fetched

1. Instructions of the CPU1 and CPU2 can be fetched from the code flash, local RAM area, and global RAM area.
2. The reset vector (RBASE value after reset) of the CPU1 and CPU2:

- When the startup area is the user boot area, its head address is 0100_0000_H.
- When the startup area is the user area, its head address is 0000_0000_H

4.1.2.2 Data Space Accessible by CPU1

See **Figure 4.1** for the accessible spaces from CPU1.

4.1.2.3 Data Space Accessible by CPU2

See **Figure 4.1** for the accessible spaces from CPU2.

4.1.2.4 Data Space Accessible by DMA (DMAC, DTS)

See **Figure 4.1** for the accessible spaces from the DMA.

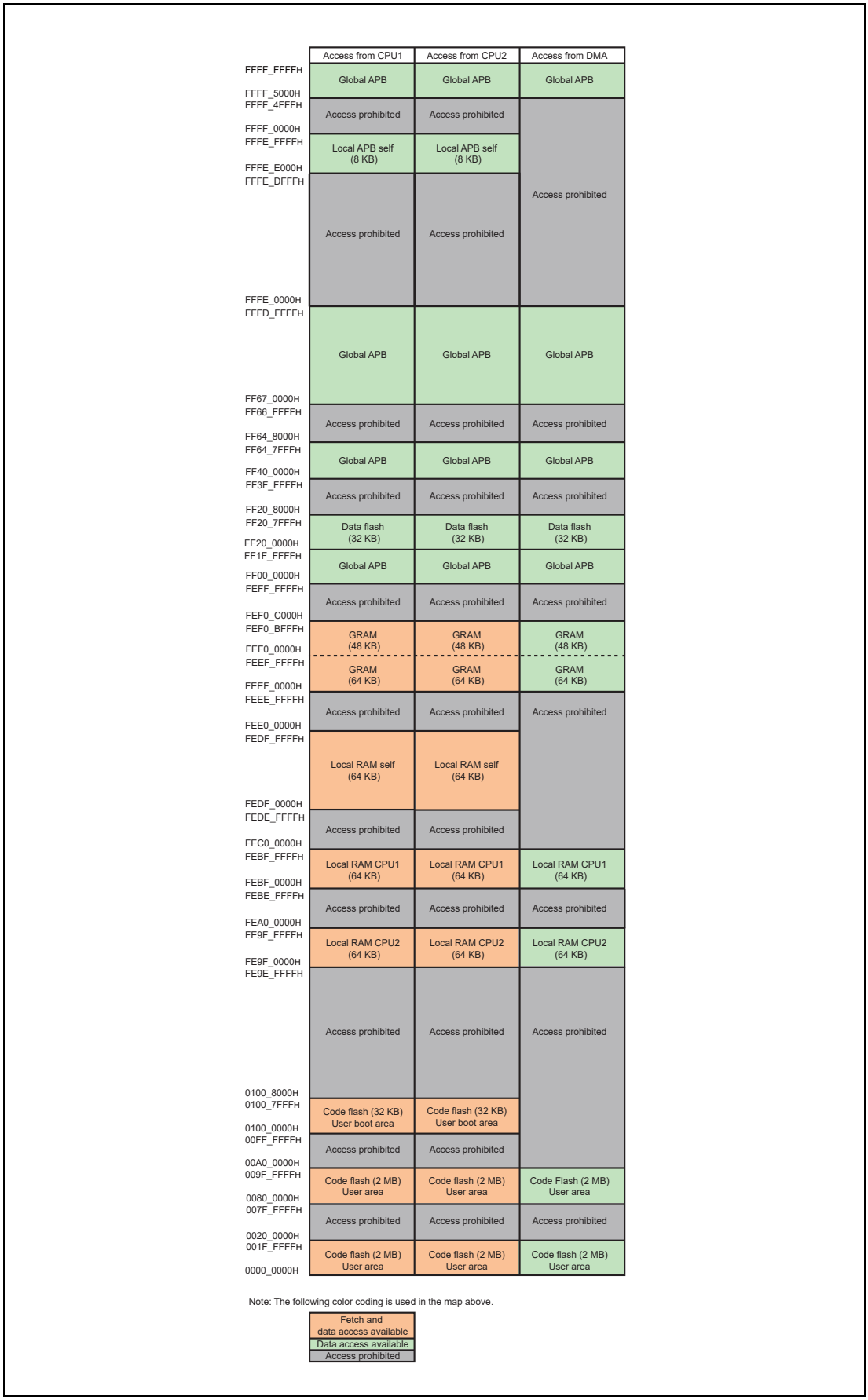


Figure 4.1 Address Space Viewed from Each Bus Master (C1H)

4.1.3 Global RAM Area

The global RAM is divided into two, bank A and bank B. Different banks can be accessed concurrently.

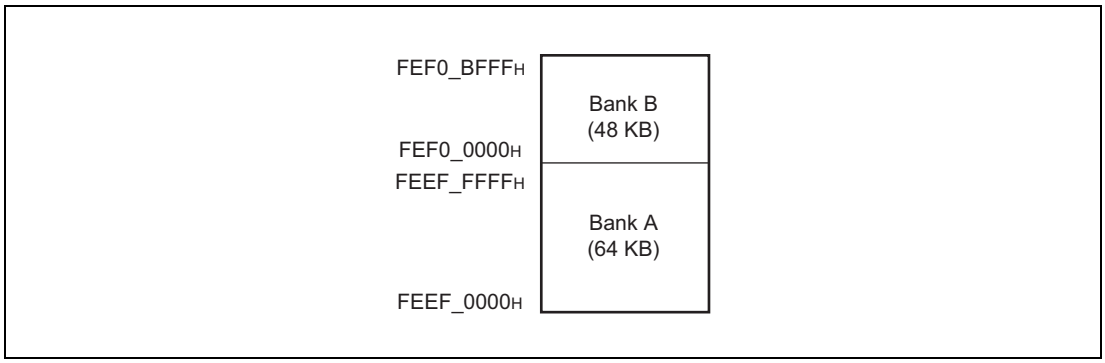


Figure 4.2 GRAM Bank Area (C1H)

4.2 Address Space (C1M)

4.2.1 Address Space

Table 4.2 shows the address space of the RH850/C1M.

Do not access an address with which no register is mapped the On-chip I/O register space. Do not access any address that is not specified in **Table 4.2** and any reserved area. If an unspecified address or reserved area is accessed, operation is not guaranteed.

Table 4.2 Address Space (C1M)

Address	Address Space Type	Size
0000 0000_H to 001F $FFFF_H$ (0001 7000_H to 0001 $7FFF_H$)	Code flash (user area read) (FCU firmware area (Map is switched by the FCUFAREA register.)) 2 MB provided in C1M (1-bank configuration)*1	2.0 MB (4 KB)
0020 0000_H to 00FF $FFFF_H$	Reserved area	
0100 0000_H to 0100 $7FFF_H$	Code flash (user boot area read)	32 KB
0100 8000_H to FEBE $FFFF_H$	Reserved area	
FEBF 0000_H to FEBF $FFFF_H$	Local RAM (CPU1 area)	64 KB
FEC0 0000_H to FEDE $FFFF_H$	Reserved area	
FEDF 0000_H to FEDF $FFFF_H$	Local RAM (self area)	64 KB
FEE0 0000_H to FEEE $FFFF_H$	Reserved area	
FEEF 0000_H to FEEF $FFFF_H$	Global RAM	64 KB
FEF0 0000_H to FEFF $FFFF_H$	Reserved area	
FF00 0000_H to FFFD $FFFF_H$ (FF20 0000_H to FF20 $7FFF_H$) (FFA1 2000_H to FFA1 $2FFF_H$)	On-chip I/O register (Data flash (read/write) (FCU RAM area)	16 MB-128 KB (32 KB) (4 KB)
FFFE 0000_H to FFFE $DFFF_H$	Reserved area	
FFFE $E000_H$ to FFFE $FFFF_H$	On-chip I/O register (self area)	8 KB
FFFF 0000_H to FFFF $4FFF_H$	Reserved area	
FFFF 5000_H to FFFF $FFFF_H$	On-chip I/O register	44 KB

Note 1. For details, see the *RH850/C1x Flash Memory User's Manual: Hardware Interface.*

4.2.2 Address Space Viewed from Each Bus Master

Figure 4.3 shows address spaces viewed from each bus master.

4.2.2.1 Space in which Instructions can be Fetched

1. Instructions of the CPU1 can be fetched from the code flash, local RAM area, and global RAM area.
2. The reset vector (RBASE value after reset) of the CPU1:

- When the startup area is the user boot area, its head address is 0100_0000_H.
- When the startup area is the user area, its head address is 0000_0000_H.

4.2.2.2 Data Space Accessible by CPU1

See **Figure 4.3** for the accessible spaces from CPU1.

4.2.2.3 Data Space Accessible by DMA (DMAC, DTS)

See **Figure 4.3** for the accessible spaces from the DMA.

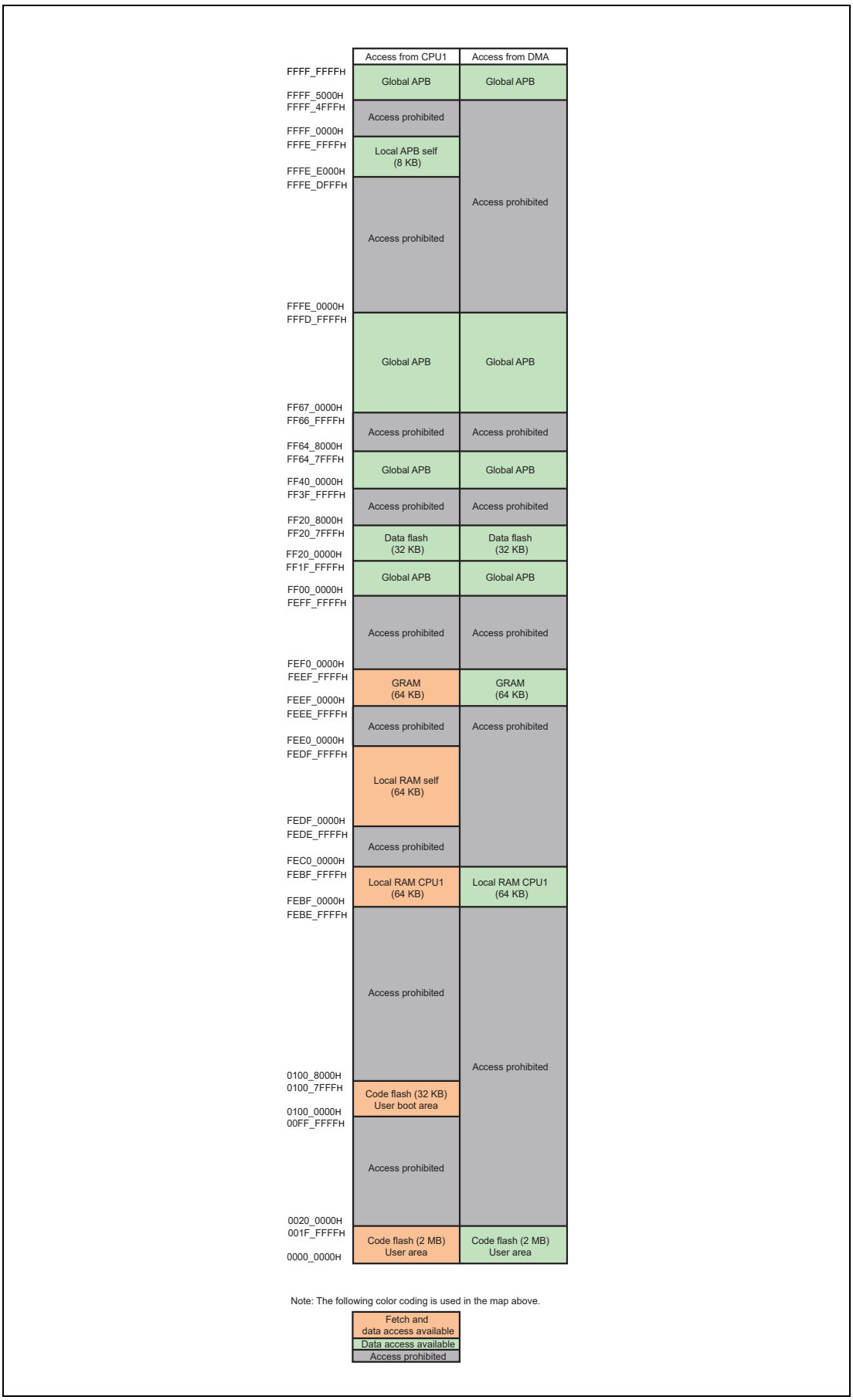


Figure 4.3 Address Space Viewed from Each Bus Master (C1M)

4.2.3 GRAM Bank Area

The global RAM consists of only Bank A.

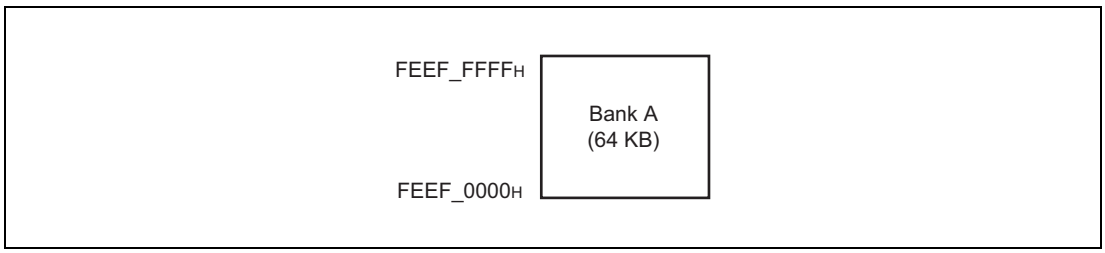


Figure 4.4 GRAM Bank Area (C1M)

Section 5 Operating Mode

5.1 Features

- This LSI has three mode pins to determine the operating mode (MD1, MD0, and FLMODE).

5.2 Operating Mode

This LSI has multiple operating modes, which can be selected with the three pins (MD1, MD0, and FLMODE) and the setting of STMSEL1/STMSEL0 in option byte 0. For the procedures to set STMSEL1/STMSEL0, see **Section 31, Flash Memory**. **Table 5.1** shows the list of the operating modes.

Table 5.1 Selection of Operating Mode

Value Set in The Pin			Value Set in Option Byte 0		Operating Mode	Startup Area	Types of I/F*1	Remarks
MD1	MD0	FLMODE	STM SEL1	STM SEL0				
0	0	0	0	0	User boot mode	User area	It is possible to select I/F by OPBT2 in option byte. For details, see **Section 31.10.2, OPBT2 — Option Byte 2 Register**.	On-chip debug is available.
			0	1	User boot mode	User boot area		
			1	×	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available.
0	0	1	×	×	Boundary scan mode	—	JTAG	Boundary scan is available.
0	1	0	×	×	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available.
0	1	1	×	×	Serial programming mode	Boot area	Writer I/F (3-line clock synchronization)	Serial programming is available.

Note: X = Don't care.

Note 1. For the correspondence between the pin function and pin state in each interface, see **Section 2.4.3, Pin State**.

5.2.1 User Boot Mode

After release from the reset state, instruction fetch is carried out from the user boot area or the user area.

5.2.2 Serial Programming Mode

After release from the reset state, the LSI boots up from the on-chip boot program and starts connection in the specified transmission method. For details, see **Section 31, Flash Memory**.

5.2.3 Boundary Scan Mode

This is a mode to use the boundary scan function in accordance with the standard of IEEE1149.1. For details, see **Section 34, Boundary Scan**.

Section 6 Interrupt

The interrupt controllers (INTC) judge the priority levels of interrupt sources and control interrupt requests for the CPU. The INTC has registers for setting the order of priority for all of the interrupts, and processing of the interrupts is in accord with the order of priority set by the user in these registers.

6.1 Overview

- Distribution of interrupt requests from one source to more than one core is supported.
  - Interrupt requests from one source are distributable to more than one CPU core (specifically, to the CPU1 and the CPU2)
  - Target interrupt sources: 1 non-maskable interrupt, 1 FE level interrupt, and 21 EI level interrupts
- Interrupt sources
  - Non-maskable interrupts:
    External pin NMI interrupt (FENMI) (1 source)
  - FE level interrupts
    ECM interrupt (FEINT) (1 source)
  - EI level interrupts (maskable) (EIINT)

    High-speed interrupts (EIINT0 to EIINT31) C1H: 21 sources; C1M: 16 sources
    - Inter-processor interrupts
    - ECM interrupts
    - External interrupts (IRQ)
    - Software interrupts (SINT)
    - Interrupts from the fixed-period timer (OSTM)
    - DMA error notification interrupts

    Low-speed interrupts (EIINT32 to EIINT255) C1H: 188 sources; C1M: 175 sources
    - Interrupts from timers
    - Transfer-related interrupts
    - Interrupts from the ADC
    - Interrupts from the DMAC and DTS
    - Others
- Able to set up the order of priority for interrupts
  256 interrupt control registers are provided so that one of 16 priority levels is assignable to the IRQ (external) and other maskable interrupts thus placing each of the interrupt requests in an order of priority.
- Sensing of external interrupts (NMI/IRQ)
  For NMI sources, falling edges and rising edges are selectable.
  For IRQ sources, the low level, the high level, falling edges, and rising edges are selectable.
- Two ways to specify the addresses of the interrupt handlers: through direct branching or with reference to a table.

- Inter-processor interrupts
  High-speed interrupts between the processors are available.
- Software interrupts (SINT)
  The software interrupt registers are accessible from programs and can be used to generate changes to the order of priority as desired.
- Sharing interrupt sources
  Exception handler addresses are reduced by merging multiple interrupt sources.

Interrupts are controlled by the following interrupt controllers.

- INTC1
  CPU1 and CPU2 have their own interrupt controllers.
  Each CPU accesses to the INTC1 register that corresponds to respective CPUs.
  INTC1 controls high-speed interrupts, and has the following functions:
  - Priority order setting
  - Interrupt mask setting
- INTC2
  INTC2 is a common interrupt controller that CPU1 and CPU2 share.
  INTC2 controls low-speed interrupts, and has the following functions:
  - Priority order setting
  - Interrupt mask setting
  - Binding setting

6.2 Register Specifications

The INTC contains the registers listed in the table below. These registers are used to specify the interrupt priority and control the detection of external interrupt input signals.

6.2.1 Configuration of Registers

Table 6.1 Interrupt Control

Module Name	Address	Register Symbol	Register Name	R/W	Value after Reset
INTC1 (EIC0 to EIC31)	FFFE EA00$_H$ to FFFE EA3E$_H$ (EIC0 to EIC31)	EICn*1	EI level interrupt control register	R/W	008F$_H$*5 808F$_H$*6
INTC2 (EIC32 to EIC255)	FFFF B040$_H$ to FFFF B1FE$_H$ (EIC32 to EIC255)				
INTC1 (IMR0)	FFFE EAF0$_H$ (IMR0)	IMRn*2	EI level interrupt mask register	R/W	FFFF FFFF$_H$
INTC2 (IMR1 to IMR7)	FFFF B404$_H$ to FFFF B41C$_H$ (IMR1 to IMR7)				
INTC1 (EIBD0 to EIBD31)	FFFE EB00$_H$ to FFFE EB7C$_H$ (EIBD0 to EIBD31)	EIBDn*3	EI level interrupt bind register	R/W	*4
INTC2 (EIBD32 to EIBD255)	FFFF B880$_H$ to FFFF BBFC$_H$ (EIBD32 to EIBD255)				

Note 1. n = 0 to 255
Note 2. n = 0 to 7
Note 3. n = 0 to 255
Note 4. n = 0 to 31: Same as the PEID bits
n = 32 to 255: 0000 0001$_H$
Note 5. This is the case when an edge is detected.
Note 6. This is the case when a level signal is detected.

Among the registers shown in **Table 6.1**, the EIC0 to EIC31, IMR0, and EIBD0 to EIBD31 are located in INTC1 of the CPU Peripheral area included in each CPU. Each register of these only can be accessed from CPU1 or CPU2 which includes it. Writing is only possible in supervisor mode (PSW.UM = 0).

Of the registers listed in **Table 6.1**, EIC32 to EIC255, IMR1 to IMR7, and EIBD32 to EIBD255 are located in INTC2, the controller for interrupts from peripheral group 0. Writing to these registers is possible when the following conditions are satisfied.

- CPU1 is in supervisor mode (UM = 0).
- CPU2 is in supervisor mode (UM = 0) and EIBDn (n = 32 to 255) are bound to CPU2.

However, when writing to IMR1 to IMR7 by CPU2, only the bits corresponding to those in EIBDn (n = 32 to 255) that are bound to CPU2 are updated; other bits are not updated.

In the register areas listed in **Table 6.1**, the values of those listed as reserved for the given channel numbers in **Table 6.11, Interrupt Exception Handlers and Orders of Priority** must retain their values after a reset.

Table 6.2 External Interrupts, Software Interrupts, and NMI

Module Name	Address	Register Symbol	Register Name	R/W	Initial Value
EINT	FFC0 0000$_H$	NMICTL	NMI interrupt control register	R/W	00$_H$
EINT	FFC0 0010$_H$	EXINTCTL	External interrupt control register	R/W	0000$_H$
EINT	FFC0 0014$_H$	EXINTSTR	External interrupt status register	R	00$_H$
EINT	FFC0 0018$_H$	EXINTSTC	External interrupt status clear register	W	00$_H$
EINT	FFC0 0020$_H$	SINTR0	Software interrupt register 0	R/W	00$_H$
EINT	FFC0 0024$_H$	SINTR1	Software interrupt register 1	R/W	00$_H$
EINT	FFC0 0028$_H$	SINTR2	Software interrupt register 2	R/W	00$_H$
EINT	FFC0 002C$_H$	SINTR3	Software interrupt register 3	R/W	00$_H$

Table 6.3 Merging of Interrupts

Module Name	Address	Register Symbol	Register Name	R/W	Initial Value
INTIF	FFF9 8000$_H$	PINT0	Peripheral interrupt status register 0	R	0000 0000$_H$
INTIF	FFF9 8004$_H$	PINT1	Peripheral interrupt status register 1	R	0000 0000$_H$
INTIF	FFF9 8008$_H$	PINT2	Peripheral interrupt status register 2	R	0000 0000$_H$
INTIF	FFF9 800C$_H$	PINT3	Peripheral interrupt status register 3	R	0000 0000$_H$
INTIF	FFF9 8010$_H$	PINT4	Peripheral interrupt status register 4	R	0000 0000$_H$
INTIF	FFF9 8014$_H$	PINT5	Peripheral interrupt status register 5	R	0000 0000$_H$
INTIF	FFF9 8018$_H$	PINT6	Peripheral interrupt status register 6	R	0000 0000$_H$
INTIF	FFF9 801C$_H$	PINT7	Peripheral interrupt status register 7	R	0000 0000$_H$
INTIF	FFF9 8020$_H$	PINTCLR0	Peripheral interrupt status clear register 0	W	0000 0000$_H$
INTIF	FFF9 8024$_H$	PINTCLR1	Peripheral interrupt status clear register 1	W	0000 0000$_H$
INTIF	FFF9 8028$_H$	PINTCLR2	Peripheral interrupt status clear register 2	W	0000 0000$_H$
INTIF	FFF9 802C$_H$	PINTCLR3	Peripheral interrupt status clear register 3	W	0000 0000$_H$
INTIF	FFF9 8030$_H$	PINTCLR4	Peripheral interrupt status clear register 4	W	0000 0000$_H$
INTIF	FFF9 8034$_H$	PINTCLR5	Peripheral interrupt status clear register 5	W	0000 0000$_H$
INTIF	FFF9 8038$_H$	PINTCLR6	Peripheral interrupt status clear register 6	W	0000 0000$_H$
INTIF	FFF9 803C$_H$	PINTCLR7	Peripheral interrupt status clear register 7	W	0000 0000$_H$

6.2.2 EI Level Interrupt Control Registers 0 to 255 (EIC0 to EIC255)

One of these registers is provided for every EI level interrupt source. The registers are used to configure the conditions of control for each source. For each source, see **Table 6.11, Interrupt Exception Handlers and Orders of Priority**.

CAUTION

If 0 is written to the EIRFn bit immediately after a peripheral module generates the corresponding interrupt request upon detection of an edge (before an interrupt is accepted by the CPU), the request may be lost.

Writing 1 to the EIRFn bit immediately after an interrupt is accepted by the CPU may lead to re-issuing of the request.

This includes the use of bit-manipulation instructions (set1, clr1, and not1) for writing to any of these registers. For bit-manipulation instructions, see also Section 3.4.2, Access to Registers by Bit-Manipulation Instructions.

Executing a bit-manipulation instruction to the lower bytes including the EIMKn bit has no effect on the EIRFn bit.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EICTn	—	—	EIRFn	—	—	—	—	EIMKn	EITBn	—	—	EIPn			
Value after reset	*1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Note 1. 0: Edge detection
1: Level detection

Table 6.4 EIC Register Contents (1/2)

Bit Position	Bit Name	Function
15	EICTn	This is the interrupt channel type bit. The values listed below are read in accord with the interrupt input interface. The bit is read-only. Following a read modify write sequence for access to change this bit as a single bit is prohibited. 0: Edge detection 1: Level detection When writing to this bit, write the value after reset.
14, 13	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
12	EIRFn	This is an interrupt request flag. The operation of the flag depends on the interrupt input interface. 0: No interrupt request is made (value after reset). 1: Interrupt request is made. • Edge detection When the CPU core has accepted an interrupt request on its own channel, the request is automatically cleared. This bit may be set and cleared by software. When the EIRFn bit is set to 1, an EI level maskable interrupt n (EIINTn) is generated in the same way as acceptance of an interrupt request. • Level detection Setting and clearing of the bit by software is not possible. That is, the bit is read-only in this case.
11 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.

Table 6.4 EIC Register Contents (2/2)

Bit Position	Bit Name	Function
7	EIMKn	This is an interrupt mask bit. Setting this bit masks interrupt requests set in the interrupt request flag (EIRFn), i.e. it may be used to obstruct interrupt requests from the given channel to the CPU core. Furthermore, clearing of the PMEI bit in the ICSR does not proceed in response to notification of the presence of an interrupt that has not been processed on the channel for which this bit has been set. This bit does not mask input from an interrupt input pin, so the interrupt request flag still gets set even if this bit is set. The setting of this bit also reflects the setting in the interrupt mask register (IMR). 0: Interrupt processing is enabled. 1: Interrupt processing is disabled (value after reset).
6	EITBn	This bit is used to select the method of detection for the interrupt. 0: Direct jumping to an address determined from the level of priority 1: Reference to a table
5, 4	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
3 to 0	EIPn	These bits specify the interrupt priority as one of 16 levels, with 0 as the highest and 15 as the lowest. When multiple EI level-interrupt requests are made simultaneously, the interrupt from the source with the highest priority setting in these bits is selected and conveyed to the CPU core. When these bits specify the same priority level for simultaneously occurring interrupt requests, the source with the lower channel number takes priority. This order is fixed.

Note 1. n = 0 to 255

Note 2. Leave registers corresponding to channel numbers listed as reserved in **Section Table 6.11, Interrupt Exception Handlers and Orders of Priority**, at their values following a reset.

6.2.3 EI Level Interrupt Mask Registers 0 to 7 (IMR0 to IMR7)

These registers are a collection of the EIMK bits of the EIC registers. Each bit of IMRn reflects the setting of the corresponding EIMK bit. Setting IMRn is reflected in the corresponding EIMK bit.

IMR0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR0H	EIMK31	EIMK30	EIMK29	EIMK28	EIMK27	EIMK26	EIMK25	EIMK24	EIMK23	EIMK22	EIMK21	EIMK20	EIMK19	EIMK18	EIMK17	EIMK16
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR0L	EIMK15	EIMK14	EIMK13	EIMK12	EIMK11	EIMK10	EIMK9	EIMK8	EIMK7	EIMK6	EIMK5	EIMK4	EIMK3	EIMK2	EIMK1	EIMK0
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IMR1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR1H	EIMK63	EIMK62	EIMK61	EIMK60	EIMK59	EIMK58	EIMK57	EIMK56	EIMK55	EIMK54	EIMK53	EIMK52	EIMK51	EIMK50	EIMK49	EIMK48
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR1L	EIMK47	EIMK46	EIMK45	EIMK44	EIMK43	EIMK42	EIMK41	EIMK40	EIMK39	EIMK38	EIMK37	EIMK36	EIMK35	EIMK34	EIMK33	EIMK32
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

:

:

IMR7

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR7H	EIMK255	EIMK254	EIMK253	EIMK252	EIMK251	EIMK250	EIMK249	EIMK248	EIMK247	EIMK246	EIMK245	EIMK244	EIMK243	EIMK242	EIMK241	EIMK240
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR7L	EIMK239	EIMK238	EIMK237	EIMK236	EIMK235	EIMK234	EIMK233	EIMK232	EIMK231	EIMK230	EIMK229	EIMK228	EIMK227	EIMK226	EIMK225	EIMK224
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. EIMK bits corresponding to channel numbers listed as reserved in **Table 6.11, Interrupt Exception Handlers and Orders of Priority** must be set to 1.

6.2.4 EI Level Interrupt Bind Registers 0 to 255 (EIBD0 to EIBD255)

An EIBD register is provided per EI level interrupt and handles support for the PE and all sources. For each source, see **Table 6.11, Interrupt Exception Handlers and Orders of Priority**.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIBDnL	—	—	—	—	—	—	—	—	—	—	—	—	—	PEID		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	*1	*1	*1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 6.5 EIBD Register Contents

Bit Position	Bit Name	Function
31 to 18	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
17, 16	GPID	These bits are only present in EIBD32 to EIBD255. Used them to check the PEID settings as listed below. 00: The PEID bits are selecting the CPU1 as the target for binding. 01: The PEID bits are selecting the CPU2 as the target for binding. These bits are reserved in registers EIBD0 to EIBD31. When writing, set the value for these bits to 0. When read, the bits are always read as 0.
15 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2 to 0	PEID	These bits specify the destination to which the interrupt request is bound. However, the PEID bits, which are multiplexed per PE, in the EIBD0 to EIBD31 registers are fixed to each PE number, so the values of these bits cannot be changed in those registers. 001: The interrupt is bound to the CPU1. 010: The interrupt is bound to the CPU2. Always make either of the above settings if interrupts from the corresponding source are to be executed.

Note 1. EIBD32 to EIBD255: 001

Note: Leave registers corresponding to channel numbers listed as reserved in **Table 6.11, Interrupt Exception Handlers and Orders of Priority**, at their values following a reset.

CAUTION

Changing the value of the corresponding EIBDn register during the processing of an EIINT request is prohibited.

6.2.5 NMICTL — NMI Interrupt Control Register

This eight-bit register is used to specify the falling edges or rising edges as the sense for detection of the NMI input through the NMI pin.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	NMIS	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 6.6 NMICTL Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1, 0	NMIS	NMI Interrupt Sense Select These bits are used to select the falling edges or rising edges as the sense for detection of the NMI input through the NMI pin. 00: Interrupt requests are detected as falling edges on the NMI input. 01: Interrupt requests are detected as rising edges on the NMI input. Setting prohibited other than the above.

6.2.6 EXINTCTL — External Interrupt Control Register

This 16-bit register is used to specify the low level, the high level, falling edges, or rising edges as the sense for detection of the individual external interrupts input through the IRQ0 to IRQ7 pins.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7S		IRQ6S		IRQ5S		IRQ4S		IRQ3S		IRQ2S		IRQ1S		IRQ0S	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 6.7 EXINTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	IRQ7S	External Interrupt Sense Select Each pair of bits is used to select the low level, the high level, falling edges, or rising edges as the sense for detection of interrupt signals on a pin from among IRQ0 to IRQ7. 00: Interrupt requests are detected as the low level on the IRQn input.*1 01: Interrupt requests are detected as the high level on the IRQn input.*1 10: Interrupt requests are detected as falling edges on the IRQn input. 11: Interrupt requests are detected as rising edges on the IRQn input.
13, 12	IRQ6S	
11, 10	IRQ5S	
9, 8	IRQ4S	
7, 6	IRQ3S	
5, 4	IRQ2S	
3, 2	IRQ1S	
1, 0	IRQ0S	

Note 1. When the sense setting is for level detection, the active level should be retained until an interrupt request is accepted.

6.2.7 EXINTSTR — External Interrupt Status Register

EXINTSTR is an eight-bit register that indicates interrupt requests via the external interrupt input pins (IRQ0 to IRQ7). When the sense setting for a given interrupt from among IRQ0 to IRQ7 is for edge detection, register EXINTSTC can be used to clear the corresponding flag in this register.

Bit	7	6	5	4	3	2	1	0
	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 6.8 EXINTSTR Register Contents

Bit Position	Bit Name	Function
7	IRQ7F	External Interrupt Request These bits form a table that indicates the states of interrupt requests from IRQ0 to IRQ7. • When level detection is selected 0: An interrupt request is not present on IRQn. 1: An interrupt request is present on IRQn. [Clearing condition] The level selected by the IRQn1S and IRQn0S bits in the EXINTCTL register not being on the corresponding IRQn. [Setting condition] The level selected by the IRQn1S and IRQn0S bits in the EXINTCTL register being on the corresponding IRQn. • When edge detection is selected 0: An IRQn interrupt request has not been detected. 1: An IRQn interrupt request has been detected. [Clearing condition] 1 is written to an IRQnC bit in the EXINTSTC register. [Setting condition] The edge selected by the IRQn1S and IRQn0S bits in the EXINTCTL register having been generated.
6	IRQ6F	
5	IRQ5F	
4	IRQ4F	
3	IRQ3F	
2	IRQ2F	
1	IRQ1F	
0	IRQ0F	

Note: n = 0 to 7

6.2.8 EXINTSTC — External Interrupt Status Clear Register

Register EXINTSTC is an eight-bit register that is used to clear IRQnF bits in EXINTSTR for which edge detection has been selected as the method of sensing. When this is the case, writing 1 to an IRQnC bit clears the corresponding IRQnF in EXINTSTR if the interrupt has been detected.

Bit	7	6	5	4	3	2	1	0
	IRQ7C	IRQ6C	IRQ5C	IRQ4C	IRQ3C	IRQ2C	IRQ1C	IRQ0C
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 6.9 EXINTSTC Register Contents

Bit Position	Bit Name	Function
7	IRQ7C	External Interrupt Request Clear When edge detection has been selected as the method of sensing for an IRQ0 to IRQ7, the corresponding bit is used to clear the status flag that indicates the interrupt request. • When level detection is selected The bits have no function. • When edge detection is selected Writing 1 to an IRQnC bit clears the corresponding IRQnF in EXINTSTR.
6	IRQ6C	
5	IRQ5C	
4	IRQ4C	
3	IRQ3C	
2	IRQ2C	
1	IRQ1C	
0	IRQ0C	

Note: n = 0 to 7

6.2.9 SINTR0 to SINTR3 — Software Interrupt Register

These eight-bit registers are used to control software interrupts 0 to 3 (SINT0 to SINT3).

Writing 01H to one of these registers increments the value in the counter; writing 00H decrements it. If the resulting counter value is one or more, the corresponding interrupt from among software interrupts 0 to 3 (SINT0 to SINT3) is generated.

Reading produces the current value of the counter.

Bit	7	6	5	4	3	2	1	0
	SINTCn[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 6.10 SINTR Register Contents

Bit Position	Bit Name	Function
7 to 0	SINTCn[7:0]	Software Interrupt Request These bits are used to generate software interrupt requests. [Reading] The value counted number of times counted by the SINTn interrupt request counter is read out. [Writing] Writing 01_H increments the counter.*1 Writing 00_H decrements the counter.*2

Note 1. When 01_H is written to the register while the value of the counter is FF_H, the counter is not incremented and its value remains FF_H.

Note 2. When 00_H is written to the register while the value of the counter is 00_H, the counter is not decremented and its value remains 00_H.

6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers

The DTS transfer completion interrupts and transfer count match interrupts are merged in 32-channel units. PINT0 to PINT7 contain the interrupt status flags to indicate the originating channels for the actual interrupts. When multiple interrupt sources are generated within the same register among PINT0 to PINT7, only the single bit on the lower-order side is set.

Interrupt request flags that have been set are cleared from within the interrupt handlers by reading the value of the interrupt register from among PINT0 to PINT7 and writing to the register from among PINTCLR0 to PINTCLR7 that corresponds to the interrupt channel.

PINTn + x (n = 0 to 3; x = 0)

Bit	31	30	29	28	27	26	25	24
	INTDTS [31+32*n]	INTDTS [30+32*n]	INTDTS [29+32*n]	INTDTS [28+32*n]	INTDTS [27+32*n]	INTDTS [26+32*n]	INTDTS [25+32*n]	INTDTS [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTDTS [23+32*n]	INTDTS [22+32*n]	INTDTS [21+32*n]	INTDTS [20+32*n]	INTDTS [19+32*n]	INTDTS [18+32*n]	INTDTS [17+32*n]	INTDTS [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTDTS [15+32*n]	INTDTS [14+32*n]	INTDTS [13+32*n]	INTDTS [12+32*n]	INTDTS [11+32*n]	INTDTS [10+32*n]	INTDTS [9+32*n]	INTDTS [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTDTS [7+32*n]	INTDTS [6+32*n]	INTDTS [5+32*n]	INTDTS [4+32*n]	INTDTS [3+32*n]	INTDTS [2+32*n]	INTDTS [1+32*n]	INTDTS [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINTn + x (n = 0 to 3; x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTDTS [31+32*n]	INTCTDTS [30+32*n]	INTCTDTS [29+32*n]	INTCTDTS [28+32*n]	INTCTDTS [27+32*n]	INTCTDTS [26+32*n]	INTCTDTS [25+32*n]	INTCTDTS [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTCTDTS [23+32*n]	INTCTDTS [22+32*n]	INTCTDTS [21+32*n]	INTCTDTS [20+32*n]	INTCTDTS [19+32*n]	INTCTDTS [18+32*n]	INTCTDTS [17+32*n]	INTCTDTS [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTCTDTS [15+32*n]	INTCTDTS [14+32*n]	INTCTDTS [13+32*n]	INTCTDTS [12+32*n]	INTCTDTS [11+32*n]	INTCTDTS [10+32*n]	INTCTDTS [9+32*n]	INTCTDTS [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTCTDTS [7+32*n]	INTCTDTS [6+32*n]	INTCTDTS [5+32*n]	INTCTDTS [4+32*n]	INTCTDTS [3+32*n]	INTCTDTS [2+32*n]	INTCTDTS [1+32*n]	INTCTDTS [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINTCLRn + x (n = 0 to 3; x = 0)

Bit	31	30	29	28	27	26	25	24
	INTCLR [31+32*n]	INTCLR [30+32*n]	INTCLR [29+32*n]	INTCLR [28+32*n]	INTCLR [27+32*n]	INTCLR [26+32*n]	INTCLR [25+32*n]	INTCLR [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCLR [23+32*n]	INTCLR [22+32*n]	INTCLR [21+32*n]	INTCLR [20+32*n]	INTCLR [19+32*n]	INTCLR [18+32*n]	INTCLR [17+32*n]	INTCLR [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCLR [15+32*n]	INTCLR [14+32*n]	INTCLR [13+32*n]	INTCLR [12+32*n]	INTCLR [11+32*n]	INTCLR [10+32*n]	INTCLR [9+32*n]	INTCLR [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCLR [7+32*n]	INTCLR [6+32*n]	INTCLR [5+32*n]	INTCLR [4+32*n]	INTCLR [3+32*n]	INTCLR [2+32*n]	INTCLR [1+32*n]	INTCLR [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINTCLRn + x (n = 0 to 3; x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTCLR [31+32*n]	INTCTCLR [30+32*n]	INTCTCLR [29+32*n]	INTCTCLR [28+32*n]	INTCTCLR [27+32*n]	INTCTCLR [26+32*n]	INTCTCLR [25+32*n]	INTCTCLR [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCTCLR [23+32*n]	INTCTCLR [22+32*n]	INTCTCLR [21+32*n]	INTCTCLR [20+32*n]	INTCTCLR [19+32*n]	INTCTCLR [18+32*n]	INTCTCLR [17+32*n]	INTCTCLR [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCTCLR [15+32*n]	INTCTCLR [14+32*n]	INTCTCLR [13+32*n]	INTCTCLR [12+32*n]	INTCTCLR [11+32*n]	INTCTCLR [10+32*n]	INTCTCLR [9+32*n]	INTCTCLR [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCTCLR [7+32*n]	INTCTCLR [6+32*n]	INTCTCLR [5+32*n]	INTCTCLR [4+32*n]	INTCTCLR [3+32*n]	INTCTCLR [2+32*n]	INTCTCLR [1+32*n]	INTCTCLR [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINT0

Bit Position	Bit Name	Function
31 to 0	INTDTS[31:0]	States of DTS transfer completion interrupts from channel 31 to channel 0

PINT1

Bit Position	Bit Name	Function
31 to 0	INTDTS[63:32]	States of DTS transfer completion interrupts from channel 63 to channel 32

PINT2

Bit Position	Bit Name	Function
31 to 0	INTDTS[95:64]	States of DTS transfer completion interrupts from channel 95 to channel 64

PINT3

Bit Position	Bit Name	Function
31 to 0	INTDTS [127:96]	States of DTS transfer completion interrupts from channel 127 to channel 96

PINT4

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [31:0]	States of DTS transfer count match interrupts from channel 31 to channel 0

PINT5

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [63:32]	States of DTS transfer count match interrupts from channel 63 to channel 32

PINT6

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [95:64]	States of DTS transfer count match interrupts from channel 95 to channel 64

PINT7

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [127:96]	States of DTS transfer count match interrupts from channel 127 to channel 96

PINTCLR0

Bit Position	Bit Name	Function
31 to 0	INTCLR[31:0]	The respective bits clear the corresponding flags for the DTS channel 31 to channel 0 transfer completion interrupts. Write the value read from PINT0 to this register from within the interrupt handler.

PINTCLR1

Bit Position	Bit Name	Function
31 to 0	INTCLR[63:32]	The respective bits clear the corresponding flags for the DTS channel 63 to channel 32 transfer completion interrupts. Write the value read from PINT1 to this register from within the interrupt handler.

PINTCLR2

Bit Position	Bit Name	Function
31 to 0	INTCLR[95:64]	The respective bits clear the corresponding flags for the DTS channel 95 to channel 64 transfer completion interrupts. Write the value read from PINT2 to this register from within the interrupt handler.

PINTCLR3

Bit Position	Bit Name	Function
31 to 0	INTCLR[127:96]	The respective bits clear the corresponding flags for the DTS channel 127 to channel 96 transfer completion interrupts. Write the value read from PINT3 to this register from within the interrupt handler.

PINTCLR4

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[31:0]	The respective bits clear the corresponding flags for the DTS channel 31 to channel 0 transfer count match interrupts. Write the value read from PINT4 to this register from within the interrupt handler.

PINTCLR5

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [63:32]	The respective bits clear the corresponding flags for the DTS channel 63 to channel 32 transfer count match interrupts. Write the value read from PINT5 to this register from within the interrupt handler.

PINTCLR6

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [95:64]	The respective bits clear the corresponding flags for the DTS channel 95 to channel 64 transfer count match interrupts. Write the value read from PINT6 to this register from within the interrupt handler.

PINTCLR7

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [127:96]	The respective bits clear the corresponding flags for the DTS channel 127 to channel 96 transfer count match interrupts. Write the value read from PINT7 to this register from within the interrupt handler.

6.3 Interrupt Sources

The five types of interrupt source are external interrupts (NMI, IRQ), ECM interrupt, inter-processor interrupt, software interrupt, and peripheral module interrupt.

6.3.1 NMI Interrupts

NMI refers to external interrupt requests input via the NMI pins. The settings of the NMI interrupt sense select bits (NMIS) in the NMI interrupt control register (NMICTL) select the falling edges, or rising edges.

NMI interrupts are accepted at the highest priority even when other FE level interrupt has occurred and cannot be masked regardless of the state of the PSW.NP in the CPU system register. They are non-maskable interrupts from which return or recovery is not possible.

6.3.2 IRQ Interrupts

IRQ interrupts refers to interrupt requests input via pins IRQ0 to IRQ7. The settings of the external interrupt sense select bits (IRQ7S to IRQ0S) in the external interrupt control (EXINTCTL) register select the low level, the high level, falling edges, or rising edges for detection as the IRQ interrupt on each of the corresponding pins. Furthermore, the interrupt control registers set the priority up to 16 levels for each interrupt source.

When the low level is selected for detection as an IRQ interrupt, an interrupt request signal is conveyed to the interrupt controller (INTC) while the low level is on the corresponding pin from among IRQ0 to IRQ7. The request signal is not conveyed when the level becomes high. The active level should be retained until acceptance of the interrupt request. Checking for requests can be done by reading the corresponding IRQ interrupt request bit (IRQ0F to IRQ7F) in the external interrupt status register (EXINTSTR).

When edge detection is selected for an IRQ interrupt, an interrupt request signal is conveyed to the interrupt controller (INTC) when the corresponding change in the level on the given pin from among IRQ0 to IRQ7 occurs.

Reading bits EXINTSTR.IRQ7F to EXINTSTR.IRQ0F checks whether the corresponding IRQ interrupt request is detected. In addition, the IRQ interrupt request can be cleared by writing 1 to the bit corresponding to EXINTSTC when an edge is detected.

When returning from the IRQ interrupt exception handler, clear the external interrupt status register (EXINTSTR) to prevent re-acceptance by mistake, and then issue an instruction to return from interrupt.

6.3.3 ECM Interrupts

The Error Control Module (ECM) generates ECM interrupt requests by merging error interrupts from multiple sources. For details, refer to **Section 28, Error Control Module (ECM)**.

6.3.4 Inter-Processor Interrupts

Four registers (IPIR_CHn) for conveying interrupts between CPUs are provided.

IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for specific CPUs (including own CPU) can be requested by manipulating bits corresponding to respective CPUs.
The order of priority can be set for each source in 16 levels by using the interrupt control registers.

6.3.5 Software Interrupts

The software interrupt (SINT) is generated by setting the SINTR0 to SINTR3 registers. Multiple interrupt requests can be queued.
The order of priority for software interrupts is set for each source in 16 levels by using interrupt control registers.

6.3.6 Internal Peripheral Module Interrupts

Internal peripheral module interrupts are interrupts from the internal peripheral modules listed below.

- Code flash and data flash
- Serial communications interface 3 (SCI3)
- OS Timer (OSTM)
- Window watchdog timer (WDTA)
- Motor control timer (TSG3)
- A/D converter (ADCC)
- Clocked Serial Interface H (CSIH)
- CAN interface (RS-CAN)
- LIN master interface (RLIN2)
- Direct memory access controller (DMAC, DTS)
- Enhanced motor control unit (EMU2)
- R/D converter (RDC2)
- Timer array unit J (TAUJ)
- Timer array unit D (TAUD)
- Encoder timer (ENCA)
- Timer option (TAPA)
- Timer pattern buffer (TPBA)
- Peripheral interconnection 1 (PIC1A)

Since each source is allocated a different interrupt vector, there is no need to determine the source from the interrupt exception handling routines. The order of priority for interrupts is set for each source in 16 levels.

6.4 Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority

Interrupt sources, the corresponding source codes, address offsets of the handlers, and interrupt priority levels are listed in **Table 6.11**.

Along with the standard specification for the addresses of handlers, where these are determined by the settings of the PSW.EBV bits and the RBASE and EBASE registers within the CPU core, an extended specification where fixed addresses of the handlers for each of the interrupts are set per channel is also available.

In the standard specification, the address of each exception handler is the offset address plus the base address (set in the RBASE and EBASE registers) within the CPU core.

Interrupt offsets are determined through the two methods described below. Put briefly, the offset address is specified or determined by the interrupt channel.

- In one method, the address is determined as the priority level set for each channel (0 to 15) plus a value in the range from 100H to 1F0H regardless of the interrupt channel number (see note 1 under **Table 6.11**).
- In the other method, the addresses are evenly spread across the range of offsets up to 100H regardless of the priority level. This method has the advantage of reducing the amount of memory taken up for exception handling (see note 2 under **Table 6.11**).

In the extended specification, a table of the addresses of the exception handlers for each of the interrupt channels is kept, and the addresses of the handlers are extracted from this table. The locations for reference in the table are obtained from the formula below (see note 3 under **Table 6.11**). INTBP is the value in the register of that name within the CPU core.

Location for reading the address of the exception handler = value in the INTBP register + the channel number × 4 (in bytes)

See *the RH850G3M User's Manual: Software for RH850G3M* exceptions.

For system error exceptions, see **Section 3, CPU System**.

The orders of priority are set per channel to values. When the same priority level is set for two sources, the channel numbers are used to provide fixed priority levels for the sources, and that with the smaller channel number is selected.

CAUTION

You must assign the SYNCP instruction to the first of exception handler of FPI, SYSERR, EIINT (Direct Vector Method), FEINT, FENMI. For the details, see *the RH850G3M User's Manual: Software for RH850G3M*.

Table 6.11 Interrupt Exception Handlers and Orders of Priority (1/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address: Direct Branching RINT = 0*2	Offset Address: Direct Branching RINT = 1*2	Offset Address: Reference to a Table*2	Interrupt Priority Order (Value after Reset)	Target Device: C1H	Target Device: C1M	Default Priority Order
Non-maskable interrupt	NMI interrupt		(FENMI)	$E0_H$	$+0E0_H$	$+0E0_H$	—		√	√	Higher priority
FE level interrupt	Error control module (ECM) FE level interrupts		(FEINT)	$F0_H$	$+0F0_H$	$+0F0_H$	—		√	√	
Inter-processor interrupts	IPIR_CH0		0	1000	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+000_H$	0 to 15 (15)	√	×	
	IPIR_CH1		1	1001			$+004_H$	0 to 15 (15)	√	×	
	IPIR_CH2		2	1002			$+008_H$	0 to 15 (15)	√	×	
	IPIR_CH3		3	1003			$+00C_H$	0 to 15 (15)	√	×	
	Reserved		4	1004					×	×	
	Reserved		5	1005					×	×	
	Reserved		6	1006					×	×	
	Reserved		7	1007					×	×	
Error control module	Error control module (ECM) maskable interrupt		8	1008			$+020_H$	0 to 15 (15)	√	√	
IRQ (External interrupts)	IRQ0 interrupt	√	9	1009			$+024_H$	0 to 15 (15)	√	√	
	IRQ1 interrupt	√	10	100A			$+028_H$	0 to 15 (15)	√	√	
	IRQ2 interrupt	√	11	100B			$+02C_H$	0 to 15 (15)	√	√	
	IRQ3 interrupt	√	12	100C			$+030_H$	0 to 15 (15)	√	√	
	IRQ4 interrupt	√	13	100D			$+034_H$	0 to 15 (15)	√	√	
	IRQ5 interrupt	√	14	100E			$+038_H$	0 to 15 (15)	√	√	
	IRQ6 interrupt	√	15	100F			$+03C_H$	0 to 15 (15)	√	√	
	IRQ7 interrupt	√	16	1010			$+040_H$	0 to 15 (15)	√	√	
SINT (software interrupts)	SINT0 interrupt	√	17	1011			$+044_H$	0 to 15 (15)	√	√	
	SINT1 interrupt	√	18	1012			$+048_H$	0 to 15 (15)	√	√	
	SINT2 interrupt	√	19	1013			$+04C_H$	0 to 15 (15)	√	√	
	SINT3 interrupt	√	20	1014			$+050_H$	0 to 15 (15)	√	√	
	Reserved		21	1015					×	×	
	Reserved		22	1016					×	×	
	Reserved		23	1017					×	×	
	Reserved		24	1018					×	×	
OSTM	OSTM0 interrupt (OSTM0TINT)		25	1019			$+064_H$	0 to 15 (15)	√	√	
	OSTM1 interrupt (OSTM1TINT)		26	101A			$+068_H$	0 to 15 (15)	√	√	
	OSTM2 interrupt (OSTM2TINT)		27	101B			$+06C_H$	0 to 15 (15)	√	×	
	Reserved		28	101C					×	×	
DMA	DMA transfer error (DMAERR)		29	101D			$+074_H$	0 to 15 (15)	√	√	
	Reserved		30	101E					×	×	
	Reserved		31	101F					×	×	
Code flash, data flash	Reserved		32	1020					×	×	
	Flash sequencer processing completion interrupt*3		33	1021			$+084_H$	0 to 15 (15)	√	√	
	Reserved		34	1022					×	×	
	Reserved		35	1023					×	×	
	Reserved		36	1024					×	×	
	Reserved		37	1025					×	×	
DMAC	Channel 0 transfer completion interrupt/ channel 0 transfer count match interrupt		38	1026			$+098_H$	0 to 15 (15)	√	√	
	Channel 1 transfer completion interrupt/ channel 1 transfer count match interrupt		39	1027			$+09C_H$	0 to 15 (15)	√	√	
	Channel 2 transfer completion interrupt/ channel 2 transfer count match interrupt		40	1028			$+0A0_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (2/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address: Direct Branching RINT = 0*2	Offset Address: Direct Branching RINT = 1*2	Offset Address: Reference to a Table*2	Interrupt Priority Order (Value after Reset)	Target Device C1H	Target Device C1M	Default Priority Order
DMAC	Channel 3 transfer completion interrupt/ channel 3 transfer count match interrupt		41	1029	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+0A4_H$	0 to 15 (15)	√	√	Higher priority
	Channel 4 transfer completion interrupt/ channel 4 transfer count match interrupt		42	102A			$+0A8_H$	0 to 15 (15)	√	√	
	Channel 5 transfer completion interrupt/ channel 5 transfer count match interrupt		43	102B			$+0AC_H$	0 to 15 (15)	√	√	
	Channel 6 transfer completion interrupt/ channel 6 transfer count match interrupt		44	102C			$+0B0_H$	0 to 15 (15)	√	√	
	Channel 7 transfer completion interrupt/ channel 7 transfer count match interrupt		45	102D			$+0B4_H$	0 to 15 (15)	√	√	
	Channel 8 transfer completion interrupt/ channel 8 transfer count match interrupt		46	102E			$+0B8_H$	0 to 15 (15)	√	√	
	Channel 9 transfer completion interrupt/ channel 9 transfer count match interrupt		47	102F			$+0BC_H$	0 to 15 (15)	√	√	
	Channel 10 transfer completion interrupt/ channel 10 transfer count match interrupt		48	1030			$+0C0_H$	0 to 15 (15)	√	√	
	Channel 11 transfer completion interrupt/ channel 11 transfer count match interrupt		49	1031			$+0C4_H$	0 to 15 (15)	√	√	
	Channel 12 transfer completion interrupt/ channel 12 transfer count match interrupt		50	1032			$+0C8_H$	0 to 15 (15)	√	√	
	Channel 13 transfer completion interrupt/ channel 13 transfer count match interrupt		51	1033			$+0CC_H$	0 to 15 (15)	√	√	
	Channel 14 transfer completion interrupt/ channel 14 transfer count match interrupt		52	1034			$+0D0_H$	0 to 15 (15)	√	√	
	Channel 15 transfer completion interrupt/ channel 15 transfer count match interrupt		53	1035			$+0D4_H$	0 to 15 (15)	√	√	
WDTA	WDTA0TIT interval timer interrupt (75% interrupt)		54	1036			$+0D8_H$	0 to 15 (15)	√	√	
	WDTA1TIT interval timer interrupt (75% interrupt)		55	1037			$+0DC_H$	0 to 15 (15)	√	×	
EMU2	EMU20 interrupt 0		56	1038			$+0E0_H$	0 to 15 (15)	√	√	
	EMU20 interrupt 1		57	1039			$+0E4_H$	0 to 15 (15)	√	√	
	EMU20 interrupt 2		58	103A			$+0E8_H$	0 to 15 (15)	√	√	
	EMU20 interrupt 3		59	103B			$+0EC_H$	0 to 15 (15)	√	√	
	EMU20 interrupt 4		60	103C			$+0F0_H$	0 to 15 (15)	√	√	
	EMU21 interrupt 0		61	103D			$+0F4_H$	0 to 15 (15)	√	√	
	EMU21 interrupt 1		62	103E			$+0F8_H$	0 to 15 (15)	√	√	
	EMU21 interrupt 2		63	103F			$+0FC_H$	0 to 15 (15)	√	√	
	EMU21 interrupt 3		64	1040			$+100_H$	0 to 15 (15)	√	√	
	EMU21 interrupt 4		65	1041			$+104_H$	0 to 15 (15)	√	√	
RDC_0	RDC20 Z-phase interrupt		66	1042			$+108_H$	0 to 15 (15)	√	√	
	RDC20 error interrupt		67	1043			$+10C_H$	0 to 15 (15)	√	√	
	RDC20 Compare 0 match interrupt		68	1044			$+110_H$	0 to 15 (15)	√	√	
	RDC20 Compare 1 match interrupt		69	1045			$+114_H$	0 to 15 (15)	√	√	
	RDC20 Compare 2 match interrupt		70	1046			$+118_H$	0 to 15 (15)	√	√	
	RDC20 Excitation timer (ET) interrupt		71	1047			$+11C_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (3/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address: Direct Branching RINT = 0*2	Offset Address: Direct Branching RINT = 1*2	Offset Address: Reference to a Table*2	Interrupt Priority Order (Value after Reset)	Target Device: C1H	Target Device: C1M	Default Priority Order
RDC_1	RDC21 Z-phase interrupt		72	1048	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+120_H$	0 to 15 (15)	√	×	Higher priority
	RDC21 error interrupt		73	1049			$+124_H$	0 to 15 (15)	√	×	
	RDC21 Compare 0 match interrupt		74	104A			$+128_H$	0 to 15 (15)	√	×	
RDC_1	RDC21 Compare 1 match interrupt		75	104B			$+12C_H$	0 to 15 (15)	√	×	
	RDC21 Compare 2 match interrupt		76	104C			$+130_H$	0 to 15 (15)	√	×	
	RDC21 Excitation timer (ET) interrupt		77	104D			$+134_H$	0 to 15 (15)	√	×	
TAUJ_0	INTTAUJ0I0 interrupt		78	104E			$+138_H$	0 to 15 (15)	√	√	
	INTTAUJ0I1 interrupt		79	104F			$+13C_H$	0 to 15 (15)	√	√	
	INTTAUJ0I2 interrupt		80	1050			$+140_H$	0 to 15 (15)	√	√	
	INTTAUJ0I3 interrupt		81	1051			$+144_H$	0 to 15 (15)	√	√	
TAUD	INTTAUD0I0 interrupt		82	1052			$+148_H$	0 to 15 (15)	√	√	
	INTTAUD0I1 interrupt		83	1053			$+14C_H$	0 to 15 (15)	√	√	
	INTTAUD0I2 interrupt		84	1054			$+150_H$	0 to 15 (15)	√	√	
	INTTAUD0I3 interrupt		85	1055			$+154_H$	0 to 15 (15)	√	√	
	INTTAUD0I4 interrupt		86	1056			$+158_H$	0 to 15 (15)	√	√	
	INTTAUD0I5 interrupt		87	1057			$+15C_H$	0 to 15 (15)	√	√	
	INTTAUD0I6 interrupt		88	1058			$+160_H$	0 to 15 (15)	√	√	
	INTTAUD0I7 interrupt		89	1059			$+164_H$	0 to 15 (15)	√	√	
	INTTAUD0I8 interrupt		90	105A			$+168_H$	0 to 15 (15)	√	√	
	INTTAUD0I9 interrupt		91	105B			$+16C_H$	0 to 15 (15)	√	√	
	INTTAUD0I10 interrupt		92	105C			$+170_H$	0 to 15 (15)	√	√	
	INTTAUD0I11 interrupt		93	105D			$+174_H$	0 to 15 (15)	√	√	
	INTTAUD0I12 interrupt		94	105E			$+178_H$	0 to 15 (15)	√	√	
	INTTAUD0I13 interrupt		95	105F			$+17C_H$	0 to 15 (15)	√	√	
	INTTAUD0I14 interrupt		96	1060			$+180_H$	0 to 15 (15)	√	√	
	INTTAUD0I15 interrupt		97	1061			$+184_H$	0 to 15 (15)	√	√	
	INTTAUD1I0 interrupt		98	1062			$+188_H$	0 to 15 (15)	√	√	
	INTTAUD1I1 interrupt		99	1063			$+18C_H$	0 to 15 (15)	√	√	
	INTTAUD1I2 interrupt		100	1064			$+190_H$	0 to 15 (15)	√	√	
	INTTAUD1I3 interrupt		101	1065			$+194_H$	0 to 15 (15)	√	√	
	INTTAUD1I4 interrupt		102	1066			$+198_H$	0 to 15 (15)	√	√	
	INTTAUD1I5 interrupt		103	1067			$+19C_H$	0 to 15 (15)	√	√	
	INTTAUD1I6 interrupt		104	1068			$+1A0_H$	0 to 15 (15)	√	√	
	INTTAUD1I7 interrupt		105	1069			$+1A4_H$	0 to 15 (15)	√	√	
	INTTAUD1I8 interrupt		106	106A			$+1A8_H$	0 to 15 (15)	√	√	
	INTTAUD1I9 interrupt		107	106B			$+1AC_H$	0 to 15 (15)	√	√	
	INTTAUD1I10 interrupt		108	106C			$+1B0_H$	0 to 15 (15)	√	√	
	INTTAUD1I11 interrupt		109	106D			$+1B4_H$	0 to 15 (15)	√	√	
	INTTAUD1I12 interrupt		110	106E			$+1B8_H$	0 to 15 (15)	√	√	
	INTTAUD1I13 interrupt		111	106F			$+1BC_H$	0 to 15 (15)	√	√	
	INTTAUD1I14 interrupt		112	1070			$+1C0_H$	0 to 15 (15)	√	√	
	INTTAUD1I15 interrupt		113	1071			$+1C4_H$	0 to 15 (15)	√	√	
	Reserved		114	1072					×	×	
	Reserved		115	1073					×	×	
PIC1A	ENCA0 compare 0 match or capture 0 interrupt		116	1074			$+1D0_H$	0 to 15 (15)	√	√	
	ENCA1 compare 0 match or capture 0 interrupt		117	1075			$+1D4_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (4/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address: Direct Branching RINT = 0*2	Offset Address: Direct Branching RINT = 1*2	Offset Address: Reference to a Table*2	Interrupt Priority Order (Value after Reset)	Target Device: C1H	Target Device: C1M	Default Priority Order
ENCA_0	Overflow interrupt		118	1076	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+1D8_H$	0 to 15 (15)	√	√	Higher priority
	Compare 1 match or capture 1 interrupt		119	1077			$+1DC_H$	0 to 15 (15)	√	√	
	Underflow interrupt		120	1078			$+1E0_H$	0 to 15 (15)	√	√	
	Encoder clear interrupt		121	1079			$+1E4_H$	0 to 15 (15)	√	√	
ENCA_1	Overflow interrupt		122	107A			$+1E8_H$	0 to 15 (15)	√	√	
	Compare 1 match or capture 1 interrupt		123	107B			$+1EC_H$	0 to 15 (15)	√	√	
	Underflow interrupt		124	107C			$+1F0_H$	0 to 15 (15)	√	√	
	Encoder clear interrupt		125	107D			$+1F4_H$	0 to 15 (15)	√	√	
TAPA	TAPA0 peak interrupt 0		126	107E			$+1F8_H$	0 to 15 (15)	√	√	
	TAPA0 trough interrupt 0		127	107F			$+1FC_H$	0 to 15 (15)	√	√	
	TAPA1 peak interrupt 0		128	1080			$+200_H$	0 to 15 (15)	√	√	
	TAPA1 trough interrupt 0		129	1081			$+204_H$	0 to 15 (15)	√	√	
TPBA_0	Cycle match detection interrupt		130	1082			$+208_H$	0 to 15 (15)	√	√	
	Duty match detection interrupt		131	1083			$+20C_H$	0 to 15 (15)	√	√	
	Pattern count match detection interrupt		132	1084			$+210_H$	0 to 15 (15)	√	√	
TPBA_1	Cycle match detection interrupt		133	1085			$+214_H$	0 to 15 (15)	√	×	
	Duty match detection interrupt		134	1086			$+218_H$	0 to 15 (15)	√	×	
	Pattern count match detection interrupt		135	1087			$+21C_H$	0 to 15 (15)	√	×	
TSG3_0	TSG30 compare match interrupt 1 (INTTSG30I1)		136	1088			$+220_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 2 (INTTSG30I2)		137	1089			$+224_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 3 (INTTSG30I3)		138	108A			$+228_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 4 (INTTSG30I4)		139	108B			$+22C_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 5 (INTTSG30I5)		140	108C			$+230_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 6 (INTTSG30I6)		141	108D			$+234_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 7 (INTTSG30I7)		142	108E			$+238_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 8 (INTTSG30I8)		143	108F			$+23C_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 9 (INTTSG30I9)		144	1090			$+240_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 10 (INTTSG30I10)		145	1091			$+244_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 11 (INTTSG30I11)		146	1092			$+248_H$	0 to 15 (15)	√	√	
	TSG30 compare match interrupt 12 (INTTSG30I12)		147	1093			$+24C_H$	0 to 15 (15)	√	√	
	TSG30 error interrupt (INTTSG30IER)		148	1094			$+250_H$	0 to 15 (15)	√	√	
	TSG30 warning interrupt (INTTSG30IWN)		149	1095			$+254_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (5/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address: Direct Branching RINT = 0*2	Offset Address: Direct Branching RINT = 1*2	Offset Address: Reference to a Table*2	Interrupt Priority Order (Value after Reset)	Target Device: C1H	Target Device: C1M	Default Priority Order
TSG3_1	TSG31 compare match interrupt 1 (INTTSG31I1)		150	1096	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+258_H$	0 to 15 (15)	√	√	Higher priority
	TSG31 compare match interrupt 2 (INTTSG31I2)		151	1097			$+25C_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 3 (INTTSG31I3)		152	1098			$+260_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 4 (INTTSG31I4)		153	1099			$+264_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 5 (INTTSG31I5)		154	109A			$+268_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 6 (INTTSG31I6)		155	109B			$+26C_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 7 (INTTSG31I7)		156	109C			$+270_H$	0 to 15 (15)	√	√	
TSG3_1	TSG31 compare match interrupt 8 (INTTSG31I8)		157	109D			$+274_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 9 (INTTSG31I9)		158	109E			$+278_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 10 (INTTSG31I10)		159	109F			$+27C_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 11 (INTTSG31I11)		160	10A0			$+280_H$	0 to 15 (15)	√	√	
	TSG31 compare match interrupt 12 (INTTSG31I12)		161	10A1			$+284_H$	0 to 15 (15)	√	√	
	TSG31 error interrupt (INTTSG31IER)		162	10A2			$+288_H$	0 to 15 (15)	√	√	
	TSG31 warning interrupt (INTTSG31IWN)		163	10A3			$+28C_H$	0 to 15 (15)	√	√	
TSG3_0	TSG30 compare match interrupt 0 (INTTSG30I0)		164	10A4			$+290_H$	0 to 15 (15)	√	√	
	TSG30 peak interrupt (INTTSG30IPEK)		165	10A5			$+294_H$	0 to 15 (15)	√	√	
	TSG30 trough interrupt (INTTSG30IVLY)		166	10A6			$+298_H$	0 to 15 (15)	√	√	
TSG3_1	TSG31 compare match interrupt 0 (INTTSG31I0)		167	10A7			$+29C_H$	0 to 15 (15)	√	√	
	TSG31 peak interrupt (INTTSG31IPEK)		168	10A8			$+2A0_H$	0 to 15 (15)	√	√	
	TSG31 trough interrupt (INTTSG31IVLY)		169	10A9			$+2A4_H$	0 to 15 (15)	√	√	
	Reserved		170	10AA					×	×	
	Reserved		171	10AB					×	×	
ADCC	ADI00 ADCC0 scan group 0 completion interrupt		172	10AC			$+2B0_H$	0 to 15 (15)	√	√	
	ADI01 ADCC0 scan group 1 completion interrupt		173	10AD			$+2B4_H$	0 to 15 (15)	√	√	
	ADI02 ADCC0 scan group 2 completion interrupt		174	10AE			$+2B8_H$	0 to 15 (15)	√	√	
	ADI03 ADCC0 scan group 3 completion interrupt		175	10AF			$+2BC_H$	0 to 15 (15)	√	√	
	ADI04 ADCC0 scan group 4 completion interrupt		176	10B0			$+2C0_H$	0 to 15 (15)	√	√	
	ADI10 ADCC1 scan group 0 completion interrupt		177	10B1			$+2C4_H$	0 to 15 (15)	√	√	
	ADI11 ADCC1 scan group 1 completion interrupt		178	10B2			$+2C8_H$	0 to 15 (15)	√	√	
	ADI12 ADCC1 scan group 2 completion interrupt		179	10B3			$+2CC_H$	0 to 15 (15)	√	√	
	ADI13 ADCC1 scan group 3 completion interrupt		180	10B4			$+2D0_H$	0 to 15 (15)	√	√	
	ADI14 ADCC1 scan group 4 completion interrupt		181	10B5			$+2D4_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (6/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address			Interrupt Priority Order (Value after Reset)	Target Device		Default Priority Order
					Direct Branching		Reference to a Table*2				
					RINT = 0*2	RINT = 1*2			C1H	C1M	
	Reserved		182	10B6	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.			×	×	Higher priority
	Reserved		183	10B7					×	×	
ADCC	ADE0 ADCC0AD error interrupt		184	10B8			$+2E0_H$	0 to 15 (15)	√	√	
	ADE1 ADCC1AD error interrupt		185	10B9			$+2E4_H$	0 to 15 (15)	√	√	
RS-CAN	Reception FIFO interrupt	√	186	10BA			$+2E8_H$	0 to 15 (15)	√	√	
	Global error interrupt	√	187	10BB			$+2EC_H$	0 to 15 (15)	√	√	
RS-CAN-ch0	transmit/receive FIFO receive complete interrupt	√	188	10BC			$+2F0_H$	0 to 15 (15)	√	√	
	Error interrupt	√	189	10BD			$+2F4_H$	0 to 15 (15)	√	√	
	Transmission interrupt	√	190	10BE			$+2F8_H$	0 to 15 (15)	√	√	
RS-CAN-ch1	transmit/receive FIFO receive complete interrupt	√	191	10BF			$+2FC_H$	0 to 15 (15)	√	√	
	Error interrupt	√	192	10C0			$+300_H$	0 to 15 (15)	√	√	
	Transmission interrupt	√	193	10C1			$+304_H$	0 to 15 (15)	√	√	
RS-CAN-ch2	Transmit/receive FIFO receive complete interrupt	√	194	10C2			$+308_H$	0 to 15 (15)	√	√	
	Error interrupt	√	195	10C3			$+30C_H$	0 to 15 (15)	√	√	
	Transmission interrupt	√	196	10C4			$+310_H$	0 to 15 (15)	√	√	
RS-CAN-ch3	Transmit/receive FIFO receive complete interrupt	√	197	10C5			$+314_H$	0 to 15 (15)	√	√	
	Error interrupt	√	198	10C6			$+318_H$	0 to 15 (15)	√	√	
	Transmission interrupt	√	199	10C7			$+31C_H$	0 to 15 (15)	√	√	
RLIN2_0	RLIN20 transmit complete/receive complete/error detect interrupt	√	200	10C8			$+320_H$	0 to 15 (15)	√	×	
RLIN2_1	RLIN21 transmit complete/receive complete/error detect interrupt	√	201	10C9			$+324_H$	0 to 15 (15)	√	×	
RLIN2_2	RLIN22 transmit complete/receive complete/error detect interrupt	√	202	10CA			$+328_H$	0 to 15 (15)	√	×	
SCI_0	ERI (receive error)	√	203	10CB			$+32C_H$	0 to 15 (15)	√	√	
	RXI (receive data full)		204	10CC			$+330_H$	0 to 15 (15)	√	√	
	TXI (transmit data empty)		205	10CD			$+334_H$	0 to 15 (15)	√	√	
	TEI (transmission completion)	√	206	10CE			$+338_H$	0 to 15 (15)	√	√	
SCI_1	ERI (receive error)	√	207	10CF			$+33C_H$	0 to 15 (15)	√	√	
	RXI (receive data full)		208	10D0			$+340_H$	0 to 15 (15)	√	√	
	TXI (transmit data empty)		209	10D1			$+344_H$	0 to 15 (15)	√	√	
	TEI (transmission completion)	√	210	10D2			$+348_H$	0 to 15 (15)	√	√	
SCI_2	ERI (receive error)	√	211	10D3			$+34C_H$	0 to 15 (15)	√	√	
	RXI (receive data full)		212	10D4			$+350_H$	0 to 15 (15)	√	√	
	TXI (transmit data empty)		213	10D5			$+354_H$	0 to 15 (15)	√	√	
	TEI (transmission completion)	√	214	10D6			$+358_H$	0 to 15 (15)	√	√	
CSIH_0	Communication status interrupt (INTCSIHTIC)		215	10D7			$+35C_H$	0 to 15 (15)	√	√	
	Reception status interrupt (INTCSIHTIR)		216	10D8			$+360_H$	0 to 15 (15)	√	√	
	Communication error interrupt (INTCSIHTIRE)		217	10D9			$+364_H$	0 to 15 (15)	√	√	
	Job completion interrupt (INTCSIHTIJC)		218	10DA			$+368_H$	0 to 15 (15)	√	√	

Table 6.11 Interrupt Exception Handlers and Orders of Priority (7/7)

Function/Module	Interrupt	Level interrupt *1	EIINT Interrupt Channel	Source Code	Offset Address			Interrupt Priority Order (Value after Reset)	Target Device		Default Priority Order
					Direct Branching		Reference to a Table*2		C1H	C1M	
					RINT = 0*2	RINT = 1*2					
CSIH_1	Communication status interrupt (INTCSIHTIC)		219	10DB	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels	The offset address is always $+100_H$ regardless of priority	$+36C_H$	0 to 15 (15)	√	√	Higher priority
	Reception status interrupt (INTCSIHTIR)		220	10DC			$+370_H$	0 to 15 (15)	√	√	
	Communication error interrupt (INTCSIHTIRE)		221	10DD			$+374_H$	0 to 15 (15)	√	√	
	Job completion interrupt (INTCSIHTIJC)		222	10DE			$+378_H$	0 to 15 (15)	√	√	
DTS	ch31-0 transfer completion interrupt	√	223	10DF			$+37C_H$	0 to 15 (15)	√	√	
	ch63-32 transfer completion interrupt	√	224	10E0			$+380_H$	0 to 15 (15)	√	√	
	ch95-64 transfer completion interrupt	√	225	10E1			$+384_H$	0 to 15 (15)	√	√	
	ch127-96 transfer completion interrupt	√	226	10E2			$+388_H$	0 to 15 (15)	√	√	
DTS	ch31-0 transfer count match interrupt	√	227	10E3			$+38C_H$	0 to 15 (15)	√	√	
	ch63-32 transfer count match interrupt	√	228	10E4			$+390_H$	0 to 15 (15)	√	√	
	ch95-64 transfer count match interrupt	√	229	10E5			$+394_H$	0 to 15 (15)	√	√	
	ch127-96 transfer count match interrupt	√	230	10E6			$+398_H$	0 to 15 (15)	√	√	

Note 1. Withdrawing an interrupt request in the form of a level interrupt requires clearing the given bit in the status register of the corresponding module from within the interrupt processing routing. Furthermore, the value of the EICn.EICTn bit will be 1. The EICn.EIRFn bit cannot be directly cleared by software.

Note 2. See **Section 6.4, Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority**.

Note 3. See the *RH850/C1x Flash Memory User's Manual: Hardware Interface*.

6.5 Operation

6.5.1 External Interrupts (NMI and IRQ)

The two types of externally input interrupt are the NMI and IRQ. Two sensing methods are available for NMI whereas four sensing methods are available for IRQ. Regarding the flow of interrupt detection, see **Section 6.5.5, Flow of Interrupt Processing**.

6.5.2 Inter-Processor Interrupts

For operational description, see **Section 3.2.3.1, Inter-Processor Interrupt Control Registers** and **Section 6.5.5, Flow of Interrupt Processing**.

6.5.3 Software Interrupts

Regarding operations in relation to these interrupts, see **Section 6.2.9, SINTR0 to SINTR3 — Software Interrupt Register** for a description of the registers, and **Section 6.5.5, Flow of Interrupt Processing**.

6.5.4 Merging of DTS Interrupts

Each set of 32 of the 128 transfer completion interrupts and 128 transfer count match interrupts from the DTS is gathered to produce a single interrupt signal.

When multiple interrupt source conditions are satisfied, only the single bit on the lower-order side of the corresponding bits in the status registers (PINT0 to PINT7) is set to judge which interrupt is accepted. For the flow of interrupts in the merging of DTS interrupts, see **Section 6.5.5.5, Flow of Processing for DTS Interrupts**.

Table 6.12 Interrupt Related Registers of the DTS

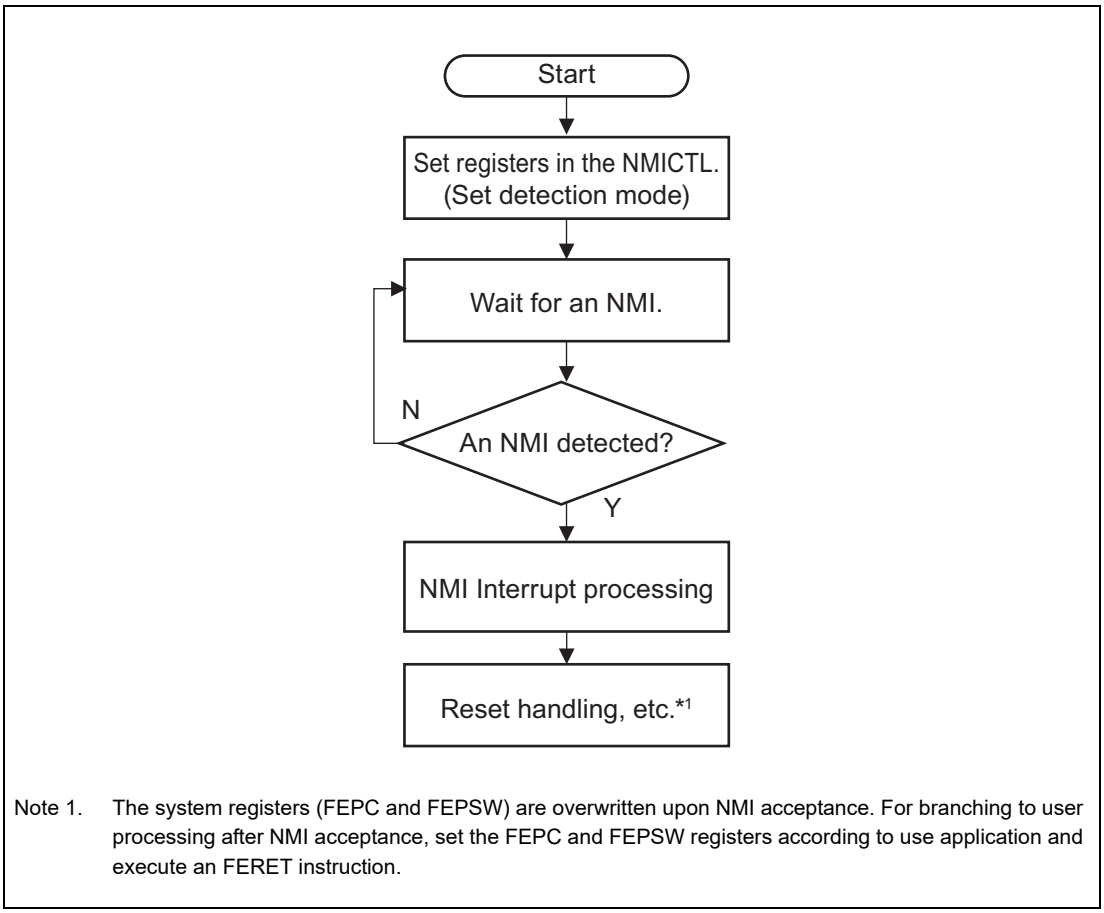
Interrupt Sources	Channel	Status Register	Clear Register
DTS transfer completion interrupt	0 to 31	PINT0	PINTCLR0
	32 to 63	PINT1	PINTCLR1
	64 to 95	PINT2	PINTCLR2
	96 to 127	PINT3	PINTCLR3
DTS transfer count match interrupt	0 to 31	PINT4	PINTCLR4
	32 to 63	PINT5	PINTCLR5
	64 to 95	PINT6	PINTCLR6
	96 to 127	PINT7	PINTCLR7

6.5.5 Flow of Interrupt Processing

6.5.5.1 Flow of Processing for NMI

Figure 6.1 shows an example of the flow of NMI processing.

- The settings of the NMICTL register select the method of detection (fall edge or rise edge) for the NMI.
- After detection of an NMI, the interrupt request is output to the INTC.
- NMI interrupts are accepted at high priority even when other FE level interrupt has occurred and cannot be masked regardless of the state of the PSW.NP in the CPU system register. They are non-maskable interrupts from which return or recovery is not possible.



Note 1. The system registers (FEPC and FEPSW) are overwritten upon NMI acceptance. For branching to user processing after NMI acceptance, set the FEPC and FEPSW registers according to use application and execute an FERET instruction.

Figure 6.1 Example of NMI Processing Flow

6.5.5.2 Flow of Processing for External Interrupts

Figure 6.2 shows an example of the flow of IRQ (external interrupt) processing.

- Set the EXINTCTL register to select the method of detection (edge detection or level detection) for the IRQ.
- After detection of an IRQ, an interrupt request is issued to the INTC.
- After recovery on completion of interrupt processing within the INTC in the case of level detection, confirm that the IRQn pin is negated before issuing the instruction to return from the interrupt.
- After recovery on completion of interrupt processing within the INTC in the case of edge detection, clear the interrupt request bit in the EXINTSTR register, and then issue the instruction to return from the interrupt.

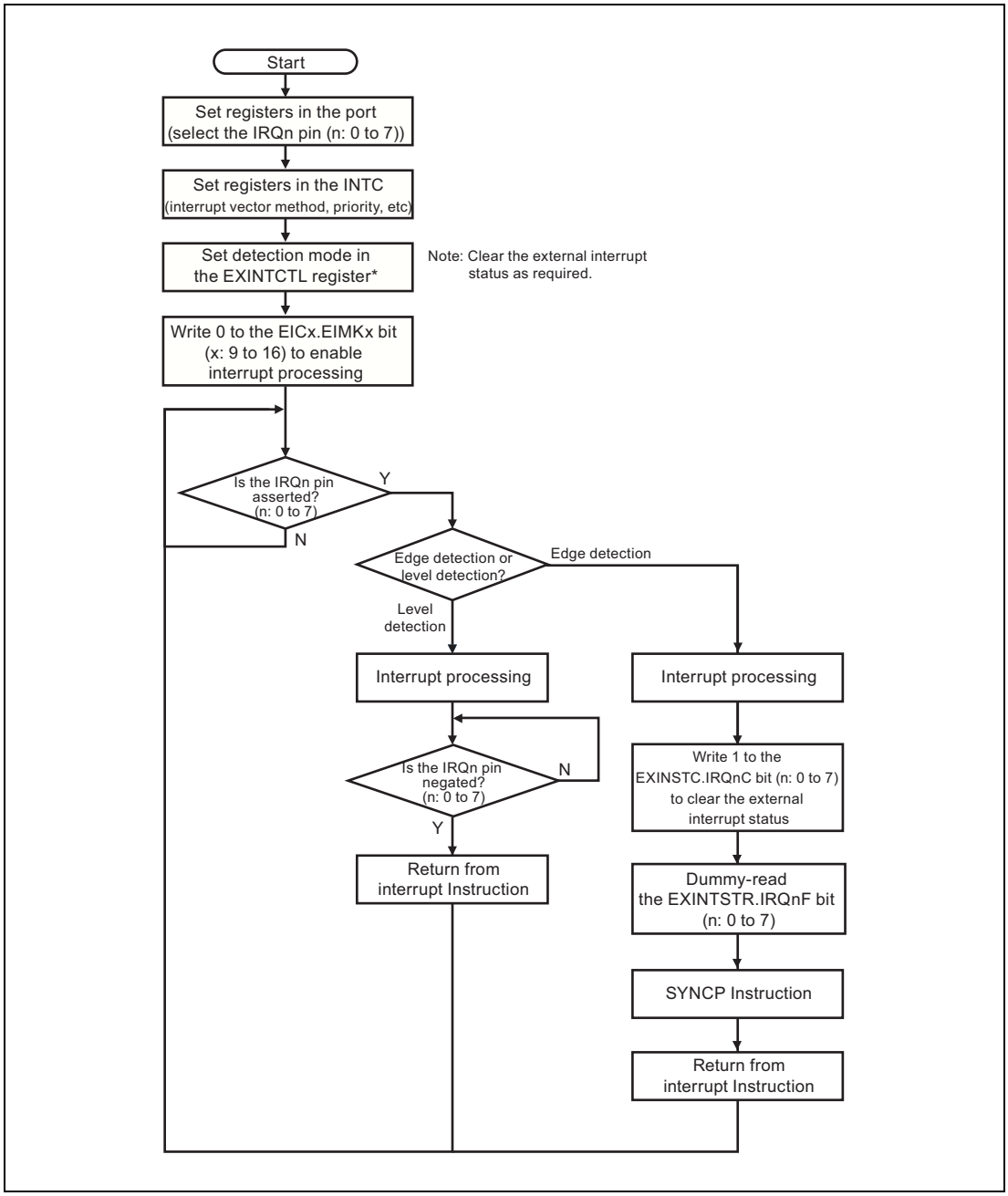


Figure 6.2 Example of External Interrupt Processing Flow

6.5.5.3 Inter-Processor Interrupt Flow

Figure 6.3 shows an example of the flow of inter-processor interrupt processing.

- Inter-processor interrupting generates an interrupt request by writing 1 to applicable bits of PE to which interrupts of the inter-PE interrupt registers (IPIR0 to IPIR3) are requested.
- The settings of interrupt request of the inter-PE interrupt registers (IPIR0 to IPIR3) are automatically cleared to 0 after notification of an interrupt request is complete.

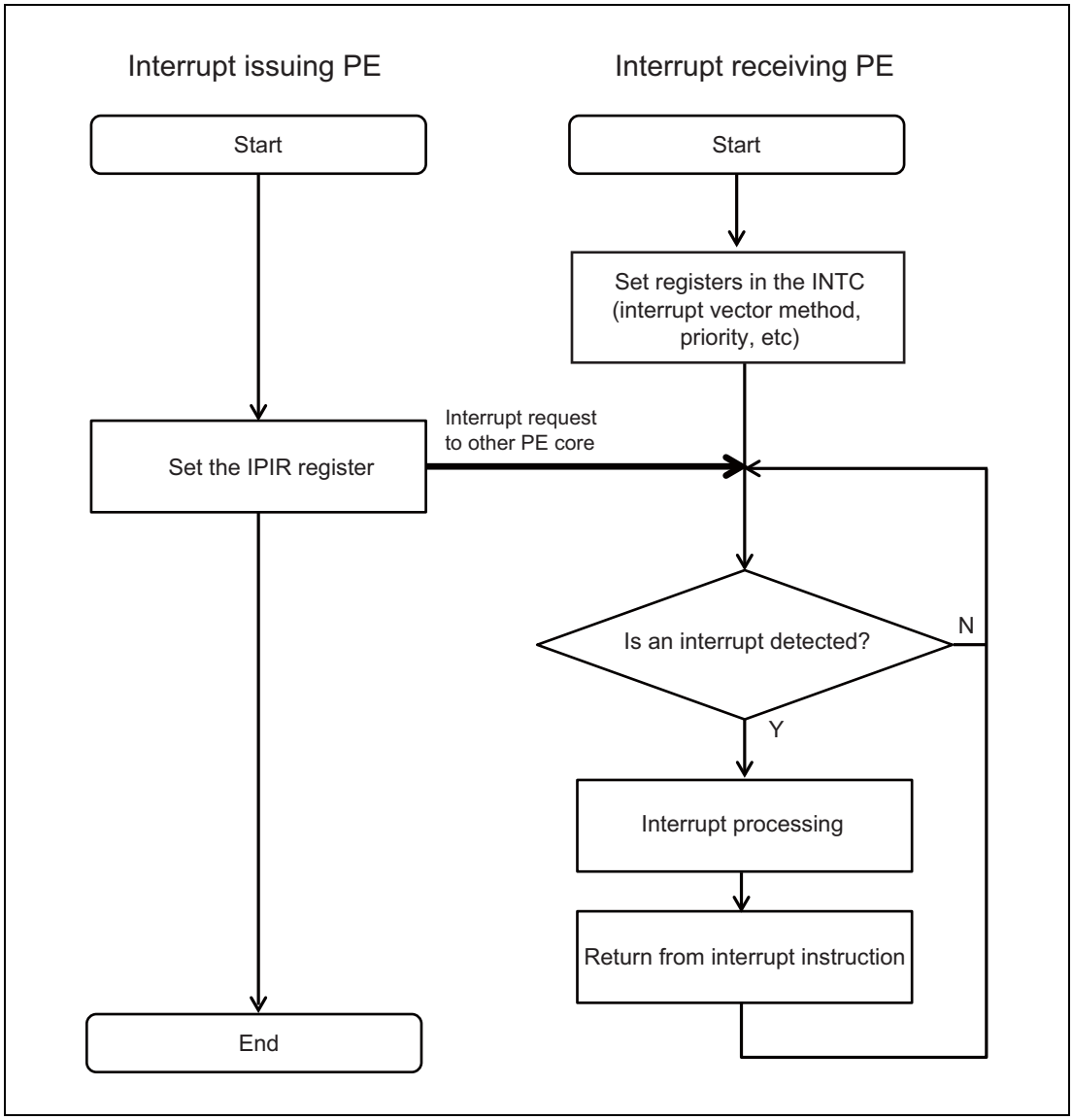


Figure 6.3 Example of Inter-Processor Interrupt Processing Flow

6.5.5.4 Software Interrupt Processing Flow

Figure 6.4 shows an example of the flow of software interrupt processing.

- Software interrupt requests are controlled by writing 00_H or 01_H to the counter registers (SINTR0 to SINTR3).
- Writing 00_H leads to the counter's value being decremented by 1.
- Writing 01_H leads to the counter's value being incremented by 1.
- If the incremented counter value is 1 or greater, an interrupt request for the INTC is generated.
- Decrement the counter by 1 during interrupt processing in the INTC, and if SINTRn is 00_H after issuing the instruction to return from the interrupt, wait for the writing of 01_H to SINTRn.

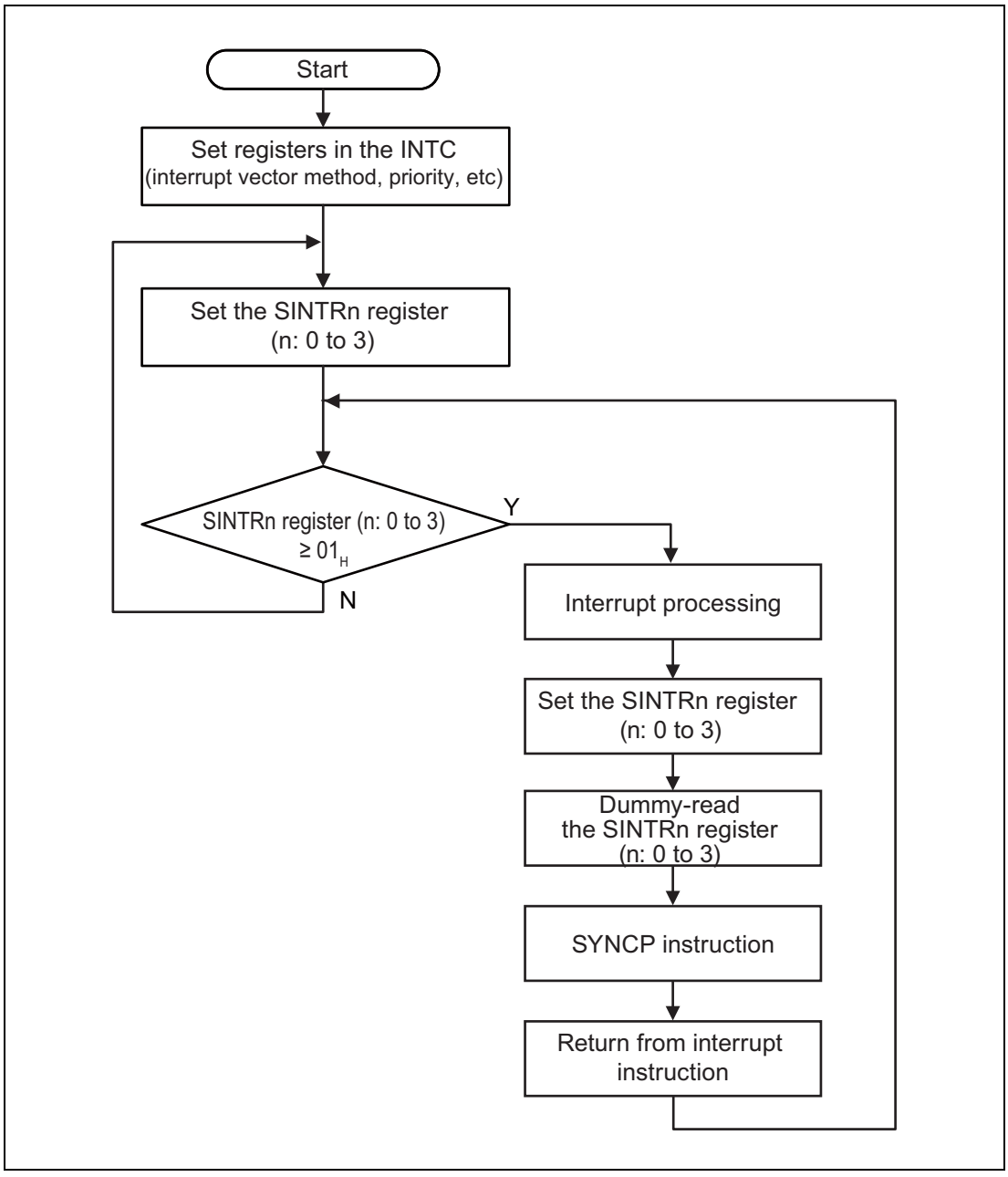


Figure 6.4 Example of Software Interrupt Processing Flow

6.5.5.5 Flow of Processing for DTS Interrupts

Figure 6.5 shows an example of the flow of DTS interrupt processing.

- In the case of a single interrupt request from a set of 32 actual sources
  - After the bit corresponding to the interrupt request in a PINTn register is set to 1, the interrupt request is output.
  - On completion of interrupt processing, 1 is written to the corresponding bit in a clearing register (PINTCLRn), the interrupt request is cleared before issuing the return from interrupt instruction, then waiting for a next interrupt commences.
- In the case of multiple interrupt requests from a set of 32 actual sources
  - The bit on the lower-order side among the multiple bits for the interrupt requests is extracted, only the corresponding bit in a PINTn register is set to 1, and the interrupt request is output.
  - On completion of interrupt processing, 1 is written to the corresponding bit in a clearing register (PINTCLRn), clearing the interrupt request bits, then issuing the return from interrupt instruction.
  - After clearing the interrupt request bits for which interrupt processing has proceeded, in the same way as the previous time, the bit on the lower-order side of the PINTn register that corresponds to an interrupt request is extracted and an interrupt request is issued.
  - This process is repeated until none of the 32 bits for the set of interrupt sources remains set.

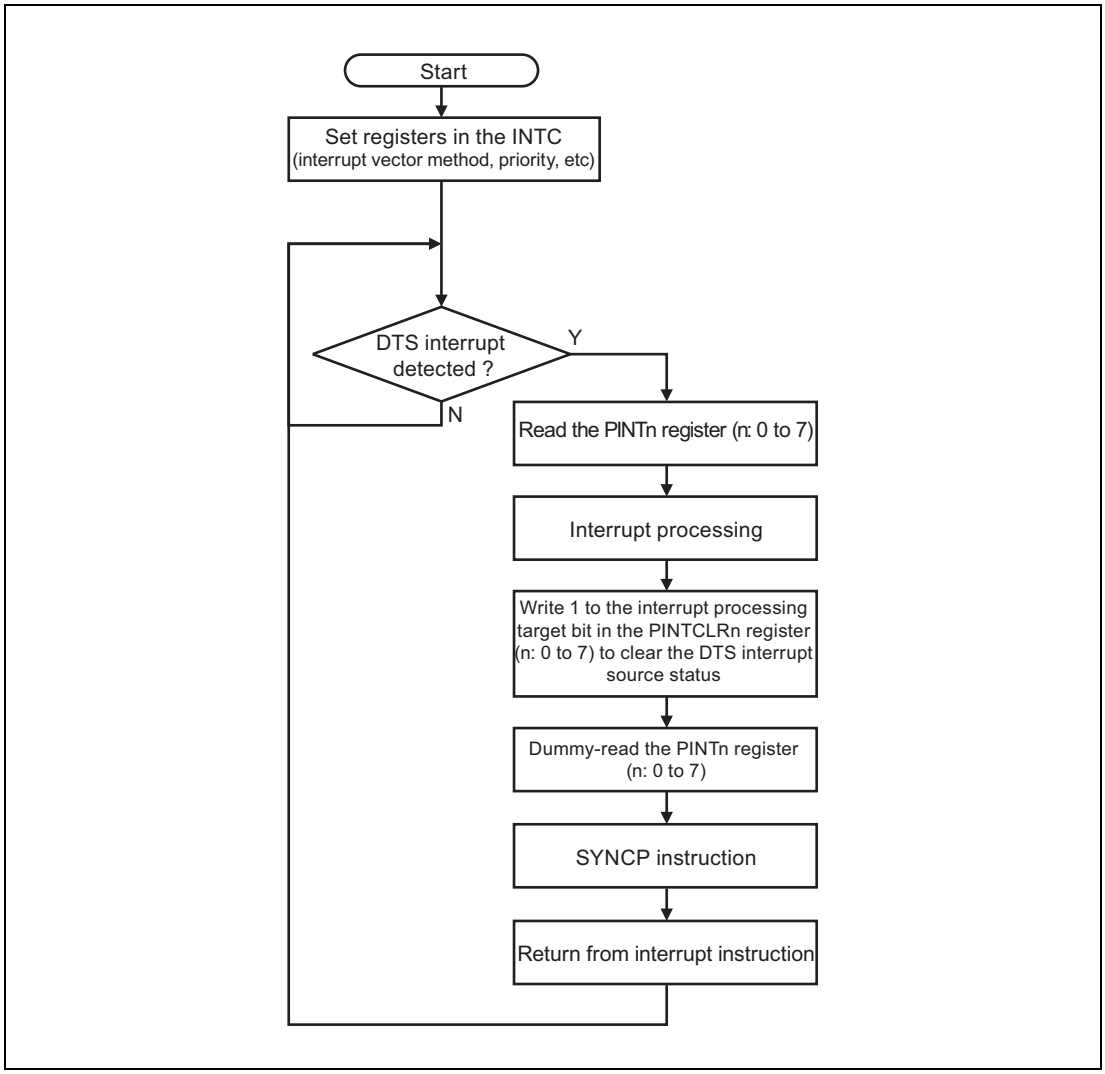


Figure 6.5 Example of DTS Interrupt Processing Flow

6.6 Interrupt Response Times

Table 6.13 Interrupt Response Times (min.)

	Interrupt Request Source		Number of Cycles for Processing				
Target	INTC Connection	Operating Clock	Synchro -nization	INTC2	INTC1	in CPU1/ in CPU2	Total*1
CPU	Directly input to INTC1	High-speed / low-speed peripheral clock	0	—	2 × Iϕ <1 × Iϕ>	See the description under “in CPU1/CPU2” below.	7 × Iϕ
		Unmodulated high-speed peripheral clock	—				—
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				5 × Pϕ + 7 × Iϕ
	Input via INTC2	High-speed / low-speed peripheral clock	0	3 × Pϕ + 1 × Iϕ <2 × Pϕ + 1 × Iϕ>	—		3 × Pϕ + 6 × Iϕ
		Unmodulated high-speed peripheral clock	4 × Pϕ <2 × Pϕ>				7 × Pϕ + 6 × Iϕ
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				8 × Pϕ + 6 × Iϕ

Note: The numbers in < > indicate the numbers of cycles in the case of level detection.
Iϕ = CLK_CPU, Pϕ = 3Iϕ at 240 MHz

Note 1. In the case of edge detection by fixed vector method a).

Vector Method	Cache HIT/MISS	in CPU1/CPU2 (240 MHz)	Vector method
Fixed vector method	a) ISR ENTRY I$ HIT	5 × Iϕ	a) Fixed vector method
	b) ISR ENTRY I$ MISS	10 × Iϕ	
Vector table reference method	c) Vector code flash assigned, ISR ENTRY I$ HIT	14 × Iϕ	b) Vector table reference method Code flash assigned
	d) Vector code flash assigned, ISR ENTRY I$ MISS	19 × Iϕ	

Note: Iϕ = CLK_CPU, Pϕ = 3Iϕ at 240 MHz

6.7 Data Transfer in Response to Interrupt Request Signals

An interrupt request signal can be used to activate the DMAC or DTS for handling data transfer. For details, see **Section 7, DMA Controller**.

Section 7 DMA Controller

7.1 Overview

7.1.1 Overview

The direct memory access (DMA) controller is used to move data without using the CPU.

DMA consists of two types of DMA transfer modules: DMAC and DTS. A DMAC includes registers for storing transfer information, and a DTS stores transfer information in the dedicated RAM (DTSRAM). DMA has two 8-channel DMAC modules and one 128-channel DTS module.

In this manual, DTFR denotes the function to select among hardware DMA transfer sources for a DMAC and retain the DMA transfer request, and DTSFSL denotes the function to retain a DMA transfer request for each DTS channel. The DTFR and DTSFSL can respectively handle 128 types of hardware DMA transfer sources.

The address space that can be used for DMA transfer is a 4 GB address space represented by a 32-bit address. For information about which resource is assigned to a particular area in the 4 GB address space and which area is accessible from DMA, see **Section 4, Address Space**.

7.1.2 Term Definition

Table 7.1 shows the terms used in this section.

Table 7.1 List of Term Definitions

Term	Meaning
DMA transfer	A general term for data transfer carried out by DMA.
DMA cycle	A series of actions that consist of reading an amount of data specified by the transfer size (8/16/32/64/128 bits) from the address specified by the source address and writing it to the address specified by the destination address. The first half of the DMA cycle (reading part) is called a read cycle, and the second half (writing part) is called a write cycle.
Hardware DMA transfer source	A trigger for a DMA transfer request given by an internal peripheral device.
Hardware DMA transfer request	A DMA transfer request generated by a hardware DMA transfer source
Software DMA transfer request	A DMA transfer request generated by software writing to a register.
DMA transfer request	A trigger to start DMA transfer with a DMAC and DTS.
Transfer information (TI)	The information required for DMA transfer, including the source address, destination address, transfer data size, and transfer count. The transfer information for a DTS is specifically termed as TI.
DTSRAM	RAM used by a DTS to store the transfer information.
Single transfer	DMA transfer consisting of one DMA cycle started by one DMA transfer request.
Block transfer 1	DMA transfer consisting of a number of DMA cycles specified by the transfer count in the transfer information, started by one DMA transfer request.
Block transfer 2	DMA transfer consisting of a number of DMA cycles specified by the address reload count in the transfer information, started by one DMA transfer request.
Block transfer	A general term for both block transfer 1 and block transfer 2.
Last transfer	The DMA cycle carried out when the transfer count in the transfer information is 1.
Address reload transfer	The DMA cycle carried out when the address reload count in the transfer information is 1 if the reload function 2 is used.
Suspension	An action of pausing DMA transfer during block transfer. You can resume DMA transfer after suspension.
Resume	An action of resuming suspended DMA transfer.
Transfer abort	An action of aborting DMA transfer in the middle. You cannot resume DMA transfer after that.

7.2 DMA Function

7.2.1 Basic Operation of DMA Transfer

7.2.1.1 Transfer Mode

DMA has three transfer modes.

Single Transfer

One DMA cycle is executed when a DMA transfer request is accepted.

Block Transfer 1

A number of DMA cycles specified in the transfer count register are executed when a DMA transfer request is accepted.

Block Transfer 2

A number of DMA cycles specified by the address reload count are executed when a DMA transfer request is accepted. If the address reload count is larger than the value in the transfer count register, a number of DMA cycles specified in the transfer count register are executed.

7.2.1.2 Executing a DMA Cycle

In dual-address DMA, the data transfer takes two cycles. DMA always executes a write cycle after a read cycle is complete. For example, if the transfer data size is 128 bits, a write cycle is executed after a read cycle for the 128-bit data is finished. A write cycle never starts in the middle of a read cycle.

7.2.1.3 Updating Transfer Information

When a DMA cycle is executed, DMA updates transfer information as follows.

Source Address and Destination Address

Transfer information will be updated as described in **Table 7.2** according to the settings for the source address and destination address and the settings in the transfer control register such as the direction of counting for source and destination addresses and transfer data size.

Table 7.2 Updating the Source Address and the Destination Address

Direction of Counting	Transfer Data Size	Address after Update
Up	8 bits	(address before update) + 0000_0001_H
	16 bits	(address before update) + 0000_0002_H
	32 bits	(address before update) + 0000_0004_H
	64 bits	(address before update) + 0000_0008_H
	128 bits	(address before update) + 0000_0010_H
Down	8 bits	(address before update) - 0000_0001_H
	16 bits	(address before update) - 0000_0002_H
	32 bits	(address before update) - 0000_0004_H
	64 bits	(address before update) - 0000_0008_H
	128 bits	(address before update) - 0000_0010_H
Fixed	—	Same as the address before update.

When you use the reload function, a special update rule is applied other than the one described in **Table 7.2** for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Transfer Count and Address Reload Count

The transfer count is decremented by one for every DMA cycle.

The address reload count is decremented by one for every DMA cycle when the reload function 2 or block transfer 2 is used. When the reload function 2 or block transfer 2 is not used, it is not updated.

When you use the reload function, a special update rule is applied for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Other transfer information

Other transfer information is not updated during execution of a DMA cycle.

7.2.1.4 Last Transfer and Address Reload Transfer

The last transfer means a DMA cycle executed when the value in the transfer count register, which shows the remaining number of transfers, is one. The last transfer differs in operation compared to other DMA cycles as follows.

- The transfer completion flag (DCSTn.TC) is set when the last transfer is complete. (Only applicable for a DMAC)
- The channel operation enable (DCENn.DTE) bit is cleared when the last transfer is complete. (Only applicable for a DMAC. When the continuous transfer is disabled.)
- When the transfer completion interrupt output enable is set, a transfer completion interrupt is output when the last transfer is complete.
- When the reload function 1 is enabled, the reload function 1 is executed at the timing of the last transfer. For details, see **Section 7.2.3, Reload Function**.

The address reload transfer means a DMA cycle executed when the reload function 2 is enabled and the address reload count is one. The reload function 2 is executed during the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

7.2.1.5 Transfer Completion Interrupt and Transfer Count Match Interrupt Outputs

DMA can output a transfer completion interrupt and a transfer count match interrupt.

Transfer Completion Interrupt Output

When the transfer completion interrupt output enable (DTCTn.TCE) is set in the transfer control register, a DMAC requests a DMAC transfer completion interrupt when the last transfer is complete.

When the transfer completion interrupt output enable (DTTCTn.TCE) is set in the transfer control register, a DTS requests a DTS transfer completion interrupt when the last transfer is complete.

Transfer Count Match Interrupt Output

When the transfer count match interrupt enable (DTCTn.CCE) is set in the transfer control register, a DMAC requests a DMAC transfer count match interrupt when the DMA cycle in which the transfer count compare register and the transfer count have the same value is complete.

When the transfer count match interrupt enable (DTTCTn.CCE) is set in the transfer control register, a DTS requests a DTS transfer count match interrupt at the completion of the DMA cycle in which the transfer count compare register and the transfer count have the same value.

Figure 7.1 shows the operation of the transfer completion interrupt and the transfer count match interrupt.

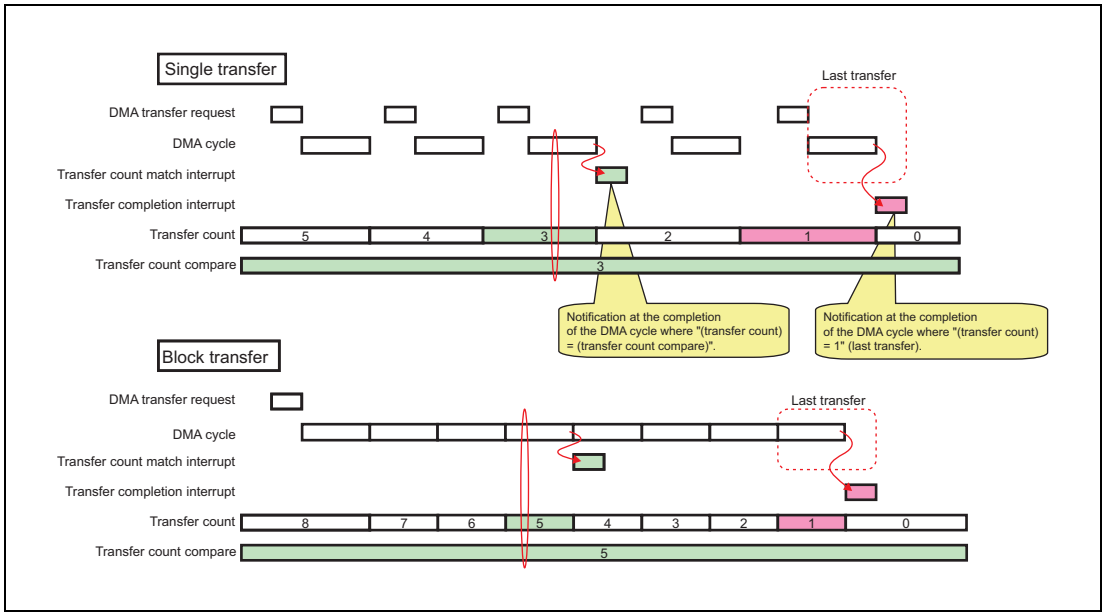


Figure 7.1 Transfer Completion Interrupt and Transfer Count Match Interrupt

7.2.1.6 Continuous Transfer

If the continuous transfer is not used, a DMAC sets the transfer completion flag (DCSTn.TC) and clears the channel operation enable (DCENn.DTE) bit when the last transfer is complete. In this case, a DMA transfer request is not accepted when the request is generated after the completion of the last transfer.

If the continuous transfer is used, the channel operation enable (DCENn.DTE) bit is not cleared when the last transfer is complete, and a DMA transfer request can be accepted even when the transfer completion flag is set. If DMA is used for a case where a specified number of DMA transfers are executed repetitively, software overhead associated with clearing the transfer completion flag and setting the channel operation enable bit after the completion of the last transfer can be reduced by using the continuous transfer.

The continuous transfer is enabled by setting the continuous transfer enable (DTCTn.MLE) in the DMAC transfer control register.

The continuous transfer is designed to work with the reload function 1. The continuous transfer function itself does not update the source address register, destination address register, and transfer count register. If, after the last transfer is complete, you want to restore the source address register, destination address register, and transfer count register to the state before the DMA transfer starts, use the reload function 1 and set the values of the source address register, destination address register, and the transfer count register before the DMA transfer starts to the reload source address register, reload destination address register, and reload transfer count register respectively.

A DTS does not have a setting corresponding to the continuous transfer enable (DTCTn.MLE) for a DMAC. This is because a DTS does not have bits like the transfer completion flag (DCSTn.TC) and the channel operation enable (DCENn.DTE) a DMAC has.

A DTS does not start DMA transfer even when a DMA transfer request is generated while the transfer count is 0.

If the reload function 1 is used for a DTS and the value other than 0 is reloaded to the transfer count when the last transfer is complete, DMA transfer can start when the next DMA transfer request is accepted. (This corresponds to the case for a DMAC where the continuous transfer is used.)

Figure 7.2 shows an operation of continuous transfer by a DMAC.

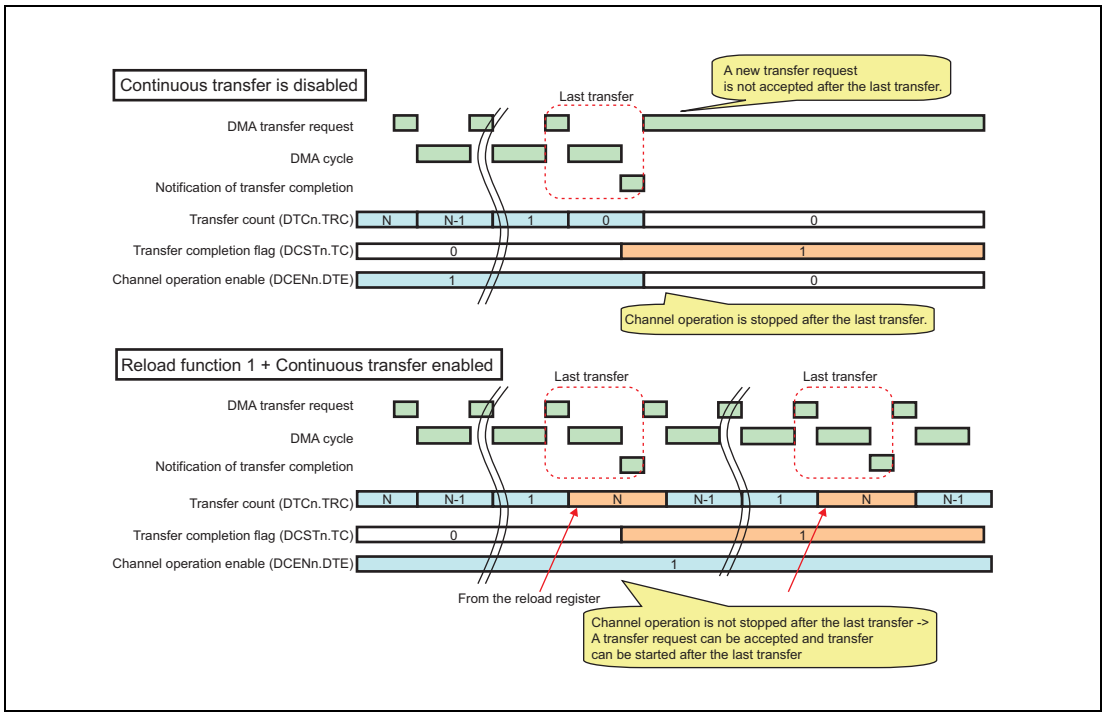


Figure 7.2 Operation of Continuous Transfer by a DMAC

7.2.2 Channel Priority Order

This subsection explains arbitration between multiple DMA channels.

7.2.2.1 DMAC Channel Arbitration

A DMAC select one channel out of eight channels with arbitration. Arbitration is done according to the fixed priority order. The priority order is “channel 0 > channel 1 > channel 2 > channel 3 > channel 4 > channel 5 > channel 6 > channel 7” for DMAC0, and “channel 8 > channel 9 > channel 10 > channel 11 > channel 12 > channel 13 > channel 14 > channel 15” for DMAC1.

Arbitration is done for every DMA cycle. No arbitration occurs between the read and write of a DMA cycle.

If, at the timing when one DMA cycle in the middle of a block transfer of a channel is complete, there is a DMA transfer request from a channel with a higher priority than the channel, a DMA cycle of the channel with the higher priority will be executed next as the result of arbitration.

If a DMAC executes the block transfer 1 or block transfer 2, DMAC channel arbitration is done for every DMA cycle, and possibly a DMA cycle of another DMAC channel with a higher priority can cut in.

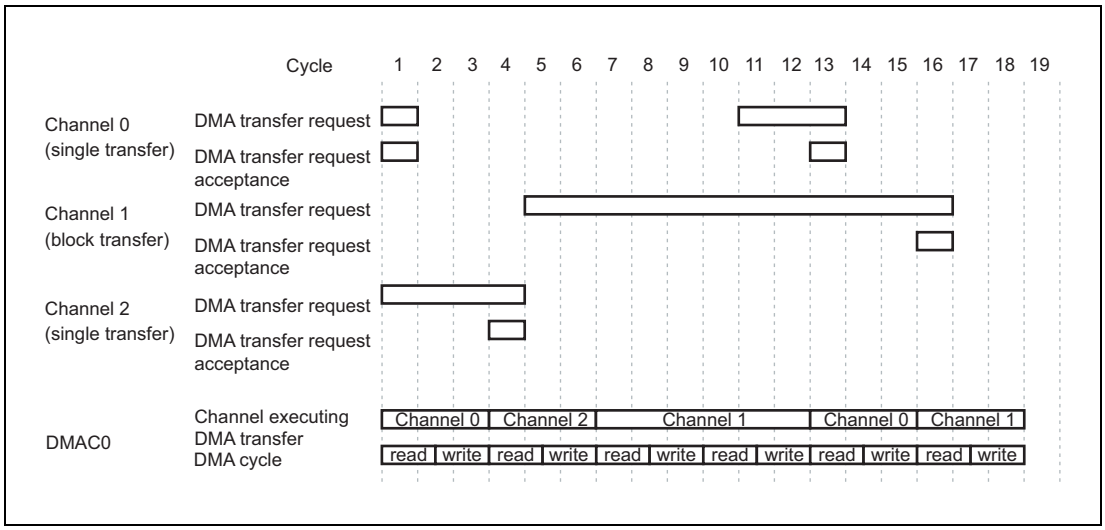


Figure 7.3 DMAC Channel Arbitration

Cycle numbers shown in **Figure 7.3** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.3**, DMA transfer requests for channels 0 and 2 are generated at Cycle 1. As a result of arbitration, a DMA cycle for channel 0 starts because its priority is higher. At Cycle 4, a DMA cycle for channel 2 starts. At Cycle 5, a DMA transfer request for channel 1 is generated. The DMA cycle for channel 2 is still ongoing and no arbitration is done at this point. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done at this point. At Cycle 12, the DMA cycle for channel 1 is complete. At Cycle 13, a DMA cycle for channel 0 starts as a result of arbitration between DMA channels 0 and 1.

It should be noted that, even though a block transfer of channel 1 has been already started, a DMA cycle of not channel 1 but channel 0 is executed at Cycle 13 because the priority of the latter is higher. At Cycle 15, the DMA cycle for channel 0 is complete. At Cycle 16, a DMA cycle for channel 1 starts again. At Cycle 18, the last DMA cycle of the block transfer of channel 1 is complete.

7.2.2.2 DTS Channel Arbitration

If there are DMA transfer requests from multiple DTS channels, the DTSFSL arbitrates those DTS channels. For each DTS channel, a priority can be selected from four levels using DTS channel priority setting registers.

If there are DMA transfer request from multiple DTS channels, the arbitration is done as follows.

1. A channel with a higher priority level in the setting of DTS channel priority setting registers has a priority.
2. If two channels have the same priority level in the setting of DTS channel priority setting registers, a channel with a lower channel number has a priority.

The DTSFSL sends the DTS a DMA transfer request for the channel selected by arbitration. The DTS executes DMA transfer when it accepts the DMA transfer request.

Unlike DMA transfer with a DMAC, DMA transfer with a DTS does not allow arbitration between DTS channels in the middle of a block transfer. That means, even if a DMA transfer request with a higher priority comes during a block transfer for a channel with a lower priority, the DMA transfer with a higher priority does not start until the current block transfer for the channel with a lower priority is complete[*1].

Note 1. The timing of completion of the block transfer is when the last transfer for the block transfer 1 or the last transfer or address reload transfer for the block transfer 2 occurs.

When a DTS executes the block transfer 1 or block transfer 2, a DMA cycle of a DTS channel with a higher priority does not take over the ongoing block transfer until the last transfer.

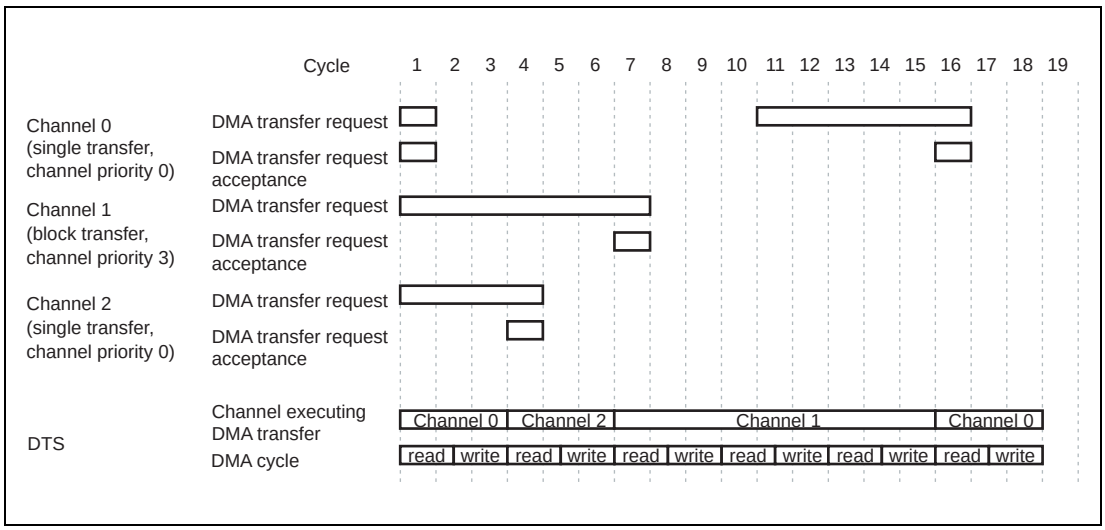


Figure 7.4 DMAC Channel Arbitration

Cycle numbers shown in **Figure 7.4** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.4**, DMA transfer requests for channels 0, 1, and 2are generated at Cycle 1. The channel priority for channels 0 and 2 is 0 and is higher than the channel priority for channel 1, which is 3. In addition, if two channels have the same priority, the channel with the smaller channel number has a higher priority. Consequently, the priority order for arbitration is “channel 0 > channel 2 > channel 1”, and a DMA cycle for channel 0 starts because its priority is the highest At Cycle 4, as a result of arbitration between channels 1 and 2, a DMA cycle for channel 2 starts. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done until the block transfer of channel 1 is complete.

At Cycle 15, the block transfer of channel 1 is complete. At Cycle 16, a DMA cycle for channel 0 starts.

7.2.2.3 Interface Arbitration

DMAC0, DMAC1, and DTS work independently and execute DMA transfer.

If the request to DMAC0, DMAC1, and DTS occurs simultaneously, arbitration is performed on a round-robin basis.

7.2.3 Reload Function

7.2.3.1 Overview of the Reload Function

The reload function updates a portion of transfer information, more specifically, the source address, destination address, transfer count, and address reload count, to the predefined values during DMA transfer.

The reload function has two types of functions: reload function 1 and reload function 2.

7.2.3.2 Operation of Reload Function 1

When the reload function 1 is enabled, actions described in **Table 7.3** are executed at the timing of the last transfer according to the reload function 1 setting.

Table 7.3 Operation of Reload Function 1

Reload Function 1 Setting (DTCTn.RLD1M[1:0])	Register	Action at the Last Transfer
00 (Reload function 1 disabled)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Transfer count	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 1 enabled. Reloading source address and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
10 (Reload function 1 enabled. Reloading destination address and transfer count.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
11 (Reload function 1 enabled. Reloading source address, destination address, and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded • If the reload function 2 is enabled: The reload address reload count is copied to this.

Figure 7.5 shows an operation of the reload function 1.

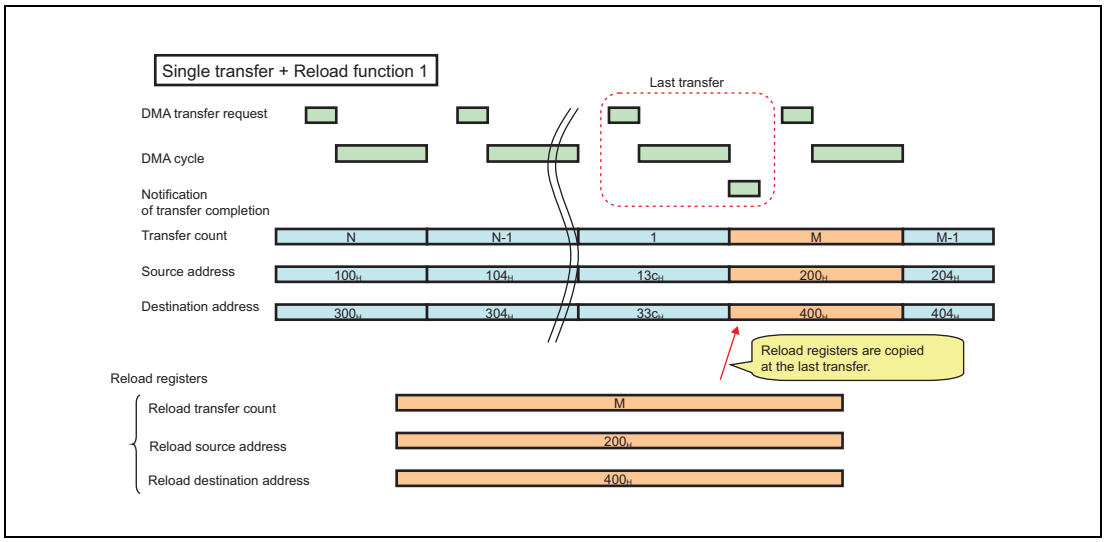


Figure 7.5 Operation of Reload Function 1

7.2.3.3 Reload Function 2

When the reload function 2 is enabled, actions described in **Table 7.4** are executed at the timing of the address reload transfer according to the reload function 2 setting.

Table 7.4 Operation of Reload Function 2

Reload Function 2 Setting (DTCTn.RLD2M[1:0])	Register	Action at the Address Reload Transfer
00 (Reload function 2 disabled)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 2 enabled. Reloading source address.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Address reload count	The reload address reload count is copied to this.
10 (Reload function 2 enabled. Reloading destination address.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.
11 (Reload function 2 enabled. Reloading source address and destination address.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.

Figure 7.6 shows an operation of the reload function 2.

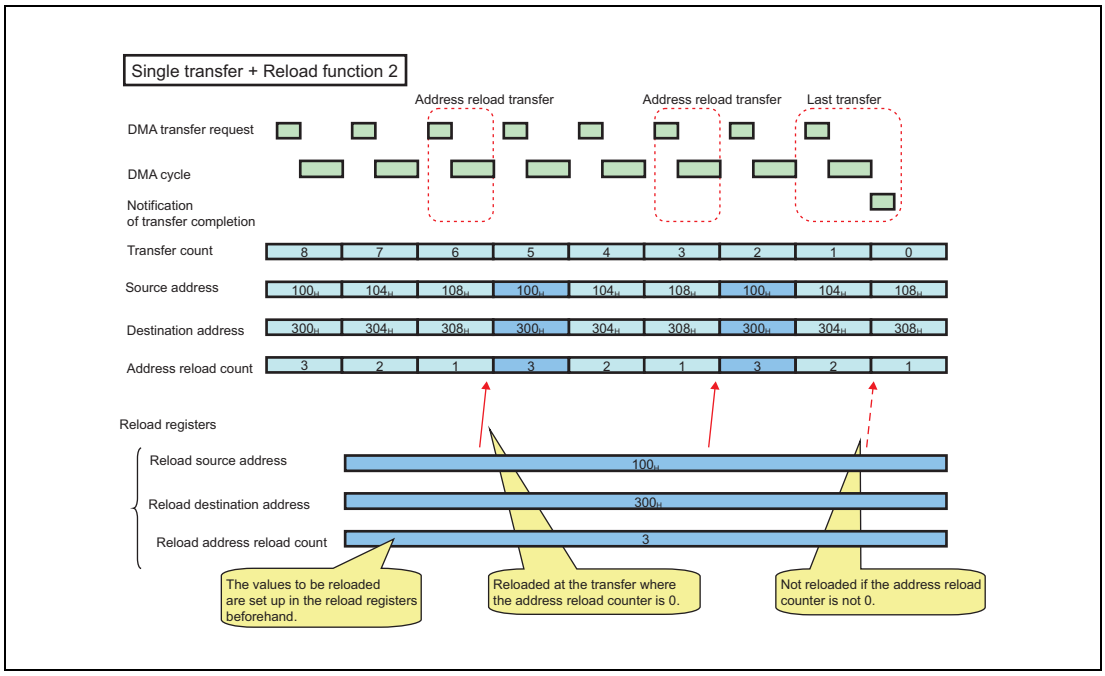


Figure 7.6 Operation of Reload Function 2

Figure 7.7 shows an operation when both the reload function 1 and the reload function 2 are used simultaneously.

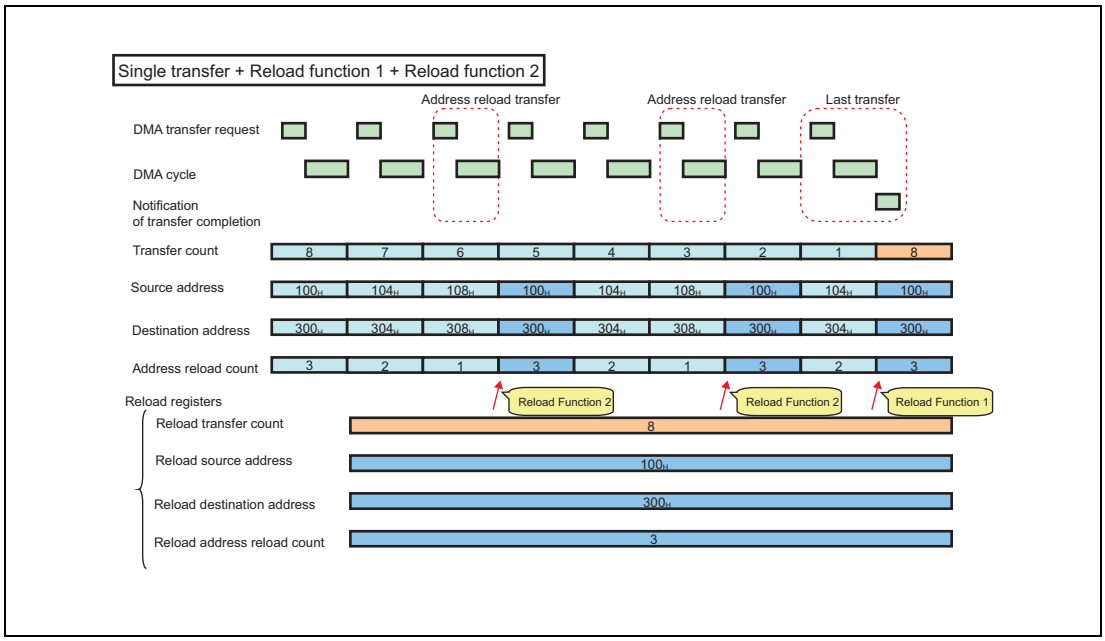


Figure 7.7 Operation when Combining Reload Function 1 and Reload Function 2

7.2.3.4 Timing of Setting DMAC Reload Registers

You can set up the reload source address register, reload destination address register, and reload transfer count register any time (even during DMA transfer). As an exception, if you update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, there may be a conflict between reloading at the last transfer or address reload transfer and updating the reload register. In order to avoid this conflict, setting up reload registers must be completed before the last transfer or address reload transfer starts.

If you need to update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, one way to know the right timing of update is to use a DMA transfer count match interrupt. In this case, you must set up the DMA transfer count compare register (DTCCn) so that you can have enough margin for the time necessary to update the reload registers.

7.2.3.5 Timing of Setting DTS Reload Registers

It should be noted that the right timing of setting up the reload source address information, reload destination address information, and reload transfer count information differs depending on the transfer mode.

In single-transfer mode, the TI fetched at the beginning of the last transfer or address reload transfer is used for reload at the completion of the DMA cycle. Therefore, if you use the reload function for single transfer, the reload source address information, reload destination address information, and reload transfer count information in the TI must be set up before the beginning of the last transfer or address reload transfer.

During block transfer, TI is fetched only at the beginning of DMA transfer. The TI fetched at the beginning of the DMA transfer is used for reload at the last transfer or address reload transfer. Therefore, if you use the reload function for block transfer, the reload source address information, reload destination address information, and reload transfer count information in the TI must be set up before the beginning of the DMA transfer. If you update the reload source address information, reload destination address information, and reload transfer count information in the TI in the middle of a block transfer, those new settings will not be used for reload at the completion of the block transfer.

7.2.4 Chain Function

7.2.4.1 Overview

DMA offers a function called chain function. If you use the chain function, the completion of the DMA cycle or last transfer for one channel can trigger a DMA transfer request for another channel. A DMA transfer request for another channel initiated by the chain function is called a chain request.

You can select the condition for generating a chain request from the following two options.

- Always chain: A chain request is generated at the completion of every DMA cycle.
- Chain at the last transfer: A chain request is generated at the completion of the last transfer.

Figure 7.8 shows an operation of the case “always chain”.

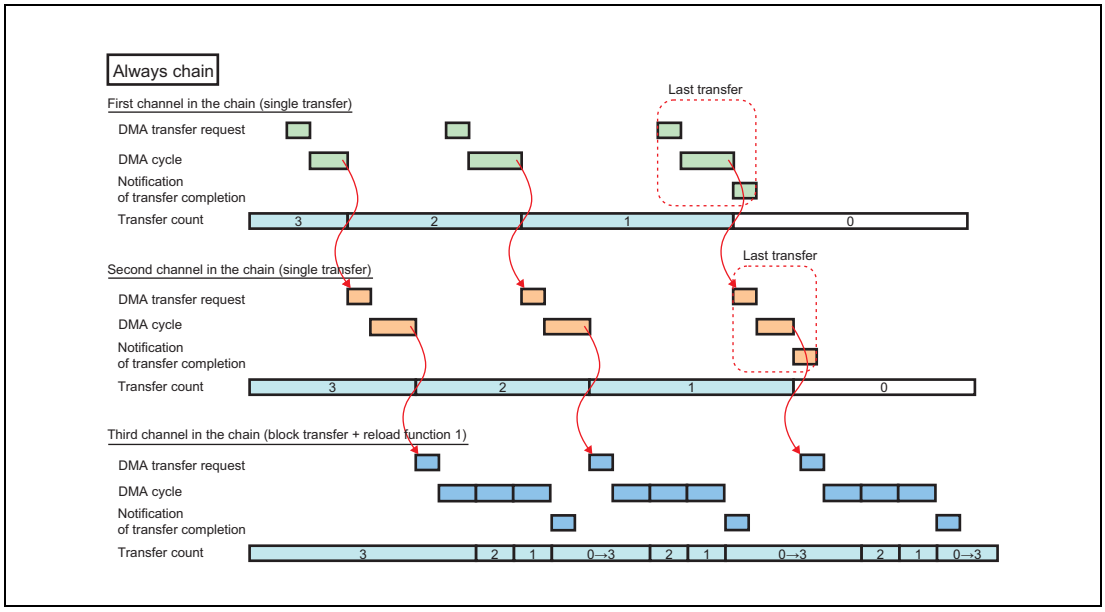


Figure 7.8 Operation of the Case “Always Chain”

Figure 7.9 shows an operation of the case “chain at the last transfer”.

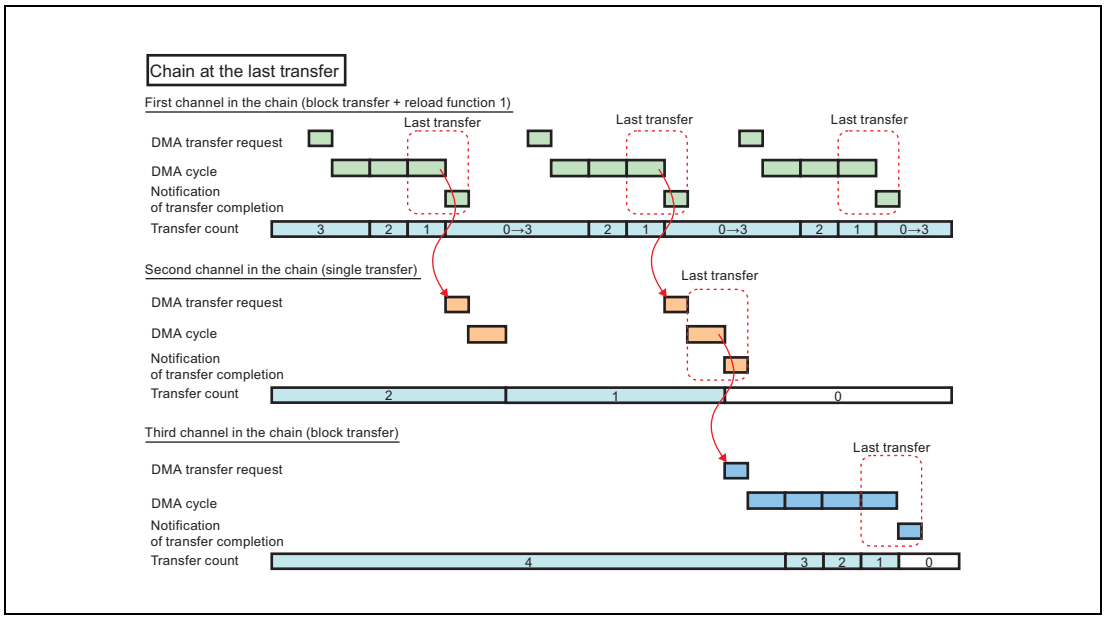


Figure 7.9 Operation of the Case “Chain at the Last Transfer”

7.2.4.2 Setting Up the Chain Function

For a DMAC, you need to write to the chain enable (DTCTn.CHNE) and the next channel in the chain selection (DTCTn.CHNSEL) in the DMAC transfer control register in order to set up the type of chain function and the next channel number in the chain.

For a DTS, you need to write to the chain enable (DTTCTnnn.CHNE) and the next channel in the chain selection (DTTCTnnn.CHNSEL) in the DTS transfer control register in order to set up the type of chain function and the next channel number in the chain.

7.2.4.3 Caution for Using the Chain Function

The chain function sets the software DMA transfer request flag of the next channel in the chain as a part of its function. Therefore, you need to set up the channel settings of the next channel in the chain in the same way as when the software DMA transfer request is used. If you specify a channel using the hardware DMA transfer request for the next channel in the chain, the chain function does not work.

A channel and its next channel in the chain must belong to the same module (DMAC0, DMAC1, and DTS). You cannot specify a channel in another module for its next channel in the chain.

7.2.5 DMAC Operation

7.2.5.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DMAC starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request. The DMA transfer request selection assignment (DRS) bit in the DMAC transfer control register (DTCTn) determines whether a hardware DMA transfer request or a software DMA transfer request is used.

In the case of a hardware DMA transfer request for a DMAC, one out of 128 hardware DMA transfer sources is selected and assigned for each channel of the DMAC in the DTFR. This assignment is configured in the DTFR setting registers.

7.2.5.2 Generating and Accepting a Software DMA Transfer Request

By setting the software DMA transfer request flag (SR) in the DMAC transfer status register (DCSTn) using the DMAC transfer status set register (DCSTSn), a software DMA transfer request can be generated.

The software DMA transfer request flag is automatically cleared when the DMAC processes the DMA transfer request. The timing when the software DMA transfer request flag is automatically cleared differs depending on the transfer mode of the DMA transfer to be executed.

- In single-transfer mode, the software DMA transfer request flag is cleared whenever the software DMA transfer request is accepted.
- In block transfer 1 mode, the software DMA transfer request flag is cleared when the last transfer starts.
- In block transfer 2 mode, the software DMA transfer request flag is cleared when the last transfer or address reload transfer starts.

The software DMA transfer request flag can also be cleared by software using the DMAC transfer status clear register (DTSTCn). When you abort a DMA transfer of a DMAC channel, you must clear the software DMA transfer request flag.

7.2.6 DTS Operation

7.2.6.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DTS starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request.

A transfer request for a DTS is retained in the transfer request pending bit of the DTSFSL for each channel. As for the DTSFSL, both a hardware DMA transfer request and a software DMA transfer request are retained in the same transfer request pending bit. When executing DMA transfer, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request.

In the case of a hardware DMA transfer request for a DTS, each of 128 hardware DMA transfer sources is assigned to one of 128 channels in the DTSFSL in the fixed manner. You cannot change this assignment by, for example, register settings.

7.2.6.2 Generating and Accepting a DMA Transfer Request

When the DTSFSL detects a hardware DMA transfer source input, the DTSFSL sets the transfer request pending bit and retains the hardware DMA transfer source as a DMA transfer request. If the transfer request pending bit is set and the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register is set, the DTSFSL notifies the DTS of the DMA transfer request.

Software can also generate a DMA transfer request by setting the transfer request pending bit (DTFSTnnn.DRQ) using the DTSFSL transfer request set register (DTFSSnnn).

The DTSFSL can retain only one DMA transfer request per channel. If, while the transfer request pending bit for a channel is set, a new hardware DMA transfer source input for the same channel comes, the new hardware DMA transfer source input is ignored.

When the DTS accepts a DMA transfer request, it notifies of the acceptance of the DMA transfer request.

The transfer request pending bit is automatically cleared when the DTS accepts the DMA transfer request. The DTSFSL clears the transfer request pending bit automatically when the DTS accepts the DMA transfer request regardless of the type of the DMA transfer to be executed by the DTS.

The transfer request pending bit can also be cleared using the DTSFSL transfer request clear register (DTFSCnnn). If the transfer request pending bit of a channel is cleared before the DTS accepts the DMA transfer request, DMA transfer of the channel is not executed.

7.2.6.3 Executing DMA Transfer

When the DTS accepts a DMA transfer request for a channel, the DTS executes DMA transfer of the channel. If there are DMA transfer requests from multiple channels, the DTSFSL arbitrates the DTS channels and picks up one channel for a DMA transfer request.

While the DTS is executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is set. In addition, the channel number of the currently ongoing DMA transfer is set in the DTS transfer channel (DTSSTS.DTSACH).

When the DMA transfer is complete or aborted because of DMA transfer error or writing to registers and no channel is currently executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit is cleared.

7.2.6.4 DTSRAM Access

A DTS accesses the DTSRAM when DMA transfer starts and finishes.

A DTS's action of reading transfer information from the DTSRAM when DMA transfer starts is called TI fetch.

A DTS's action of updating the transfer information on the DTSRAM when DMA transfer finishes is called TI write back.

A single transfer performs a TI fetch at the beginning of a DMA cycle and a TI write back at the end of a DMA cycle.

A block transfer performs a TI fetch at the beginning of the first DMA cycle and a TI write back at the end of the DMA cycle that satisfies the block transfer completion condition (the last transfer or address reload transfer).

Therefore, in the case of single transfer, the transfer information on the DTSRAM is updated for each DMA cycle. In the case of block transfer, the transfer information on the DTSRAM is updated after the completion of the block transfer. If software reads the transfer information on the DTSRAM during execution of a block transfer, the transfer information at the beginning of the block transfer is read.

7.3 Suspension, Resume, Transfer Abort, and Clearing a DMA Transfer Request

7.3.1 DMA Suspension and Resume by Software Control

The DMA control register (DMACTL) is used to suspend DMA transfer for all channels.

When the DMA suspension bit (DMACTL.DMASPD) in the DMA control register is set, DMA puts all channels into the suspended state If all channels are in the suspended state and the DMA suspension bit (DMACTL.DMASPD) in the DMA control register is cleared, DMA restores all channels from the suspended state to the normal state and resumes the DMA transfer of the suspended channel.

When all channels are put into the suspended state, DMA transfer is suspended for all channels without changing the value of the DCENn.DTE bit of each DMAC channel and the DTSCTL1.DTSUST bit of the DTS.

CAUTION

Execute the enable / disable operation for the DMA transfer request enable bit (DTFSLnnn.REQEN) to suspend/resume DMA transfer executed by a DTS. For details, see the notes in Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS.

7.3.2 Suspension, Resume, and Transfer Abort of a DMAC Channel

You can suspend the DMA transfer of a DMAC channel by setting the DMA suspension bit (DMACTL.DMASPD) in the DMA control register or clearing the channel operation enable bit (DCENn.DTE) in the DMAC channel operation enable setting register for the channel. If a DMA cycle is ongoing, the DMA transfer of a channel is suspended after the currently ongoing DMA cycle is finished. If you clear the bit (DMACTL.DMASPD) or reset the DCENn.DTE bit for a channel while the DMA transfer of the channel is suspended, the DMA transfer of the suspended DMA channel is resumed.

If you want to abort the currently ongoing DMA transfer of a DMAC channel, similarly clear the channel operation enable bit (DCENn.DTE) in the DMAC channel operation enable setting register, and then clear the hardware DMA transfer request in the DTFR in the case of a hardware DMA transfer request, and clear the software DMA transfer request flag (DCSTn.SR) using the DMAC transfer request flag clear bit (DCSTCn.SRC) in the DMAC transfer status clear register in the case of a software DMA transfer request.

The channel operation enable bit (DCENn.DTE) is kept being set while the continuous transfer enable bit (DTCTn.MLE) is set. Even if the channel operation enable bit (DCENn.DTE) is cleared by software during the last DMA cycle, the function of continuous transfer enable bit (DTCTn.MLE) has priority. Thus, the channel operation enable bit (DCENn.DTE) is set again after the last transfer is complete.

If you want to suspend the DMA transfer of a DMAC channel while the continuous transfer function is in use, clear the continuous transfer enable bit (DTCTn.MLE) in the DMAC transfer control register, and then clear the channel operation enable bit (DCENn.DTE) in the DMAC channel operation enable setting register to suspend the DMA transfer of a DMAC channel. Only in this case, writing to the DMAC transfer control register (DTCTn) is enabled in the channel operation enable state (DCENn.DTE = 1).

Figure 7.10 shows an example of suspension, resume, and transfer abort of a DMAC channel.

In **Figure 7.10**, both channels 0 and 1 are executing block transfer. At time tick 1, DMA transfer of channel 1 starts. At time tick 2, a DMA transfer request for channel 0 is accepted. As a result of DMAC channel arbitration, DMA transfer of channel 0 starts because channel 0 has a higher priority than channel 1. At time tick 3, the last transfer of channel 0 is complete, and the remaining DMA transfer in the block transfer of channel 1 starts. At time tick 4, the last transfer of channel 1 is complete. After time tick 5, DMA transfer of channel 0 and DMA transfer of channel 1 are executed similarly. At time tick 7, the DMA transfer of channel 0 is suspended and, as a result of DMAC channel arbitration, the DMA transfer of channel 1 starts. At time tick 8, the last transfer of channel 1 is complete, and then, at time tick 9, the DMA transfer of channel 0 resumes. At time tick 10, the last transfer of channel 0 is suspended again, and then, at time tick 11, the DMA transfer of channel 0 is aborted. At time tick 12, the suspended state for channel 0 is cleared, but DMA transfer is not executed because the DMA transfer is aborted at time tick 11.

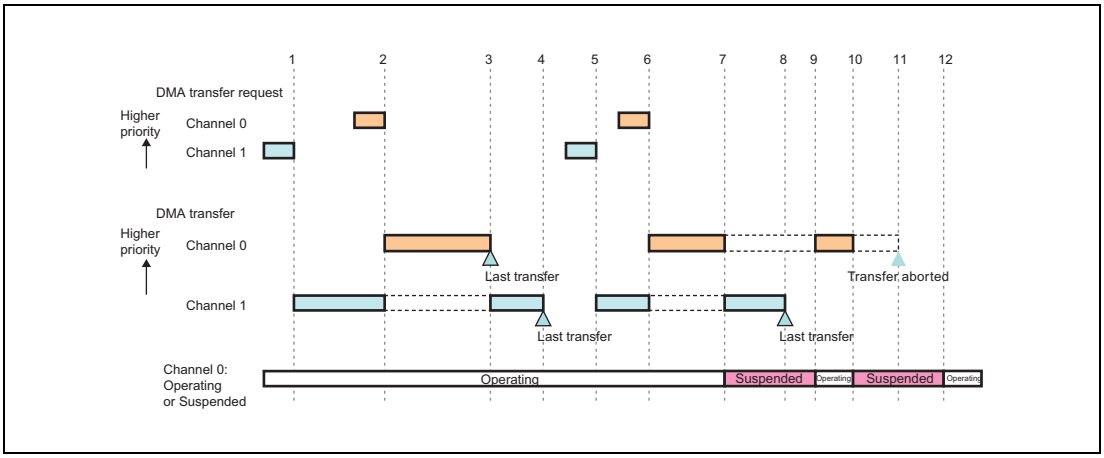


Figure 7.10 Example of Suspension, Resume, and Transfer Abort of a DMAC Channel

7.3.3 Suspension, Resume, and Transfer Abort of a DTS

You can suspend*1 the DMA transfer executed by a DTS by setting the DTS suspend bit (DTSCTL1.DTSUST) in the DTS control register 1 and the DMA suspend bit (DMACTL.DMASPD) in the DMA control register. If a DMA cycle is ongoing, the DMA transfer is suspended at the timing when the DMA cycle is finished. If the ongoing DMA cycle is a single transfer or a transfer that completes a block transfer (the last transfer or address reload transfer), the DMA transfer is suspended after a TI write back after the completion of the DMA cycle. If the ongoing DMA cycle is a type other than the above, the DMA transfer is suspended after the completion of the DMA cycle without a TI write back. If you resume the DMA transfer while the DMA transfer is suspended, clear*1 the DTS suspend bit in the DTS control register 1 or the DMA suspend bit in the DMA control register.

Note 1. Before the suspension/resumption of DMA transfer during the execution by a DTS, you need to execute the operation of enable/disable for a DMA transfer request enable bit (DTFSLnnn.REQEN). Follow the procedures below for the suspension/resumption.

(a) Procedure to suspend a DTS

1. Clear the DMA transfer request enable bit (DTFSLnnn.REQEN) of all the DTS channels set up with DMA transfer request enable bit (DTFSLnnn.REQEN).
2. Set a DMACTL.DMASPD bit or DTSCTL1.DTSUST bit.

(b) Procedure to resume a DTS

1. Set a DMA transfer request enable bit (DTFSLnnn.REQEN) of the DTS channel which was cleared in 1 of (a) above.
2. Clear the DMACTL.DMASPD bit or DTSCTL1.DTSUST bit which was set in 2 of (a) above.

If you want to abort the currently ongoing DMA transfer executed by a DTS, use the DTS suspend bit ?DTSCTL1.DTS0UST?in the DTS control register 1 and suspend the DTS as described above, and then set the DTS transfer abort request bit (DTSCTL2.DTSTIT) in the DTS control register 2 to abort the currently suspended DMA transfer. If transfer is aborted, no TI write back is executed. In addition, aborting the DMA transfer does not change the value of the DTS suspend bit (DTSCTL1.DTSUST). If you want the DTS to accept another DMA transfer request after the abort, after setting the DMA transfer request enable bit (DTFSLnnn.REQEN) of DTS channel, clear the DTS suspend bit.

Figure 7.11 shows an example of suspension, resume, and transfer abort of a DTS.

In **Figure 7.11**, channels 0, 1, and 2 are executing block transfer. At time tick 1, a DMA transfer request for channel 1 is accepted and DMA transfer starts. At time tick 2, DMA transfer requests for channels 0 and 2 are generated. At time tick 3, the last transfer of channel 1 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted, and DMA transfer of channel 0 starts because channel 0 has a higher priority than channel 2. At time tick 4, the last transfer of channel 0 is complete, and DMA transfer of channel 2 starts. At time tick 5, the DTS is put into the suspended state, and the DMA transfer of channel 2 is suspended. At time tick 6, DMA transfer requests for channels 0 and 1 are generated. At time tick 7, the suspended state for the DTS is cleared, and the DMA transfer of channel 2, which has been suspended in the middle of a block transfer, is resumed. If DMA transfer is suspended in the middle of a block transfer, no DTS channel arbitration is done when it is resumed. At time tick 8, the last transfer of channel 2 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted and the DMA transfer starts because channel 0 has a higher priority than channel 2. At time tick 9, the DTS is put into the suspended state, and at time tick 10, the suspended DMA transfer of channel is aborted. When the suspended state of the DTS 0 is cleared at time tick 11, DMA transfer of channel 1 starts because there is no currently ongoing DMA transfer and channel 1 is the only channel with a DMA transfer request.

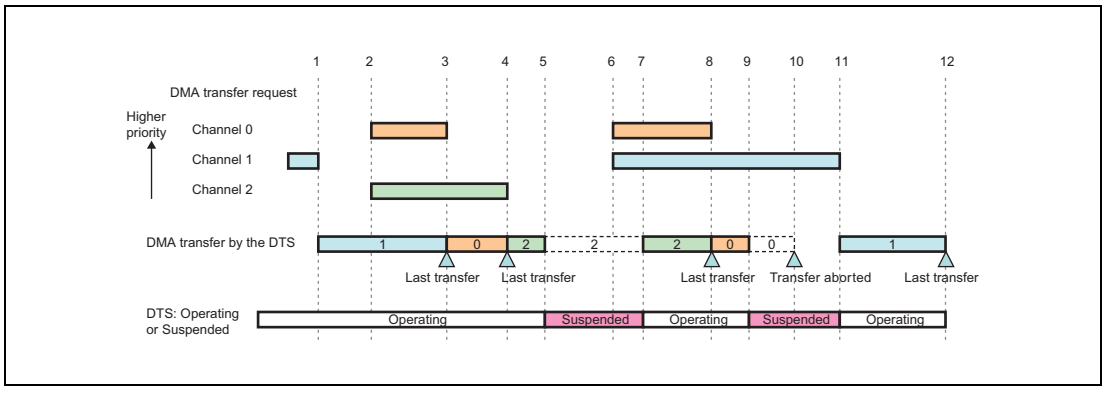


Figure 7.11 Example of Suspension, Resume, and Transfer Abort of a DTS

7.3.4 Masking and Clearing a Hardware DMA Transfer Request by the DTFR

If a DMAC uses a hardware DMA transfer request, you can temporarily disable (mask) the hardware DMA transfer request output from the DTFR to the DMAC by clearing the hardware DMA transfer source selection enable bit (DTFRn.REQEN) in the DTFR setting register.

Also, if a hardware DMA transfer source is used, you can clear a hardware DMA transfer request retained in the DTFR by using the hardware DMA transfer request clear (DTFRRQCn.DRQC) bit in the DTFR transfer request clear register.

Even if you suspend or abort DMA transfer of a DMAC channel, the hardware DMA transfer request selection/hold circuit of the DTFR is still running, and consequently, the DTFR may retain a hardware DMA transfer request that came to the DTFR during the suspension or transfer abort period of the DMAC channel. When you resume or start DMA transfer of a DMAC channel, clear the hardware DMA transfer request retained in the DTFR as required.

When DMAC uses a hardware transfer request and the block transfer 1 or block transfer 2, if software disables the hardware transfer source selection bit of DTFR (DTFRn.REQEN = 0) while DMAC is executing block transfer, the block transfer in progress is suspended.

7.3.5 Masking and Clearing a Hardware DMA Transfer Request by the DTSFSL

As for a DTS, you can temporarily disable (mask) a DMA transfer request from a channel to the DTS by clearing the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register. (The masking is actually done by excluding the channel from the candidates in DTS channel arbitration in the DTSFSL.)

Also, you can clear a DMA transfer request retained in the DTSFSL by using the transfer request clear (DTFSCnnn.DRQC) bit in the DTSFSL transfer request clear register.

Regardless of the state of the DTS and the transfer request enable bit (DTFSLnnn.REQEN) of the DTSFSL, the DTSFSL always monitors the hardware transfer source input, and a DMA transfer request for a channel is set when a hardware transfer source for the channel is input to the DTSFSL. When you resume or start DTS transfer, clear the hardware DMA transfer request retained in the DTSFSL as required.

7.3.6 List of Suspend, Resume, and Transfer Abort Functions

Table 7.5 List of Suspend, Resume, and Transfer Abort Functions

Function	How to execute the function	Operation	Possibility of DMA transfer abort	Master that can execute the function (See Section 7.5, Reliability Function.)
DMA suspension and resume by software control	Setting and clearing*2 the DMACTL.DMASPD.	All channels are in the suspended state.	Not possible*1	Special master
Suspension and resume of a DMAC channel	Clearing and setting the DCENn.DTE in each channel register.*3	DMA transfer of a channel is suspended.	Possible (by clearing the DMA transfer request flag during suspension)	Special master, and general master assigned to the channel.
Suspension and resume of a DTS	Setting and clearing*2 the DTSUST.DTS0UST.	DMA transfer of a DTS is suspended.	Possible (by setting the DTSCTL2.DTSTIT during suspension)	Special master

Note 1. In order to abort DMA transfer, you need to either abort transfer for the DMAC channel or abort transfer for the DTS.

Note 2. In order to suspend resume DMA transfer during the execution by a DTS, you need to execute the operation of enable/disable for a DMA transfer request enable bit (DTFSLnnn.REQEN). For the details, see the notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

Note 3. While the continuous transfer function is in use, set or clear the continuous transfer enable bit (DTCTn.MLE) in advance.

7.4 Error Control

7.4.1 Type of Error

DMA can generate the following two types of errors.

- DMA Transfer Error
  This error is generated when error is detected in the read cycle or write cycle in a DMA cycle.
  This error can be generated in all DMAC and DTS channels during execution of DMA transfer.
- DTSRAM Error
  This error is generated when ECC error is detected in the DTSRAM read access by a DTS.
  This error can be generated in the TI fetch during execution of DMA transfer for a DTS or while software is accessing the DTS channel registers.

7.4.2 DMA Transfer Error

When DMA transfer error occurs, a DMA transfer error interrupt (DMAERR) is generated.

Though DMA transfer error is detected independently by each DMAC channel and a DTS, a DMA transfer error interrupt (DMAERR) is common to both the DMAC and DTS.

7.4.2.1 Operation of a DMAC When DMA Transfer Error Occurs

When DMA transfer error occurs in a DMAC, the transfer error flag (DCSTn.ER) in the DMAC transfer status register of the channel with the DMA transfer error. The DMAC error register (DMACER) shows the transfer error flags of all 16 DMAC channels.

While the transfer error flag of a channel is set, a new DMA cycle is not executed if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is set. On the other hand, a DMA cycle is executed regardless of the value of the transfer error flag if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is cleared.

If you want to abort the DMA transfer of a channel with DMA transfer error, follow the procedure to abort DMA transfer of the DMAC channel.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated.

7.4.2.2 Operation of a DTS When DMA Transfer Error Occurs

When DMA transfer error occurs in a DTS, the DTS error flag (DTSER1.DTSER) in the DTS error register is set, and the DTS channel number with the DMA transfer error is stored in the DTS error channel (DTSER1.DTSERCH) in the same register.

If DMA transfer error occurs in a single transfer, a TI write back is executed to finish the DMA cycle.

If DMA transfer error occurs in the middle of a block transfer and the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is set, the remaining DMA cycles in the block transfer are not executed, but a TI write back is executed to finish the DMA cycle. At the same time, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is cleared. If DMA transfer error occurs in the middle of a block transfer and the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is cleared, the block transfer continues regardless of the DMA transfer error.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated, and the TI is updated by a TI write back.

If the DTS error flag in the DTS error register is set, a TI fetch is executed when the DTS accepts a DMA transfer request for the channel with the same channel number as the one stored in the DTS error channel. If, as a result of the TI fetch, the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is found to be set, a DMA cycle and a TI write back are not executed. If the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is cleared, DMA transfer is executed.

If the DTS error flag in the DTS error register is set, DMA transfer is executed when the DTS accepts a DMA transfer request for a channel with a channel number other than the one stored in the DTS error channel.

7.4.3 DTSRAM Error

There are two types of DTSRAM errors detected in the DTSRAM read access: 1-bit and 2-bit ECC error.

If a 1-bit ECC error is detected during a TI fetch, error corrected data is used, and DMA transfer continues. If a 1-bit ECC error is detected during DTS channel register access from software, error corrected data is returned as read data. In either case, the DTSRAM1 bit error flag (DTSER2.RAMSED) in the DTS error register 2 is set, and the address of the error location in the DTSRAM is stored to the DTSRAM1 bit error address (DTSER2.RAMSEDAD).

In addition, the error is notified to the ECM.

If a 2-bit ECC error is detected during a TI fetch, handling of the DMA transfer request is terminated without executing a DMA cycle and TI write back. If a 2-bit ECC error is detected during DTS channel register access from software, peripheral bus error is notified. In either case, the DTSRAM2 bit error flag (DTSER2.RAMDED) in the DTS error register 2 is set, and the address of the error location is stored to the DTSRAM2 bit error address (DTSER2.RAMDEDAD). In addition, the error is notified to the ECM.

7.5 Reliability Function

7.5.1 Overview

In this product, DMA is a resource used by multiple masters (CPU1 or CPU2). In order for DMA to support multi-core configuration, the following reliability functions are offered.

- Register access protection function
- Master information inherit function

7.5.2 Register Access Protection Function

This product is designed to assign each DMA channel to a CPU1 or CPU2.

The register access protection function allows access to the transfer information of each DMA channel from the master (CPU1 or CPU2) assigned to the channel but prohibits access from other masters.

The register access protection function enables you, for example, to prevent the settings of the channel from being read or updated by masters other than the one assigned to the channel.

7.5.2.1 Identifying the Accessing Master

DMA identifies a master based on the ID of the accessing CPU (PEID), and whether the CPU is in supervisor mode (PSW.UM = 0) or user mode (PSW.UM = 1).

7.5.2.2 Master Access

There are the following two types of master accesses:

- Special master access (the CPU is in supervisor mode (UM = 0))
- General master access (access other than special master access)

In special master access, access to all registers is allowed.

In general master access, access to the following registers is allowed:

- The following global registers
  DMACER, DTSER1, DTSER2, and DTSSTS
- Channel registers of the channels assigned by the channel assignment. (For details, see **Section 7.5.2.3, Channel Assignment**.)

In general master access, access to registers other than the above is not allowed.

7.5.2.3 Channel Assignment

To each channel, DMA can assign a master (CPU1 or CPU2) so that the master is allowed to use the channel. Channel assignment is configured in the channel master setting register (DMnnCM in the case of a DMAC and DTSnnnCM in the case of a DTS) in supervisor mode by the CPU (UM = 0).

In general master access, the master assigned to a channel by channel assignment is allowed to access the channel registers of the channel. If the channel registers of a channel is accessed by a master other than the master assigned to the channel, the access is called illegal access. For information about illegal access, see **Section 7.5.2.4, Illegal Access**.

7.5.2.4 Illegal Access

DMA handles the following access as illegal access.

(a) General master access to the global registers
Except the following registers: DMACER, DTSER1, DTSER2, and DTSSTS

(b) General master access to the channel registers of a channel by a master other than the master assigned to the channel

DMA's actions against illegal access are as follows.

For both cases (a) and (b),

- Write access is ignored.
- Read access returns 0 as read data.

Only for the case (b),

- The information about the illegal access is stored in a register access protection violation register.
- The DMAC0, DMAC1, and DTS have their own register access protection violation registers (DM0CMV, DM1CMV, and DTSCMV respectively).

Only the special master can access the register access protection violation registers. The special master can check whether illegal access has occurred by checking the register access protection violation registers periodically.

In addition, it is recommended that, when a master tries to use DMA and configures transfer information in the channel registers, the master should check whether the configuration has been successfully completed without illegal access by, for example, reading back the settings.

7.5.3 Master Information Inherit Function

In this product, DMA access is assumed to inherit and use master information that is equivalent to the master information of the CPU1 or CPU2 assigned to the DMA channel. The master information assigned to the DMA channel is set and used in the DMnnCM register / DTSnnnCM register.

The master information that is output from DMA is as in **Table 7.6**.

Table 7.6 Master Information Output from DMA

Meaning	Value that is output from DMA
UM	Same as the UM bit value in the channel master setting register
SPID	Same as the SPID bit value in the channel master setting register
PEID	Same as the PEID bit value in the channel master setting register
DMA access	1

7.5.4 Other Reliability Functions

7.5.4.1 Restriction on the Next Channel in the Chain

The reliability function limits the channels you can select as the next channel in the chain.

When you use the chain function, the channel master settings of a channel and its next channel in the chain must be the same.

The chain function is designed so that a channel and its next channel in the chain are managed by the same master.

When DMA detects that different masters are assigned to a channel and its next channel in the chain, it is deemed illegal and the chain function is suppressed. More specifically, when DMA tries to execute the chain function, DMA compares the chain master settings of the channel and its next channel in the chain, and if the settings are all the same for PEID and UM, the chain function is allowed and a chain request is sent to the next channel. If the settings are not the same for PEID or UM, a chain request is not sent.

7.6 Setting Up DMA Transfer

7.6.1 Overview of Setting Up DMA

Table 7.7 Channel Assignment (1/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
1	Special master (supervisor mode of CPU (UM = 0))	Overall DMA operation setting	DTSPR0 to DTSPR7	DTS channel priority setting register	Mandatory (if a DTS is used)
2			DM00CM to DM17CM	DMAC channel master setting register	Mandatory (if a DMAC is used)
3			DTS0CM to DTS127CM	DTS channel master setting register	Mandatory (if a DTS is used)
4		Status clear	DTSERC	DTS error clear register	Recommended
5			CMVC	Channel protection violation clear register	Recommended
6	Master assigned to the DMAC channel	Channel setting	DSAn	DMAC source address register	Mandatory
7			DDAn	DMAC destination address register	Mandatory
8			DTCn	DMAC transfer count register	Mandatory
9			DTCTn	DMAC transfer control register	Mandatory
10			DRSAn	DMAC reload source address register	Mandatory if the reload function is used
11			DRDAn	DMAC reload destination address register	Mandatory if the reload function is used
12			DRTCn	DMAC reload transfer count register	Mandatory if the reload function is used
13			DTCCn	DMAC transfer count compare register	Mandatory if the transfer count match interrupt is used
14			DTFRn	DTFR setting register	Mandatory
15		Status clear	DCSTCn	DMAC transfer status clear register	Mandatory
16			DTFRRQCn	DTFR transfer request clear register	Recommended
17		Channel operation enable	DCENn	DMAC channel operation enable setting register	Mandatory

Table 7.7 Channel Assignment (2/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
18	Master assigned to the DTS channel	Channel setting	DTSAnnn	DTS source address register	Mandatory
19			DTDAnnn	DTS destination address register	Mandatory
20			DTTCnnn	DTS transfer count register	Mandatory
21			DTTCTnnn	DTS transfer control register	Mandatory
22			DTRSAnnn	DTS reload source address register	Mandatory if the reload function is used
23			DTRDAnnn	DTS reload destination address register	Mandatory if the reload function is used
24			DTRTCnnn	DTS reload transfer count register	Mandatory if the reload function is used
25			DTTCCnnn	DTS transfer count compare register	Mandatory if the transfer count match interrupt is used
26		Status clear	DTFSCnnn	DTSFSL transfer request clear register	Recommended
27		Transfer request enable	DTFSLnnn	DTSFSL operation setting register	Mandatory

7.6.2 Setting Up the Overall DMA Operation

You need to set up the overall DMA operation before you start using DMA.

To configure the overall DMA operation, the special master (supervisor mode of the CPU (UM = 0)) needs to set up global registers. Global registers can be set up only by special master access. For details, see **Section 7.5, Reliability Function**.

The following registers must be set up to configure the overall DMA operation.

- DTS channel priority setting registers (DTSPRn, n = 0 to 7)
  Those registers configure the priority level of each DTS channel used for DTS channel arbitration.
- DMAC channel master setting registers (DMnnCM)
- DTS channel master setting registers (DTSnnnCM)
  Those registers configure channel assignment. (For details, see **Section 7.5, Reliability Function**.)
  If the DMAC channel master setting registers and the DTS channel master setting registers are not properly set, DMA channel setting and DMA transfer cannot be executed properly.

Also, if errors are detected in the following registers while the overall DMA operation is set up, clearing the errors is recommended.

- DTS error register 1 (DTSER1)
- DTS error register 2 (DTSER2)
- DMAC0 register access protection violation register (DM0CMV)
- DMAC1 register access protection violation register (DM1CMV)
- DTS register access protection violation register (DTSCMV)

7.6.3 Setting Up the DMA Channel Setting

The DMA channel setting defines the transfer information and transfer source for each DMAC and DTS channel.

To configure the DMA channel setting, the master assigned to each channel by the channel assignment needs to set up channel registers.

7.6.3.1 Setting Up the DMAC Channel Setting

Follow the procedure below to set up the DMAC channel setting and use the DMAC.

(1) Disabling the DMAC Channel Operation

If the channel operation enable (DTE) in the DMAC channel operation enable setting register (DCENn) is set, clear the DTE bit to disable the channel operation.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DMAC, the following registers need to be set up.

- DMAC source address register (DSAn)
- DMAC destination address register (DDAn)
- DMAC transfer count register (DTCn)
- DMAC transfer control register (DTCTn)
- DMAC reload source address register (DRSAn)
- DMAC reload destination address register (DRDAn)
- DMAC reload transfer count register (DRTCn)
- DMAC transfer count compare register (DTCCn)

(3) Setting Up the DMA Transfer Request

While setting the transfer information, you need to set up the DMA transfer request selection assignment (DTCTn.DRS) bit in the DMAC transfer control register (DTCTn) to define whether the hardware or software DMA transfer request is used.

You cannot use both the hardware and software DMA transfer requests for the same channel at the same time.

If you use the hardware DMA transfer request, you need to select one source used as the hardware DMA transfer request out of 128 hardware DMA transfer sources using the hardware DMA transfer source selection (DTFRn.REQSEL) in the DTFR setting register. Also, you need to enable the hardware DMA transfer source selection (DTFRn.REQEN) in the same register.

The DTFR may retain a hardware DMA transfer request that came before the hardware DMA transfer source is selected. Clear the hardware DMA transfer request (DTFRRQn.DRQ) retained in the DTFR using the DTFR transfer request clear register (DTFRRQCn) if necessary.

If you use the software DMA transfer request, disable the hardware DMA transfer source selection (DTFRn.REQEN) in the DTFR setting register.

(4) Clearing the Transfer Status

The DMAC transfer status register (DCSTn) may retain the result of the previous DMA transfer. You need to clear the flags in the DMAC transfer status register using the DMAC transfer status clear register (DCSTCn).

(5) Enabling the DMAC Channel Operation

Set the channel operation enable (DCENn.DTE) bit in the DMAC channel operation enable setting register to enable the channel operation.

After the channel operation enable bit is set, the DMAC can accept a DMA transfer request and start DMA transfer.

7.6.3.2 Setting Up the DTS Channel Setting

Follow the procedure below to set up the DTS channel setting and use the DTS.

(1) Disabling the Transfer Request by the DTSFSL

Clear the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register of the DTS channel you want to set up the channel setting for. This procedure is not mandatory but recommended in order to prevent a DMA transfer request from being sent mistakenly to the DTS channel currently being configured.

It is also recommended to check the DTS status register (DTSSTS) and confirm that DMA transfer is not ongoing for the DTS channel currently being configured.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DTS, the following registers need to be set up to configure the transfer information.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- DTS transfer count compare register (DTTCCnnn)

(3) Setting Up the DMA Transfer Request

Unlike a DMAC, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request. A DTS has a transfer request pending bit for each channel in the DTSFSL, and both a hardware and software DMA transfer requests are retained in the same transfer request pending bit (DTFSTnnn.DRQ). Therefore, a DTS has no setting for selecting whether the hardware or software transfer request is used.

The DTSFSL may retain a DMA transfer request that came before the transfer information is set up. Clear the DMA transfer request (DTFSTnnn.DRQ) retained in the DTSFSL if necessary, using the DTSFSL transfer request clear register (DTFSCnnn).

(4) Enabling the Transfer Request by the DTSFSL

Set the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register to enable the DMA transfer request for the DTS channel.

After the transfer request enable bit for the DTSFSL is set, the DTS can accept a DMA transfer request and start DMA transfer.

7.7 DMA Trigger Source

7.7.1 List of DMA Trigger Sources

The DMA trigger source assignment for DMA channel n is set in the DTFR setting register (DTFRn).

Table 7.8 List of DMA Trigger Sources (1/4)

DMA Trigger Source		Function/Module	Target Device	
			C1H	C1M
DMACTRG0	Overflow interrupt	ENCA_0	√	√
DMACTRG1	Compare 1 match or capture 1 interrupt		√	√
DMACTRG2	Underflow interrupt		√	√
DMACTRG3	Encoder clear interrupt		√	√
DMACTRG4	Overflow interrupt	ENCA_1	√	√
DMACTRG5	Compare 1 match or capture 1 interrupt		√	√
DMACTRG6	Underflow interrupt		√	√
DMACTRG7	Encoder clear interrupt		√	√
DMACTRG8	CH0 INTTAUD0I0 interrupt	TAUD	√	√
DMACTRG9	CH1 INTTAUD0I1 interrupt		√	√
DMACTRG10	CH2 INTTAUD0I2 interrupt		√	√
DMACTRG11	CH3 INTTAUD0I3 interrupt		√	√
DMACTRG12	CH4 INTTAUD0I4 interrupt		√	√
DMACTRG13	CH5 INTTAUD0I5 interrupt		√	√
DMACTRG14	CH6 INTTAUD0I6 interrupt		√	√
DMACTRG15	CH7 INTTAUD0I7 interrupt		√	√
DMACTRG16	CH8 INTTAUD0I8 interrupt		√	√
DMACTRG17	CH9 INTTAUD0I9 interrupt		√	√
DMACTRG18	CH10 INTTAUD0I10 interrupt		√	√
DMACTRG19	CH11 INTTAUD0I11 interrupt		√	√
DMACTRG20	CH12 INTTAUD0I12 interrupt		√	√
DMACTRG21	CH13 INTTAUD0I13 interrupt		√	√
DMACTRG22	CH14 INTTAUD0I14 interrupt		√	√
DMACTRG23	CH15 INTTAUD0I15 interrupt		√	√
DMACTRG24	CH0 INTTAUD1I0 interrupt		√	√
DMACTRG25	CH1 INTTAUD1I1 interrupt		√	√
DMACTRG26	CH2 INTTAUD1I2 interrupt		√	√
DMACTRG27	CH3 INTTAUD1I3 interrupt		√	√
DMACTRG28	CH4 INTTAUD1I4 interrupt		√	√
DMACTRG29	CH5 INTTAUD1I5 interrupt		√	√
DMACTRG30	CH6 INTTAUD1I6 interrupt		√	√
DMACTRG31	CH7 INTTAUD1I7 interrupt		√	√
DMACTRG32	CH8 INTTAUD1I8 interrupt		√	√
DMACTRG33	CH9 INTTAUD1I9 interrupt		√	√
DMACTRG34	CH10 INTTAUD1I10 interrupt		√	√
DMACTRG35	CH11 INTTAUD1I11 interrupt		√	√
DMACTRG36	CH12 INTTAUD1I12 interrupt		√	√

Table 7.8 List of DMA Trigger Sources (2/4)

DMA Trigger Source		Function/Module	Target Device	
			C1H	C1M
DMACTRG37	CH13 INTTAUD1I13 interrupt	TAUD	√	√
DMACTRG38	CH14 INTTAUD1I14 interrupt		√	√
DMACTRG39	CH15 INTTAUD1I15 interrupt		√	√
DMACTRG40	TAPA0 peak interrupt 0	TAPA	√	√
DMACTRG41	TAPA0 trough interrupt 0		√	√
DMACTRG42	TAPA1 peak interrupt 0		√	√
DMACTRG43	TAPA1 trough interrupt 0		√	√
DMACTRG44	Cycle match detection interrupt	TPBA_0	√	√
DMACTRG45	Duty match detection interrupt		√	√
DMACTRG46	Pattern number match detection interrupt		√	√
DMACTRG47	Cycle match detection interrupt	TPBA_1	√	×
DMACTRG48	Duty match detection interrupt		√	×
DMACTRG49	Pattern number match detection interrupt		√	×
DMACTRG50	ENCA0 compare 0 match or capture 0 interrupt	ENCA_0	√	√
DMACTRG51	ENCA1 compare 0 match or capture 0 interrupt	ENCA_1	√	√
DMACTRG52	CH0 INTTAUJ0I0 interrupt	TAUJ_0	√	√
DMACTRG53	CH1 INTTAUJ0I1 interrupt		√	√
DMACTRG54	CH2 INTTAUJ0I2 interrupt		√	√
DMACTRG55	CH3 INTTAUJ0I3 interrupt		√	√
DMACTRG56	TSG30 compare match interrupt 11 (INTTSG30I11)	TSG3_0	√	√
DMACTRG57	TSG30 compare match interrupt 12 (INTTSG30I12)		√	√
DMACTRG58	TSG30 peak interrupt (INTTSG30IPEK)		√	√
DMACTRG59	TSG30 trough interrupt (INTTSG30IVLY)		√	√
DMACTRG60	TSG31 compare match interrupt 11 (INTTSG31I11)	TSG3_1	√	√
DMACTRG61	TSG31 compare match interrupt 12 (INTTSG31I12)		√	√
DMACTRG62	TSG31 peak interrupt (INTTSG31IPEK)		√	√
DMACTRG63	TSG31 trough interrupt (INTTSG31IVLY)		√	√
DMACTRG64	ADI00 ADCC0 scan group 0 end interrupt	ADCC	√	√
DMACTRG65	ADI01 ADCC0 scan group 1 end interrupt		√	√
DMACTRG66	ADI02 ADCC0 scan group 2 end interrupt		√	√
DMACTRG67	ADI03 ADCC0 scan group 3 end interrupt		√	√
DMACTRG68	ADI04 ADCC0 scan group 4 end interrupt		√	√
DMACTRG69	ADI10 ADCC1 scan group 0 end interrupt		√	√
DMACTRG70	ADI11 ADCC1 scan group 1 end interrupt		√	√
DMACTRG71	ADI12 ADCC1 scan group 2 end interrupt		√	√
DMACTRG72	ADI13 ADCC1 scan group 3 end interrupt		√	√
DMACTRG73	ADI14 ADCC1 scan group 4 end interrupt		√	√
DMACTRG74	Communication status interrupt (INTCSIHTIC)	CSIH_0	√	√
DMACTRG75	Reception status interrupt (INTCSIHTIR)		√	√
DMACTRG76	Job completion interrupt (INTCSIHTIJC)		√	√
DMACTRG77	Communication status interrupt (INTCSIHTIC)	CSIH_1	√	√
DMACTRG78	Reception status interrupt (INTCSIHTIR)		√	√
DMACTRG79	Job completion interrupt (INTCSIHTIJC)		√	√

Table 7.8 List of DMA Trigger Sources (3/4)

DMA Trigger Source		Function/Module	Target Device	
			C1H	C1M
DMACTRG80	Reserved		×	×
DMACTRG81	Reserved		×	×
DMACTRG82	Reserved		×	×
DMACTRG83	Reserved		×	×
DMACTRG84	Reserved		×	×
DMACTRG85	Reserved		×	×
DMACTRG86	RDC20 Z-phase interrupt	RDC_0	√	√
DMACTRG87	RDC20 Compare 0 match interrupt		√	√
DMACTRG88	RDC20 Compare 1 match interrupt		√	√
DMACTRG89	RDC20 Compare 2 match interrupt		√	√
DMACTRG90	RDC20 Excitation timer (ET) DMA request		√	√
DMACTRG91	RDC21 Z-phase interrupt	RDC_1	√	×
DMACTRG92	RDC21 Compare 0 match interrupt		√	×
DMACTRG93	RDC21 Compare 1 match interrupt		√	×
DMACTRG94	RDC21 Compare 2 match interrupt		√	×
DMACTRG95	RDC21 Excitation timer (ET) DMA request		√	×
DMACTRG96	EMU20 interrupt 0	EMU2	√	√
DMACTRG97	EMU20 interrupt 1		√	√
DMACTRG98	EMU20 interrupt 2		√	√
DMACTRG99	EMU20 interrupt 3		√	√
DMACTRG100	EMU20 interrupt 4		√	√
DMACTRG101	EMU21 interrupt 0		√	√
DMACTRG102	EMU21 interrupt 1		√	√
DMACTRG103	EMU21 interrupt 2		√	√
DMACTRG104	EMU21 interrupt 3		√	√
DMACTRG105	EMU21 interrupt 4		√	√
DMACTRG106	Reserved		×	×
DMACTRG107	RXI (receive data full)	SCI_0	√	√
DMACTRG108	TXI (transmit data empty)		√	√
DMACTRG109	RXI (receive data full)	SCI_1	√	√
DMACTRG110	TXI (transmit data empty)		√	√
DMACTRG111	RXI (receive data full)	SCI_2	√	√
DMACTRG112	TXI (transmit data empty)		√	√
DMACTRG113	Reserved		×	×
DMACTRG114	Reserved		×	×
DMACTRG115	Reserved		×	×
DMACTRG116	Reserved		×	×
DMACTRG117	Reserved		×	×
DMACTRG118	Reserved		×	×
DMACTRG119	Reserved		×	×
DMACTRG120	Reserved		×	×
DMACTRG121	Reserved		×	×
DMACTRG122	Reserved		×	×

Table 7.8 List of DMA Trigger Sources (4/4)

DMA Trigger Source		Function/Module	Target Device	
			C1H	C1M
DMACTRG123	Reserved		×	×
DMACTRG124	Reserved		×	×
DMACTRG125	DMA programming command signal	Data flash memory	√	√
DMACTRG126	Reserved		×	×
DMACTRG127	Reserved		×	×

Note 1. The name of DMA trigger source may be different from that of output signal in each module.

7.8 DTS Trigger Source

7.8.1 List of DTS Trigger Sources

Table 7.9 shows the DTS trigger source assignment for DTS channel n.

Table 7.9 List of DTS Trigger Sources (1/4)

DMA (DTS) Trigger Source		Function/Module	Target Device	
			C1H	C1M
DTSTRG0	ADI00 ADCC0 scan group 0 end interrupt	ADCC	√	√
DTSTRG1	ADI01 ADCC0 scan group 1 end interrupt		√	√
DTSTRG2	ADI02 ADCC0 scan group 2 end interrupt		√	√
DTSTRG3	ADI03 ADCC0 scan group 3 end interrupt		√	√
DTSTRG4	ADI04 ADCC0 scan group 4 end interrupt		√	√
DTSTRG5	ADI10 ADCC1 scan group 0 end interrupt		√	√
DTSTRG6	ADI11 ADCC1 scan group 1 end interrupt		√	√
DTSTRG7	ADI12 ADCC1 scan group 2 end interrupt		√	√
DTSTRG8	ADI13 ADCC1 scan group 3 end interrupt		√	√
DTSTRG9	ADI14 ADCC1 scan group 4 end interrupt		√	√
DTSTRG10	Reserved		×	×
DTSTRG11	Reserved		×	×
DTSTRG12	CH0 INTTAUJ0I0 interrupt	TAUJ_0	√	√
DTSTRG13	CH1 INTTAUJ0I1 interrupt		√	√
DTSTRG14	CH2 INTTAUJ0I2 interrupt		√	√
DTSTRG15	CH3 INTTAUJ0I3 interrupt		√	√
DTSTRG16	CH0 INTTAUD0I0 interrupt	TAUD	√	√
DTSTRG17	CH1 INTTAUD0I1 interrupt		√	√
DTSTRG18	CH2 INTTAUD0I2 interrupt		√	√
DTSTRG19	CH3 INTTAUD0I3 interrupt		√	√
DTSTRG20	CH4 INTTAUD0I4 interrupt		√	√
DTSTRG21	CH5 INTTAUD0I5 interrupt		√	√
DTSTRG22	CH6 INTTAUD0I6 interrupt		√	√
DTSTRG23	CH7 INTTAUD0I7 interrupt		√	√
DTSTRG24	CH8 INTTAUD0I8 interrupt		√	√
DTSTRG25	CH9 INTTAUD0I9 interrupt		√	√
DTSTRG26	CH10 INTTAUD0I10 interrupt		√	√
DTSTRG27	CH11 INTTAUD0I11 interrupt		√	√
DTSTRG28	CH12 INTTAUD0I12 interrupt		√	√
DTSTRG29	CH13 INTTAUD0I13 interrupt		√	√
DTSTRG30	CH14 INTTAUD0I14 interrupt		√	√
DTSTRG31	CH15 INTTAUD0I15 interrupt		√	√
DTSTRG32	CH0 INTTAUD1I0 interrupt		√	√
DTSTRG33	CH1 INTTAUD1I1 interrupt		√	√
DTSTRG34	CH2 INTTAUD1I2 interrupt		√	√
DTSTRG35	CH3 INTTAUD1I3 interrupt		√	√
DTSTRG36	CH4 INTTAUD1I4 interrupt		√	√

Table 7.9 List of DTS Trigger Sources (2/4)

DMA (DTS) Trigger Source		Function/Module	Target Device	
			C1H	C1M
DTSTRG37	CH5 INTTAUD1I5 interrupt	TAUD	√	√
DTSTRG38	CH6 INTTAUD1I6 interrupt		√	√
DTSTRG39	CH7 INTTAUD1I7 interrupt		√	√
DTSTRG40	CH8 INTTAUD1I8 interrupt		√	√
DTSTRG41	CH9 INTTAUD1I9 interrupt		√	√
DTSTRG42	CH10 INTTAUD1I10 interrupt		√	√
DTSTRG43	CH11 INTTAUD1I11 interrupt		√	√
DTSTRG44	CH12 INTTAUD1I12 interrupt		√	√
DTSTRG45	CH13 INTTAUD1I13 interrupt		√	√
DTSTRG46	CH14 INTTAUD1I14 interrupt		√	√
DTSTRG47	CH15 INTTAUD1I15 interrupt		√	√
DTSTRG48	ENCA0 compare 0 match or capture 0 interrupt	ENCA_0	√	√
DTSTRG49	ENCA1 compare 0 match or capture 0 interrupt	ENCA_1	√	√
DTSTRG50	TAPA0 peak interrupt 0	TAPA	√	√
DTSTRG51	TAPA0 trough interrupt 0		√	√
DTSTRG52	TAPA1 peak interrupt 0		√	√
DTSTRG53	TAPA1 trough interrupt 0		√	√
DTSTRG54	Overflow interrupt	ENCA_0	√	√
DTSTRG55	Compare 1 match or capture 1 interrupt		√	√
DTSTRG56	Underflow interrupt		√	√
DTSTRG57	Encoder clear interrupt		√	√
DTSTRG58	Overflow interrupt	ENCA_1	√	√
DTSTRG59	Compare 1 match or capture 1 interrupt		√	√
DTSTRG60	Underflow interrupt		√	√
DTSTRG61	Encoder clear interrupt		√	√
DTSTRG62	Cycle match detection interrupt	TPBA_0	√	√
DTSTRG63	Duty match detection interrupt		√	√
DTSTRG64	Pattern number match detection interrupt		√	√
DTSTRG65	Cycle match detection interrupt	TPBA_1	√	×
DTSTRG66	Duty match detection interrupt		√	×
DTSTRG67	Pattern number match detection interrupt		√	×
DTSTRG68	RDC20 Z-phase interrupt	RDC_0	√	√
DTSTRG69	RDC20 Compare 0 match interrupt		√	√
DTSTRG70	RDC20 Compare 1 match interrupt		√	√
DTSTRG71	RDC20 Compare 2 match interrupt		√	√
DTSTRG72	RDC20 Excitation timer (ET) DMA request		√	√
DTSTRG73	RDC21 Z-phase interrupt	RDC_1	√	×
DTSTRG74	RDC21 Compare 0 match interrupt		√	×
DTSTRG75	RDC21 Compare 1 match interrupt		√	×
DTSTRG76	RDC21 Compare 2 match interrupt		√	×
DTSTRG77	RDC21 Excitation timer (ET) DMA request		√	×

Table 7.9 List of DTS Trigger Sources (3/4)

DMA (DTS) Trigger Source		Function/Module	Target Device	
			C1H	C1M
DTSTRG78	EMU20 interrupt 0	EMU2	√	√
DTSTRG79	EMU20 interrupt 1		√	√
DTSTRG80	EMU20 interrupt 2		√	√
DTSTRG81	EMU20 interrupt 3		√	√
DTSTRG82	EMU20 interrupt 4		√	√
DTSTRG83	EMU21 interrupt 0		√	√
DTSTRG84	EMU21 interrupt 1		√	√
DTSTRG85	EMU21 interrupt 2		√	√
DTSTRG86	EMU21 interrupt 3		√	√
DTSTRG87	EMU21 interrupt 4		√	√
DTSTRG88	Reserved		×	×
DTSTRG89	Reserved		×	×
DTSTRG90	Reserved		×	×
DTSTRG91	Reserved		×	×
DTSTRG92	TSG30 compare match interrupt 11 (INTTSG30I11)	TSG3_0	√	√
DTSTRG93	TSG30 compare match interrupt 12 (INTTSG30I12)		√	√
DTSTRG94	TSG30 peak interrupt (INTTSG30IPEK)		√	√
DTSTRG95	TSG30 trough interrupt (INTTSG30IVLY)		√	√
DTSTRG96	TSG31 compare match interrupt 11 (INTTSG31I11)	TSG3_1	√	√
DTSTRG97	TSG31 compare match interrupt 12 (INTTSG31I12)		√	√
DTSTRG98	TSG31 peak interrupt (INTTSG31IPEK)		√	√
DTSTRG99	TSG31 trough interrupt (INTTSG31IVLY)		√	√
DTSTRG100	Communication status interrupt (INTCSIHTIC)	CSIH_0	√	√
DTSTRG101	Reception status interrupt (INTCSIHTIR)		√	√
DTSTRG102	Job completion interrupt (INTCSIHTIJC)		√	√
DTSTRG103	Communication status interrupt (INTCSIHTIC)	CSIH_1	√	√
DTSTRG104	Reception status interrupt (INTCSIHTIR)		√	√
DTSTRG105	Job completion interrupt (INTCSIHTIJC)		√	√
DTSTRG106	Reserved		×	×
DTSTRG107	Reserved		×	×
DTSTRG108	Reserved		×	×
DTSTRG109	Reserved		×	×
DTSTRG110	Reserved		×	×
DTSTRG111	Reserved		×	×
DTSTRG112	RXI (receive data full)	SCI_0	√	√
DTSTRG113	TXI (transmit data empty)		√	√
DTSTRG114	RXI (receive data full)	SCI_1	√	√
DTSTRG115	TXI (transmit data empty)		√	√
DTSTRG116	RXI (receive data full)	SCI_2	√	√
DTSTRG117	TXI (transmit data empty)		√	√
DTSTRG118	Reserved		×	×
DTSTRG119	Reserved		×	×
DTSTRG120	Reserved		×	×

Table 7.9 List of DTS Trigger Sources (4/4)

DMA (DTS) Trigger Source		Function/Module	Target Device	
			C1H	C1M
DTSTRG121	Reserved		×	×
DTSTRG122	Reserved		×	×
DTSTRG123	Reserved		×	×
DTSTRG124	Reserved		×	×
DTSTRG125	Reserved		×	×
DTSTRG126	Reserved		×	×
DTSTRG127	Reserved		×	×

7.9 Global Register

7.9.1 List of Global Register Addresses

Address = Base address "FFFF 8000_H" + Offset address

Table 7.10 List of Global Register Addresses (1/2)

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission	
				Special Master	General Master
DMASS	0000_H	DMACTL	DMA control register	√	×
DMASS	0010_H	DTSCTL1	DTS control register 1	√	×
DMASS	0014_H	DTSCTL2	DTS control register 2	√	×
DMASS	0018_H	DTSSTS	DTS status register	√	√
DMASS	0020_H	DMACER	DMAC error register	√	√
DMASS	0024_H	DTSER1	DTS error register 1	√	√
DMASS	0028_H	DTSER2	DTS error register 2	√	√
DMASS	$002C_H$	DTSERC	DTS error clear register	√	×
DMASS	0030_H	DM0CMV	DMAC0 register access protection violation register	√	×
DMASS	0034_H	DM1CMV	DMAC1 register access protection violation register	√	×
DMASS	0038_H	DTSCMV	DTS register access protection violation register	√	×
DMASS	$003C_H$	CMVC	Register access protection violation clear register	√	×
DMASS	0060_H	DTSPR0	DTS channel priority setting register 0	√	×
DMASS	0064_H	DTSPR1	DTS channel priority setting register 1	√	×
DMASS	0068_H	DTSPR2	DTS channel priority setting register 2	√	×
DMASS	$006C_H$	DTSPR3	DTS channel priority setting register 3	√	×
DMASS	0070_H	DTSPR4	DTS channel priority setting register 4	√	×
DMASS	0074_H	DTSPR5	DTS channel priority setting register 5	√	×
DMASS	0078_H	DTSPR6	DTS channel priority setting register 6	√	×
DMASS	$007C_H$	DTSPR7	DTS channel priority setting register 7	√	×
DMASS	0080_H	DTRECCTL	DTSRAM ECC control register	√	×
DMASS	0084_H	DTRERINT	DTSRAM Error notification control register	√	×
DMASS	0094_H	DTRTSCTL	DTSRAM test control register	√	×
DMASS	0098_H	DTRTWDAT	DTSRAM test write data register	√	×
DMASS	$009C_H$	DTRTRDAT	DTSRAM test read data register	√	×
DMASS	0100_H	DM00CM	DMAC channel 0 channel master setting register	√	×
DMASS	0104_H	DM01CM	DMAC0 channel 1 channel master setting register	√	×
DMASS	0108_H	DM02CM	DMAC0 channel 2 channel master setting register	√	×
DMASS	$010C_H$	DM03CM	DMAC0 channel 3 channel master setting register	√	×
DMASS	0110_H	DM04CM	DMAC0 channel 4 channel master setting register	√	×
DMASS	0114_H	DM05CM	DMAC0 channel 5 channel master setting register	√	×
DMASS	0118_H	DM06CM	DMAC0 channel 6 channel master setting register	√	×
DMASS	$011C_H$	DM07CM	DMAC0 channel 7 channel master setting register	√	×
DMASS	0120_H	DM10CM	DMAC1 channel 0 channel master setting register	√	×
DMASS	0124_H	DM11CM	DMAC1 channel 1 channel master setting register	√	×
DMASS	0128_H	DM12CM	DMAC1 channel 2 channel master setting register	√	×
DMASS	$012C_H$	DM13CM	DMAC1 channel 3 channel master setting register	√	×

Table 7.10 List of Global Register Addresses (2/2)

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission	
				Special Master	General Master
DMASS	0130_H	DM14CM	DMAC1 channel 4 channel master setting register	√	×
DMASS	0134_H	DM15CM	DMAC1 channel 5 channel master setting register	√	×
DMASS	0138_H	DM16CM	DMAC1 channel 6 channel master setting register	√	×
DMASS	$013C_H$	DM17CM	DMAC1 channel 7 channel master setting register	√	×
DMASS	0200_H + 4 × [DTS channel number] *1 (0200_H - $03FC_H$)	DTSnnnCM *1	DTS channel nnn channel master setting register*1	√	×

Note 1. [DTS channel number] and "nnn" in the register symbols and meanings are numbers in the range from 000 to 127.

7.9.2 Details of Global Registers

7.9.2.1 DMACTL — DMA Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8000_H

Value after reset: 0000 0000_H

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

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

Table 7.11 DMACTL Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DMASPD	DMA suspension This bit shows whether DMA transfer for all channels is suspended. If a user writes 1 to this bit, DMA transfer for all channels can be suspended. If a user writes 0 to this bit, suspension of DMA transfer for all channels can be cleared. The suspension controlled by this bit is independent from the suspension controlled by the transfer enable bit (DTE) of each DMAC channel and the suspension setting bit (DTSUST) for a DTS. That means, if this bit is 1, all DMA transfers are suspended regardless of the values of the DTE bit of each channel and the DTSUST bit of the DTS. Writing to this bit does not affect the DTE bit of each channel and the DTSUST bit of the DTS. 0: DMA suspension cleared 1: DMA suspension request/DMA suspension ongoing

CAUTION

In order to abort/resume DMA transfer during the execution by a DTS, you need to execute the operation of enable/disable for a DMA transfer request enable bit (DTFSLnnn.REQEN). For the details, see the notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

7.9.2.2 DTSCTL1 — DTS Control Register 1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8010_H

Value after reset: 0000 0000_H

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

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

Table 7.12 DTSCTL1 Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSUST	DTS suspension This bit shows whether DMA transfer of a DTS is suspended. If a user writes 1 to this bit, DMA transfer of a DTS can be suspended. 0: DTS suspension cleared 1: DTS suspension request/DTS suspension ongoing

CAUTION

In order to abort/resume DMA transfer during the execution by a DTS, you need to execute the operation of enable/disable for a DMA transfer request enable bit (DTFSLnnn.REQEN). For the details, see the notes in **Section 7.3.3, Suspension, Resume, and Transfer Abort of a DTS**.

7.9.2.3 DTSCTL2 — DTS Control Register 2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8014_H

Value after reset: $0000\ 0000_H$

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

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

Table 7.13 DTSCTL2 Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSTIT	DTS transfer abort request While the DTS is suspended, a user can write 1 to this bit to abort the suspended DMA transfer. When the suspended DMA transfer of a DTS is aborted, the DTSSTS.DTSACT bit is cleared to 0. This bit is always read as 0.

7.9.2.4 DTSSTS — DTS Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8018$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DTS CYC	DTSACH[6:0]							DTS ACT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.14 DTSSTS Register Contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is returned.
8	DTSCYC	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in the DTS. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7 to 1	DTSACH[6:0]	DTS transfer channel If there is a channel in the DTS executing DMA transfer, the channel number is shown. If there is no channel in the DTS executing DMA transfer, the channel number of the last DMA transfer is shown.
0	DTSACT	DTS transfer status This bit shows whether there is a channel in the DTS executing DMA transfer. 0: There is a channel in the DTS executing DMA transfer. 1: There is no channel in the DTS executing DMA transfer. If the DTS is put into the suspended state while there is a channel executing DMA transfer, this bit remains 1. If a DTS transfer abort request is made using the DTSCTL2.DTSTIT bit, the suspended DTS transfer is aborted, and this bit is cleared to 0. When DMA transfer error occurs and the DMA transfer is aborted, this bit is cleared.

7.9.2.5 DMACER — DMAC Error Register

Access: This register can be read in 32-bit units.

Address: FFFF 8020$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM1ER[7:0]								DM0ER[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.15 DMACER Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned.
15 to 8	DM1ER[7:0]	DMAC1 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC1. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC1 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated
7 to 0	DM0ER[7:0]	DMAC0 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC0. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC0 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated

7.9.2.6 DTSER1 — DTS Error Register 1

Access: This register can be read in 32-bit units.

Address: FFFF 8024_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DTSERCH[6:0]							—	—	—	—	—	—	DTSERWR	DTSER
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.16 DTSER1 Register Contents

Bit Position	Bit Name	Function
31 to 15	Reserved	When read, the value after reset is returned.
14 to 8	DTSERCH[6:0]	DTS error channel These bits show the DTS channel number of the first DMA transfer error after the DTSER bit is cleared to 0. These bits are read-only and cannot be cleared.
7 to 2	Reserved	When read, the value after reset is returned.
1	DTSERWR	DTS DMA transfer error occurring cycle This bit is updated at the same time as setting of the DTS DMA transfer error flag (DTSER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the DTSER bit has been set. If the DTSER bit is cleared, this bit is also cleared to 0. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
0	DTSER	DTS DMA transfer error flag This bit shows whether DMA transfer error is generated in the DTS. 0: DMA transfer error is not generated 1: DMA transfer error is generated If DMA transfer error is generated in the DTS while this bit is 0, this bit is set, and DTSERCH6 to DTSERCH0 retain the DTS channel number of the DMA transfer error. If DMA transfer error is generated in the DTS while this bit is 1, this bit remains 1, and TSERCH6 to DTSERCH0 do not change. This bit can be cleared by using the DTSERC register.

7.9.2.7 DTSER2 — DTS Error Register 2

Access: This register can be read in 32-bit units.

Address: FFFF 8028$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMADED	RAMDEDOV	—	—	RAMDEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSED	RAMSEDOV	—	—	RAMSEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.17 DTSER2 Register Contents (1/2)

Bit Position	Bit Name	Function
31	RAMDED	DTSRAM 2-bit error flag This bit shows whether the 2-bit error is generated in the read access to the DTSRAM. 0: 2-bit error is not generated in the DTSRAM 1: 2-bit error is generated in the DTSRAM If the 2-bit error occurs in DTSRAM read access while this bit is 0, this bit is set, and RAMDEDAD11 to RAMDEDAD0 retain the DTSRAM address of the error. If the 2-bit error occurs in DTSRAM read access while this bit is 1, this bit remains 1, and RAMDEDAD11 to RAMDEDAD0 do not change. This bit can be cleared by using the DTSERC register.
30	RAMDEDOV	DTSRAM 2-bit error overflow flag This bit is set when the RAMDED bit is 1 and the 2-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMDEDAD11 to RAMDEDAD0 bits. This bit can be cleared by operation of the DTSERC register.
29, 28	Reserved	When read, the value after reset is returned.
27 to 16	RAMDEDAD [11:0]	DTSRAM 2-bit error address These bits show the DTSRAM address at which the first 2-bit error occurs in DTSRAM read access after the RAMDED bit is cleared to 0. These bits are read-only and cannot be cleared.
15	RAMSED	DTSRAM 1-bit error flag This bit shows whether the 1-bit error is generated in the read access to the DTSRAM. 0: 1-bit error is not generated in the DTSRAM 1: 1-bit error is generated in the DTSRAM If the 1-bit error occurs in the DTSRAM while this bit is 0, this bit is set, and RAMSEDAD11 to RAMSEDAD0 retain the DTSRAM address of the error. If the 1-bit error occurs in DTSRAM read access while this bit is 1, this bit remains 1, and RAMSEDAD11 to RAMSEDAD0 do not change. This bit can be cleared by using the DTSERC register.
14	RAMSEDOV	DTSRAM 1-bit error overflow flag This bit is set when the RAMSED bit is 1 and the 1-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMSEDAD11 to RAMSEDAD0 bits. This bit can be cleared by operation of the DTSERC register.
13, 12	Reserved	When read, the value after reset is returned.

Table 7.17 DTSER2 Register Contents (2/2)

Bit Position	Bit Name	Function
11 to 0	RAMSEDAD [11:0]	DTSRAM 1-bit error address These bits show the DTSRAM address of the first DTSRAM 1-bit error after the RAMSED bit is cleared to 0. These bits are read-only and cannot be cleared.

7.9.2.8 DTSERC — DTS Error Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 802C$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSEDC	RAMSEDOVC	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSERC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.18 DTSERC Register Contents

Bit Position	Bit Name	Function
31	RAMDEDC	DTSRAM 2-bit error flag clear If a user writes 1 to this bit, the DTSRAM 2-bit error flag (DTSER2.RAMDED) is cleared. 0 is always read from this bit.
30	RAMDEDOVC	DTSRAM 2-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 2-bit error overflow flag (DTSER2.RAMDEDOV) is cleared. The read value of this bit is always 0.
29 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15	RAMSEDC	DTSRAM 1-bit error flag clear If a user writes 1 to this bit, the DTSRAM 1-bit error flag (DTSER2.RAMSED) is cleared. 0 is always read from this bit.
14	RAMSEDOVC	DTSRAM 1-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 1-bit error overflow flag (DTSER2.RAMSEDOV) is cleared. The read value of this bit is always 0.
13 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTSERC	DTS error flag clear If a user writes 1 to this bit, the DTS DMA error flag (DTSER1.DTSER) is cleared. 0 is always read from this bit.

7.9.2.9 DM0CMV — DMAC0 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8030_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 7.19 DM0CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC0. 0: No illegal access has occurred in the DMAC0 1: Illegal access has occurred in the DMAC0 If an illegal access occurs in the DMAC0 while this bit is 0, this bit is set, and MINF6 to MINF0 and VCH2 to VCH0 store their respective information. If an illegal access occurs in the DMAC0 while this bit is 1, this bit remains 1, and MINF6 to MINF0 and VCH2 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.10 DM1CMV — DMAC1 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8034_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 7.20 DM1CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	Reserved	When read, the value after reset is returned.
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC1. 0: No illegal access has occurred in the DMAC1 1: Illegal access has occurred in the DMAC1 If an illegal access occurs in the DMAC1 while this bit is 0, this bit is set, and MINF6 to MINF0 and VCH2 to VCH0 store their respective information. If an illegal access occurs in the DMAC1 while this bit is 1, this bit remains 1, and MINF6 to MINF0 and VCH2 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.11 DTSCMV — DTS Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8038$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 7.21 DTSCMV Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is returned.
22 to 17	MINF[6:1]	Illegal access master information These bits show the master information of the originator of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 11	Reserved	When read, the value after reset is returned.
10 to 4	VCH[6:0]	Illegal access channel These bits show the channel number (0 to 127) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	Reserved	When read, the value after reset is returned.
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DTS. 0: No illegal access has occurred in the DTS 1: Illegal access has occurred in the DTS If an illegal access occurs in the DTS while this bit is 0, this bit is set, and MINF6 to MINF0 and VCH2 to VCH0 store their respective information. If an illegal access occurs in the DTS while this bit is 1, this bit remains 1, and MINF6 to MINF0 and VCH2 to VCH0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.12 CMVC — Register Access Protection Violation Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 803C$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSVC	DM1VC	DM0VC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 7.22 CMVC Register Contents

Bit Position	Bit Name	Function
31 to 3	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
2	DTSVC	DTS illegal access flag clear The DTS illegal access flag (DTSCMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
1	DM1VC	DMAC1 illegal access flag clear The DMAC1 illegal access flag (DM1CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
0	DM0VC	DMAC0 illegal access flag clear The DMAC0 illegal access flag (DM0CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.9.2.13 DTSPRn — DTS Channel Priority Setting Register (n = 0 to 7)

- DTSPR0

Access: This register can be read/written in 32-bit units.

Address: FFFF 8060$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS15PR[1:0]		DTS14PR[1:0]		DTS13PR[1:0]		DTS12PR[1:0]		DTS11PR[1:0]		DTS10PR[1:0]		DTS9PR[1:0]		DTS8PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS7PR[1:0]		DTS6PR[1:0]		DTS5PR[1:0]		DTS4PR[1:0]		DTS3PR[1:0]		DTS2PR[1:0]		DTS1PR[1:0]		DTS0PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.23 DTSPR0 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[15:0] PR[1:0]	DTS channel [15:0] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8064$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS31PR[1:0]		DTS30PR[1:0]		DTS29PR[1:0]		DTS28PR[1:0]		DTS27PR[1:0]		DTS26PR[1:0]		DTS25PR[1:0]		DTS24PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS23PR[1:0]		DTS22PR[1:0]		DTS21PR[1:0]		DTS20PR[1:0]		DTS19PR[1:0]		DTS18PR[1:0]		DTS17PR[1:0]		DTS16PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.24 DTSPR1 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[31:16] PR[1:0]	DTS channel [31:16] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8068$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS47PR[1:0]		DTS46PR[1:0]		DTS45PR[1:0]		DTS44PR[1:0]		DTS43PR[1:0]		DTS42PR[1:0]		DTS41PR[1:0]		DTS40PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS39PR[1:0]		DTS38PR[1:0]		DTS37PR[1:0]		DTS36PR[1:0]		DTS35PR[1:0]		DTS34PR[1:0]		DTS33PR[1:0]		DTS32PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.25 DTSPR2 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[47:32] PR[1:0]	DTS channel [47:32] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR3

Access: This register can be read/written in 32-bit units.

Address: FFFF 806C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS63PR[1:0]		DTS62PR[1:0]		DTS61PR[1:0]		DTS60PR[1:0]		DTS59PR[1:0]		DTS58PR[1:0]		DTS57PR[1:0]		DTS56PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS55PR[1:0]		DTS54PR[1:0]		DTS53PR[1:0]		DTS52PR[1:0]		DTS51PR[1:0]		DTS50PR[1:0]		DTS49PR[1:0]		DTS48PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.26 DTSPR3 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[63:48] PR[1:0]	DTS channel [63:48] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR4

Access: This register can be read/written in 32-bit units.

Address: FFFF 8070$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS79PR[1:0]		DTS78PR[1:0]		DTS77PR[1:0]		DTS76PR[1:0]		DTS75PR[1:0]		DTS74PR[1:0]		DTS73PR[1:0]		DTS72PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS71PR[1:0]		DTS70PR[1:0]		DTS69PR[1:0]		DTS68PR[1:0]		DTS67PR[1:0]		DTS66PR[1:0]		DTS65PR[1:0]		DTS64PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.27 DTSPR4 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[79:64] PR[1:0]	DTS channel [79:64] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR5

Access: This register can be read/written in 32-bit units.

Address: FFFF 8074$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS95PR[1:0]		DTS94PR[1:0]		DTS93PR[1:0]		DTS92PR[1:0]		DTS91PR[1:0]		DTS90PR[1:0]		DTS89PR[1:0]		DTS88PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS87PR[1:0]		DTS86PR[1:0]		DTS85PR[1:0]		DTS84PR[1:0]		DTS83PR[1:0]		DTS82PR[1:0]		DTS81PR[1:0]		DTS80PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.28 DTSPR5 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[95:80] PR[1:0]	DTS channel [95:80] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR6

Access: This register can be read/written in 32-bit units.

Address: FFFF 8078$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS111PR[1:0]		DTS110PR[1:0]		DTS109PR[1:0]		DTS108PR[1:0]		DTS107PR[1:0]		DTS106PR[1:0]		DTS105PR[1:0]		DTS104PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS103PR[1:0]		DTS102PR[1:0]		DTS101PR[1:0]		DTS100PR[1:0]		DTS99PR[1:0]		DTS98PR[1:0]		DTS97PR[1:0]		DTS96PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.29 DTSPR6 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[111:96] PR[1:0]	DTS channel [111:96] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR7

Access: This register can be read/written in 32-bit units.

Address: FFFF 807C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS127PR[1:0]		DTS126PR[1:0]		DTS125PR[1:0]		DTS124PR[1:0]		DTS123PR[1:0]		DTS122PR[1:0]		DTS121PR[1:0]		DTS120PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS119PR[1:0]		DTS118PR[1:0]		DTS117PR[1:0]		DTS116PR[1:0]		DTS115PR[1:0]		DTS114PR[1:0]		DTS113PR[1:0]		DTS112PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.30 DTSPR7 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[127:112] PR[1:0]	DTS channel [127:112] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

7.9.2.14 DTRECCTL — DTSRAM ECC Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8080_H

Value after reset: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEC DIS	ECC DIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.31 DTRECCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCDIS and SECDIS bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	SECDIS	DTSRAM SEC error correction disable This bit enables or disables SEC error correction when the ECCDIS bit is 0. 1-bit ECC error detection operation is always executed when ECCDIS bit is 0 regardless of the state of this bit. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error correction is enabled when the SEC error is detected. 1: Error correction is disabled when the SEC error is detected.
0	ECCDIS	DTSRAM ECC disable This bit enables or disables DTSRAM ECC error detection and correction. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error detection and correction are enabled. 1: Error detection and correction are disabled. **NOTE** The encoding function is effective when error detection and correction are disabled.

7.9.2.15 DTRERINT — DTSRAM Error Notification Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8084_H

Value after reset: 0000 0002_H

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

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

Table 7.32 DTRERINT Register Contents

Bit Position	Bit Name	Function
31 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	DEDIE	DTSRAM 2-bit error notification enable This bit enables or disables notification of 2-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 2-bit error to ECM is disabled. 1: Notification of 2-bit error to ECM is enabled.
0	SEDIE	DTSRAM 1-bit error notification enable This bit enables or disables notification of 1-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 1-bit error to ECM is disabled. 1: Notification of 1-bit error to ECM is enabled.

7.9.2.16 DTRTSCTL — DTSRAM Test Control Register

This register is used for ECC test (self-diagnosis). It enables setting of ECC test mode and selection of ECC data to be written to the DTSRAM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 8094$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC TST	DAT SEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.33 DTRTSCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCTST and DATSEL bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
1	ECCTST	DTSRAM ECC Test Mode This bit enables or disables DTSRAM ECC test mode. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC test mode is enabled. 1: ECC test mode is disabled.
0	DATSEL	ECC Test Data Selection This bit is valid when ECCTST is 1 and selects ECC data to be written to the DTSRAM. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC encoded from the written data is used. 1: The value specified by the DTSRAM test write data register (DTRTWDAT) is used.

7.9.2.17 DTRTWDAT — DTSRAM Test Write Data Register

This register is used for ECC test (self-diagnosis). It specifies ECC data to be written to the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read/written in 32-bit units.

Address: FFFF 8098$_H$

Value after reset: 0000 0000$_H$

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

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

Table 7.34 DTRTWDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 0	TWDAT[6:0]	ECC Test Write Data These bits specify ECC data to be written to the DTSRAM when DTRTSCTL.ECCTST = 1 and DTRTSCTL.DATSEL = 1. Writing to these bits is enabled when DTRTSCTL.ECCTST = 1 When DTRTSCTL.ECCTST = 0, these bits cannot be written and their read value is 0.

7.9.2.18 DTRTRDAT — DTSRAM Test Read Data Register

This register is used for ECC test (self-diagnosis). It reads out ECC data of the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read in 32-bit units.

Address: FFFF 809C$_H$

Value after reset: 0000 0000$_H$

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

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

Table 7.35 DTRTRDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is returned.
6 to 0	TRDAT[6:0]	ECC Test Read Data These bits retain the last ECC data read out from the DTSRAM when DTRTSCTL.ECCTST = 1. When DTRTSCTL.ECCTST = 0, the read value of these bits is 0.

7.9.2.19 DMnnCM — DMAC Channel Master Setting Register (nn = 00 to 07, 10 to17)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8100_H + 4 × Ch. No. n (n = 0 to 7)
FFFF 8120_H + 4 × Ch. No. n - 10 (n = 10 to 17)

Value after reset: 0000 0010_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 7.36 DMnnCM Register Contents

Bit Position	Bit Name	Function ion
31 to 7	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
6 to 4	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
3, 2	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
1	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
0	Reserved	When read, the value after reset is returned. When writing to this bit, write the value after reset.

CAUTION

DM00CM to DM07CM configure the channel master information of the DMAC0 channels 0 to 7 respectively.

DM10CM to DM17CM configure the channel master information of the DMAC1 channels 0 to 7 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8200_H + 4_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

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

Table 7.37 DTS Channel Master Setting Register Contents

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
22 to 20	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
19, 18	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
17	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
16	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
15 to 0	CMC[15:0]	Transfer count compare In terms of contents, this field is the same as bits [15:0] of the **Section 7.11.3.8, DTTCCnnn — DTS Transfer Count Compare Register**.

CAUTION

DTS000CM to DTS127CM configure the channel master information of the DTS channels 0 to 127 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

CAUTION

The 16 low-order bits of this register are shared with the DTS transfer count compare register, one of the DTS channel registers.

If you write to this register, the DTS transfer count compare register is updated as well.

Recommended setup procedure of the DTS channel master setting register

When the special master configures the overall DMA operation, the channel master setting must be configured in bits 22 to 17 in this register, and 0 must be specified in bits 15 to 0 and reserved bits 31 to 23 and 16.

When the master assigned to the channel updates the value of the transfer count compare, use the DTS transfer count compare register instead.

Bits 31 to 26 of this register are reserved, but you can read and write those bits. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value by software.

7.10 DMAC Channel Register

7.10.1 DMAC Channel Register Address

Address = Base address "FFFF 8000_H" + Offset address

Table 7.38 DMAC Channel Register Address

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission	
				Special Master	General Master
DMASS	$0400_H + 40_H \times$ [channel number]	DSAn	DMAC source address	√	√
DMASS	$0404_H + 40_H \times$ [channel number]	DDAn	DMAC destination address	√	√
DMASS	$0408_H + 40_H \times$ [channel number]	DTCn	DMAC transfer count	√	√
DMASS	$040C_H + 40_H \times$ [channel number]	DTCTn	DMAC transfer control	√	√
DMASS	$0410_H + 40_H \times$ [channel number]	DRSAn	DMAC reload source address	√	√
DMASS	$0414_H + 40_H \times$ [channel number]	DRDAn	DMAC reload destination address	√	√
DMASS	$0418_H + 40_H \times$ [channel number]	DRTCn	DMAC reload transfer count	√	√
DMASS	$041C_H + 40_H \times$ [channel number]	DTCCn	DMAC transfer count compare	√	√
DMASS	$0420_H + 40_H \times$ [channel number]	DCENn	DMAC channel operation enable setting	√	√
DMASS	$0424_H + 40_H \times$ [channel number]	DCSTn	DMAC transfer status	√	√
DMASS	$0428_H + 40_H \times$ [channel number]	DCSTSn	DMAC transfer status set	√	√
DMASS	$042C_H + 40_H \times$ [channel number]	DCSTCn	DMAC transfer status clear	√	√
DMASS	$0430_H + 40_H \times$ [channel number]	DTFRn	DTFR setting	√	√
DMASS	$0434_H + 40_H \times$ [channel number]	DTFRRQn	DTFR transfer request status	√	√
DMASS	$0438_H + 40_H \times$ [channel number]	DTFRRQCn	DTFR transfer request clear	√	√

Note 1. The [channel number] in the offset addresses and "n" in the register symbols are numbers in the range from 0 to 15, and the correspondence between the channel number n and the channel is as follows.

Channel number n	Channel
0	DMAC0 channel 0
1	DMAC0 channel 1
2	DMAC0 channel 2
3	DMAC0 channel 3
4	DMAC0 channel 4
5	DMAC0 channel 5
6	DMAC0 channel 6
7	DMAC0 channel 7
8	DMAC1 channel 0
9	DMAC1 channel 1
10	DMAC1 channel 2
11	DMAC1 channel 3
12	DMAC1 channel 4
13	DMAC1 channel 5
14	DMAC1 channel 6
15	DMAC1 channel 7

7.10.2 Details of DMAC Channel Registers

The “n” in the register symbols indicates the DMA channel number (n = 0 to 15).

7.10.2.1 DSAn — DMAC Source Address Register

Access: This register can be read/written in 32-bit units.

Address: $FFFF8400_H + 40_H \times$ Ch. No. n (n = 0 to 15)

Value after reset: $0000\ 0000_H$

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

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

Table 7.39 DSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specify the DMA transfer source address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer source address for the next DMA cycle is read.

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. The address must be set up while the DTE bit is 0.
3. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).
   Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.2 DDAn — DMAC Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8404_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.40 DDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specify the DMA transfer destination address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer destination address for the next DMA cycle is read.

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. The address must be set up while the DTE bit is 0.
3. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.
4. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit).
   Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.3 DTCn — DMAC Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8408_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 7.41 DTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specify the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. If you read from those bits during DMA transfer, the address reload count for the next DMA cycle is read. When the reload function 2 or block transfer 2 is used, ARC[15:0] is decremented by one for every DMA cycle. When the reload function 2 or block transfer 2 is not used, ARC[15:0] bits are not updated. If the value is 0000H, it means that the number of transfers until the address reload when the reload function 2 is used and the number of transfers when the block transfer 2 is used are 65536.
15 to 0	TRC[15:0]	Transfer count Configure the number of transfers. TRC[15:0] is decremented by one whenever a DMA cycle is executed. If you read from those bits, the remaining number of transfers for the next DMA cycle is read. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained.

TRC15-0	Operation
0000_H	The number of transfers is 65536, or the transfer is complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTIONS

1. Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.
2. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.

7.10.2.4 DTCTn — DMAC Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 840C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	DRS	—	—	—	—	—	CHNSEL[2:0]			CHNE[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	MLE	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.42 DTCTn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
27	ESE	Transfer error case DMA transfer disable setting Configures whether a DMA cycle is executed when the DCSTn.ER bit is set due to DMA transfer error. If this bit is cleared to 0, even when the DCSTn.ER bit is set due to DMA transfer error, the following DMA cycles can be executed. If this bit is set to 1, the following DMA cycles are not executed when the DCSTn.ER bit is set due to DMA transfer error. 0: DMA cycles are executed while the DCSTn.ER bit is set. 1: DMA cycles are not executed while the DCSTn.ER bit is set.
26	DRS	DMA transfer request selection assignment Selects the type of DMA transfer requests to be accepted. 0: Software DMA transfer request 1: Hardware DMA transfer request
25 to 21	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
20 to 18	CHNSEL[2:0]	Next channel in the chain Specify the next channel in the chain. The next channel must be another channel in the same DMAC. You cannot specify a channel in the different DMAC or in the DTS. You cannot specify the same channel for the next channel. (If you do, correct operation is not guaranteed.)
17, 16	CHNE[1:0]	Chain enable Select the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: Setting prohibited (No guarantee of operation) 11: Always chain A chain request is generated at the completion of every DMA cycle.
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.

Table 7.42 DTCTn Register Contents (2/3)

Bit Position	Bit Name	Function
13	MLE	Continuous transfer enable If this bit is set, the DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request. 0: The DTE bit is cleared at the completion of DMA transfer. In addition, the next DMA transfer can start only after the TC bit is cleared. 1: The DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request.
12, 11	RLD2M[1:0]	Reload function 2 setting Configure the reload function 2. 00: Reload function 2 is disabled. 01: Reload function 2 is enabled. The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 10: Reload function 2 is enabled. The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 11: Reload function 2 is enabled. The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10, 9	RLD1M[1:0]	Reload function 1 setting Configure the reload function 1. 00: Reload function 1 is disabled. 01: Reload function 1 is enabled. The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 10: Reload function 1 is enabled. The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 11: Reload function 1 is enabled. The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
8, 7	DACM[1:0]	Destination address count direction Specify the direction of counting for the destination address.

DACM1	DACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

6, 5	SACM[1:0]	Source address count direction Specify the direction of counting for the source address.

SACM1	SACM0	Direction of Counting
0	0	Up
0	1	Down
1	0	Fixed
1	1	Setting prohibited (No guarantee of operation)

Table 7.42 DTCTn Register Contents (3/3)

Bit Position	Bit Name	Function
4 to 2	DS[2:0]	Transfer data size Specify the transfer data size.

DS2	DS1	DS0	Transfer Data Size
0	0	0	8 bits
0	0	1	16 bits
0	1	0	32 bits
0	1	1	64 bits
1	0	0	128 bits
Other than the above			Setting prohibited (No guarantee of operation)

1, 0	TRM[1:0]	Transfer mode Specify the DMA transfer mode. 00: Single transfer 01: Block transfer 1 (The number of transfers is specified by the transfer count.) 10: Block transfer 2 (The number of transfers is specified by the address reload count.) 11: Setting prohibited (No guarantee of operation)

CAUTIONS

1. Setting the MLE bit from 1 to 0 is only enabled while the channel operation is enabled (the DTE bit = 1). (Otherwise, correct operation is not guaranteed.)
2. If prohibited settings are made to some of the bits, correct operation is not guaranteed.

7.10.2.5 DRSAn — DMAC Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8410_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.43 DRSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specify the source address to be reloaded to the DMA source address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.6 DRDAn — DMAC Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8414_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.44 DRDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specify the destination address to be reloaded to the DMA destination address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.7 DRTCn — DMAC Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8418_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 7.45 DRTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specify the value to be loaded to the address reload count in the transfer count register at the timing of reload when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specify the value to be loaded to the transfer count in the transfer count register at the timing of reload when the reload function 1 is used.

7.10.2.8 DTCCn — DMAC Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 841C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

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

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

Table 7.46 DTCCn Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 0	CMC[15:0]	Transfer count compare Configure the transfer count to be compared to the transfer count register. At the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register, the transfer count match flag (DCSTn.CC) in the DMAC transfer status register is set. Furthermore, if the transfer count match interrupt enable (DTCTn.CCE) bit is 1, a transfer count match interrupt is generated. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, the transfer count match flag in the DMAC transfer status register is never set, and a transfer count match interrupt is never generated.

CAUTION

Writing to this register is prohibited while the channel operation is enabled (DTE bit = 1). If it is written, correct operation is not guaranteed.

7.10.2.9 DCENn — DMAC Channel Operation Enable Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8420_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.47 DCENn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DTE	Channel operation enable Specifies whether to enable or disable the transfer operation of the channel. If the DTE bit is 1, DMA transfer starts when there is a DMA transfer request. If the MLE bit is 0, the DTE bit is cleared automatically at the completion of the DMA transfer. In addition, if 0 is written to the DTE bit during DMA transfer, the DMA transfer is suspended. If 1 is written to the DTE bit during suspension, the suspension is cleared and the DMA transfer resumes. 0: Channel operation is disabled/Channel suspended 1: Channel operation is enabled/Channel suspension cleared

7.10.2.10 DCSTn — DMAC Transfer Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8424_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ERWR	—	—	CY	ER	—	CC	TC	—	—	DR	SR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.48 DCSTn Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 12	Reserved	When read, the value after reset is returned.
11	ERWR	DMA Transfer Error occurring cycle This bit is updated at the same time as setting of the DMA transfer error flag (ER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the ER bit has been set. If the ER bit is cleared, this bit is also cleared to 0. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
10, 9	Reserved	When read, the value after reset is returned.
8	CY	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in this channel. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7	ER	Transfer error flag This bit is set when DMA transfer error is generated. If this bit is 1 and the DTCTn.ESE bit is set, a DMA cycle is not executed when a DMA transfer request is generated. 0: No DMA transfer error is generated 1: DMA transfer error is generated
6	Reserved	When read, the value after reset is returned.
5	CC	Transfer count match flag This bit is set at the completion of the DMA cycle in which the remaining transfer count is the same as the value set in the transfer compare register. 0: No compare match has occurred with the transfer count compare register. 1: Compare match has occurred with the transfer count compare register.
4	TC	Transfer completion flag This bit is set at the completion of the last transfer and shows whether the DMA transfer is complete. If the MLE bit is 0 and this bit is 1, a DMA cycle is not executed when a DMA transfer request is generated. 0: DMA transfer incomplete 1: DMA transfer complete
3, 2	Reserved	When read, the value after reset is returned.

Table 7.48 DCSTn Register Contents (2/2)

Bit Position	Bit Name	Function
1	DR	Hardware DMA transfer request status This bit shows whether there is a hardware DMA transfer request (DMARQ) from the DTFR. This bit changes regardless of the value of the DTE bit when a hardware DMA transfer request from the DTFR is generated. If the software DMA transfer request is selected by the transfer request selection bit (DRS) in the DMAC transfer control register, this bit is not set even when a hardware DMA transfer request is input from the DTFR. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request
0	SR	Software DMA transfer request flag This bit shows whether there is a software DMA transfer request (DMARQ). This bit is automatically cleared when DMA transfer is executed. A user can set this bit by writing 1 to the SRS bit in the DMAC transfer status set register (DCSTSn). In addition, a user can clear this bit by writing 1 to the SRC bit in the DMAC transfer status clear register (DCSTCn), but if this is done, the ongoing DMA transfer is aborted and cannot be resumed. 0: There is no software DMA transfer request 1: There is a software DMA transfer request

7.10.2.11 DCSTSn — DMAC Transfer Status Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8428_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.49 DCSTSn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	SRS	Software DMA transfer request flag A user can set the software DMA transfer request flag (SR) by writing 1 to this bit. 0 is always read from this bit.

7.10.2.12 DCSTCn — DMAC Transfer Status Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 842C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ERC	—	CCC	TCC	—	—	—	SRC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R	R	R/W

Table 7.50 DCSTCn Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
7	ERC	Transfer error flag clear The DMA transfer error flag (ER) can be cleared by writing 1 to this bit. 0 is always read from this bit.
6	Reserved	When read, the value after reset is returned. When writing to this bit, write the value after reset.
5	CCC	Transfer count match flag clear The transfer count match flag (CC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
4	TCC	Transfer completion flag clear The transfer completion flag (TC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
3 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	SRC	Software DMA transfer request flag clear The software DMA transfer request flag (SR) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.10.2.13 DTFRn — DTFR Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8430_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: $0000\ 0000_H$

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

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

Table 7.51 DTFRn Register Contents

Bit Position	Bit Name	Function
31 to 8	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
7 to 1	REQSEL[6:0]	Hardware DMA transfer source selection Select one out of 128 hardware DMA transfer sources as the hardware DMA transfer request. 000_0000: Selecting the DMACTRG[0] input : 111_1111: Selecting the DMACTRG[127] input
0	REQEN	Hardware DMA transfer source selection enable This bit enables or disables the hardware DMA transfer source selection. 0: Hardware DMA transfer source selection is disabled. 1: Hardware DMA transfer source selection is enabled. If this bit is 0, even when the hardware DMA transfer source selected by the REQSEL6 to REQSEL0 bits is activated, it is not recognized as a hardware DMA transfer request, and a hardware DMA transfer request is not generated.

7.10.2.14 DTFRRQn — DTFR Transfer Request Status Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8434_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.52 DTFRRQn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned.
0	DRQ	Hardware DMA transfer request status If this bit is set, it means that a hardware DMA transfer request exists or is retained. • If the hardware DMA transfer request is an edge detection type*1 This bit shows whether a hardware DMA transfer request generated by edge detection is retained. When the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is automatically cleared. A user can clear this bit by writing 1 to the DTFRRQC.DRQC bit. • If the hardware DMA transfer request is a level input type*1 This bit shows whether there is a hardware DMA transfer request input. Even when the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is not automatically cleared. In addition, this bit is not cleared even when a user writes to the DTFRRQC.DRQC bit. This bit changes regardless of the value of the DTFRn.REQEN bit when a hardware DMA transfer request is generated. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.10.2.15 DTFRRQCn — DTFR Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8438_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

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

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

Table 7.53 DTFRRQCn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQC	Hardware DMA transfer request clear If the hardware DMA transfer request is an edge detection type*1, a user can clear the DTFRRQ.DRQ bit by writing 1 to this bit. If the hardware DMA transfer request is a level input type*1, the DTFRRQ.DRQ bit cannot be cleared by writing to this bit. 0 is always read from this bit.

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.11 DTS Channel Register

7.11.1 Transfer information of the DTS (TI)

7.11.1.1 Structure of the TI

The transfer information of the DTS is called TI. One set of TI consists of 32 bits. 8 sets of TI are assigned for each channel. The 8 sets of TI is called TI-A, TI-B, TI-C, TI-D, TI- E, TI-F, TI-G, and TI-H.

Figure 7.12 shows the structure of the TI.

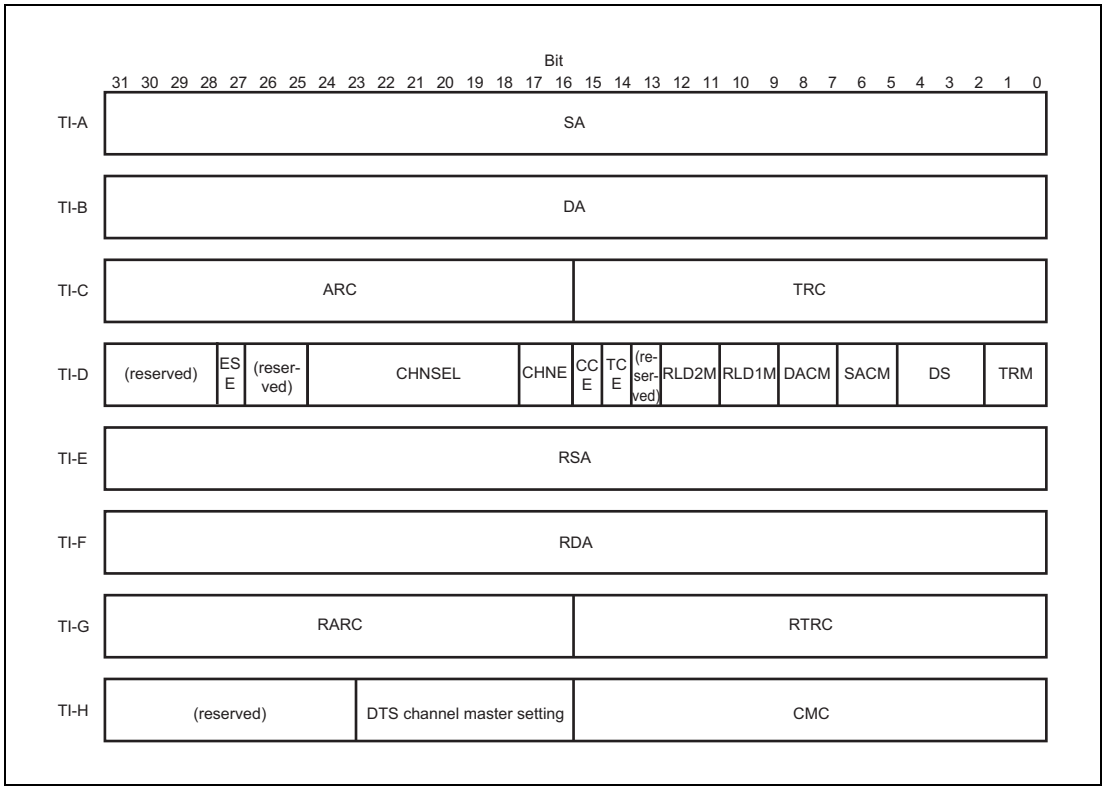


Figure 7.12 Structure of the TI

7.11.1.2 Organization of the TI in the DTSRAM

A user indirectly accesses the DTSRAM by way of the DTS channel registers for each channel and the DTS channel master setting registers.

Therefore, usually, a user does not have to think about the address organization of the TI in the DTSRAM. As an exception, when ECC error occurs while the DTSRAM is read, the address of the ECC error in the DTSRAM is stored to the DTSRAM error register 2 (DTSER2), one of the global registers. In order to know which channel and TI have generated the error, you need to understand the address organization of the TI in the DTSRAM.

Figure 7.13 shows the address organization of the TI in the DTSRAM.

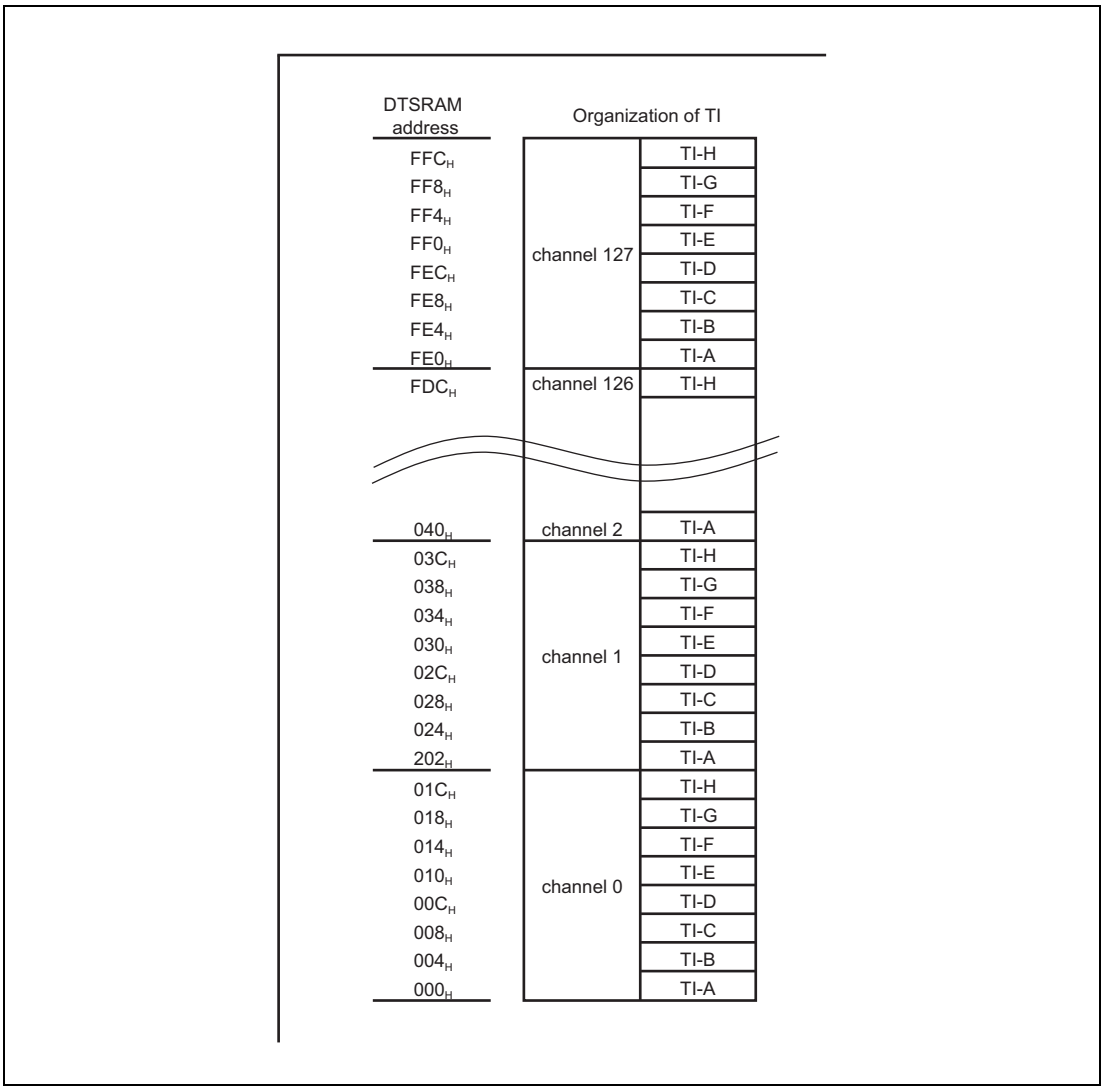


Figure 7.13 Organization of the TI in the DTSRAM

7.11.1.3 Accessing the TI

TI-A can be accessed by way of the DTS source address register (DTSAnnn) for each channel.

TI-B can be accessed by way of the DTS destination address register (DTDAnnn) for each channel.

TI-C can be accessed by way of the DTS transfer count register (DTTCnnn) for each channel.

TI-D can be accessed by way of the DTS transfer control register (DTTCTnnn) for each channel.

TI-E can be accessed by way of the DTS reload source address register (DTRSAnnn) for each channel.

TI-F can be accessed by way of the DTS reload destination address register (DTRDAnnn) for each channel.

TI-G can be accessed by way of the DTS reload transfer count register (DTRTCnnn) for each channel.

TI-H can be accessed by way of the channel master setting register (DTSnnnCM), which is a global register, and the transfer count compare register (DTTCCnnn) for each channel.

7.11.1.4 Caution about accessing the TI

The data of the DTS channel master setting register and the data of the DTS transfer count compare register are stored to the same TI-H.

Access to the DTS channel master setting register (DTSnnnCM) is actually 32-bit access to the whole TI-H. Therefore, when you write to the DTS channel master setting register, the values of the 16 lower-order bits of the DTS transfer count compare (CMC) are updated at the same time. When you read from the DTS channel master setting register, the value of the DTS transfer count compare (CMC) is read into the 16 lower-order bits.

When you read from the DTS transfer count compare register (DTTCCnnn), 32-bit data is read from the TI-H, but only data of the 16 lower-order bits is actually seen in the result of the register read. When you write to the DTS transfer count compare register (DTTCCnnn), read-modify-write access to the 16 lower-order bits for the 32-bit TI-H is made. Data in the TI immediately after the reset is undefined. It should be noted that, if you try to write to the DTS transfer count compare register (DTTCCnnn) before setting up the DTS channel master setting register, ECC error may be detected during the read of the read-modify-write access.

Bits 31 to 23 of the TI-H are not used, but you can read and write those bits by accessing the DTS channel master setting register. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value by software.

After the reset, the values in the DTSRAM, which stores the TI, are undefined. After the reset, if you read TI before you write to the TI, ECC error is generated. Therefore, the first access to the following registers after the reset must be write access. The first access after the reset should never be read access.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- Channel master setting registers (DTSnnnCM)

In addition, after the reset, the first access to the DTS transfer count compare register (DTTCCnnn) must be always after write access to the channel master setting register (DTSnnnCM).

You can access the TI from a CPU while the DTS is executing DMA transfer. But if you do so, the following should be noted.

- While a channel is executing DMA transfer, the TI of the channel should not be updated by TI access from a CPU. If this situation happens, the result of the DMA transfer and the contents of the TI may mismatch.
- If TI access is requested from a CPU while a TI fetch or TI write back is executed, the TI access is executed after the completion of the TI fetch or TI write back. If a TI fetch or TI write back is requested while a TI access request from a CPU is processed, the TI fetch or TI write back is executed after the processing of the TI access is complete.

7.11.2 DTS Channel Register Address

Address = Base address "FFFF 9000_H" + Offset address

Module Name	Offset Address	Register Symbol	Meaning	Accessed Permission	
				Special Master	General Master
DMASS	$0000_H + 40_H \times$ [channel number]	DTSAnnn	DTS source address	√	√
DMASS	$0004_H + 40_H \times$ [channel number]	DTDAnnn	DTS destination address	√	√
DMASS	$0008_H + 40_H \times$ [channel number]	DTTCnnn	DTS transfer count	√	√
DMASS	$000C_H + 40_H \times$ [channel number]	DTTCTnnn	DTS transfer control	√	√
DMASS	$0010_H + 40_H \times$ [channel number]	DTRSAnnn	DTS reload source address	√	√
DMASS	$0014_H + 40_H \times$ [channel number]	DTRDAnnn	DTS reload destination address	√	√
DMASS	$0018_H + 40_H \times$ [channel number]	DTRTCnnn	DTS reload transfer count	√	√
DMASS	$001C_H + 40_H \times$ [channel number]	DTTCCnnn	DTS transfer count compare	√	√
DMASS	$0020_H + 40_H \times$ [channel number]	DTFSLnnn	DTSFSL operation setting	√	√
DMASS	$0024_H + 40_H \times$ [channel number]	DTFSTnnn	DTSFSL transfer request status	√	√
DMASS	$0028_H + 40_H \times$ [channel number]	DTFSSnnn	DTSFSL transfer request set	√	√
DMASS	$002C_H + 40_H \times$ [channel number]	DTFSCnnn	DTSFSL transfer request clear	√	√

Note: The [channel number] in the offset addresses is a number in the range from 0 to 127.
The "nnn" in the register symbols is a 3-digit number in the range from 000 to 127.

7.11.3 Details of DTS Channel Registers

The “nnn” in the register symbols indicates the DTS channel number (nnn = 000 to 127).

7.11.3.1 DTSAnnn — DTS Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9000_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

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

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

Table 7.54 DTSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specify the DMA transfer source address. SA[31:0] is updated at the timing of the TI write back and retains the DMA transfer source address for the next DMA transfer.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.2 DTDAnnn — DTS Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9004_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

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

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

Table 7.55 DTDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specify the DMA transfer destination address. DA[31:0] is updated at the timing of the TI write back and retains the DMA transfer destination address for the next DMA transfer.

CAUTIONS

1. If DMA transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.
2. DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.3 DTTCnnn — DTS Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9008_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 7.56 DTTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specify the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. When the reload function 2 or block transfer 2 is used, ARC[15:0] is decremented by one for every DMA cycle and updated at the timing of the TI write back. When the reload function 2 or block transfer 2 is not used, ARC[15:0] is not updated. If 0000_H is set, address reload is disabled. If the value at the start of a DMA cycle is 0000_H, subtraction from the address reload count does not proceed even after the DMA cycle.
15 to 0	TRC[15:0]	Transfer count Configure the number of transfers. TRC[15:0] is decremented by one whenever a DMA cycle is executed. It is updated at the timing of the TI write back. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained. If 0000_H is set, DMA transfer is not executed even when a DMA transfer request is accepted.

TRC15 to 0	Operation
0000_H	Transfer is disabled or complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTIONS

1. If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.
2. Unlike a DMAC, "0000_H" in the transfer count of the DTS does not mean "65536 transfers" but means that transfer is disabled or complete.

7.11.3.4 DTTCTnnn — DTS Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 900C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: XXXX XXXX$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	—	—	CHNSEL[6:0]							CHNE[1:0]	
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	—	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.57 DTTCTnnn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
27	ESE	Transfer error case DMA transfer abort setting Specifies whether to abort DMA transfer when DMA transfer error is generated. If this bit is cleared to 0, DMA transfer continues when DMA transfer error is generated. If this bit is set to 1, DMA transfer is aborted when DMA transfer error is generated. 0: DMA transfer continues when DMA transfer error is generated. 1: DMA transfer is aborted when DMA transfer error is generated.
26, 25	Reserved	When read, the value returned is undefined. When writing to these bits, write 0.
24 to 18	CHNSEL[6:0]	Next channel in the chain Specify the next channel in the chain. The next channel must be another channel in the DTS. You cannot specify a channel in a DMAC. You cannot specify the same channel for the next channel. (If you do, correct operation is not guaranteed.)
17, 16	CHNE[1:0]	Chain enable Select the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: Setting prohibited (No guarantee of operation) 11: Always chain A chain request is generated at the completion of every DMA cycle.
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.
13	Reserved	When read, the value returned is undefined. When writing to this bit, write 0.

Table 7.57 DTTCTnnn Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>12, 11</td><td>RLD2M[1:0]</td><td>Reload function 2 setting
Configure the reload function 2.
00: Reload function 2 is disabled.
01: Reload function 2 is enabled.
The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10: Reload function 2 is enabled.
The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
11: Reload function 2 is enabled.
The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.</td></tr>
<tr><td>10, 9</td><td>RLD1M[1:0]</td><td>Reload function 1 setting
Configure the reload function 1.
00: Reload function 1 is disabled.
01: Reload function 1 is enabled.
The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
10: Reload function 1 is enabled.
The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
11: Reload function 1 is enabled.
The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)</td></tr>
<tr><td>8, 7</td><td>DACM[1:0]</td><td>Destination address count direction
Specify the direction of counting for the destination address.
<table>
<tr><th>DACM1</th><th>DACM0</th><th>Direction of Counting</th></tr>
<tr><td>0</td><td>0</td><td>Up</td></tr>
<tr><td>0</td><td>1</td><td>Down</td></tr>
<tr><td>1</td><td>0</td><td>Fixed</td></tr>
<tr><td>1</td><td>1</td><td>Setting prohibited (No guarantee of operation)</td></tr>
</table></td></tr>
<tr><td>6, 5</td><td>SACM[1:0]</td><td>Source address count direction
Specify the direction of counting for the source address.
<table>
<tr><th>SACM1</th><th>SACM0</th><th>Direction of Counting</th></tr>
<tr><td>0</td><td>0</td><td>Up</td></tr>
<tr><td>0</td><td>1</td><td>Down</td></tr>
<tr><td>1</td><td>0</td><td>Fixed</td></tr>
<tr><td>1</td><td>1</td><td>Setting prohibited (No guarantee of operation)</td></tr>
</table></td></tr>
<tr><td>4 to 2</td><td>DS[2:0]</td><td>Transfer data size
Specify the transfer data size.
<table>
<tr><th>DS2</th><th>DS1</th><th>DS0</th><th>Transfer Data Size</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>8 bits</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>16 bits</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>32 bits</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>64 bits</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>128 bits</td></tr>
<tr><td colspan="3">Other than the above</td><td>Setting prohibited (No guarantee of operation)</td></tr>
</table></td></tr>
</table>

Table 7.57 DTTCTnnn Register Contents (3/3)

Bit Position	Bit Name	Function
1, 0	TRM[1:0]	Transfer mode Specify the DMA transfer mode. 00: Single transfer 01: Block transfer 1 (The number of transfers is specified by the transfer count.) 10: Block transfer 2 (The number of transfers is specified by the address reload count.) 11: Setting prohibited (No guarantee of operation)

CAUTIONS

1. If prohibited settings are made to some of the bits, correct operation is not guaranteed.
2. Bits 31 to 28, 26, 25, and 13 are unused, but you can read and write those bits. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value.

7.11.3.5 DTRSAnnn — DTS Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9010_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

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

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

Table 7.58 DTRSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specify the source address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.6 DTRDAnnn — DTS Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9014_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

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

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

Table 7.59 DTRDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specify the destination address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer of misaligned data is not supported. The 4 lower-order bits of addresses corresponding to the transfer data size are as follows (× indicates any bit). Correct operation is not guaranteed if settings do not comply with the following table.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.7 DTRTCnnn — DTS Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9018_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: XXXX $XXXX_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 7.60 DTRTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specify the value to be reloaded to the address reload count when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specify the value to be reloaded to the transfer count when the reload function 1 is used.

RTRC[15:0]	Operation
0000_H	No DMA transfer
0001_H	1 transfer
:	:
$FFFF_H$	65535 transfers

7.11.3.8 DTTCCnnn — DTS Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 901C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: XXXX XXXX$_H$

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

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

Table 7.61 DTTCCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
15 to 0	CMC[15:0]	Transfer count compare Configure the transfer count to be compared to the transfer count register. If the transfer count match interrupt enable (DTTCTnnn.CCE) bit in the DTS transfer control register is 1, a transfer count match interrupt is generated at the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, a transfer count match interrupt is not generated.

CAUTION

This register must be accessed after the DTS channel master setting register is set up.

If you access this register without setting up the DTS channel master setting register after the reset, ECC error may be generated during access to this register.

7.11.3.9 DTFSLnnn — DTSFSL Operation Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9020_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

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

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

Table 7.62 DTFSLnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	REQEN	DMA transfer request enable This bit selects whether a DMA transfer request from this channel retained in the DTSFSL is used as a candidate in DTS channel arbitration. 0: A DMA transfer request from this channel is not used as a candidate in DTS channel arbitration. 1: A DMA transfer request from this channel is used as a candidate in DTS channel arbitration. If this bit is 0, even if the DTSFSL retains a DMA transfer request, this channel is not a candidate in DTS channel arbitration inside the DTSFSL, and consequently a DMA transfer request from this channel is not generated.

7.11.3.10 DTFSTnnn — DTSFSL Transfer Request Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 9024_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

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

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

Table 7.63 DTFSTnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned.
0	DRQ	DMA transfer request pending This bit shows whether a DMA transfer request of this channel is pending. This bit is set when input of a hardware transfer source is detected, or when software writes “1” to the DTFSSnnn.DRQ bit. This bit is automatically cleared when the DTS receives the DMA transfer request signal while the DTSFSL is requesting DMA transfer of this channel. Alternatively, software can clear this bit by writing 1 to the DTFSCnnn.DRQC bit. 0: A DMA transfer request is not pending. 1: A DMA transfer request is pending.

7.11.3.11 DTFSSnnn — DTSFSL Transfer Request Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9028_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

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

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

Table 7.64 DTFSSnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQS	DMA transfer request set A user can set the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

7.11.3.12 DTFSCnnn — DTSFSL Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF $902C_H$ + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

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

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

Table 7.65 DTFSCnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	Reserved	When read, the value after reset is returned. When writing to these bits, write the value after reset.
0	DRQC	DMA transfer request clear A user can clear the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

Section 8 Resets

8.1 Features of the RH850/C1x Reset

- The $\overline{\text{RESET}}$ pin incorporates a noise canceller.
- The source of a reset can be determined by referring to the reset source determination register.
- The CPU can assert a reset signal by setting a register.

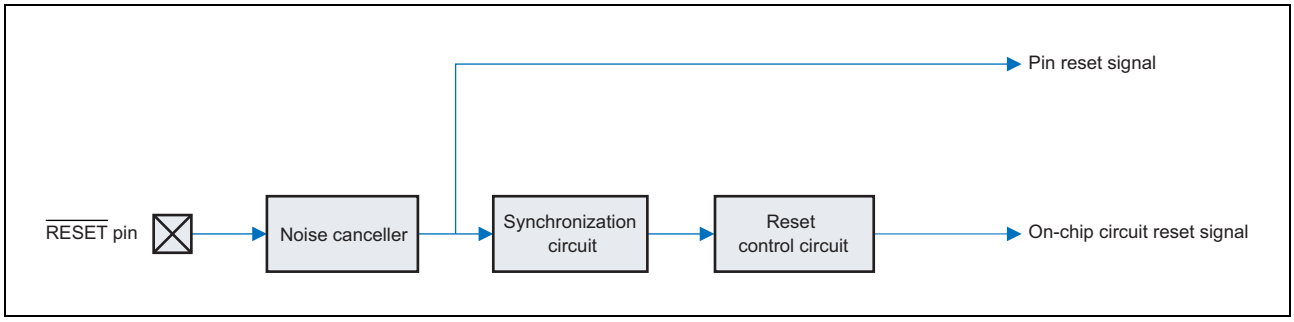


Figure 8.1 Reset Circuit

8.2 Reset State

8.2.1 External Reset State

When a low-level pulse longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{\text{RESET}}$ pin, external reset is received and this LSI transits to the external reset state. When external reset is received, individual pins transit to the external reset state. Refer to **Table 2.61**, in **Section 2, Pins**, for the state of individual pins during external reset.

The low-level pulse width of the input signal must be longer than t_{RESW} (t_{cyc}) because the $\overline{\text{RESET}}$ pin incorporates a noise cancelling circuit. This LSI transits to the internal reset state by driving the $\overline{\text{RESET}}$ pin to the high-level after the required low-level period.

CAUTION

Refer to Section 35, Electrical Characteristics, for t_{RESNCW} and t_{RESW}.

8.2.2 Internal Reset State

When a high-level signal longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{\text{RESET}}$ pin in the external reset state, this product transits to the internal reset state. Refer to **Table 2.61** in **Section 2, Pins**, for the state of individual pins.

CAUTION

Refer to Section 35, Electrical Characteristics, for t_{RESNCW}.

8.3 Reset Sources

The following are the reset sources of this product.

Some registers are only initialized by the external reset state. That is, they are only initialized by input of the low level on the $\overline{\text{RESET}}$ pin.

However, most registers are initialized by either an external or internal reset state. This means that they are initialized by a reset from any source.

Regarding sources for the initialization of registers, see the descriptions of the registers in the relevant sections. Where there is no explicit description of the sources for resetting and initializing a register, the register is initialized by either an external or internal reset state. That is, such registers are initialized by a reset from any source. However, the value after resets of some registers are undefined. Take care since the values of such registers following a reset are not fixed.

Source	Description
When the $\overline{\text{RESET}}$ pin is driven to the low level	Transits to the external reset state.
When a reset request is issued from ECM	Transits to the internal reset state.
When a reset is issued from the debugger (When a forced reset is issued)	Transits to the external reset state.
When the CPU sets the software reset request register	Transits to the internal reset state

8.4 Register Specifications

8.4.1 List of Registers

Reset registers are listed in the following table.

Table 8.1 Registers

Module Name	Register Name	Symbol	Address	Access Protection
SYS	Reset source determination register	RESF	FFF8 2800_H	
SYS	Reset Source Clear Register	RESFC	FFF8 2808_H	
SYS	Software reset request register	SWRESA	FFF8 $AC18_H$	PROT1PHCMD

8.4.2 RESF — Reset Source Determination Register

This register determines the reset source.

The flags of this register can be cleared by the RESFC register and can only be initialized by an external reset, but not an internal reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESF1	RESF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 8.2 RESF Register Contents

Bit Position	Bit Name	Function
1	RESF1	ECM Reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred
0	RESF0	Software Reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred

8.4.3 RESFC — Reset Source Clear Register

This register clears the reset source indicated by the RESF register.

This register is always read as 0000 0000_H, and can only be initialized by an external reset, but not an internal reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESFC1	RESFC0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 8.3 RESFC Register Contents

Bit Position	Bit Name	Function
1	RESFC1	ECM reset Clears the status bit. 0: Do not clear 1: Clear
0	RESFC0	Software reset Clears the status bit. 0: Do not clear 1: Clear

8.4.4 SWRESA — Software Reset Request Register

This register generates an internal reset when accessed. A software reset is issued when 1 is written to SWRESA.

This register can be protected by the PROT1PHCMD register.

This register can be reset by either an internal or external reset.

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

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

Table 8.4 SWRESA Register Contents

Bit Position	Bit Name	Function
0	SWRESA	0: — (default) 1: Internal reset enabled (internal reset triggered)

Example) Sequence of writing to the SWRESA register

Writing to SWRESA requires the special sequence to unlock protection described below.

A protection error is displayed in the PROTIPS register if the following steps are not observed.

For details of the PROT1PS register, see **Section 10.2.11, PROT1PS — Protect 1 Status Register**.

Step 1. Write the fixed value (0000 $00A5_H$) to the PROT1PHCMD register.

Step 2. Write the setting value (0000 0001_H) to the SWRESA register.

Step 3. Write the inverse (FFFF $FFFE_H$) to the SWRESA register.

Step 4. Write the new setting value (0000 0001_H) to the SWRESA register.

The setting value (0000 0001_H) can only be written to register SWRESA by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting value (0000 0001_H) is not written to register SWRESA, and the PROT1PS.PROTERR bit is set to 1.

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. For a target register, see the note in **Section 10.2.10, PROT1PHCMD Protect 1 Command Register**.

8.5 Procedure

8.5.1 Software Reset

Figure 8.2 shows the flow for setting a software reset.

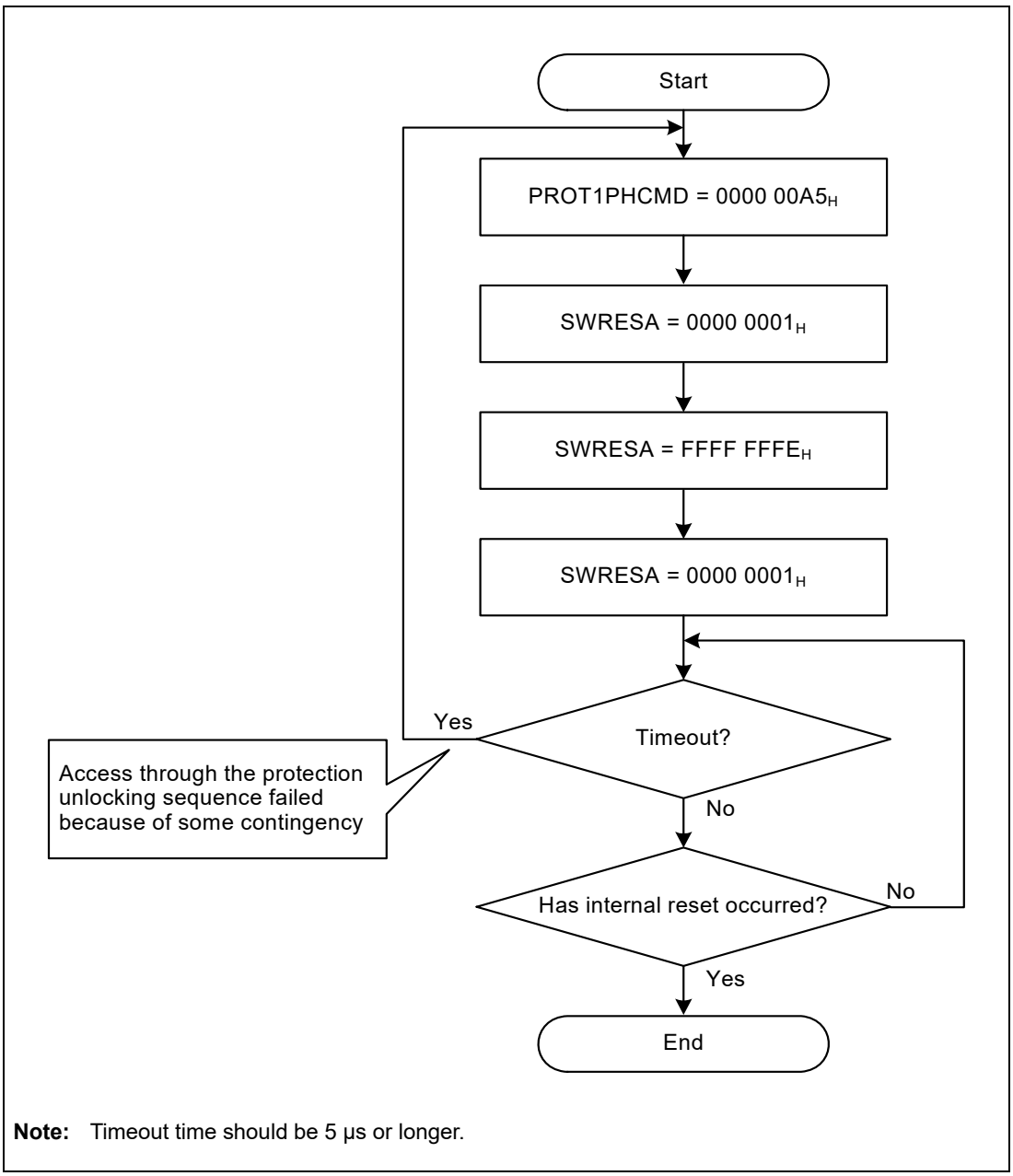


Figure 8.2 Flow of Software Reset

8.6 Notes

The debug function includes a reset masking function. To prevent unexpected operations of this function, fix the $\overline{\text{DCUTRST}}$ pin to the low level when the debug function is not used.

Section 9 Power Supply Circuit

This section contains a generic description of the RH850/C1x power supply circuit.

9.1 Features of the RH850/C1x Power Supply Circuit

- Connection of an external pass transistor (EPT), but only in the case of the C1H.
  Since an EPT cannot be used with the C1M, be sure to make settings for the C1M as described in **Section 9.4.2, Settings when an EPT is not in Use**.

With an EPT	Without an EPT
Operation with a two-output power management IC (5 V, 3.3 V)	Operation with a three-output power management IC (5 V, 3.3 V, 1.25 V)

- Power supply
  The power supply pins and the power supply voltages to be used are listed in the following table. During operations, supply the specified voltages to all power lines. When stopping operation, turn off all of the power supplies.
  Connect VCC and PLLVCC must be connected to the same electric potential.
  And then, A0VCC, A1VCC and RVCC must be connected to the same electric potential.

Power Supply Name for Power Supply Pin	Pin Name	Power Supply Voltage in Operation	Use of Power Supply
SYSVCC		3.0 V to 3.6 V	Power for system logic
VCC		3.0 V to 3.6 V	Power for oscillator and flash programming
PLLVCC*1		3.0 V to 3.6 V	PLL
VDD		1.15 V to 1.35 V	Power for the core (direct power supply) and connection of the stabilizing capacitor for core power
EVCC		4.5 V to 5.5 V	Ports (5 V)
A0VCC/A1VCC		4.5 V to 5.5 V	Power for SAR AD
	A0VREFH/ A1VREFH	4.5 V to 5.5 V	Reference voltage for SAR AD
	EPTVOUT	—	Controls the EPT to supply VDD Note: The VDD voltage is not externally applied if an EPT is connected.
RVCC		4.5 V to 5.5 V	Power for RDC

Note 1. PLLVCC is included only in the C1H. In the C1M, PLLVCC is integrated into VCC.

9.2 Examples of Connection of Power Management ICs

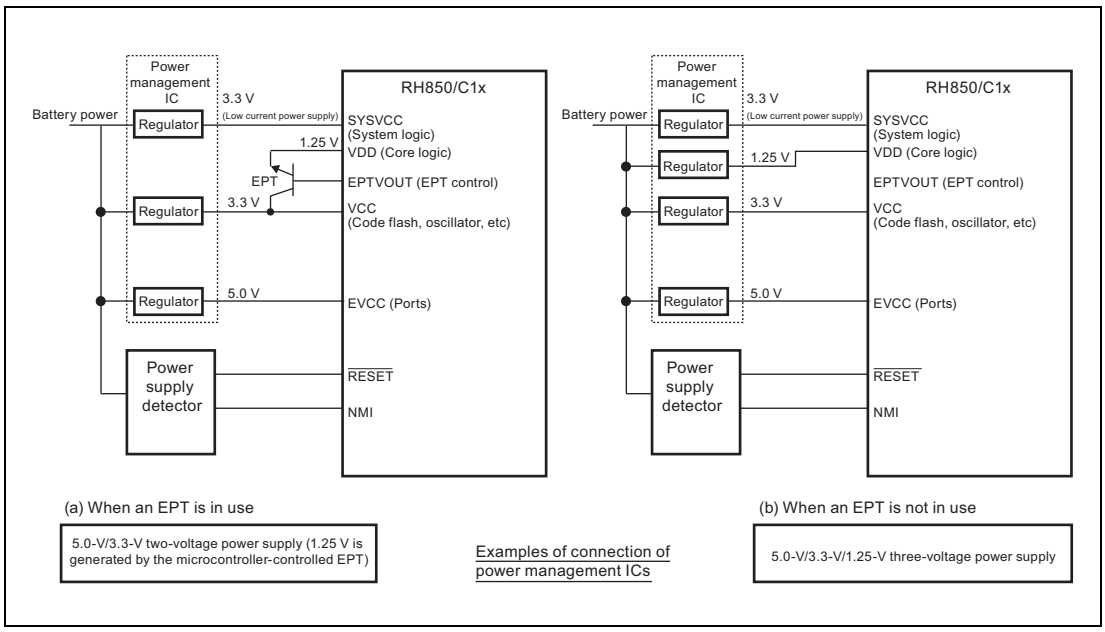


Figure 9.1 Power: Examples of Connection of Power Management ICs

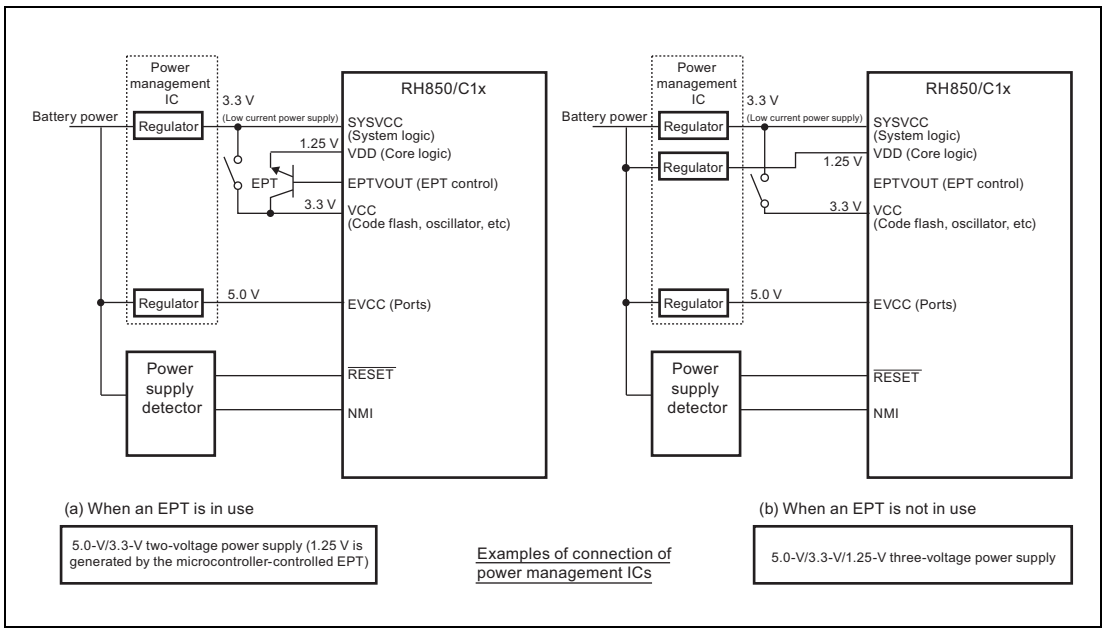


Figure 9.2 Power: Examples of Connection of Power Management ICs (RH850/C1x Switched Off)

9.3 Registers

9.3.1 List of Registers

EPT registers are listed in the following table.

Table 9.1 List of Registers

Module Name	Register Name	Symbol	Address	Access Protection
SYS	EPT control register	EPTCNT	FFF8 2C0C$_H$	PROT0PHCMD
SYS	Protect 0 command register	PROT0PHCMD	FFF8 3000$_H$	
SYS	Protect 0 status register	PROT0PS	FFF8 3004$_H$	

9.3.2 EPTCNT — EPT Control Register

This register enables or disables the EPT.

This register is protected by the PROT0PHCMD register.

This register can only be initialized by an external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 2C0C$_H$

Value after reset: 0000 0000$_H$

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

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

Table 9.2 EPTCNT register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The write value should be the value after reset.
0	EPTCNT	0: EPT is enabled. 1: EPT is disabled.

When the EPT is not in use, setting the above bit to 1 controls the EPT to stop it supplying power, which reduces power consumption.

9.3.3 PROT0PHCMD — Protect 0 Command Register

PROT0PHCMD is a protection command register to start the protection unlocking sequence necessary for access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 3000_H

Value after reset: 0000 0000_H

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

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

Table 9.3 PROT0PHCMD register contents

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should be the value after reset.
7 to 0	PCMD[7:0]	Write protection command register

This register is used in protecting against inadvertent access to write-protected registers, i.e. registers to which writing raises the possibility of serious effects on application systems, such as programs crashing and the like.

Access to write-protected registers is in the sequence described below (protection unlocking sequence).

Step 1:

Write $A5_H$ to this protection command register.

When the write data is $A5_H$, the protection error flag (PROTERR) in the protect 0 status register (PROT0PS) is cleared to 0. When the write data is not A5H, the protection error flag in PROT0PS is set to 1.

Step 2:

Write the value to the register to be protected. If the data is written to a register not to be protected, the error flag is set.

Step 3:

Write the inverse of the value written in step 2 to the register to be protected. If the write value is not the inverse value, the error flag is set. In addition, if the data is written to a register not to be protected, the error flag is set as well.

Step 4:

Re-execute Step 2 to complete the setting.

Start over again from Step 1 when the error flag has been set.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register*1 in the same module, writing to the protected register fails and the PROTERR bit in the PROT0PS register is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. With respect to the PROTOPHCMD register, this applies to registers allocated at FFF8 2410_H to FFF8 3004_H. For details on the register name, register symbol, and module name, see the lists of registers in **Table 8.1**, **Table 9.1**, and **Table 31.10**.

9.3.4 PROT0PS — Protect 0 Status Register

PROT0PS is a status register of the protection unlocking sequence. This register indicates whether an error occurs or not in access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.
Address: FFF8 3004_H
Value after reset: 0000 0000_H

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

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

Table 9.4 PROT0PS register contents

Bit Position	Bit Name	Function
31 to 1	Reserved	The write value should be the value after reset.
0	PROTERR	Protection Error Flag 0: Protection error has not occurred. 1: Protection error has occurred.

Operating conditions of the PROTERR bit

[Setting Condition]

Access to a write-protected register without executing the step of the protection unlocking sequence that involves the PROT0PHCMD register

[Clearing Condition]

Writing of $A5_H$ to the PROT0PHCMD register (step 1 in the protection unlocking sequence)

9.4 Procedure

9.4.1 Power-On Sequence

For power-on and -off specifications, see **Section 35, Electrical Characteristics**.

9.4.2 Settings when an EPT is not in Use

When an EPT is not in use, set the EPT control register (EPTCNT) to 1 (disabling the EPT). Disabling the EPT cuts the power for EPT control, which reduces power consumption.

The following figure is the flowchart for disabling the EPT. Perform the steps in this flowchart before starting the sequence for shifting the clock gear up. Regarding the sequence for shifting the clock gear up, see **Section 10.3.1, Operation When the Divide Function is Used**.

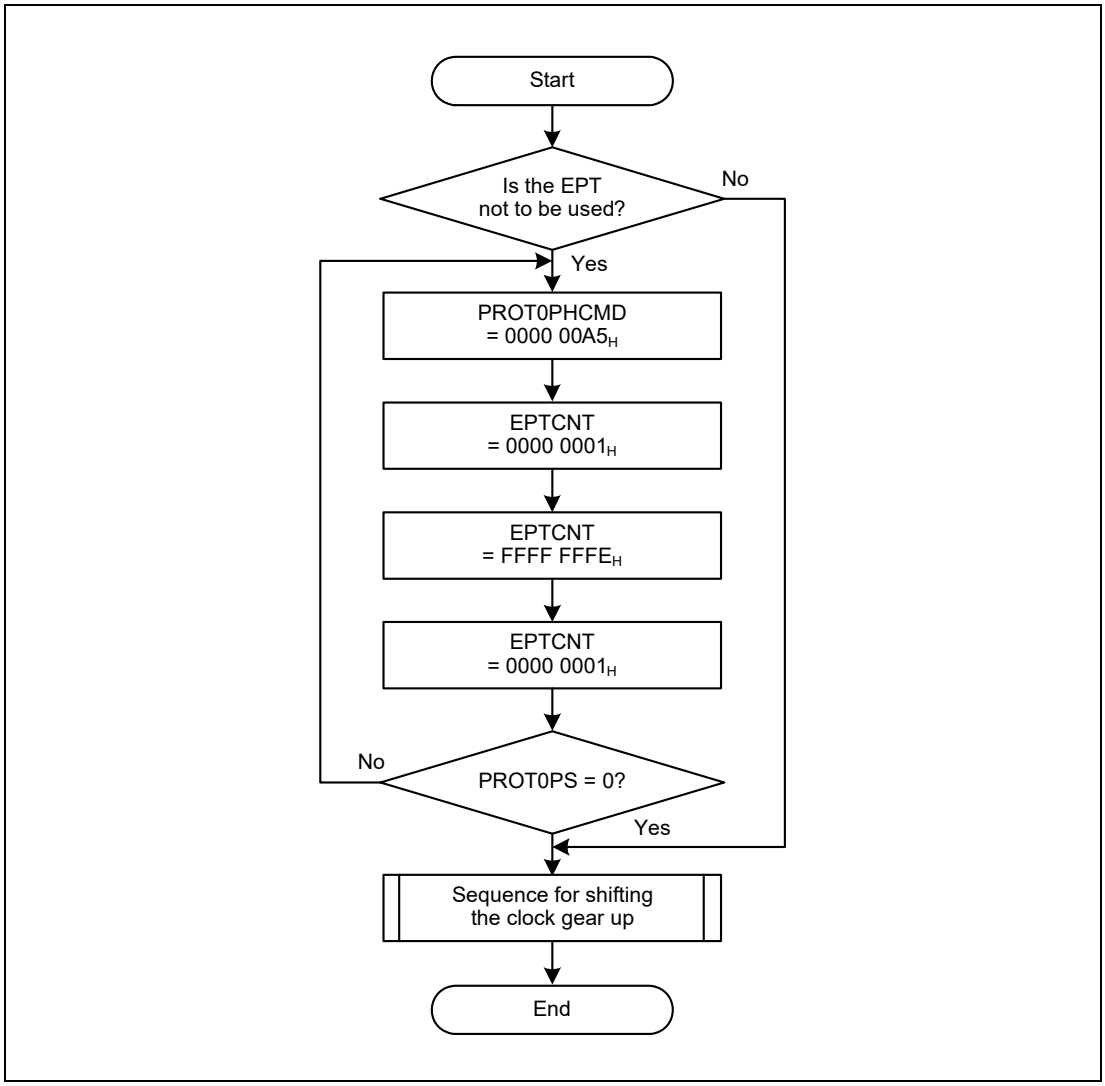


Figure 9.3 Flow for Disabling the EPT

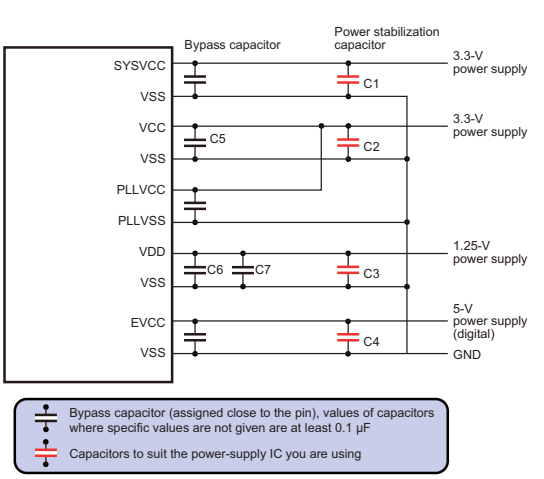
9.5 Notes

9.5.1 Usage with an EPT

If an EPT is to be used, external components such as the NPN transistor itself and a power-smoothing capacitor are required. Certain precautions must also be taken in connecting external components to the RH850/C1x device. For detailed information, make a separate request to a Renesas Electronics sale office.

9.5.2 Example of Connections between Power-Supply Pins and External Capacitors

The example shown below is for reference only. Take the voltage fluctuations seen under the actual conditions of your system into consideration in evaluating and assigning optimal capacitors in terms of bringing the power supply pins of the microcomputer into accord with the voltage levels in the product specifications.



The following are reference values of the power stabilization capacitors for the power-supply IC.
Determine the optimal capacitances to suit the power-supply IC you are using.

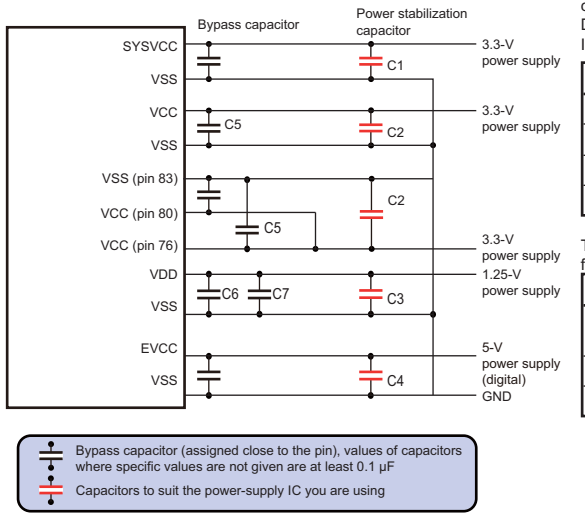
Essential Component	Reference Value
C1	≥ 10 µF
C2	≥ 10 µF
C3	≥ 10 µF
C4	= 1 to 10 µF

The following are reference values of the bypass capacitors for this device.

Essential Component	Reference Value
C5	≥ 0.22 µF or 0.1 µF × 2
C6*1	≥ 2.2 µF
C7*1	≥ 2.2 µF

Note 1: We recommend adding two 2.2-µF capacitors (C6 and C7) directly under the package.

Figure 9.4 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1H (BGA))



The following are reference values of the power stabilization capacitors for the power-supply IC.
Determine the optimal capacitances to suit the power-supply IC you are using.

Essential Component	Reference Value
C1	≥ 10 µF
C2	≥ 10 µF
C3	≥ 10 µF
C4	= 1 to 10 µF

The following are reference values of the bypass capacitors for this device.

Essential Component	Reference Value
C5	≥ 0.22 µF or 0.1 µF × 2
C6*1	≥ 2.2 µF
C7*1	≥ 2.2 µF

Note 1: We recommend adding a 2.2-µF capacitor (C6) for each VDD pin.

Figure 9.5 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1M (QFP))

Section 10 Clock Controller

10.1 Features of the RH850/C1x Clock Controller

- An oscillation circuit is included (Main OSC).
- Oscillation at 20 MHz is possible without using external capacitors (only with limited oscillators).
- On-chip PLLs are included and can be used to multiply clock frequencies.
- PLL0 is capable of operating as a spread spectrum clock generator (SSCG) for reducing radiated noise.
  The clock signal produced by PLL1 is not frequency modulated (i.e. is a clean clock signal) and so is suitable for use with timers and communications modules.
- A low-speed internal oscillator (LS IntOSC) is included.
- Interrupts can be generated when the Main OSC goes outside the frequency range specified by the clock monitor.
- Inrush currents after release from the reset state can be suppressed by having software increase the clock frequencies in a stepwise manner. The division ratios for the CPU clock and the peripheral clocks can be selected with register settings (1/4, 1/2, and 1/1).*1

Note 1. Correct operations of ADCC and RDC2 are not guaranteed if 1/4 or 1/2 is selected. Other peripheral circuits operate by using the clock frequency that was input, but their electrical characteristics are not guaranteed.
Therefore, only use the product after the sequence for shifting the clock gear up has reached the point where the division ratio for dividers 0A and 1A is 1/1 (no division).

10.1.1 Type of Clocks

Table 10.1 shows the list of clocks, **Table 10.2** shows the operation clocks of each functional module, and **Figure 10.1** shows the block diagram of clocks.

Table 10.1 List of Clocks

When operating at 20 MHz input frequency (Main OSC)

Clock Name	Symbol	Clock Frequency*1			Remarks
		Division Ratio of Divider 0A/ Divider 1A			
		1/1	1/2	1/4	
CPU clock	CLK_CPU	240 MHz	120 MHz	60 MHz	PLL0 (SSCG can be selected)
GRAM clock	CLK_GRAM	120 MHz	60 MHz	30 MHz	
High speed peripheral clock	CLK_HSB	80 MHz	40 MHz	20 MHz	
Low speed peripheral clock (peripheral clock)	CLK_LSB	40 MHz	20 MHz	10 MHz	
Unmodulated high speed peripheral clock	CLKC_HSB	80 MHz	40 MHz	20 MHz	PLL1 (SSCG cannot be selected)
Unmodulated low speed peripheral clock	CLKC_LSB	40 MHz	20 MHz	10 MHz	
Low speed internal clock	CLK_LIOSC	240 kHz			
WDTA counter clock	WDTACLKI	250 kHz			1/80 of the main OSC

Note 1. Only use the product after the sequence for shifting the clock gear up has reached the point where the division ratio for dividers 0A and 1A is 1/1 (no division).

Table 10.2 Clocks and Functional Modules

Clock Name	Functional Module Name
CPU clock	CPU1, CPU2
GRAM clock	GRAM
High speed peripheral clock	INTC, INTIF, DMAC, DTS, CSIH, RS-CSN (PCLK), EMU2 (PCLK)
Low speed peripheral clock (peripheral clock)	ECM, DCRA, PORT, data flash, flash controller, FLSCI3, EINT, RLIN2 (PCLK), ADCC (PCLK)
Unmodulated high speed peripheral clock	TAUD, TAUJ, TSG3, TAPA, TPBA, PIC1A, PIC2B, ENCA, EMU2
Unmodulated low speed peripheral clock	RS-CAN, RLIN2, SCI3, WDTA (PCLK), OSTM, RDC2, ADCC
Low speed internal clock	CLMA
WDTA counter clock	WDTA

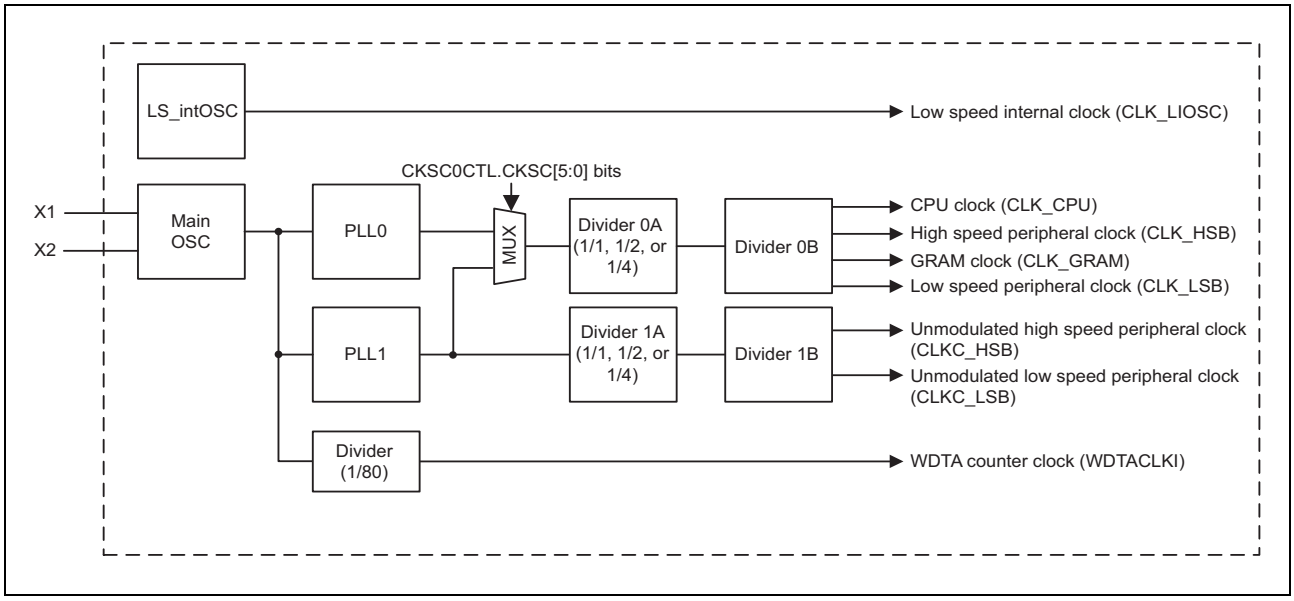


Figure 10.1 Clock Controller Block Diagram

10.1.2 External Input/Output Pins

Table 10.3 shows the pins related to the clock controller.

Table 10.3 Pins Related to the Clock Controller

Name	Pin Name	Input/Output	Function
Crystal input	X1	Input	Connected to a crystal oscillator
Crystal output	X2	Output	Connected to a crystal oscillator
PLL power supply*1	PLLVCC	Input	Power supply for the PLL multiply circuit
PLL ground*1	PLLVSS	Input	Power supply for the PLL multiply circuit

Note 1. PLLVCC and PLLVSS are included only in the C1H. In the C1M, PLLVCC and PLLVSS are integrated into VCC and VSS, respectively.

10.1.3 How to Connect a Crystal Oscillator

Figure 10.2 shows how to connect a crystal oscillator. When the oscillator recommended by the company is used (the detail information is separately provided), any external component such as a load capacitor and a dumping resistor is not necessarily required for oscillation in general. However, proper operation should be verified under actual conditions prior to use.

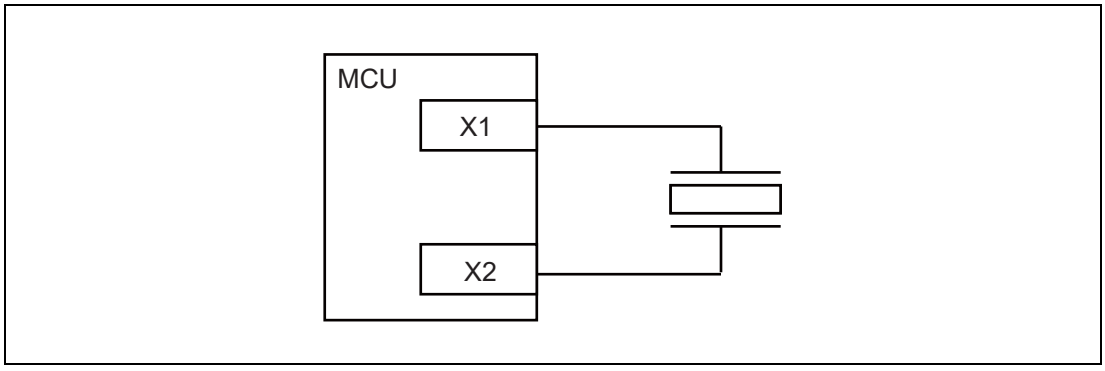


Figure 10.2 Connection Example of Crystal Oscillator

10.2 Register

10.2.1 List of Registers

Table 10.4 shows the list of registers.

Table 10.4 Registers Related to the Clock Controller

Module Name	Register Name	Symbol	Address	Access Protection
SYS	PLL0 status register	PLL0CLKS	FFF8 8004$_H$	
SYS	PLL0 control register 1	PLL0CLKC1	FFF8 8200$_H$	PROT1PHCMD
SYS	Clock 0 selection control register	CKSC0CTL	FFF8 9000$_H$	PROT1PHCMD
SYS	Clock 0 selection active register	CKSC0ACT	FFF8 9008$_H$	
SYS	Clock 0 division register	CLKD0DIV	FFF8 8800$_H$	PROT1PHCMD
SYS	Clock 0 division status register	CLKD0STAT	FFF8 8804$_H$	
SYS	Clock 1 selection control register	CKSC1CTL	FFF8 9040$_H$	PROT1PHCMD
SYS	Clock 1 selection active register	CKSC1ACT	FFF8 9048$_H$	
SYS	Protect 1 command register	PROT1PHCMD	FFF8 B000$_H$	
SYS	Protect 1 status register	PROT1PS	FFF8 B004$_H$	

10.2.2 PLL0CLKS — PLL0 Status Register

This register indicates the state of the PLL0 clock effective, active, or stable.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8004$_H$

Value after reset: 0000 0001$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK ACT	CLK STAB	CLKEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0*1	0*1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. Depends on the read timing after the CPU starts operation.

Table 10.5 PLL0CLKS Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0.
2	CLKACT	PLL0 Clock Source State 0: PLL0 clock source is inactive. 1: PLL0 clock source is active.
1	CLKSTAB	PLL0 Clock Stable State 0: PLL0 is unstable. 1: PLL0 is stable.
0	CLKEN	PLL0 Clock Operating State 0: PLL0 is stopped. 1: PLL0 is operating.

10.2.3 PLL0CLKC1 — PLL0 Control Register 1

This register controls the operation of the spread spectrum clock generator (SSCG) for PLL0.

This register can be set when PLL0 is running.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8200$_H$

Value after reset: 0000 0000$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSMODE1	—	SELMFREQ[4:0]					SELMPERCENT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 10.6 PLL0CLKC1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 10	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
9	SSMODE1	SSCG Control 0: Modulation disabled 1: Modulation enabled
8	—	Reserved This bit is always read as 0. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 3	SELMFREQ [4:0]	SSCG Cycle Setting The modulation cycle of SSCG can be selected by setting these bits. Selectable settings are as follows: 1 0 0 0 0 $_B$: 80.65 kHz 1 0 0 0 1 $_B$: 75.76 kHz 1 0 0 1 0 $_B$: 69.44 kHz 1 0 0 1 1 $_B$: 65.79 kHz 1 0 1 0 0 $_B$: 62.50 kHz 1 0 1 0 1 $_B$: 59.52 kHz 1 0 1 1 0 $_B$: 58.14 kHz 1 0 1 1 1 $_B$: 50.00 kHz 1 1 0 0 0 $_B$: 41.67 kHz 1 1 0 0 1 $_B$: 39.68 kHz 1 1 0 1 0 $_B$: 37.31 kHz 1 1 0 1 1 $_B$: 33.33 kHz 1 1 1 0 0 $_B$: 30.12 kHz 1 1 1 0 1 $_B$: 25.00 kHz 1 1 1 1 0 $_B$: 20.00 kHz When SSMODE1 is set to 1, do not set the values other than the above.

Table 10.6 PLL0CLKC1 Register Contents (2/2)

Bit Position	Bit Name	Function
2 to 0	SELMPERCENT [2:0]	SSCG Modulation Range Setting The modulation range of SSCG can be selected by setting these bits Selectable settings are as follows: $1\ 0\ 0_B$: –5.0% When SSMODE1 is set to 1, do not set the values other than the above.

When changing the modulation depth of PLL0, turn off the SSCG and then wait for at least 1.6 ms.

10.2.4 CKSC0CTL — Clock 0 Selection Control Register

This register is used for selection of a clock source for the frequency divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9000_H

Value after reset: 0000 0020_H

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

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

Table 10.7 CKSC0CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select a clock source for the frequency divider 0A. 100000_B: Selects the PLL1 clock. 100011_B: Selects the PLL0 clock. Setting other than the above is prohibited.

10.2.5 CKSC0ACT — Clock 0 Selection Active Register

This register indicates the state of a clock source for the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9008_H

Value after reset: 0000 0020_H

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

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

Table 10.8 CKSC0ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of a clock source for divider 0A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC0CTL register.

10.2.6 CLKD0DIV — Clock 0 Division Register

This register specifies a frequency division ratio of divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8800_H

Value after reset: 0000 0004_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0DIV[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 10.9 CLKD0DIV Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	CLKD0DIV[2:0]	These bits select a frequency division ratio of divider 0A. 0 0 1_B: No division 0 1 0_B: Divided by 2 1 0 0_B: Divided by 4 Setting other than the above is prohibited.

10.2.7 CLKD0STAT — Clock 0 Division Status Register

This register indicates a clock state of the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8804$_H$

Value after reset: 0000 0001$_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0 SYNC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.10 CLKD0STAT Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	CLKD0SYNC	Divider synchronization status 0: Divider 0A is switching to the division ratio specified in CLKD0DIV. 1: Divider 0A is running at the division ratio specified in CLKD0DIV.

10.2.8 CKSC1CTL — Clock 1 Selection Control Register

This register is used for selection of an output clock for the frequency divider 1A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9040_H

Value after reset: 0000 0021_H

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

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

Table 10.11 CKSC1CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select an output clock for the frequency divider 1A. 100001_B: Selects 1/4 clock of PLL1. 100010_B: Selects 1/2 clock of PLL1. 100100_B: Selects 1/1 clock of PLL1. Setting other than the above is prohibited.

10.2.9 CKSC1ACT — Clock 1 Selection Active Register

This register indicate a clock state of the frequency divider 1A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9048$_H$

Value after reset: 0000 0021$_H$

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

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

Table 10.12 CKSC1ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of an output clock for the frequency divider 1A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC1CTL register. 100001_B: 1/4 of PLL1 100010_B: 1/2 of PLL1 100100_B: 1/1 of PLL1

10.2.10 PROT1PHCMD Protect 1 Command Register

PROT1PHCMD is a protection command register to start the protection unlocking sequence necessary for access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 $B000_H$

Value after reset: 0000 0000_H

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

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

Table 10.13 PROT1PHCMD Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0.
7 to 0	PCMD[7:0]	Write protection command register

This register is used in protecting against inadvertent access to write-protected registers, i.e. registers to which writing raises the possibility of serious effects on application systems, such as programs crashing and the like.

Writing to the write-protected registers is only possible with the protection unlocking sequence described below.

Step 1. Write the fixed value (0000 $00A5_H$) to the PROT1PHCMD register.

Step 2. Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.

Step 3. Write the bitwise inverse of the setting value to the same register as in step 2. Write the inverse of the value after a reset to the reserved bits.

Step 4. Write the new setting to the same register as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the PROT1PS.PROTERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the PROT1PS.PROTERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows.

Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. With respect to the PROT1PHCMD register, this applies to registers allocated at FFF8 8004_H to FFF8 $B004_H$. For details on the register name, register symbol, and module name, see the lists of registers in **Table 8.1**, **Table 10.4**, **Table 27.118** to **Table 27.120**, and **Table 31.10**

10.2.11 PROT1PS — Protect 1 Status Register

PROT1PS is a status register of the protection unlocking sequence. This register indicates whether an error occurs or not in access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 B004$_H$

Value after reset: 0000 0000$_H$

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

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

Table 10.14 PROT1PS Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	PROTERR	Protection Error Flag 0: Protection error does not occur. 1: Protection error occurs.

Operating conditions of the PROTERR bit

[Setting Condition]

Access to a write-protected register without executing the step of the protection unlocking sequence that involves the PROT1PHCMD register.

[Clearing Condition]

Writing of 0000 00A5$_H$ to the PROT1PHCMD register (step 1 in the protection unlocking sequence).

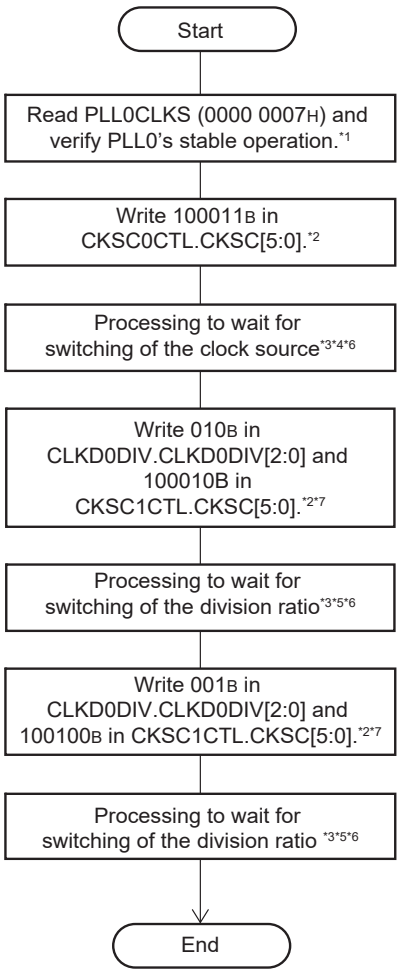
10.3 Function

10.3.1 Operation When the Divide Function is Used

Follow the procedure given below to switch the clock signals.

1. When a user program is running after release from the reset state, PLL0 and PLL1 will be oscillating and PLL1 will be in use as the internal operating clock. Also, the division ratio of divider 0A is set to 1/4.
   Read the PLL0CLKS register and verify that its value is 07_H, indicating that PLL0 operation is stable.
2. Write 23_H to the CKSC0CTL.CKSC[5:0] bits to select PLL0 as the clock source.
   Read CKSC0ACT and verify that the value of the CKSC0ACT.CLKACT[5:0] bits is 23_H.
3. Write 010_B to the CLKD0DIV.CLKD0DIV[2:0] bits to select 1/2 as the division ratio for the divider.
   Read CLKD0STAT and verify that the value of CLKD0SYNC is 1.
4. Write 001_B to the CLKD0DIV.CLKD0DIV[2:0] bits to select 1/1 as the division ratio for the divider.
   Read CLKD0STAT and verify that the value of CLKD0SYNC is 1.

Figure 10.3 shows an example of the procedure for switching the clocks (and shifting the clock gear up for both clocks). **Table 10.15** lists the clock frequencies in each step of the process.



Note 1. After release from the reset state, verify that PLL0 operation is stable before executing the sequence to shift the clock gear up. PLL0CLKS = 07_H indicates that PLL0 is stable.

Note 2. The CKSC0CTL and CLKD0DIV registers are protected by the PROT1PHCMD register. Execute the protection unlocking sequence (see the description of the protect 1 command register) when setting these registers.

Note 3. This processing is required because actual switching of the clock source or division ratio from the previous register settings takes time. The processing must produce periods of waiting of at least 105 cycles of the CPU clock.

Note 4. Verify that the value of the CKSC0ACT.CLKACT[5:0] bits is the same as that specified by the CKSC0CTL.CKSC[5:0] bits after switching the source clock for frequency divider 0A.
Similarly, verify that the value of the CKSC1ACT.CLKACT[5:0] bits is the same as that specified by the CKSC1CTL.CKSC[5:0] bits after switching the source clock for frequency divider 1A.

Note 5. Verify that the CLKD0STAT.CLKD0SYNC bit is 1 after changing the division ratio of divider 0A.

Note 6. Switching of clock sources or division ratios produces transient voltage fluctuations which last for a certain time, so measures such as processing to add more cycles of waiting may be required. Whether or not this is the case will depend on evaluation of the actual environment where the LSI chip is used.

Note 7. In the procedure described above, the division ratios of both PLL0 and PLL1 are set at the same time. When minimizing the transient voltage fluctuations during this procedure is crucial, separate the register settings and subsequent periods of waiting for PLL0 and PLL1 to achieve this (which also prolongs the procedure).

Figure 10.3 Example of Sequence for Shifting the Clock Gear Up

Table 10.15 Frequencies in the Process of Shifting the Clock Gear Up

	CLK_CPU	CLK_HSB	CLK_LSB	CLKC_HSB	CLKC_LSB
Before Step 1	60	20	10	20	10
Between Step 1 and Step 2	60	20	10	20	10
Between Step 2 and Step 3	120	40	20	40	20
After Step 3	240	80	40	80	40

10.4 Notes

Avoid using commands that employ the computing unit, such as FPU operations, while the division ratio is being changed. Increased current fluctuation may make the operation unstable.

10.4.1 Board Design Notes

As shown in **Figure 10.4**, do not cross any other signal lines over the signal lines to the X1 and X2 pins. Induction may inhibit proper oscillation.

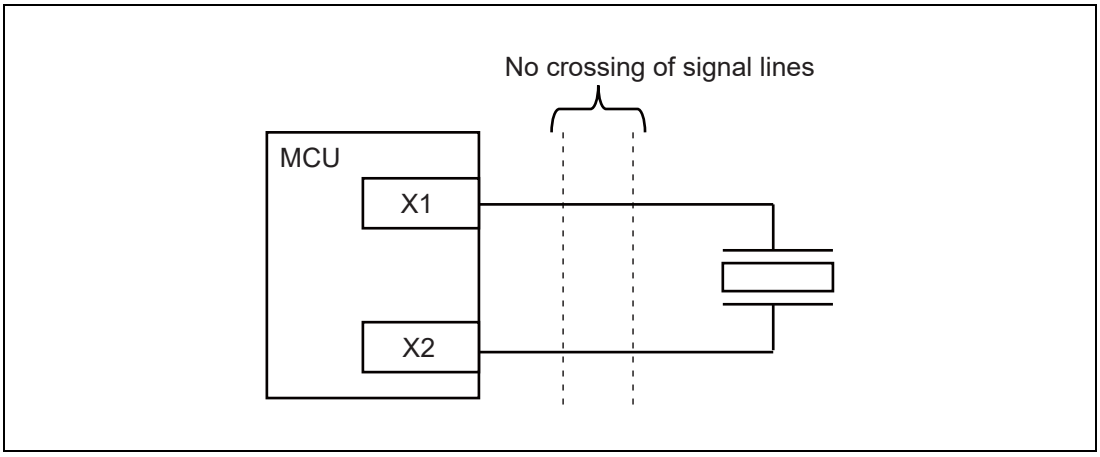


Figure 10.4 Board Design Notes

Section 11 Clocked Serial Interface H (CSIH)

This section contains a generic description of the Clocked Serial Interface H (CSIH).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of CSIH.

11.1 Features of RH850/C1x CSIH

11.1.1 Units

This LSI has the following number of CSIH units.

CSIH Each unit has one channel interface

Table 11.1 Units

Product	RH850/C1H 252 pins	RH850/C1M 144 pins
Number of Units	2	2
Name	CSIHn (n = 0, 1)	CSIHn (n = 0, 1)

Table 11.2 Index

Index	Meaning
n	Throughout this section, the individual CSIH units are identified by the index “n” (n = 0, 1); for example, CSIHnCTL0 is the CSIHn control register 0.
x	CSIHn has up to 4 chip select signals. Throughout this section, the individual chip select signals are identified by the index “x”: that is, CSx denotes a non-specified chip select signal.
y	A variable used for explanation is identified by the index “y”; for example, CSIHnBRSy is a non-specified baud rate setting register of CSIHn.

The number of chip select signals for each channel of CSIH is given in the following table:

Table 11.3 Number of Chip Select Signals

Unit Name	Number of Chip Select Signals	
	252 pins	144 pins
CSIH0	CSx (x = 0 to 3)	CSx (x = 0 to 3)
CSIH1	CSx (x = 0 to 3)	CSx (x = 0 to 3)

11.1.2 Register Base Addresses

CSIH base addresses are listed in the following table.

CSIH register addresses are given as offsets from the base addresses in general.

Table 11.4 Register Base Address

Base Address Name	Base Address
<CSIH0_base>	$FFD8\ 0000_H$
<CSIH1_base>	$FFD8\ 2000_H$

11.1.3 Clock Supply

The CSIH clock supply is shown in the following table.

Table 11.5 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
CSIHn	PCLK	CLK_HSB (high-speed peripheral clock)

11.1.4 Interrupt Requests

CSIH interrupt requests are listed in the following table.

Table 11.6 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMA Trigger Number	DTS Trigger Number
CSIH0				
INTCSIHTIC	Communication status interrupt	215	74	100
INTCSIHTIR	Receive status interrupt	216	75	101
INTCSIHTIRE	Communication error interrupt	217	—	—
INTCSIHTIJC	Job completion interrupt	218	76	102
CSIH1				
INTCSIHTIC	Communication status interrupt	219	77	103
INTCSIHTIR	Receive status interrupt	220	78	104
INTCSIHTIRE	Communication error interrupt	221	—	—
INTCSIHTIJC	Job completion interrupt	222	79	105

11.1.5 Reset Sources

The CSIHn and the CSIHn registers are initialized by the following reset signal.

Table 11.7 Reset Sources

Unit Name	Reset Sources
CSIHn	All reset sources

11.1.6 External Input/Output Signals

External input/output signals of CSIH are listed in the following table.

Table 11.8 External Input/Output Signals

CSIHn signal	Function	Alternative Port Pin Signal
CSIH0		
CSIHTSCK	Serial clock input signal	CSIH0SC
CSIHTSI	Serial data input signals	CSIH0SI
$\overline{\text{CSIHTSSI}}$	Slave select input signals	$\overline{\text{CSIH0SSI}}$
CSIHTRYI	Ready / Busy input signal	CSIH0RYI
CSIHTSO	Serial data output signals	CSIH0SO
CSIHTRYO	Ready / Busy output signal	CSIH0RYO
CSIHTCSS[3:0]	Chip select signals	CSIH0CSS[3:0]
CSIH1		
CSIHTSCK	Serial clock input signal	CSIH1SC
CSIHTSI	Serial data input signals	CSIH1SI
$\overline{\text{CSIHTSSI}}$	Slave select input signals	$\overline{\text{CSIH1SSI}}$
CSIHTRYI	Ready / Busy input signal	CSIH1RYI
CSIHTSO	Serial data output signals	CSIH1SO
CSIHTRYO	Ready / Busy output signal	CSIH1RYO
CSIHTCSS[3:0]	Chip select signals	CSIH1CSS[3:0]

11.1.7 Data Consistency Check

The following table lists data consistency checking for the port on which CSIHTSO pin functions are multiplexed. See **Section 11.4.18, Error Detection** for details on data consistency checking.

Table 11.9 Data Consistency Checking and Port Pins

CSIHn signal	Port Pin Name	Alternative Function
CSIH0		
CSIHTSO	P4_8	ALT_OUT6
CSIH1		
CSIHTSO	P4_1	ALT_OUT7

11.2 Overview

11.2.1 Functional Overview

- Three-wire serial synchronous data transfer
- Master mode and slave mode selectable
- Multiple slaves configuration plus RCB (Recessive Configuration for Broadcasting) thanks to 4 configurable chip select output signals
- Slave select input signal ($\overline{\text{CSIHTSSI}}$) included
- Built-in baud rate generator
- Transfer clock frequency adjustable in master mode; determined by input clock in slave mode
- Maximum transfer clock frequency:
  - in master mode: PCLK/8
  - in slave mode: PCLK/16
- Phase of clock and data selectable
- Data transfer with MSB or LSB first selectable
- Transfer data length selectable from 2 to 16 bits in 1-bit units
- EDL (Extended Data Length) function for transferring data with more than 16 bits included
- Three selectable transfer modes:
  - transmit-only mode
  - receive-only mode
  - transmit/receive mode
- Handshake function included
- Error detection (data consistency check, parity, time-out, overflow, and overrun) included
- Support of job concept
- 128 words I/O buffer memory
- Selectable direct access mode and memory mode (FIFO, dual buffer, and transmit-only buffer)
- Four different interrupt request signals (INTCSIHTIC, INTCSIHTIR, INTCSIHTIRE, INTCSIHTIJC)
- LBM (Loop Back Mode) function for self test included
- CPU-controlled high-priority communication function
- Enforced chip select idle setting
- RCB (Recessive Configuration for Broadcasting) bit is included

11.2.2 Functional Overview Description

The Clocked Serial Interface uses three signals for communication:

- Transmission clock CSIHTSCK (output in master mode, input in slave mode)
- Data output signal CSIHTSO
- Data input signal CSIHTSI

Additional signals are available for external control and monitoring.

- $\overline{\text{CSIHTSSI}}$: Slave select input signal
- CSIHTRYO: Ready/busy output signal (handshake signal)
- CSIHTRYI: Ready/busy input signal (handshake signal)
- CSIHTCSS[3:0]: Chip select signals

Data transmission is bit-wise and serial, and synchronous to the transmission clock.

The following table shows the most important registers for setting up the CSIH.

Table 11.10 Main Registers of CSIH

Register	Function
CSIHnCTL0	Enables/disables transmission clock, and permits/ prohibits data transmission and data reception. Defines end-of-job behavior and enables/disables buffering (bypass of the buffer).
CSIHnCTL1	Controls options like interrupt timing, extended data length, job feature, data consistency check, loop-back mode, handshake, etc.
CSIHnCTL2	Selects master or slave mode, and the transfer clock frequency of the on-chip baud rate generator (BRG) in master mode.
CSIHnBRSy	Registers to configure the transfer clock frequency for each chip select signal
CSIHnMCTL0	Selects memory mode and specifies the time-out value
CSIHnMCTL1	Controls the memory in FIFO mode
CSIHnMCTL2	Controls the memory in dual buffer mode or transmit-only buffer mode
CSIHnCFGx	Registers to configure the communication protocol for each chip select signal

11.2.3 Block Diagram

The block diagram shows the main components of the CSIH.

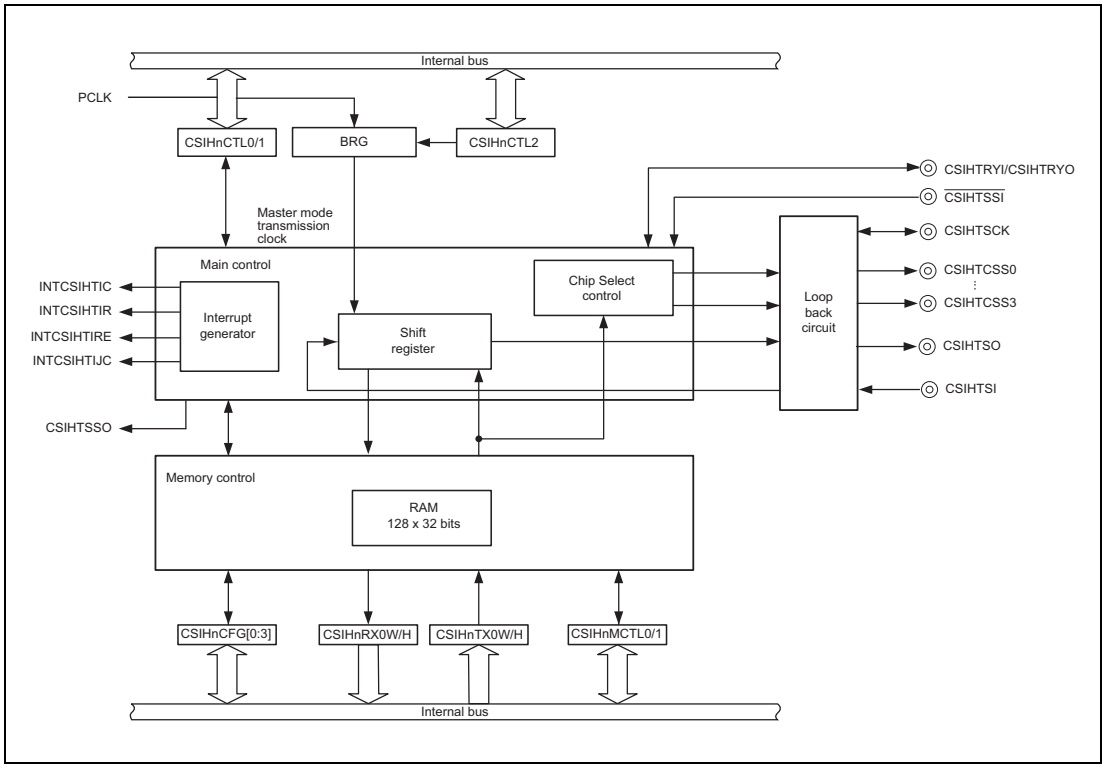


Figure 11.1 CSIH Block Diagram

In master mode, the transmission clock CSIHTSCK is generated by the built-in baud rate generator (BRG). In slave mode, the transmission clock is provided from an external source.

The built-in memory can be configured as FIFO, dual buffer (separate transmit and receive buffers), or transmit-only buffer. It can also be bypassed for data transmission and reception without buffering.

The loop back circuit disconnects the CSIH completely from the ports and supports internal self test.

NOTE

This section describes the following modes:

- The “operating mode” separates between master and slave mode. In this context, only a master can control and communicate with several slaves (for details see **Section 11.4.7, Operating Modes (Master/Slave)**).
- The “job mode” is related to the AUTOSAR job concept (for details see **Section 11.4.9.3, Job Concept**).
- The “memory mode” takes the various configurations of the associated buffer memory into account (for details see **Section 11.4.12, CSIH Buffer Memory**).
- The “data transfer mode” specifies the kind of the communication – transmit-only, receive only, or transmit/receive (for details see **Section 11.4.13, Data Transfer Modes**).

11.3 Registers

11.3.1 List of Registers

CSIH registers are listed in the following table.

For information on <CSIHn_base>, see **Section 11.1.2, Register Base Addresses**.

Table 11.11 List of Registers

Module Name	Register Name	Symbol	Address
CSIHn	CSIHn control register 0	CSIHnCTL0	<CSIHn_base> + 0000_H
CSIHn	CSIHn control register 1	CSIHnCTL1	<CSIHn_base> + 0010_H
CSIHn	CSIHn control register 2	CSIHnCTL2	<CSIHn_base> + 0014_H
CSIHn	CSIHn status register 0	CSIHnSTR0	<CSIHn_base> + 0004_H
CSIHn	CSIHn status clear register 0	CSIHnSTCR0	<CSIHn_base> + 0008_H
CSIHn	CSIHn memory control register 0	CSIHnMCTL0	<CSIHn_base> + 1040_H
CSIHn	CSIHn memory control register 1	CSIHnMCTL1	<CSIHn_base> + 1000_H
CSIHn	CSIHn memory control register 2	CSIHnMCTL2	<CSIHn_base> + 1004_H
CSIHn	CSIHn memory read/write pointer register 0	CSIHnMRWP0	<CSIHn_base> + 1018_H
CSIHn	CSIHn configuration register 0	CSIHnCFG0	<CSIHn_base> + 1044_H
CSIHn	CSIHn configuration register 1	CSIHnCFG1	<CSIHn_base> + 1048_H
CSIHn	CSIHn configuration register 2	CSIHnCFG2	<CSIHn_base> + $104C_H$
CSIHn	CSIHn configuration register 3	CSIHnCFG3	<CSIHn_base> + 1050_H
CSIHn	CSIHn transmit data register 0 for word access	CSIHnTX0W	<CSIHn_base> + 1008_H
CSIHn	CSIHn transmit data register 0 for half word access	CSIHnTX0H	<CSIHn_base> + $100C_H$
CSIHn	CSIHn receive data register 0 for word access	CSIHnRX0W	<CSIHn_base> + 1010_H
CSIHn	CSIHn receive data register 0 for half word access	CSIHnRX0H	<CSIHn_base> + 1014_H
CSIHn	CSIHn baud rate setting register 0	CSIHnBRS0	<CSIHn_base> + 1068_H
CSIHn	CSIHn baud rate setting register 1	CSIHnBRS1	<CSIHn_base> + $106C_H$
CSIHn	CSIHn baud rate setting register 2	CSIHnBRS2	<CSIHn_base> + 1070_H
CSIHn	CSIHn baud rate setting register 3	CSIHnBRS3	<CSIHn_base> + 1074_H

11.3.2 CSIHnCTL0 — CSIHn Control Register 0

This register controls the operation clock and enables/disables transmission/reception and the memory part for transmission and/or reception. It forces the stop of communication at the end of the current job.

Access: This register can be read/written in 8-bit or 1-bit units.

Address: <CSIHn_base> + 0000_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CSIHnPWR	CSIHnTXE	CSIHnRXE	—	—	—	CSIHnJOBE	CSIHnMBS
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Table 11.12 CSIHnCTL0 Register Contents

Bit Position	Bit Name	Function
7	CSIHnPWR	Controls the operation clock. 0: Stops operation clock. 1: Provides operation clock. Clearing CSIHnPWR to 0 resets the internal circuits, stops operation, and sets CSIH to standby state. No clock is provided to internal circuits. If CSIHnPWR is cleared during communication, ongoing communication is immediately aborted. In this case, communication setting must be started over.
6	CSIHnTXE	Permits or prohibits transmission. 0: Prohibits transmission. 1: Permits transmission.
5	CSIHnRXE	Permits or prohibits reception. 0: Prohibits reception. 1: Permits reception.
4 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	CSIHnJOBE	Stops communication at the end of the current job (Communication ends if data is written to the transmission buffer when CSIHnTX0W.CSIHnEOJ = 1 (job completion)): 0: Communication stop is not requested. 1: Stops communication. This bit can be used to abort an ongoing job. This bit is cleared to 0 automatically. If this bit is set to 1, the read value is always 0. In FIFO mode, the pointer must be cleared by setting CSIHnSTCR0.CSIHnPCT = 1 before the next communication is started
0	CSIHnMBS	Bypasses the memory for transmission and/or reception data. 0: Memory mode CSIH memory is used for transmission and/or reception data. 1: Direct access mode CSIH memory is bypassed.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.3 CSIHnCTL1 — CSIHn Control Register 1

This register specifies the interrupt timing and the interrupt delay mode. It enables/disables extended data length control, data consistency check, loop-back mode, handshake functionality, and job mode. It selects the active output level of each chip select signal and the behavior of the chip select signals after the transfer of the final data.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 0010_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIHn PHE	CSIHn CKR	CSIHn SLIT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSIHn CSL3	CSIHn CSL2	CSIHn CSL1	CSIHn CSL0	CSIHn EDLE	CSIHn JE	CSIHn DCS	CSIHn CSRI	CSIHn LBM	CSIHn SIT	CSIHn HSE	CSIHn SSE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.13 CSIHnCTL1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 19	Reserved	When read, the value after reset is read. When writing, write the value after reset.
18	CSIHnPHE	Sets the CPU-controlled priority-based communication function. 0: The CPU-controlled high-priority communication function is disabled. 1: The CPU-controlled high-priority communication function is enabled. To enable the CPU-controlled high-priority communication function, set this bit to 1 and set CSIHnJE = 1. This bit can only be set in transmit-only buffer mode.
17	CSIHnCKR	CSIHTSCK clock inversion function 0: The default level of CSIHTSCK is high 1: The default level of CSIHTSCK is low For details, see **Section 11.3.11, CSIHnCFGx — CSIHn Configuration Register x**.
16	CSIHnSLIT	Selects the timing of interrupt INTCSIHTIC. 0: Normal interrupt timing (interrupt is generated after the transfer) 1: As soon as the contents of the CSIHnTX0W/H register are transferred to the shift register, an interrupt is generated (this function is activated only in direct access memory mode/transmit-only buffer mode). For details, see **Section 11.4.3, INTCSIHTIC (Communication Status Interrupt)**.
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11 to 8	CSIHnCSLx	Selects the active output level of chip select signal x (CSIHTCSSx) (x = 0 to 3). 0: Chip select is active low. 1: Chip select is active high. For details, see **Section 11.4.9, Chip Selection (CS) Features**.
7	CSIHnEDLE	Enables/disables extended data length (EDL) mode. 0: Disables extended data length mode. 1: Enables extended data length mode. For details, see **Section 11.4.14.2, Data Length Greater than 16 Bits**.

Table 11.13 CSIHnCTL1 Register Contents (2/2)

Bit Position	Bit Name	Function
6	CSIHnJE	Enables/disables job mode. 0: Disables job mode. 1: Enables job mode. For details, see **Section 11.4.9.3, Job Concept**. The CSIHnCTL0.CSIHnJOBE, CSIHnTX0W.CSIHnEOJ, and CSIHnTX0W.CSIHnCIRE bits are enabled only when CSIHnJE = 1. Setting this bit in slave mode is prohibited. In addition, to enable the CPU-controlled high-priority communication function, set CSIHnPHE = 1 and this bit to 1.
5	CSIHnDCS	Enables/disables data consistency check. 0: Disables data consistency check. 1: Enables data consistency check. For details, see **Section 11.4.18.1, Data Consistency Check**.
4	CSIHnCSRI	Defines chip select signal behavior after last data transfer. 0: Chip select signal holds the active level. 1: Chip select signal returns to the inactive level. The last data is determined at the interrupt timing in direct access mode or FIFO mode. When CSIHnCTL1.CSIHnSLIT = 1, the last data is determined in direct access mode.
3	CSIHnLBM	Controls loop-back mode (LBM). 0: Deactivates loop-back mode. 1: Activates loop-back mode. For details, see **Section 11.4.19, Loop-back Mode**.
2	CSIHnSIT	Selects interrupt delay mode. 0: No delay is generated. 1: Half clock delay is generated for all interrupts. This bit is only valid in master mode. In slave mode, no delay is generated. For details, see **Section 11.4.2, Interrupt Delay**.
1	CSIHnHSE	Enables/disables the handshake function. 0: Disables the handshake function. 1: Enables the handshake function. For details refer to **Section 11.4.17, Handshake Function**.
0	CSIHnSSE	Enables/disables the slave select function. 0: Input signal $\overline{\text{CSIHTSSI}}$ is disabled. 1: Input signal $\overline{\text{CSIHTSSI}}$ is enabled. If the slave select function is not used, this bit must be set to 0 (see also **Section 11.4.8, Master/Slave Connections**).

Details about CSIHnCTL1.CSIHnSSE are shown in the following tables.

Table 11.14 Operation of the Slave Select Function During Reception

CSIHnCTL0. CSIHnRXE	CSIHnCTL1. CSIHnSSE	$\overline{\text{CSIHTSSI}}$	Receive operation
0	—	—	Reception is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

Table 11.15 Operation of the Slave Select Function During Transmission

CSIHnCTL0. CSIHnTXE	CSIHnCTL1. CSIHnSSE	$\overline{\text{CSIHTSSI}}$	Transmit operation
0	—	—	Transmission is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.4 CSIHnCTL2 — CSIHn Control Register 2

This register selects operating mode and the basic clock value, and specifies the transfer clock frequency.

For details see **Section 11.4.11, Transmission Clock Selection**.

Access: This register can be read/written in 16-bit units.

Address: <CSIHn_base> + 0014_H

Value after reset: $E000_H$

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

Table 11.16 CSIHnCTL2 Register Contents

Bit Position	Bit Name	Function
15 to 13	CSIHnPRS[2:0]	These bits select the operation mode and the reference clock value.
12 to 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CSIHnPRS2	CSIHnPRS1	CSIHnPRS0	Selection of Reference Clock (PRSOUT)
0	0	0	PCLK (Master mode)
0	0	1	PCLK/2 (Master mode)
0	1	0	PCLK/4 (Master mode)
0	1	1	PCLK/8 (Master mode)
1	0	0	PCLK/16 (Master mode)
1	0	1	PCLK/32 (Master mode)
1	1	0	PCLK/64 (Master mode)
1	1	1	External clock via CSIHTSCK(in) (Slave mode)

In master mode, the following bits are used to set the transfer clock frequency:

CSIHnCTL2.CSIHnPRS[2:0],
CSIHnCFGx.CSIHnBRSS[1:0],
CSIHnBRSy.CSIHnBRS[11:0]

In addition, any of the four different transfer clock frequency settings that are specified by the CSIHnBRSy.CSIHnBRS[11:0] bits is selected according to the chip select signal. To select the transfer clock frequency setting for each chip select signal, use the CSIHnCFGx.CSIHnBRSS[1:0] bits.

The following table shows the relationship between CSIHnCFGx.CSIHnBRSS[1:0] and CSIHnBRSy.CSIHnBRS[11:0].

CSIHnCFGx.CSIHnBRSS[1:0]	Baud rate setting bit to be selected
00	CSIHnBRS0.CSIHnBRS[11:0]
01	CSIHnBRS1.CSIHnBRS[11:0]
10	CSIHnBRS2.CSIHnBRS[11:0]
11	CSIHnBRS3.CSIHnBRS[11:0]

The following table shows the relationship between the transfer clock frequency and the transfer clock frequency setting (CSIHnBRSy[11:0]) selected by the CSIHnBRSS[1:0] bits when the bit value of the CSIHnPRS[2:0] bits is α.

CSIHnBRSy[11:0]	Transfer Clock Frequency
0	BRG stopped
1	PCLK/(2^{α} × 1 × 2)
2	PCLK/(2^{α} × 2 × 2)
3	PCLK/(2^{α} × 3 × 2)
4	PCLK/(2^{α} × 4 × 2)
:	:
4095	PCLK/(2^{α} × 4095 × 2)

When a time-out error is used in slave mode, the clock selected by this setting is used. In slave mode, the CSIHnPRS[2:0] bits are set to 111_B. In this case, the prescaler has the same setting as when the CSIHnPRS[2:0] bits are set to 000_B. If you are using a time-out error, set the CSIHnBRSy.CSIHnBRS[11:0] bits to a value other than 000_H.

CAUTIONS

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.5 CSIHnSTR0 — CSIHn Status Register 0

This register indicates the status of CSIH.

Access: This register can be read in 32-bit units.

Address: <CSIHn_base> + 0004_H

Value after reset: $0000\ 0010_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHnSRP[7:0]								CSIHnSPF[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn TMOE	CSIHn OFE	—	—	—	—	—	CSIHn HPST	CSIHn TSF	—	CSIHn FLF	CSIHn EMF	CSIHn DCE	—	CSIHn PE	CSIHn OVE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.17 CSIHnSTR0 Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 24	CSIHnSRP[7:0]	Indicates the number of received data packets in FIFO mode. **CSIHnSRP[7:0]** / **Description** 00_H : 80_H — Number of received data packets (0 to 128) Other than the above — Undefined These bits are cleared by CSIHnSTCR0.CSIHnPCT. In direct access mode, dual buffer mode, or transmit-only buffer mode, this value is fixed to 00_H. In direct access mode, this bit is fixed to 0 because there is no pointer. In buffer mode, this bit is fixed to 0 because the number of data packets is managed by CSIHnMCTL2.CSIHnND[7:0].
23 to 16	CSIHnSPF[7:0]	Indicates the number of unsent data in FIFO mode. (The number of data written by the CPU is the number of sent data.) **CSIHnSPF[7:0]** / **Description** 00_H : 80_H — Number of unsent data packets (0 to 128) Other than the above — Undefined These bits are cleared by CSIHnSTCR0.CSIHnPCT. In direct access mode, dual buffer mode, or transmit-only buffer mode, this value is fixed to 00_H. In direct access mode, this bit is fixed to 0 because there is no pointer. In buffer mode, this bit is fixed to 0 because the number of data packets is managed by CSIHnMCTL2.CSIHnND[7:0].

Table 11.17 CSIHnSTR0 Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>15</td><td>CSIHnTMOE</td><td>Time-out error flag in FIFO mode
Indicates whether a time-out error was detected in FIFO mode.
0: No time out error is detected.
1: A time out error is detected.
For details, see Section 11.4.18.3, Time-out Error.
This bit is cleared by CSIHnSTCR0.CSIHnTMOEC.
When this bit is set to 1 by the detection of time-out error and cleared to 0 by CSIHnSTCR0.CSIHnTMOEC simultaneously, setting to 1 is prioritized.
This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.</td></tr>
<tr><td>14</td><td>CSIHnOFE</td><td>Overflow error flag in FIFO mode
Indicates whether an overflow error was detected in FIFO mode.
0: No overflow error is detected.
1: An overflow error is detected.
For details, see Section 11.4.18.4, Overflow Error.
This bit is cleared by CSIHnSTCR0.CSIHnOFEC.
When this bit is set to 1 by the detection of overflow error and cleared to 0 by CSIHnSTCR0.CSIHnOFEC simultaneously, setting to 1 is prioritized.
This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.</td></tr>
<tr><td>13 to 9</td><td>Reserved</td><td>When read, the value after reset is read.</td></tr>
<tr><td>8</td><td>CSIHnHPST</td><td>Communication priority indication flag
0: Indicates low-priority communication is in progress.
1: Indicates high-priority communication is in progress.
This bit always reads 0 if CPU-controlled high-priority communication is disabled (CSIHnCTL1.CSIHnPHE = 0).</td></tr>
<tr><td>7</td><td>CSIHnTSF</td><td>Transfer status flag
0: Idle state
1: Communication is in progress or being prepared.

The timing to set or clear this bit is as follows:
<table><tr><th rowspan="2">Master mode</th><th colspan="2">Timing to Set</th><th rowspan="2">Timing to Clear</th></tr><tr><th>Direct access mode, FIFO mode</th><th>Dual Buffer mode, transmit-only mode</th></tr><tr><td>Transmit-only mode</td><td rowspan="3">Data is written to a transmit register (CSIHnTX0W/ CSIHnTX0H)</td><td rowspan="3">Bit CSIHnMCTL2.CSIHnBTST is set</td><td rowspan="3">Within a half clock of the last serial clock edge</td></tr><tr><td>Transmit/receive mode</td></tr><tr><td>Receive-only mode</td></tr></table>
<table><tr><th rowspan="2">Slave mode</th><th colspan="2">Timing to Set</th><th rowspan="2">Timing to Clear</th></tr><tr><th>Direct access mode, FIFO mode</th><th>Dual Buffer mode, transmit-only mode</th></tr><tr><td>Transmit-only mode</td><td rowspan="2">Data is written to a transmit register (CSIHnTX0W/ CSIHnTX0H)</td><td rowspan="3">Bit CSIHnMCTL2.CSIHnBTST is set</td><td rowspan="3">Within a half clock of the last serial clock edge</td></tr><tr><td>Transmit/receive mode</td></tr><tr><td>Receive-only mode</td><td>Input timing of CSIHTSCK</td></tr></table></td></tr>
<tr><td>6</td><td>Reserved</td><td>When read, the value after reset is read.</td></tr>
<tr><td>5</td><td>CSIHnFLF</td><td>A flag indicating that the buffer is full in FIFO mode
0: FIFO buffer is not full.
1: FIFO buffer is full.
This bit is cleared by CSIHnSTCR0.CSIHnPCT.
The FIFO buffer might be filled with unsent data or received data.</td></tr>
</table>

Table 11.17 CSIHnSTR0 Register Contents (3/3)

Bit Position	Bit Name	Function
4	CSIHnEMF	A flag indicating that the buffer is empty in FIFO mode 0: FIFO buffer is not empty. 1: FIFO buffer is empty. This bit is set to 1 by CSIHnSTCR0.CSIHnPCT. This bit is set to 1 when value of "CSIHnSTR0.CSIHnSRP[7:0] + CSIHnSTR0.CSIHnSPF[7:0]" matches to 00_H. The FIFO buffer might be filled with unsent data or received data.
3	CSIHnDCE	Data consistency check error flag 0: No data consistency error is detected. 1: Data consistency error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnDCEC. When this bit is set to 1 by the detection of data consistency check error and cleared to 0 by CSIHnSTCR0.CSIHnDCEC simultaneously, setting to 1 is prioritized. This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.
2	Reserved	When read, the value after reset is read.
1	CSIHnPE	Parity error flag 0: No parity error is detected. 1: Parity error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnPEC. When this bit is set to 1 by the detection of parity error and cleared to 0 by CSIHnSTCR0.CSIHnPEC simultaneously, setting to 1 is prioritized. Write access to this bit is enabled when CSIHnCTL0.CSIHnPWR = 0. This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.
0	CSIHnOVE	Overrun error flag (Fixed to 0 in dual buffer mode) 0: No overrun error is detected. 1: Overrun error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnOVEC. When this bit is set to 1 by overrun error and cleared to 0 by CSIHnSTCR0.CSIHnOVEC simultaneously, setting to 1 is prioritized. This bit is Initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.

Table 11.18 Behavior in Memory Mode

Bit name	Bit position	Mode			
		Direct access	FIFO	Transmit-Only Buffer	Dual buffer
CSIHnSRP[7:0]	31 to 24	Fixed to 0	Number of received words	Fixed to 0	Fixed to 0
CSIHnSPF[7:0]	23 to 16	Fixed to 0	Number of unsent data	Fixed to 0	Fixed to 0
CSIHnTMOE	15	Fixed to 0	0: No error is detected. 1: An error is detected.	Fixed to 0	Fixed to 0
CSIHnOFE	14	Fixed to 0	0: No error is detected. 1: An error is detected.	Fixed to 0	Fixed to 0
CSIHnTSF	7	0: Idle state 1: Communication is in progress or being prepared			
CSIHnFLF	5	Fixed to 0	0: FIFO is not full 1: FIFO is full	Fixed to 0	Fixed to 0
CSIHnEMF	4	Fixed to 1	0: FIFO is not empty 1: FIFO is empty	Fixed to 1	Fixed to 1
CSIHnDCE	3	0: No error is detected. 1: An error is detected.			
CSIHnPE	1	0: No error is detected. 1: An error is detected.			
CSIHnOVE	0	0: No error is detected. 1: An error is detected.	0: No error is detected. 1: An error is detected.	0: No error is detected. 1: An error is detected.	Fixed to 0

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.6 CSIHnSTCR0 — CSIHn Status Clear Register 0

This register clears the status flags of the CSIHnSTR0 status register.

Access: This register can be read/written in 16-bit units.
When read, the value 0000_H is always returned.

Address: <CSIHn_base> + 0008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn TMOEC	CSIHn OFEC	—	—	—	—	—	CSIHn PCT	—	—	—	—	CSIHn DCEC	—	CSIHn PEC	CSIHn OVEC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R	R	R/W	R	R/W	R/W

Table 11.19 CSIHnSTCR0 register contents

Bit Position	Bit Name	Function
15	CSIHnTMOEC	Controls the time-out error flag clear command. 0: No operation. The read value is always 0. 1: Clears the time out error flag (CSIHnSTR0.CSIHnTMOE).
14	CSIHnOFEC	Controls the overflow error flag clear command. 0: No operation. The read value is always 0. 1: Clears the overflow error flag (CSIHnSTR0.CSIHnOFE).
13 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	CSIHnPCT	Controls the FIFO pointer clear command. 0: No operation. The read value is always 0. 1: Clears the FIFO buffer pointers below (in FIFO mode, dual buffer mode, and transmit-only buffer mode) and the status bits.. **FIFO buffer pointer** / **Status bit** CSIHnMRWP0.CSIHnTRWA[6:0] / CSIHnSTR0.CSIHnSPF[7:0] CSIHnMRWP0.CSIHnRRA[6:0] / CSIHnSTR0.CSIHnSRP[7:0] CSIHnMCTL2.CSIHnSOP[6:0] / CSIHnSTR0.CSIHnFLF CSIHnSTR0.CSIHnTSF
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	CSIHnDCEC	Controls the data consistency error flag clear command. 0: No operation. The read value is always 0. 1: Clears the data consistency error flag (CSIHnSTR0.CSIHnDCE).
2	Reserved	When read, the value after reset is read. Writing to this bit, write the value after reset.
1	CSIHnPEC	Controls the parity error flag clear command. 0: No operation. The read value is always 0. 1: Clears the parity error flag (CSIHnSTR0.CSIHnPE).
0	CSIHnOVEC	Controls the overrun error flag clear command. 0: No operation. The read value is always 0. 1: Clears the overrun error flag (CSIHnSTR0.CSIHnOVE).

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.7 CSIHnMCTL0 — CSIHn Memory Control Register 0

This register selects the memory mode and the time-out setting.

Access: This register can be read/written in 16-bit units.

Address: <CSIHn_base> + 1040_H

Value after reset: $001F_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CSIHn MMS[1:0]		—	—	—	CSIHnTO[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 11.20 CSIHnMCTL0 Register Contents

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9 to 8	CSIHnMMS [1:0]	Selects the memory mode. **CSIHn MMS1 / CSIHn MMS0 / Description** 0 / 0 / FIFO mode 0 / 1 / Dual buffer mode 1 / 0 / Transmit-only buffer mode 1 / 1 / Prohibited After a change of the memory mode, the respective buffer pointers must be cleared by setting the CSIHnSTCR0.CSIHnPCT bit to 1. In direct access mode, the setting of these bits is ignored.
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 0	CSIHnTO[4:0]	Selects the time-out setting in FIFO mode. **CSIHnTO[4:0] / Description** 00000_B / No time-out is detected 00001_B / Time-out is (1 × 8 × BRG output clocks) 00010_B / Time-out is (2 × 8 × BRG output clocks) : 11111_B / Time-out is (31 × 8 × BRG output clocks) **CAUTION** Changing the time-out setting is only permitted when CSIHnCTL0.CSIHnPWR = 0. Set the CSIHnTO[4:0] bit to 00000_B in direct access mode, dual buffer mode, or transmit-only buffer mode (except FIFO mode). For details about time-out detection, see also **Section 11.4.18.3, Time-out Error**.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.8 CSIHnMCTL1 — CSIHn Memory Control Register 1

This register selects the conditions to generate the interrupt requests, INTCSIHTIC and INTCSIHTIR in FIFO mode.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 1000_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CSIHnFES[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 11.21 CSIHnMCTL1 Register Contents

Bit position	Bit name	Function
31 to 23	Reserved	When read, the value after reset is read. When writing, write the value after reset.
22 to 16	CSIHnFES[6:0]	Selects the conditions to generate the INTCSIHTIC interrupt (transmit data empty) in FIFO mode. When the number of unsent data to be transmitted in FIFO (checked by the CSIHnSTR0.CSIHnSPF[7:0] bit) and CSIHnMCTL1.CSIHnFES[6:0] match, the INTCSIHTIC interrupt request is generated.
15 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 0	CSIHnFFS[6:0]	Selects the conditions to generate the INTCSIHTIR interrupt (receive data full) in FIFO mode. When the number of received data in FIFO (checked by the CSIHnSTR0.CSIHnSRP[7:0] bit) and (128 - CSIHnMCTL1.CSIHnFFS[6:0]) match, the INTCSIHTIR interrupt request is generated.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.9 CSIHnMCTL2 — CSIHn Memory Control Register 2

This register controls the operation of the memory in dual buffer or transmit-only buffer mode and triggers to start communication in buffer mode.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 1004_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHn BTST	—	—	—	—	—	—	—	CSIHnND[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 11.22 CSIHnMCTL2 Register Contents (1/2)

Bit Position	Bit Name	Function
31	CSIHnBTST	Provides a start trigger for buffer transfer. 0: No operation. 1: Issues the start transfer command. The read value is always 0. **CAUTION** This bit can only be used in dual buffer mode and transmit-only buffer mode.
30 to 24	Reserved	When read, the value after reset is read. When writing, write the value after reset.
23 to 16	CSIHnND[7:0]	Specifies the number of data for each memory mode. The read value indicates the number of remaining communication data.

CSIHn ND[7:0]	Dual buffer mode	Transmit-only buffer mode	FIFO mode	Direct access mode
00_H	Send 0 data	Send 0 data	No influence	No influence
01_H	Send 1 data	Send 1 data	No influence	No influence
:	:	:	No influence	No influence
$3F_H$	Send 63 data	Send 63 data	No influence	No influence
40_H	Send 64 data	Send 64 data	No influence	No influence
:	Prohibited	:	No influence	No influence
$7F_H$	Prohibited	Send 127 data	No influence	No influence
80_H	Prohibited	Send 128 data	No influence	No influence
Other than the above	Setting is prohibited.			

The values are automatically decremented after data transfer (Not decremented in direct access mode).

15 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 11.22 CSIHnMCTL2 Register Contents (2/2)

Bit Position	Bit Name	Function
6 to 0	CSIHnSOP[6:0]	Selects the pointer of the data to be sent. If communication is forced to stop by setting CSIHnCTL0.CSIHnPWR to 0 or CSIHnSTCR0.CSIHnPCT to 1, these bits are cleared by hardware. In FIFO mode, these bits indicate the send address.

CSIHn SOP[6:0]	Dual buffer mode	Transmit-only buffer mode	FIFO mode	Direct access mode
00_H	0000_H	0000_H	0000_H	No influence
01_H	0004_H	0004_H	0004_H	No influence
:	:	:	:	No influence
$3F_H$	$00FC_H$	$00FC_H$	$00FC_H$	No influence
40_H	Prohibited	0100_H	0100_H	No influence
:	Prohibited	:	:	No influence
$7F_H$	Prohibited	$01FC_H$	$01FC_H$	No influence

CAUTION

In direct access mode, these bits are not incremented.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.10 CSIHnMRWP0 — CSIHn Memory Read/Write Pointer Register 0

This register sets the pointers for reading from and writing to the dual or transmit-only buffer.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 1018_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CSIHnRRA[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 11.23 CSIHnMRWP0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	Reserved	When read, the value after reset is read. When writing, write the value after reset.
22 to 16	CSIHnRRA[6:0]	These bits select the read pointer of the receive buffer.

CSIHn RRA[6:0]	Dual buffer mode	Transmit-only buffer mode	FIFO mode	Direct access mode
00_H	0000_H	No influence	0000_H	No influence
01_H	0004_H	No influence	0004_H	No influence
:	:	No influence	:	No influence
$3F_H$	$00FC_H$	No influence	$00FC_H$	No influence
40_H	Prohibited	No influence	0100_H	No influence
:	Prohibited	No influence	:	No influence
$7F_H$	Prohibited	No influence	$01FC_H$	No influence

These bits are automatically incremented when received data is read.
If an overrun error is generated while reading the CSIHnRX0W or CSIHnRX0H register, the read pointer is not incremented.
These bits are cleared when CSIHnSTCR0.CSIHnPCT is set to 1.
In direct access mode and transmit-only buffer mode, these bits are not incremented.
To perform write access in transmit-only buffer mode, set 0000_H to these bits.
In FIFO mode, these bits indicate the read address of the received data.

15 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 11.23 CSIHnMRWP0 Register Contents (2/2)

Bit Position	Bit Name	Function
6 to 0	CSIHnTRWA [6:0]	These bits select the read/write pointer of the transmit buffer.

CSIHn TRWA[6:0]	Dual buffer mode	Transmit-only buffer mode	FIFO mode	Direct access mode
00_H	0000_H	0000_H	0000_H	No influence
01_H	0004_H	0004_H	0004_H	No influence
:	:	:	:	No influence
$3F_H$	$00FC_H$	$00FC_H$	$00FC_H$	No influence
40_H	Prohibited	0100_H	0100_H	No influence
:	Prohibited	:	:	No influence
$7F_H$	Prohibited	$01FC_H$	$01FC_H$	No influence

These bits are automatically incremented when the transmission data is written or read.
These bits are cleared when CSIHnSTCR0.CSIHnPCT is set to 1.
In direct access mode, these bits are not incremented.
In FIFO mode, these bits indicate the read/write address of transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.11 CSIHnCFGx — CSIHn Configuration Register x

These four registers specify for each chip select signal CSIHTCSSx prescaler, parity, data length, recessive configuration for broadcasting, serial data direction, clock and data phase, idle enforcement configuration, idle time, hold time, inter-data time and setup time.

Slave mode

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is effective.

- CSIHnPSx[1:0]: parity usage
- CSIHnDLSx[3:0]: data length selection
- CSIHnDIRx: data direction
- CSIHnCKPx, CSIHnDAPx: clock and data phase

In slave mode, set 0 to all the bits other than above in the CSIHnCFG0 register, and the CSIHnCFG1 to CSIHnCFG3 registers.

Access: This register can be read/written in 32-bit units.

Address: CSIHnCFG0: <CSIHn_base> + 1044_H
CSIHnCFG1: <CSIHn_base> + 1048_H
CSIHnCFG2: <CSIHn_base> + $104C_H$
CSIHnCFG3: <CSIHn_base> + 1050_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHn BRSSx[1:0]		CSIHn PSx[1:0]		CSIHnDLSx[3:0]				—	—	—	—	CSIHn RCBx	CSIHn DIRx	CSIHn CKPx	CSIHn DAPx
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn IDLx	CSIHnIDx[2:0]			CSIHnHDx[3:0]				CSIHnINx[3:0]				CSIHnSPx[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.24 CSIHnCFGx Register Contents (1/5)

Bit Position	Bit Name	Function
31, 30	CSIHnBRSSx [1:0]	Selects the baud rate setting register (CSIHnBRSy). **CSIHn BRSSx1 / CSIHn BRSSx0 / Baud rate setting register selection** 0 / 0 / The baud rate is set according to the CSIHnBRS0 setting 0 / 1 / The baud rate is set according to the CSIHnBRS1 setting 1 / 0 / The baud rate is set according to the CSIHnBRS2 setting 1 / 1 / The baud rate is set according to the CSIHnBRS3 setting The maximum value for setting the transfer clock frequency, combining the CSIHnCTL2.CSIHnPRS[2:0] setting, must be as follows: Master mode: PCLK/8 Slave mode: PCLK/16

Table 11.24 CSIHnCFGx Register Contents (2/5)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>29, 28</td><td>CSIHnPSx[1:0]</td><td>Selects the parity for sending or receiving chip select signal x.
<table>
<tr><th>CSIHn PSx1</th><th>CSIHn PSx0</th><th>Transmission</th><th>Reception</th></tr>
<tr><td>0</td><td>0</td><td>No parity is transmitted</td><td>No parity is expected</td></tr>
<tr><td>0</td><td>1</td><td>Adds parity bit fixed to 0</td><td>Parity bit is expected but not judged</td></tr>
<tr><td>1</td><td>0</td><td>Adds odd parity only</td><td>Odd parity bit is expected</td></tr>
<tr><td>1</td><td>1</td><td>Adds even parity</td><td>Even parity bit is expected</td></tr>
</table></td></tr>
<tr><td>27 to 24</td><td>CSIHnDLSx [3:0]</td><td>Selects the data length for chip select signal x.
<table>
<tr><th>CSIHnDLSx[3:0]</th><th>Data length</th></tr>
<tr><td>0000_B</td><td>16 bits</td></tr>
<tr><td>0001_B</td><td>1 bit</td></tr>
<tr><td>0010_B</td><td>2 bits</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>1111_B</td><td>15 bits</td></tr>
</table>
CAUTION
When CSIHnTX0W.CSIHnEDL = 1, the setting of this bit has no effect. When CSIHnTX0W.CSIHnEDL = 0 (the data length is 16 bits), the setting of this bit is valid. Only when the previous transmit data is 16 bits while CSIHnEDL = 1, writing 1 to this bit is enabled.</td></tr>
<tr><td>23 to 20</td><td>Reserved</td><td>When read, the value after reset is read. When writing, write the value after reset.</td></tr>
<tr><td>19</td><td>CSIHnRCBx</td><td>Selects the recessive configuration for broadcasting for chip select x:
0: Dominant (higher priority)
1: Recessive (lower priority)
For details, see Section 11.4.9.1, Configuration Registers</td></tr>
<tr><td>18</td><td>CSIHnDIRx</td><td>Selects the serial data direction of chip select signal x.
0: Data is transmitted/received with MSB first.
1: Data is transmitted/received with LSB first.
For details, refer to Section 11.4.15, Serial Data Direction Selection.</td></tr>
</table>

Table 11.24 CSIHnCFGx Register Contents (3/5)

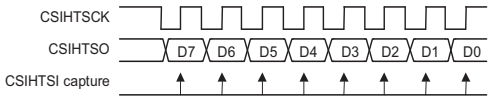
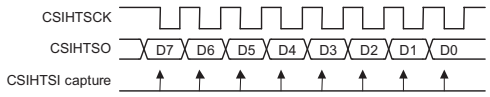
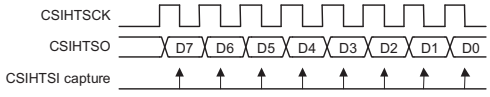
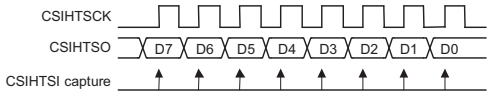
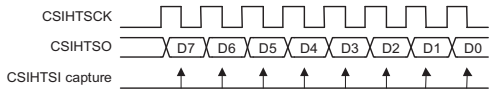
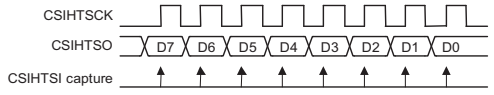
<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>17</td><td>CSIHnCKPx</td><td rowspan="2">CSIHnCKPx: Clock phase selection bit
CSIHnDAPx: Data phase selection bit
• CSIHnCTL1.CSIHnCKR = 0
<table><tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock and data phase selection</th></tr><tr><td>0</td><td>0</td><td>  </td></tr><tr><td>0</td><td>1</td><td>  </td></tr><tr><td>1</td><td>0</td><td>  </td></tr><tr><td>1</td><td>1</td><td>  </td></tr></table>• CSIHnCTL1.CSIHnCKR = 1
<table><tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock and data phase selection</th></tr><tr><td>0</td><td>0</td><td>  </td></tr><tr><td>0</td><td>1</td><td>  </td></tr><tr><td>1</td><td>×</td><td>Setting prohibited</td></tr></table></td></tr>
<tr><td>16</td><td>CSIHnDAPx</td></tr>
<tr><td>15</td><td>CSIHnIDLx</td><td>Selects the idle enforcement configuration for chip select x:
0: If the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers is different, an idle state is inserted between two transfers. If the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers is the same, an idle state is not inserted between two transfers.
1: Regardless of the CSIHnTX0W.CSIHnCSx setting of two consecutive transfers, an idle state is inserted between two transfers.
This bit is only available in master mode.
For details about the enforced idle state, see Section 11.4.21, Enforced Chip Select Idle Setting.</td></tr>
</table>

Table 11.24 CSIHnCFGx Register Contents (4/5)

Bit Position	Bit Name	Function
14 to 12	CSIHnIDx[2:0]	Selects the idle time for chip select signal x.

CSIHnIDx[2:0]	Idle time
000_B	0.5 transmission clock cycle
001_B	1.0 transmission clock cycle
010_B	1.5 transmission clock cycle
011_B	2.5 transmission clock cycle
100_B	3.5 transmission clock cycle
101_B	4.5 transmission clock cycle
110_B	6.5 transmission clock cycle
111_B	8.5 transmission clock cycle

These bits are only available in master mode.

Bit Position	Bit Name	Function
11 to 8	CSIHnHDx[3:0]	Specifies the hold time for chip select signal x in transmission clock cycles.

CSIHnHDx [3:0]	Hold time with CSIHnCTL1.CSIHnSIT = 0	Hold time with CSIHnCTL1.CSIHnSIT = 1
0000_B	0.5 transmission clock cycle	1.0 transmission clock cycle
0001_B	1.0 transmission clock cycle	1.5 transmission clock cycle
0010_B	1.5 transmission clock cycle	2.0 transmission clock cycle
0011_B	2.5 transmission clock cycle	3.0 transmission clock cycle
0100_B	3.5 transmission clock cycle	4.0 transmission clock cycle
0101_B	4.5 transmission clock cycle	5.0 transmission clock cycle
0110_B	6.5 transmission clock cycle	7.0 transmission clock cycle
0111_B	8.5 transmission clock cycle	9.0 transmission clock cycle
1000_B	9.5 transmission clock cycle	10.0 transmission clock cycle
1001_B	10.5 transmission clock cycle	11.0 transmission clock cycle
1010_B	11.5 transmission clock cycle	12.0 transmission clock cycle
1011_B	12.5 transmission clock cycle	13.0 transmission clock cycle
1100_B	14.5 transmission clock cycle	15.0 transmission clock cycle
1101_B	16.5 transmission clock cycle	17.0 transmission clock cycle
1110_B	18.5 transmission clock cycle	19.0 transmission clock cycle
1111_B	20.5 transmission clock cycle	21.0 transmission clock cycle

These bits are only available in master mode.

Table 11.24 CSIHnCFGx Register Contents (5/5)

Bit Position	Bit Name	Function
7 to 4	CSIHnINx[3:0]	Specifies the inter-data time for chip select signal x in transmission clock cycles.

CSIHnINx[3:0]	Inter-data time when CSIHnCTL1.CSIHnSIT = 0	Inter-data time when CSIHnCTL1.CSIHnSIT = 1
0000_B	0.0 transmission clock cycle	0.5 transmission clock cycle
0001_B	0.5 transmission clock cycle	1.0 transmission clock cycle
0010_B	1.0 transmission clock cycle	1.5 transmission clock cycle
0011_B	2.0 transmission clock cycle	2.5 transmission clock cycle
0100_B	3.0 transmission clock cycle	3.5 transmission clock cycle
0101_B	4.0 transmission clock cycle	4.5 transmission clock cycle
0110_B	6.0 transmission clock cycle	6.5 transmission clock cycle
0111_B	8.0 transmission clock cycle	8.5 transmission clock cycle
1000_B	9.0 transmission clock cycle	9.5 transmission clock cycle
1001_B	10.0 transmission clock cycle	10.5 transmission clock cycle
1010_B	11.0 transmission clock cycle	11.5 transmission clock cycle
1011_B	12.0 transmission clock cycle	12.5 transmission clock cycle
1100_B	14.0 transmission clock cycle	14.5 transmission clock cycle
1101_B	16.0 transmission clock cycle	16.5 transmission clock cycle
1110_B	18.0 transmission clock cycle	18.5 transmission clock cycle
1111_B	20.0 transmission clock cycle	20.5 transmission clock cycle

These bits are only available in master mode.

3 to 0	CSIHnSPx[3:0]	Specifies the setup time for chip select signal x in transmission clock cycles.

CSIHnSPx[3:0]	Setup time
0000_B	0.5 transmission clock cycle
0001_B	1.0 transmission clock cycle
0010_B	1.5 transmission clock cycle
0011_B	2.5 transmission clock cycle
0100_B	3.5 transmission clock cycle
0101_B	4.5 transmission clock cycle
0110_B	6.5 transmission clock cycle
0111_B	8.5 transmission clock cycle
1000_B	9.5 transmission clock cycle
1001_B	10.5 transmission clock cycle
1010_B	11.5 transmission clock cycle
1011_B	12.5 transmission clock cycle
1100_B	14.5 transmission clock cycle
1101_B	16.5 transmission clock cycle
1110_B	18.5 transmission clock cycle
1111_B	20.5 transmission clock cycle

These bits are only available in master mode.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.12 CSIHnTX0W — CSIHn Transmit Data Register 0 for Word Access

This register stores transmission data. In addition, it specifies the communication interrupt request, the end-of-job, the extended data length, and the chip select activation.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 1008_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHn CIRE	CSIHn EOJ	CSIHn EDL	—	—	—	—	—	—	—	—	—	CSIHnCS3	CSIHnCS2	CSIHnCS1	CSIHnCS0
Value after reset	—	—	—	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

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

Table 11.25 CSIHnTX0W Register Contents (1/2)

Bit position	Bit name	Function
31	CSIHnCIRE	Enables the communication interrupt request INTCSIHTIC in dual buffer or transmit-only buffer mode, or the job completion interrupt INTCSIHTIJC request in FIFO mode. 0: No interrupt is requested. 1: An interrupt is requested. Generates interrupt INTCSIHTIC or INTCSIHTIJC after transmission. For details, see **Section 11.4.3, INTCSIHTIC (Communication Status Interrupt)** and **Section 11.4.6, INTCSIHTIJC (job Completion Interrupt)**. **CAUTION** This bit is only valid when job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
30	CSIHnEOJ	Specifies the end of a job. 0: Indicates that the job does not end. The job continues. 1: Indicates end-of-job data. **CAUTION** This bit is only valid when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). This bit must be set to 0 in slave mode.
29	CSIHnEDL	Specifies whether the associated data requires the extended data length (EDL) option. 0: Normal operation 1: Enables the extended data length. The associated data is transmitted as a 16-bit packet. No inter-data time or idle time will be inserted after the data is transmitted. If CSIHnCTL1.CSIHnEDLE = 1 and CSIHnTX0W.CSIHnEDL = 1, the subsequent data must have the same CS selection. If CS is modified for the subsequent data, the correct operation is not assured. **CAUTION** This bit is only available if CSIHnCTL1.CSIHnEDLE = 1.
28 to 20	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 11.25 CSIHnTX0W Register Contents (2/2)

Bit position	Bit name	Function
19 to 16	CSIHnCS[3:0]	Activates one or several chip select signals. 0: Activates chip select x for the associated transmission. 1: Deactivates chip select x for the associated transmission. Setting CSIHnTX0W.CSIHnCS[3:0] = F_H is prohibited. **CAUTION** If several chip select signals are enabled for broadcasting, the configuration of one with CSIHnCFGx.CSIHnRCBx = 0 (dominant) is used. In this case, all dominant chip select signals must be set to precisely the same value. In slave mode, set the CSIHnCS[3:0] bits to E_H.
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.13 CSIHnTX0H — CSIHn Transmit Data Register 0 for Half Word Access

This register stores the transmission data. This register is the same as bits 15 to 0 of register CSIHnTX0W.

The 16 high-order bits of CSIHnTX0W are applied for transfer. Set transmit data to CSIHnTX0W before using this register because the value of CSIHnTX0W is undefined after a reset.

Access: This register can be read/written in 16-bit units.

Address: <CSIHn_base> + $100C_H$

Value after reset: Undefined

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

Table 11.26 CSIHnTX0H Register Contents

Bit position	Bit name	Function
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.14 CSIHnRX0W — CSIHn Receive Data Register 0 for Word Access

This register stores the received data.

Access: This register can be read in 32-bit units.

Address: <CSIHn_base> + 1010_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CSIHnRPE	CSIHnTDCE	—	—	—	—	CSIHnCS3	CSIHnCS2	CSIHnCS1	CSIHnCS0
Value after reset	0	0	0	0	0	0	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 11.27 CSIHnRX0W Register contents

Bit position	Bit name	Function
31 to 26	Reserved	When read, the value after reset is read. When writing, write the value after reset.
25	CSIHnRPE	Indicates whether a reception data parity error was detected. 0: No parity error was detected on the associated reception data. 1: A parity error was detected on the associated reception data.
24	CSIHnTDCE	Indicates whether a transmission data consistency error was detected. 0: No consistency error was detected on the associated transmission data. 1: A consistency error was detected on the associated transmission data.
23 to 20	Reserved	When read, the value after reset is read. When writing, write the value after reset.
19 to 16	CSIHnCSx (x = 3 to 0)	Indicates which chip select signal was activated. 0: Chip select x was activated for the associated reception. 1: Chip select x was deactivated for the associated reception.
15 to 0	CSIHnRX[15:0]	Stores the received data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.15 CSIHnRX0H — CSIHn Receive Data Register 0 for Half Word Access

This register stores the received data. This register is the same as bits 15 to 0 of register CSIHnRX0W.

Access: This register can only be read in 16-bit units.

Address: <CSIHn_base> + 1014_H

Value after reset: Undefined

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

Table 11.28 CSIHnRX0H Register Contents

Bit position	Bit name	Function
15 to 0	CSIHnRX[15:0]	Stores the received data.

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.3.16 CSIHnBRSy — CSIHn Baud Rate Setting Register y (y = 0 to 3)

This register sets the transfer clock frequency for each chip select signal.

With CSIHnCFG0 (to 3).CSIHnBRSSx[1:0] bits, one of the four types of transfer clock frequency settings can be selected for each chip select signal. For details of transfer clock frequency setting, refer to **Section 11.4.11, Transmission Clock Selection**.

Access: This register can be read/written in 16-bit units.

Address: CSIHnBRS0: <CSIHn_base> + 1068_H
CSIHnBRS1: <CSIHn_base> + $106C_H$
CSIHnBRS2: <CSIHn_base> + 1070_H
CSIHnBRS3: <CSIHn_base> + 1074_H

Value after reset: 0000_H

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

Table 11.29 CSIHnBRSy Register Contents

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11 to 0	CSIHnBRS [11:0]	0: BRG stopped 1: PCLK/(2^{α} × 1 × 2) 2: PCLK/(2^{α} × 2 × 2) 3: PCLK/(2^{α} × 3 × 2) 4: PCLK/(2^{α} × 4 × 2) . . . 4095: PCLK/(2^{α} × 4095 × 2) α is the value of CSIHnCTL2.CSIHnPRS[2:0].

CAUTION

When setting this register, see **Table 11.43, Notes on Setting Registers**.

11.4 Functions

11.4.1 Overview of Interrupt Function

CSIH can generate the interrupt requests listed in the table below:

Table 11.30 Interrupt Generation

Memory mode	Interrupt	Cause of interrupt	
		Job mode disabled CSIHnCTL1.CSIHnJE = 0	**Job mode enabled CSIHnCTL1.CSIHnJE = 1**
FIFO	INTCSIHTIC (communication status interrupt)	Tx data empty*1	Tx data empty*1 except job abortion*4
	INTCSIHTIR (reception status interrupt)	Rx data full*2 and CSIHnCTL0.CSIHnRXE = 1	Rx data full*2 and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	CSIHnTX0W.CSIHnCIRE = 1, (when it is not Tx data empty) or job abortion*4
Transmit-only buffer	INTCSIHTIC (communication status interrupt)	End of communication	CSIHnTX0W.CSIHnCIRE = 1 and (CSIHnCTL0.CSIHnJOBE = 0 or CSIHnTX0W.CSIHnEOJ = 0)
	INTCSIHTIR (reception status interrupt)	Data received and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4
Dual buffer	INTCSIHTIC (communication status interrupt)	End of communication	CSIHnTX0W.CSIHnCIRE = 1 and (CSIHnCTL0.CSIHnJOBE = 0 or CSIHnTX0W.CSIHnEOJ = 0)
	INTCSIHTIR (reception status interrupt)	End of communication and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4
Direct access	INTCSIHTIC (communication status interrupt)	1 data transferred	1 data transferred, (except the state of job abortion*4)
	INTCSIHTIR (reception status interrupt)	Data received and CSIHnCTL0.CSIHnRXE = 1	Data received and CSIHnCTL0.CSIHnRXE = 1
	INTCSIHTIRE (communication error interrupt)	Error detected	Error detected
	INTCSIHTIJC*3 (job completion interrupt)	Not applicable	Job abortion*4

Note 1. “Tx data empty” refers to the FIFO fill level, defined by CSIHnMCTL1.CSIHnFES[6:0].
Note 2. “Rx data full” refers to the FIFO fill level, defined by CSIHnMCTL1.CSIHnFFS[6:0].
Note 3. INTCSIHTIJC is not available in slave mode.
Note 4. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1
During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

The communication error interrupt INTCSIHTIRE is generated when an error is detected. The generation of the other interrupts depends on the memory mode, the job mode, and - in case of the job completion interrupt INTCSIHTIJC - also the operating mode.

The job completion interrupt INTCSIHTIJC is only generated when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). It is not available in slave mode.

11.4.2 Interrupt Delay

In master mode, all interrupts generated by the master can be delayed by one half period of the transmission clock, CSIHTSCK. This is not possible in slave mode.

The delay is specified by setting bit CSIHnCTL1.CSIHnSIT = 1.

The following example illustrates the interrupt delay function, assuming a setting of CSIHnCTL1.CSIHnSIT = 1 (interrupt delay enabled), CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0 (Normal clock and data phases), and CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B (data length 8 bits).

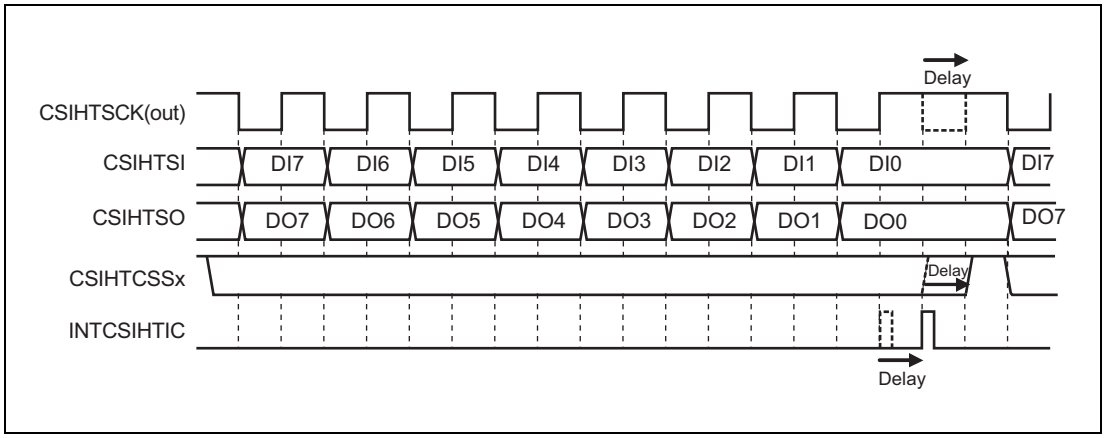


Figure 11.2 Interrupt Delay Function (CSIHnCTL1.CSIHnSIT = 1)

Setting CSIHnCTL1.CSIHnSIT = 1 adds half period delay to the transmission clock. This delays also the end of the present chip select signal (CSIHTCSSx).

11.4.3 INTCSIHTIC (Communication Status Interrupt)

Depending on the memory mode and the job mode, this interrupt is generated according to the conditions shown in the following table.

Table 11.31 INTCSIHTIC Interrupt Generation

Memory mode	Cause of interrupt	
	Job mode disabled CSIHnCTL1.CSIHnJE = 0	**Job mode enabled CSIHnCTL1.CSIHnJE = 1**
FIFO	This interrupt is generated when transmission data is about to be missing in the FIFO, indicating to the application that new data should be added. INTCSIHTIC is generated, if the number of data to be sent remaining in the FIFO (CSIHnSTR0.CSIHnSPF[7:0]) equals CSIHnMTCL1.CSIHnFES[6:0].	Similar to “when JE is 0”, an interrupt is generated when the number of transmit data remained in the FIFO CSIHnSTR0.CSIHnSPF[7:0] is the same number as CSIHnMCTL1.CSIHnFES[6:0]. At the time of job abortion, no interrupt is generated.
Transmit-only buffer, dual buffer	Generated at the End of communication. (Specified by the CSIHnMTLC2.CSIHnND[7:0] bit)	Generated when data with CSIHnTX0W.CSIHnCIRE = 1 is sent. Note that if data with CSIHnTX0W.CSIHnCIRE = 1 and job abortion*1 are sent, the INTCSIHTIJC interrupt is generated instead of INTCSIHTIC.
Direct access	Generated after every data transfer.	Generated after every data transfer, except when the communication was aborted.

Note 1. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1
During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

11.4.3.1 INTCSIHTIC in Direct Access Mode

The examples below show the INTCSIHTIC behavior in direct access mode.

The examples assume:

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases
  (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)

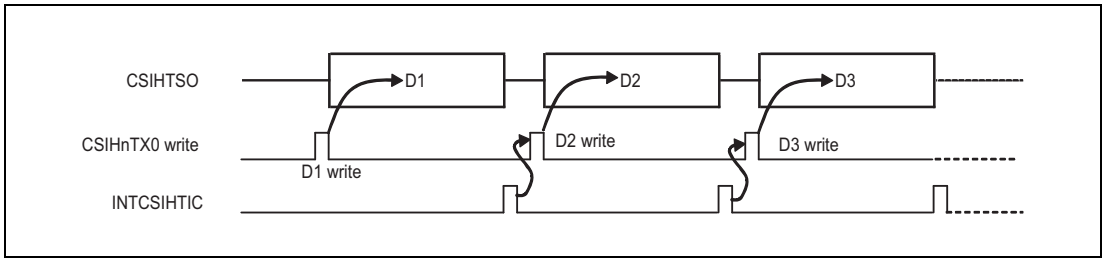


Figure 11.3 Generation of INTCSIHTIC After Transfer (CSIHnCTL1.CSIHnSLIT = 0)

If job mode is enabled (CSIHnCTL1.CSIHnJE = 1) and a job ends because data is sent with CSIHnTX0W.CSIHnEOJ = 1 and communication stop is requested (CSIHnCTL0.CSIHnJOBE = 1), then INTCSIHTIC is replaced by the job completion interrupt INTCSIHTIJC.

INTCSIHTIC can also be set up to occur as soon as the CSIHnTX0 register is free for the next data. This is specified by setting CSIHnCTL1.CSIHnSLIT = 1.

The effect is illustrated in the figure below.

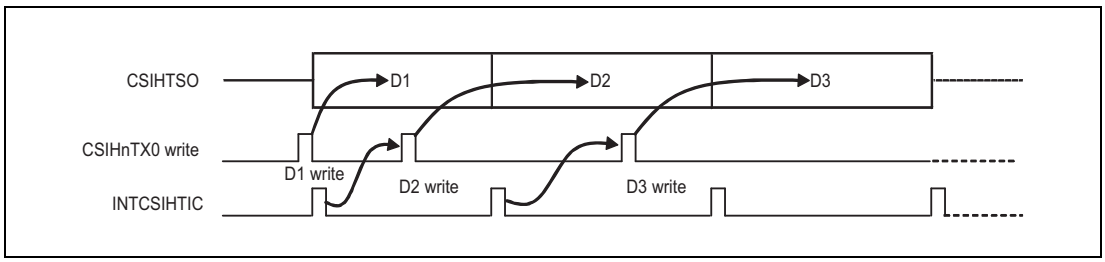


Figure 11.4 Immediate Generation of INTCSIHTIC (CSIHnCTL1.CSIHnSLIT = 1)

Thus, the new data can be written in advance.

NOTE

During high priority communication in transmit-only buffer mode, the operation is the same as that in direct access mode.

11.4.3.2 INTCSIHTIC in FIFO Mode

The example below shows the INTCSIHTIC behavior in FIFO mode.

The example assumes:

- Master mode
- FIFO mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases
  (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

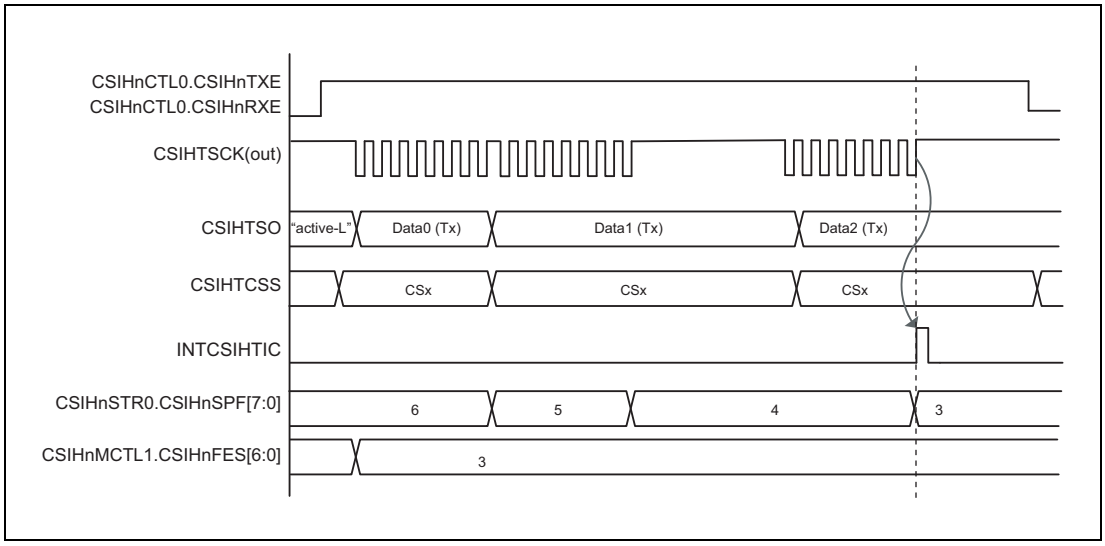


Figure 11.5 Generation of INTCSIHTIC in FIFO Memory Mode

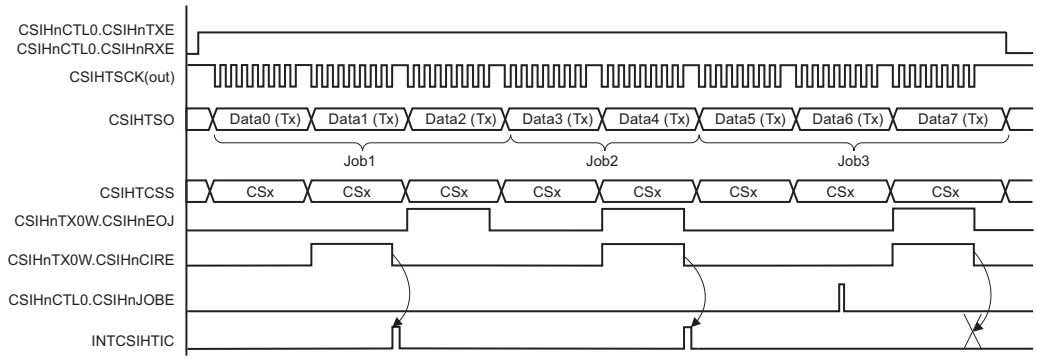
The condition for “FIFO empty” is specified in CSIHnMCTL1.CSIHnFES[6:0]. In the example of the diagram above, the number of unsent data in FIFO is set to 3. CSIHnSTR0.CSIHnSPF[7:0] indicates the number of unsent data. When both match, the interrupt INTCSIHTIC occurs.

11.4.3.3 INTCSIHTIC in job mode

The example below shows the INTCSIHTIC behavior in job mode.

The example assumes:

- Master mode
- Job mode enabled (CSIHnCTL1.CSIHnJE = 1)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases
  (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)



Note 1. CSx can be changed per data transfer.

Figure 11.6 Generation of INTCSIHTIC in job mode

The rules for generating INTCSIHTIC in job mode are shown in the following table.

Table 11.32 Generation of INTCSIHTIC in job mode

CSIHnTX0W. CSIHnEOJ	CSIHnTX0W. CSIHnCIRE	INTCSIHTIC
0	0	Not generated
0	1	Generated
1	0	Not generated
1	1	CSIHnCTL0.CSIHnJOBE = 0: Generated
		CSIHnCTL0.CSIHnJOBE = 1: Not generated, replaced by interrupt INTCSIHTIJC

11.4.4 INTCSIHTIR (Reception Status Interrupt)

Depending on the memory mode and the job mode, this interrupt is generated according to the conditions below.

Table 11.33 INTCSIHTIR Interrupt Generation

Memory mode	Cause of interrupt	
	Job mode disabled CSIHnCTL1.CSIHnJE = 0	**Job mode enabled CSIHnCTL1.CSIHnJE = 1**
FIFO	This interrupt occurs when CSIHnCTL0.CSIHnRXE is 1 and the FIFO buffer is almost full with received data, indicating to the application that the FIFO must be emptied. INTCSIHTIR is generated, if the number of received data in the FIFO (CSIHnSTR0.CSIHnSRP[7:0]) equals (128 – CSIHnMCTL1.CSIHnFFS[6:0]).	
Dual buffer	Generated when the communication has finished (as specified by the CSIHnMCTL2.CSIHnND[7:0] bit) and CSIHnCTL0.CSIHnRXE = 1.	Generated after every data transfer.
Transmit-only buffer, Direct access	Generated after every data transfer.	

11.4.4.1 INTCSIHTIR in Direct Access Mode

The example below shows the INTCSIHTIR behavior in direct access mode.

The example below assumes:

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

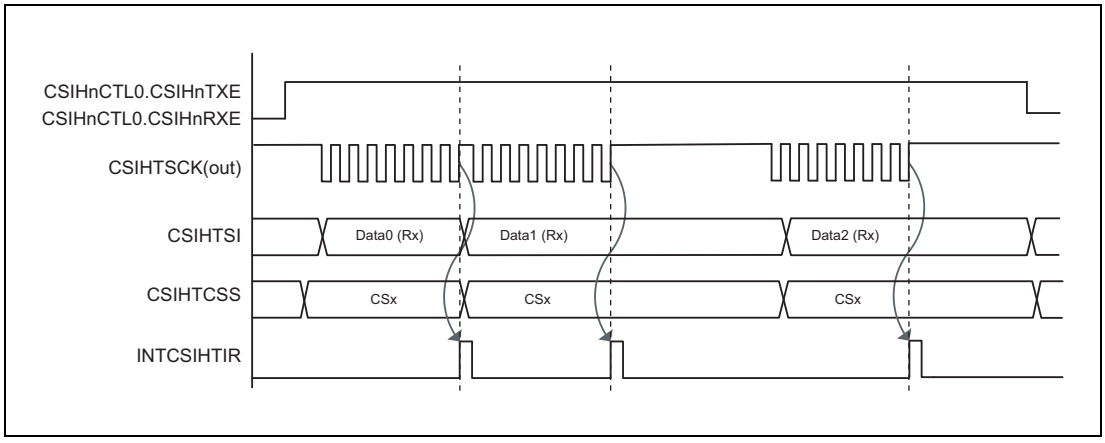


Figure 11.7 Generation of INTCSIHTIR in Direct Access Memory Mode

11.4.4.2 INTCSIHTIR in Dual Buffer Mode

The example below shows the INTCSIHTIR behavior in dual buffer mode.

The example assumes:

- Master mode
- Dual buffer mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Default clock and data phase
  (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

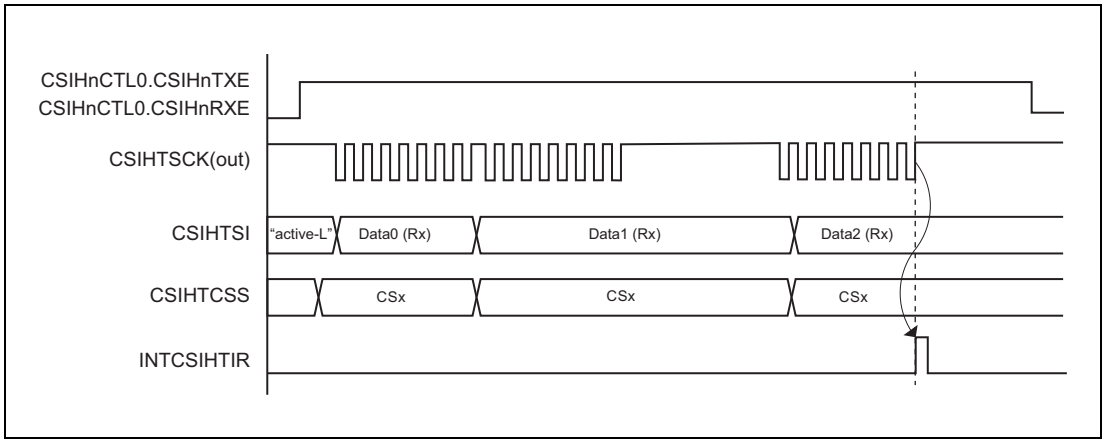


Figure 11.8 Generation of INTCSIHTIR in Dual Buffer Mode

11.4.5 INTCSIHTIRE (Reception Error Interrupt)

This interrupt is generated whenever an error is detected.

For details about generation interruption timing, see **Section 11.4.18, Error Detection**.

Table 11.34 Data Error Types

Error type	Communication Status after Communication Error Interrupt	Comment
FIFO overflow error	Interrupt is generated and communication continues	The data are not written to the FIFO buffer and the overflow of data is lost, but communications started before the error is continued.
Parity error	Interrupt is generated and communication continues	—
Data consistency error	Interrupt is generated and communication continues	—
Time-out error	Interrupt is generated and communication continues	—
Overrun error	Condition for errors 1: In FIFO mode, when the number of received data is 0 and CPU reads the CSIHnRX0W/H register, an interrupt is generated and communication continues.	—
	Condition for errors 2: In slave mode, when CSIHnCTL1.CSIHnHSE = 0 (handshake function disabled): (1) In direct access mode or transmit-only buffer mode, when reception is completed while the previous received data is remained in the CSIHnRX0W/H register, an interrupt is generated, and communication continues. (2) In FIFO mode, when reception by the FIFO buffer is completed and the buffer is in the full state, an interrupt is generated. Communication continues.	In slave mode, when CSIHnCTL1.CSIHnHSE = 1 (handshake function enabled), communication is suspended due to handshake, an overrun error is not generated.

The type of error that caused the generation of INTCSIHTIRE is flagged in register CSIHnSTR0.

Additionally a parity error flag and a data consistency error flag are attached to the received data in CSIHnRX0W.

For details about the various error types, see **Section 11.4.18, Error Detection**.

11.4.6 INTCSIHTIJC (job Completion Interrupt)

This interrupt supports the handling of jobs, see **Section 11.4.9.3, Job Concept**. This interrupt is only available in master mode.

Job mode is enabled by setting CSIHnCTL1.CSIHnJE = 1. When CSIHnCTL1.CSIHnJE = 0, INTCSIHTIJC is not generated.

Depending on the memory mode, this interrupt is generated according to the condition shown in the following table.

Table 11.35 INTCSIHTIJC Interrupt Generation

Memory mode	Job mode disabled CSIHnCTL1.CSIHnJE = 0	Job mode enabled CSIHnCTL1.CSIHnJE = 1
FIFO	Not applicable	Indicates that the communication has stopped at the end of a job after a job abortion*1 was triggered. If FIFO empty is not detected and when CSIHnCIRE is 1, INTCSIHTIJC is generated.
Transmit-only buffer		Indicates that the communication has stopped at the end of a job after a job abortion*1 was triggered
Dual buffer		
Direct access		

Note 1. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1

11.4.7 Operating Modes (Master/Slave)

For a particular CSIH module, the master or slave mode determines the source of the serial clock.

11.4.7.1 Master Mode

In master mode, the serial transmission clock is generated by the internal baud rate generator (BRG) and provided to the slave(s) by signal CSIHTSCK.

Master mode is enabled by setting CSIHnCTL2.CSIHnPRS[2:0] to any value other than 111_B. In master mode, the BRG frequency can be set by combining the CSIHnCTL2.CSIHnPRS[2:0] bits and the CSIHnBRSy.CSIHnBRS[11:0] bits.

(1) Chip Select Signals

In master mode, one or several chip select signals can be used. If several slaves are connected to the master, the chip select signals can be used to address one or several of the slaves. Only a selected slave is then enabled for communication.

The communication protocol as well as additional parameters are stored separately for each chip select signal. This makes it possible to adapt the data transfer individually to the requirements of each slave. For details, see **Section 11.4.9, Chip Selection (CS) Features**.

(2) Clock Default Setting

The default level of CSIHTSCK depends on the CSIHTSCK clock inversion function bit: It is high when CSIHnCTL1.CSIHnCKR = 0, and is low when CSIHnCTL1.CSIHnCKR = 1.

The example below shows the communication in master mode for data length 8 bits, CSIHnCTL1.CSIHnCKR = 0, CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0, and MSB first:

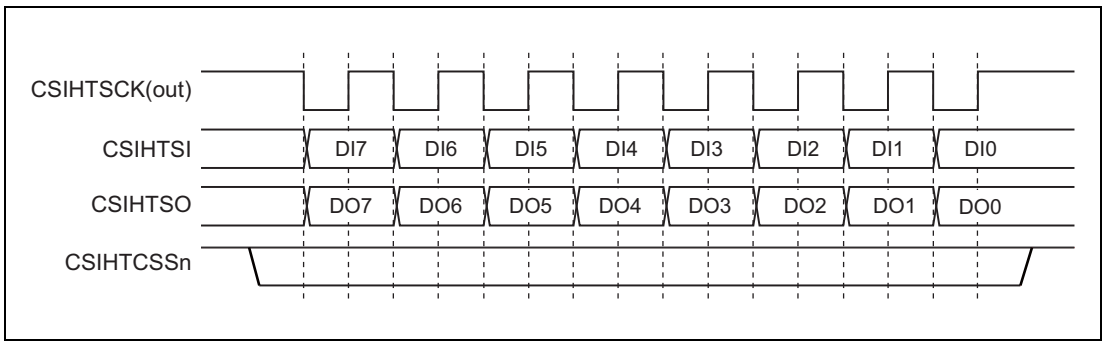


Figure 11.9 Transmission/Reception in Master Mode

11.4.7.2 Slave Mode

In slave mode, another device is the communication master and provides the transmission clock. Send/receive operation normally starts as soon as a clock signal is detected.

Slave mode is selected by setting the CSIHnCTL2.CSIHnPRS[2:0] bits to 111_B.

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is enabled (setting of the CSIHnCFG1 to CSIHnCFG3 registers is disabled).

- CSIHnPSx[1:0]: parity usage
- CSIHnDLSx[3:0]: data length selection
- CSIHnDIRx: data direction
- CSIHnCKPx, CSIHnDAPx: clock and data phase

NOTE

When using slave mode, disable the baud rate generator (BRG) by setting the CSIHnBRSy.CSIHnBRS[11:0] bits to 000_H. However, if you are using a time-out error, set the CSIHnBRSy.CSIHnBRS[11:0] bits to a value other than 000_H.

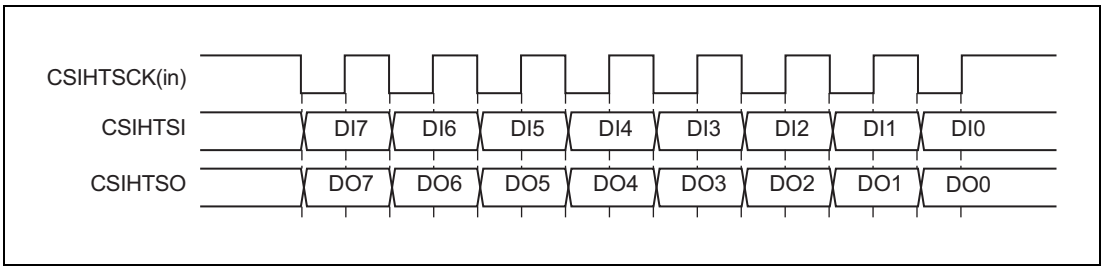


Figure 11.10 Transmission/Reception in Slave Mode

11.4.8 Master/Slave Connections

11.4.8.1 One Master and One Slave

The following figure illustrates the connections between one master and one slave.

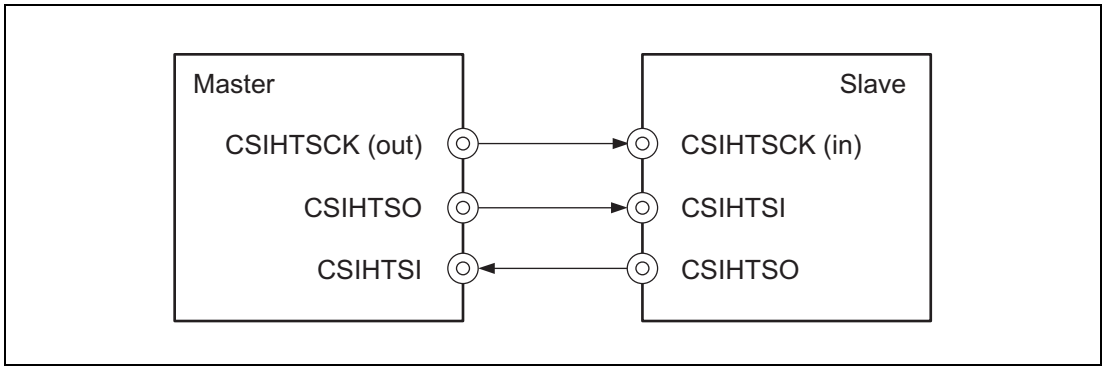


Figure 11.11 Direct Master/Slave Connection

11.4.8.2 One Master and Multiple Slaves

The following figure illustrates the connections between one master and multiple slaves. In this example, the master provides one chip select (CS) signal to each of the slaves. This signal is connected to the slave select input $\overline{\text{CSIHTSSI}}$ of the slave.

The $\overline{\text{CSIHTSSI}}$ signal can be enabled or disabled by using the CSIHnCTL1.CSIHnSSE bit.

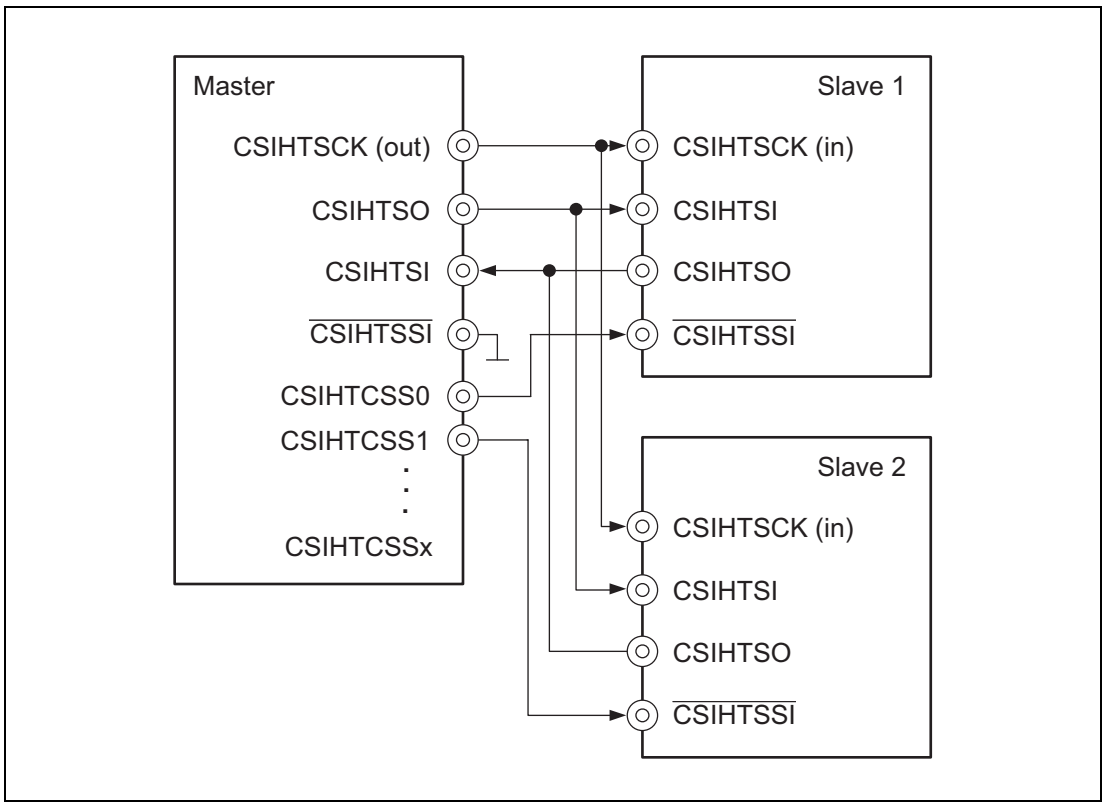


Figure 11.12 Master to Multiple Slaves Connection

By default, the chip select level is active low. That means, a slave is selected (enabled) as a CSIH slave when its $\overline{\text{CSIHTSSI}}$ signal has low level. However, to adapt the CS to other devices, the output level of each chip select signal can also be programmed to be active high.

If a slave is not selected, it will neither receive nor transmit data. In addition, its output CSIHTSO of a slave that is not selected is set to input mode in order to avoid interference with the output of another slave that was selected, in transmit-only mode or transmit/receive mode (CSIHnCTL0.CSIHnTXE = 1).

11.4.9 Chip Selection (CS) Features

The chip select signal, CSIHTCSSx can be used by the master to select one or several slaves for communication.

11.4.9.1 Configuration Registers

The parameters for each chip select signal CSIHTCSSx are defined in the corresponding configuration register CSIHnCFGx. The parameters include the communication protocol and additional CS parameters.

The communication protocol specifies:

- Data length: The number of bits to be sent or received. (CSIHnCFGx.CSIHnDLSx[3:0])
- Transfer direction: MSB or LSB first. (CSIHnCFGx.CSIHnDIRx)
- Parity usage: Odd, even, 0 parity or none. (CSIHnCFGx.CSIHnPSx[1:0])
- Clock phase and data phase. (CSIHnCFGx.CSIHnCKPx, CSIHnCFGx.CSIHnDAPx)

Additional parameters for each chip select signal that is only available in master mode are:

- Prescaler selection of the baud-rate generator separately for each chip select signal (CSIHnCFGx.CSIHnBRSSx[1:0])
- Chip select priority: Separates between “dominant” and “recessive” chip select signals. The priority applies if two or more chip selects signals with different configurations are simultaneously activated for message broadcasting. In this case, the configuration that is set as dominant is used. (CSIHnCFGx.CSIHnRCBx)

The principle is also called “Recessive Configuration for Broadcasting” (RCB).

CAUTION

It is forbidden to specify several chip select signals as dominant with different configurations unless all dominant chip selects have the same configuration.

- Chip select timing:
  - Setup time T_{setup}: The time from setting the CS signal active to starting data output. (CSIHnCFGx.CSIHnSPx[3:0])
  - Inter-data time T_{inter}: The time between one data and the next following data while the CS signal is active.
  - Hold time T_{hold}: Hold time of CS signal active level before changing the CS signal. (CSIHnCFGx.CSIHnHDx[3:0])
  - Idle time T_{idle}: Inactive time after every data transfer is completed. (CSIHnCFGx.CSIHnIDx[2:0])

The CS timings of the setup time, the inter-data time, the hold time, and the idle time are illustrated in the figure below. When the CSIHnCFGx.CSIHnIDLx bit set to 1, idle time is inserted per transfer regardless of the CS signal.

Figure 11.13 provides an example when the default CSIHTCSS1 and CSIHTCSS2 signals are active low (CSIHnCTL1.CSIHnCSL1 bit = 0, CSIHnCTL1.CSIHnCSL2 bit = 0). The active level can be specified individually for each CS.

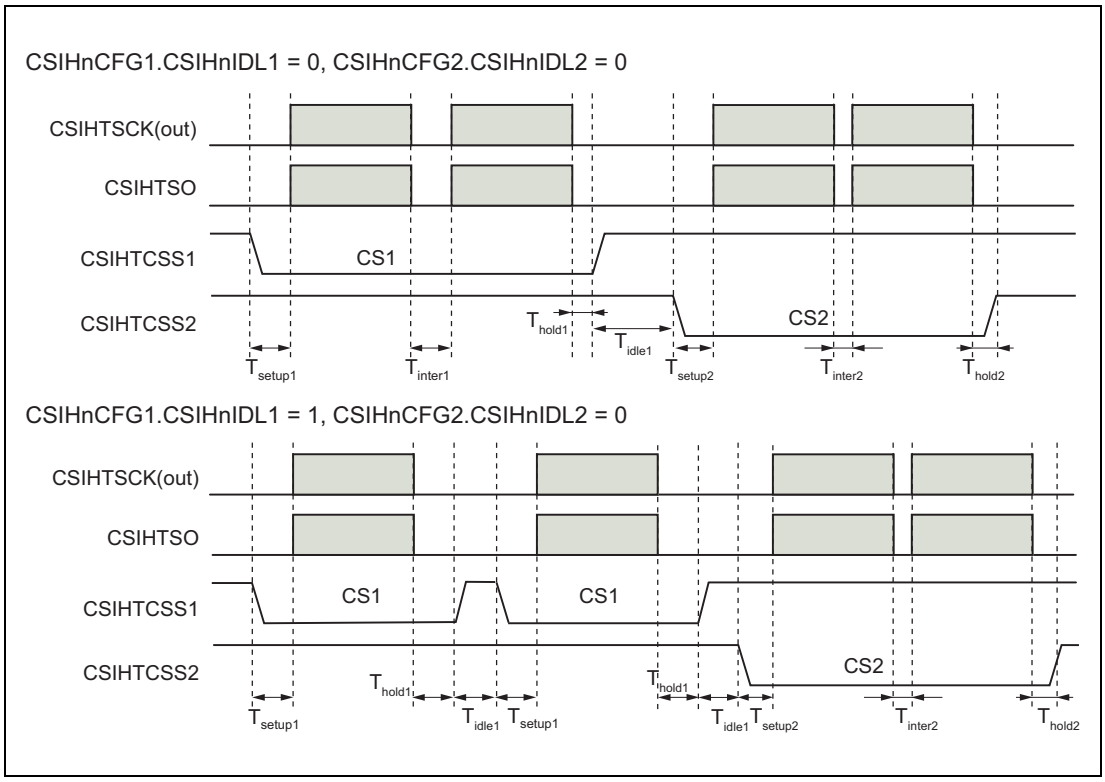


Figure 11.13 Chip Select Timings

Note that each CS signal can have a different value for setup, inter-data time, hold time, and idle time.

A particular chip select signal is activated by setting the appropriate bit in the transmission register CSIHnTX0W.CSIHnCSx.

CSIHnRX0W.CSIHnCSx in the reception register indicates the chip select signal associated with the received data.

CAUTION

If high priority communication function controlled by CPU is enabled (CSIHnCTL1.CSIHnPHE = 1), when mode is switched from low priority communication mode to high priority communication mode or from high priority communication mode to low priority communication mode, idle state is inserted regardless of the setting of the IDLx bit.

11.4.9.2 CS Example

The following figure shows an example of two consecutive data transmissions.

The first communication uses CS0 to communicate with one single slave. The second enables CS0 and CS1 to broadcast a message to two slaves. The priority of CS0 is set to “recessive: low priority”, the priority of CS1 to “dominant: high priority”. Therefore, the second communication uses CS1 set in dominant to communicate.

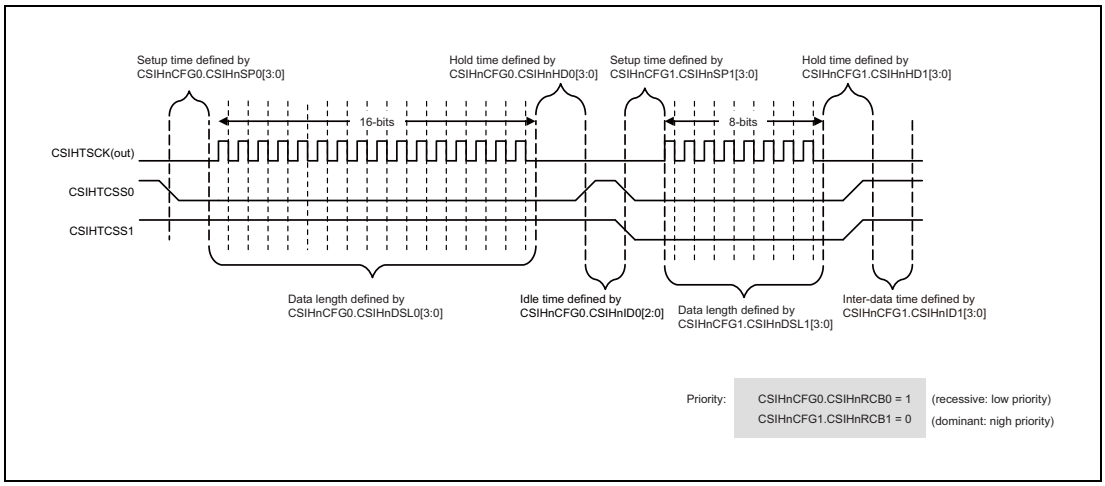


Figure 11.14 Chip Select and RCB Example

11.4.9.3 Job Concept

In terms of CSIH, a job consists of a number of data that are transferred.

Job Mode Enable

The job mode can only be enabled in master mode. The job mode is enabled and disabled by CSIHnCTL1.CSIHnJE, while the CSIH is disabled by CSIHnCTL0.CSIHnPWR = 0.

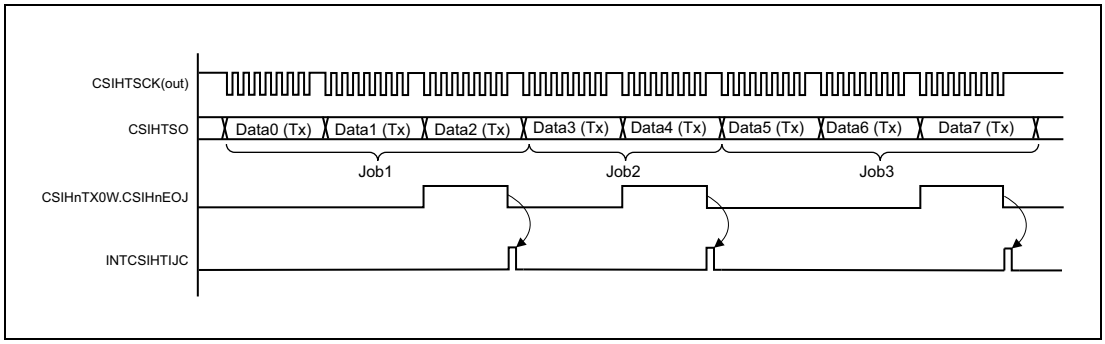


Figure 11.15 Job Examples

A job ends when data set with CSIHnTX0W.CSIHnEOJ = 1 is transmitted.

A communication stop can be specified to occur after a job has finished. This is done by setting CSIHnCTL0.CSIHnJOBE. When CSIHnJOBE is set, the communication continues until data is sent, for which the CSIHnEOJ bit was set. After this data is sent, the communication is stopped and the - interrupt INTCSIHTIJC is generated.

11.4.10 Chip Select Timing Details

11.4.10.1 Changing the Clock Phase

The serial clock level specified by CSIHnCFGx.CSIHnCKPx may be changed when communication is disabled. The minimum value of an idle time is one period of transmission clock (CSIHTSCK (out)).

If the idle time is set to 0.5 transmission clock periods (in CSIHnCFGx.CSIHnIDx[2:0]) and two consecutive data are sent with different CSIHnCFGx.CSIHnCKPx configuration, the idle time is extended to one period of CSIHTSCK (out).

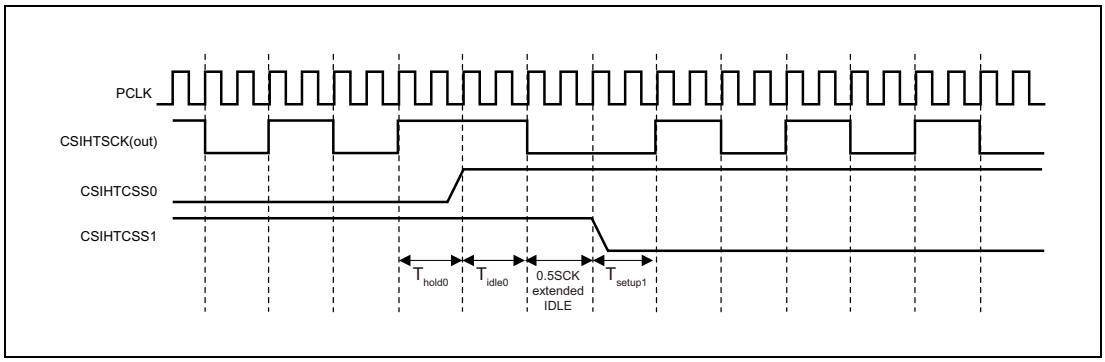


Figure 11.16 Clock Phase Timing
PCLK/4, $T_{hold}0 = T_{setup1}$ = 0.5CSIHTSCK, T_{idle0} = 0.5CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHTCSS1)

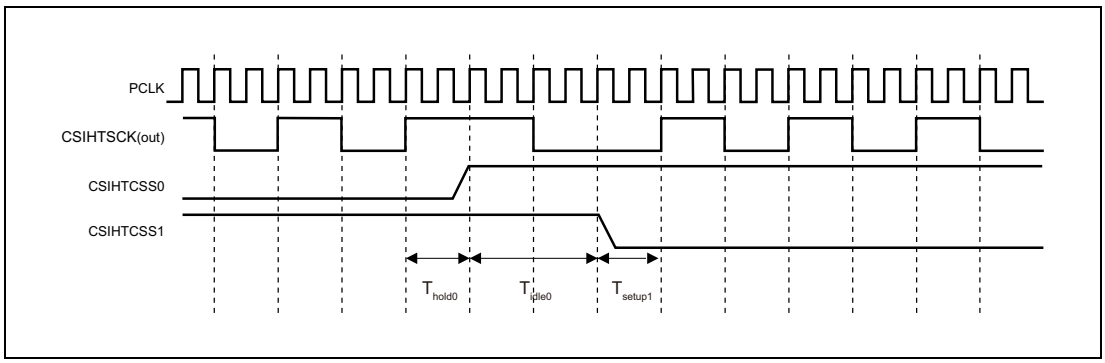


Figure 11.17 Clock Phase Timing
PCLK/4, $T_{hold}0 = T_{setup1}$ = 0.5CSIHTSCK, T_{idle0} = 1CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHTCSS1)

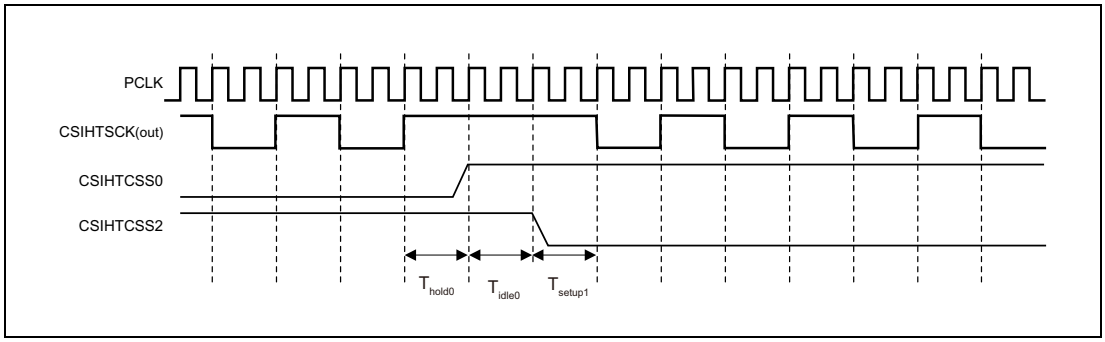


Figure 11.18 Clock Phase Timing
PCLK/4, $T_{hold}0 = T_{setup1} = 0.5$CSIHTSCK, $T_{idle0} = 0.5$CSIHTSCK, CSIHnCFG0.CSIHnCKP0 = 0 (CSIHTCSS0) → CSIHnCFG2.CSIHnCKP2 = 0 (CSIHTCSS2)

11.4.10.2 Changing the Data Phase

The CSIHnCFGx.CSIHnDAPx bit defines the phase of the data bits relative to the clock.

The value of the CSIHnCFGx.CSIHnDAPx bit affects the hold and setup times as described below.

The hold time is the time from the last edge of the serial clock (CSIHTSCK) to the point where CSIHTCSSx goes to the inactive level regardless of the CSIHnCFGx.CSIHnDAPx bit setting.

The setup time is the time from the point where CSIHTCSSx goes to the active level until output of the data for transmission (CSIHTSO).

Therefore, timing of the output of the next edge of the serial clock (CSIHTSCK) varies by 0.5 CSIHTSCK according to the setting of CSIHnCFGx.CSIHnDAPx.

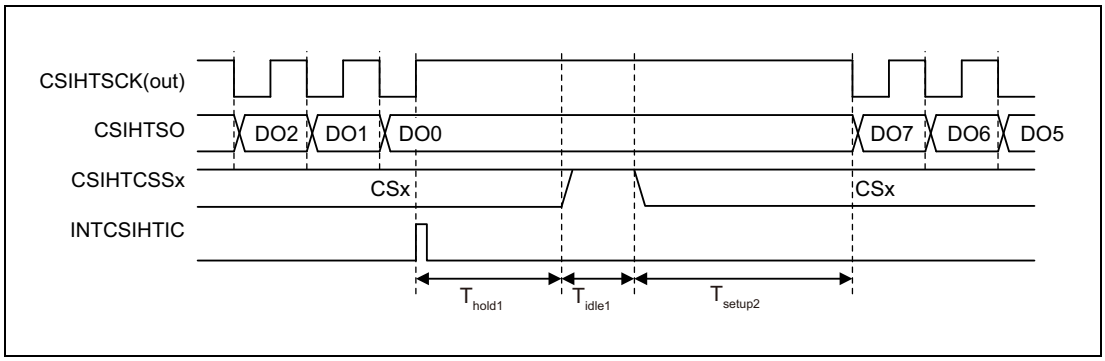


Figure 11.19 Data Phase Timing with
CSIHnCFG1.CSIHnCKP1 = 0, CSIHnCFG1.CSIHnDAP1 = 0 and CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 0

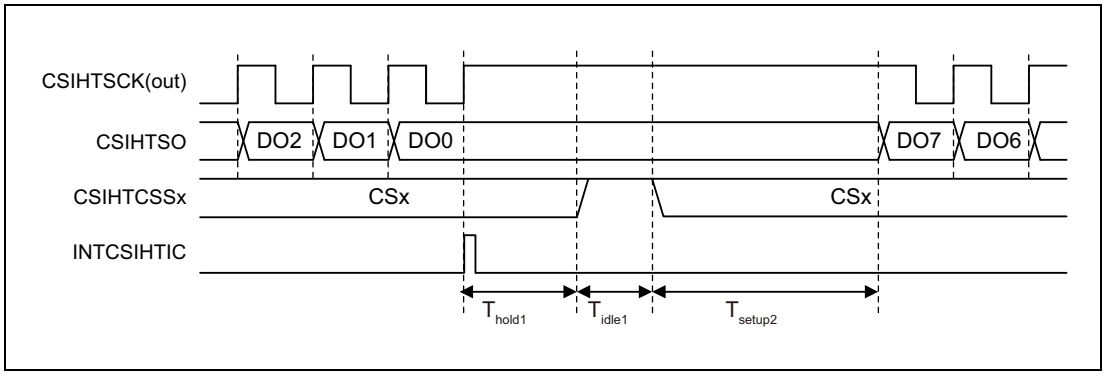


Figure 11.20 Data Phase Timing with CSIHnCFG1.CSIHnCKP1 = 1, CSIHnCFG1.CSIHnDAP1 = 0 and CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 1

11.4.11 Transmission Clock Selection

In master mode, the transfer clock frequency is selectable using the following bits:

- CSIHnCTL2.CSIHnPRS[2:0]
- CSIHnBRSy.CSIHnBRS[11:0]
- CSIHnCFGx.CSIHnBRSSx[1:0]

The transfer clock frequency of transmission clock CSIHTSCK is determined by the setting of the CSIHnCTL2.CSIHnPRS[2:0] bits and the setting of the CSIHnBRSy.CSIHnBRS[11:0] bits, but any one of CSIHnBRS3 to CSIHnBRS0 can be selected for each chip select signal with the CSIHnCFGx.CSIHnBRSSx[1:0] bits.

The following figure shows a block diagram of the baud-rate generator.

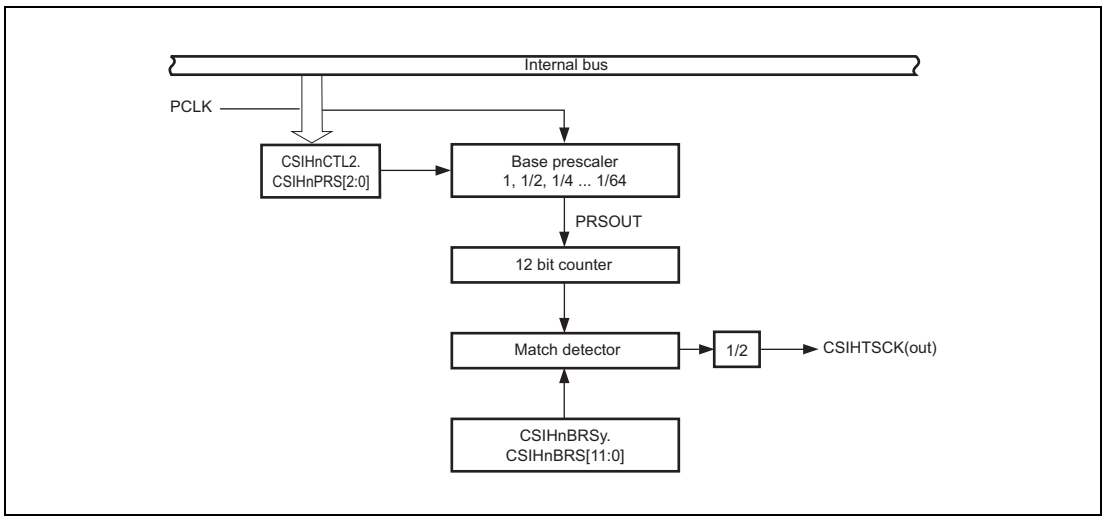


Figure 11.21 Baud Rate Generator Block Diagram

Setting CSIHnBRSy.CSIHnBRS[11:0] to 000_H disables the baud-rate generator, and thus all CSIHTSCK are stopped.

Transfer clock frequency calculation

The transfer clock frequency in master mode is calculated as:

Transfer clock frequency (CSIHTSCK) = PCLK / (division ratio of PCLK) = PCLK / ($2^{\alpha} \times k \times 2$),

where

α = CSIHnCTL2.CSIHnPRS[2:0] = 0 to 6

k = CSIHnBRS0.CSIHnBRS0[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 0)
CSIHnBRS1.CSIHnBRS1[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 1)
CSIHnBRS2.CSIHnBRS2[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 2)
CSIHnBRS3.CSIHnBRS3[11:0] = 1 to 4095 (when CSIHnCFGx.CSIHnBRSSx[1:0] = 3)

Transfer clock frequency upper and lower limits

When setting the transfer clock frequency, please note:

- The minimum transfer clock frequency in both master and slave mode is PCLK / 524160.
- The maximum transfer clock frequency is as below:
  - In master mode: PCLK / 8
  - In slave mode: PCLK / 16

11.4.12 CSIH Buffer Memory

The CSIH has a configurable RAM that can be used for buffered I/O. The size is 128 words. One word is comprised of 32 bits data plus 7 bits ECC.

The following configurations are available:

Mode	CSIHnCTL0. CSIHnMBS	CSIHnMCTL0. CSIHnMMS[1:0]
FIFO mode	0	00_B
Dual buffer mode		01_B
Transmit-only buffer mode		10_B
Direct access mode	1	X

11.4.12.1 FIFO Mode

In FIFO mode, data can be written to the CSIHnTX0W register without waiting for completion of the transmission, and data can be received without reading the CSIHnRX0W register immediately, provided the FIFO is not full.

Data to be transmitted is stored to the FIFO memory. Transmission and reception occur simultaneously – one data is sent, one data is received. That means, received data overwrites the transmitted data in the FIFO.

The CSIH automatically updates the respective FIFO memory pointers when a data is rewritten to, read from, sent to, or received from the FIFO memory:

Table 11.36 FIFO Mode

Pointer description	Control bit*1	Range
Number of unsent words	CSIHnSTR0.CSIHnSPF[7:0]	0 to 128
Number of words received and stored in the FIFO	CSIHnSTR0.CSIHnSRP[7:0]	0 to 128
Address of data to be written/read	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $01FC_H$
Address of received data to be read	CSIHnMRWP0.CSIHnRRA[6:0]	0000_H to $01FC_H$
Address to be sent	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $01FC_H$

Note 1. The value is automatically updated after each read/write or data transmit/receive operation.

The CSIH status register contains also two FIFO status flags:

- CSIHnSTR0.CSIHnFLF: FIFO full
- CSIHnSTR0.CSIHnEMF: FIFO empty

When this mode is started, bit CSIHnSTCR0.CSIHnPCT must be set. Only CSIHnSTR0.CSIHnEMF is set, but not reset.

All FIFO pointers and FIFO flags excluding CSIHnSTR0.CSIHnEMF are reset and CSIHnSTR0.CSIHnEMF is set.

11.4.12.2 Dual Buffer Mode

In this mode, the memory is divided into two parts of equal size – this means 64 words for transmit data and 64 words for received data. In dual buffer mode, the respective buffer pointers indicate the values shown in the following table.

Table 11.37 Dual Buffer Mode

Pointer description	Pointer*1	Range
Address for data written to or read from the transmit buffer	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $00FC_H$
Address of data read from the receive buffer	CSIHnMRWP0.CSIHnRRA[6:0]	0000_H to $00FC_H$
The number of transmit data remained in the transmit buffer	CSIHnMCTL2.CSIHnND[7:0]	0 to 64
Address to which data is sent to	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $00FC_H$

Note 1. Both pointers are automatically incremented after each read/write.

11.4.12.3 Transmit-only Buffer Mode

In this mode, the entire memory is used to save transmission data.

Received data must be read directly from CSIHnRX0W/H.

In transmit-only buffer mode, the respective buffer pointer indicates the values shown in the following table.

Table 11.38 Transmit-only Buffer Mode

Pointer description	Pointer*1	Range
Address for data written to or read from the transmit buffer	CSIHnMRWP0.CSIHnTRWA[6:0]	0000_H to $01FC_H$
The number of transmit data remained in the transmit buffer	CSIHnMCTL2.CSIHnND[7:0]	0 to 128
Address to which data is sent to	CSIHnMCTL2.CSIHnSOP[6:0]	0000_H to $01FC_H$

Note 1. The pointers are automatically incremented after each read/write.

11.4.12.4 Direct Access Mode

In direct access mode, the CSIH memory is completely bypassed:

- Transmission data provided by the CPU to the transmission register CSIHnTX0W or CSIHnTX0H is directly copied to the shift register.
- Reception data is directly copied from the shift register to the reception register CSIHnRX0W or CSIHnRX0H.

11.4.13 Data Transfer Modes

11.4.13.1 Transmit-only Mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 0 puts the CSIH in transmit-only mode. Start of transmission depends on the memory mode:

- In case of FIFO or direct access mode, transmission starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.
- In case of dual buffer or transmit-only buffer mode, transmission starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.2 Receive-only Mode

Setting CSIHnCTL0.CSIHnTXE = 0 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in receive-only mode.

In master mode, start of reception depends on the memory mode:

- In case of FIFO or direct access mode, reception starts when dummy data is written in the CSIHnTX0W or CSIHnTX0H register.

In slave mode, reception starts as soon as the transmission clock CSIHTSCK from the master is received. It is not necessary to write data to the CSIHnTX0W or CSIHnTX0H register of the slave.

- In case of dual buffer mode or transmit-only buffer mode, reception starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.3 Transmit/Receive Mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in transmit/receive mode.

Start of communication (transmission and reception) depends on the memory mode:

- In case of FIFO or direct access mode, communication starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.
- In case of dual buffer or transmit-only buffer mode, communication starts when the CSIHnMCTL2.CSIHnBTST bit is set.

11.4.13.4 Summary

The following table summarizes this section. It shows how data transfer is started in the various memory, operating, and transfer modes.

Table 11.39 Start of Data Transfer

Memory and operating mode		Transfer mode	
		Transmit-only Transmit/receive	Receive-only
FIFO, direct access	Master	Writing to the CSIHnTX0W register or the CSIHnTX0H register	Writing to the CSIHnTX0W register or the CSIHnTX0H register
	Slave	Reception of clock from the master	Incoming clock from the master
Transmit-only buffer, dual buffer	Master	CSIHnMCTL2.CSIHnBTST = 1	CSIHnMCTL2.CSIHnBTST = 1
	Slave	Reception of clock from the master	Incoming clock from the master

11.4.14 Data Length Selection

11.4.14.1 Data Length between 2 and 16 Bits

The length of a data packet is selectable for each chip select signal from 2 to 16 bits by using CSIHnCFGx.CSIHnDLSx[3:0]. The examples below show the communication with MSB first (CSIHnCFGx.CSIHnDIRx = 0).

Data length = 16 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 0000_B)

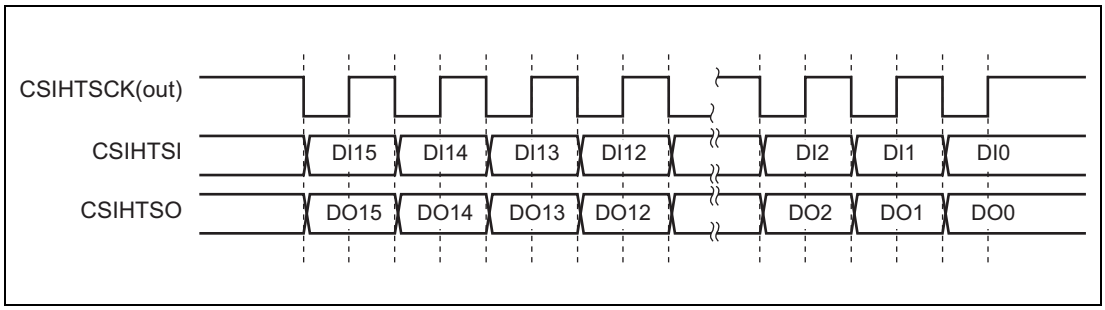


Figure 11.22 16 Bit Data Length, MSB First

Data length = 14 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1110_B):

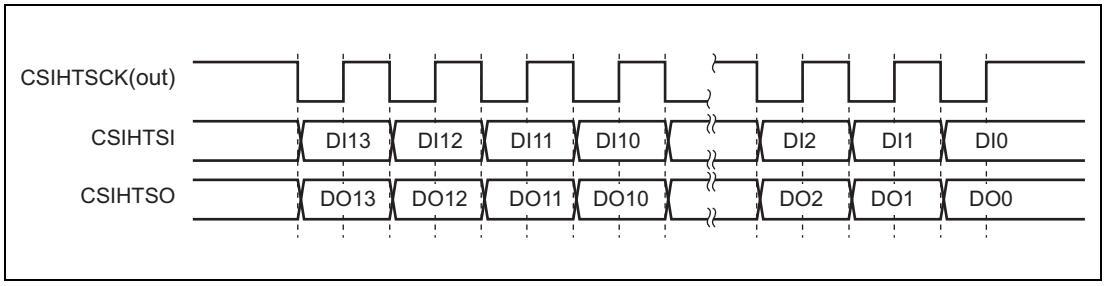


Figure 11.23 14 Bit Data Length, MSB First

11.4.14.2 Data Length Greater than 16 Bits

If the data to be sent/received exceeds 16 bits, the extended data length (EDL) feature can be used.

EDL function is enabled by setting the CSIHnCTL1.CSIHnEDLE bit to 1.

EDL function works as follows:

- The data has to be broken into 16-bit blocks plus remainder. For example, 42-bit data would be broken into two 16-bit blocks plus 10 bits.
- The bit length of the remainder is set as “data length” in CSIHnCFGx.CSIHnDLSx[3:0].
- For transmitting the 16-bit blocks, CSIHnTX0W.CSIHnEDL must be set to 1. In this case, the data written to CSIHnTX0W is sent as a 16-bit data length regardless of the CSIHnCFGx.CSIHnDLSx[3:0] bit setting.
- The transfer is complete after a block with the specified data length (the remainder of data specified with CSIHnTX0W.CSIHnEDL = 0) has been sent.

Example

Example for sending 40-bit data ($123456789A_H$) to CS0:

40 bits are split into 2 to 16 bits plus 8 bits.

- Initialize CSIHnCFG0.CSIHnDLS0[3:0] = 8_D.
- To send $123456789A_H$ with MSB first, write the following sequence to CSIHnTX0W:
  - $20FE\ 1234_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $20FE\ 5678_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $00FE\ 009A_H$ (CSIHnTX0W.CSIHnEDL = 0)

The following figure illustrates the timing.

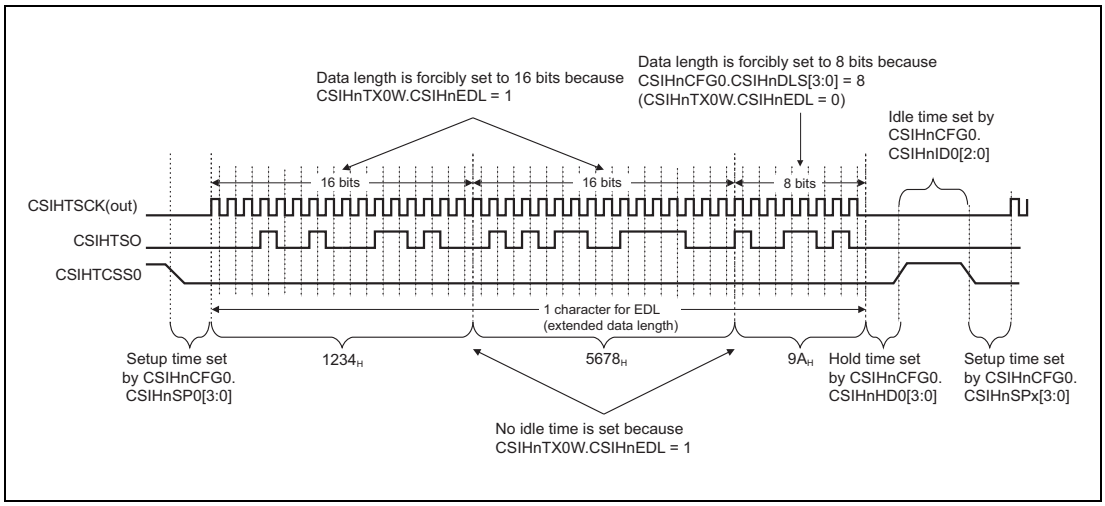


Figure 11.24 EDL Timing Diagram

NOTES

1. A data length of 1 bit can only be selected when EDL mode is in use.
2. It is not possible to send two consecutive data with a data length of less than 2 bits.
3. If parity is enabled, the parity bit is added after the last bit.
4. When extended data length (EDL) function is used, use the same chip selection signal.
5. To consider the data direction, pay attention to the following example:
   - Data to be sent: 123456_H
   - MSB first:
     Set CSIHnCFGx.CSIHnDIRx = 0
     Write CSIHnTX0W = 20FE 1234_H (EDL bit = 1)
     Write CSIHnTX0W = 00FE 0056_H (EDL bit = 0)
   - LSB first:
     Set CSIHnCFGx.CSIHnDIRx = 1
     Write CSIHnTX0W = 20FE 3456_H (EDL bit = 1)
     Write CSIHnTX0W = 00FE 0012_H (EDL bit = 0)
6. If CSIHnTX0W.CSIHnEOJ = 1 and CSIHnTX0W.CSIHnEDL = 1 are set simultaneously while CSIHnCTL1.CSIHnJE = 1 and CSIHnCTL1.CSIHnEDLE = 1, the behavior and performance are not guaranteed.
7. EDL mode cannot be used in receive-only mode of slave mode. (CSIHnCTL2.CSIHnPRS[2:0] = 111_B, CSIHnCTL0.CSIHnTXE = 0, CSIHnCTL0.CSIHnRXE = 1)

11.4.15 Serial Data Direction Selection

The serial data direction is selectable for each chip select signal by using the CSIHnDIRx bit in the CSIHnCFGx register.

The examples below show communication for a data length of 8 bit (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).

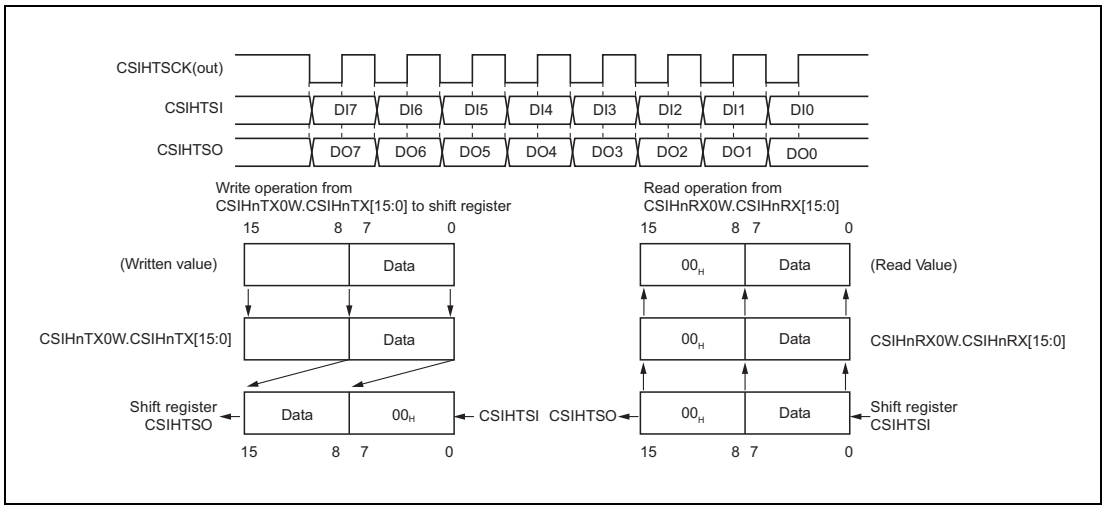


Figure 11.25 Serial Data Direction Select Function - MSB First (CSIHnDIRx = 0)

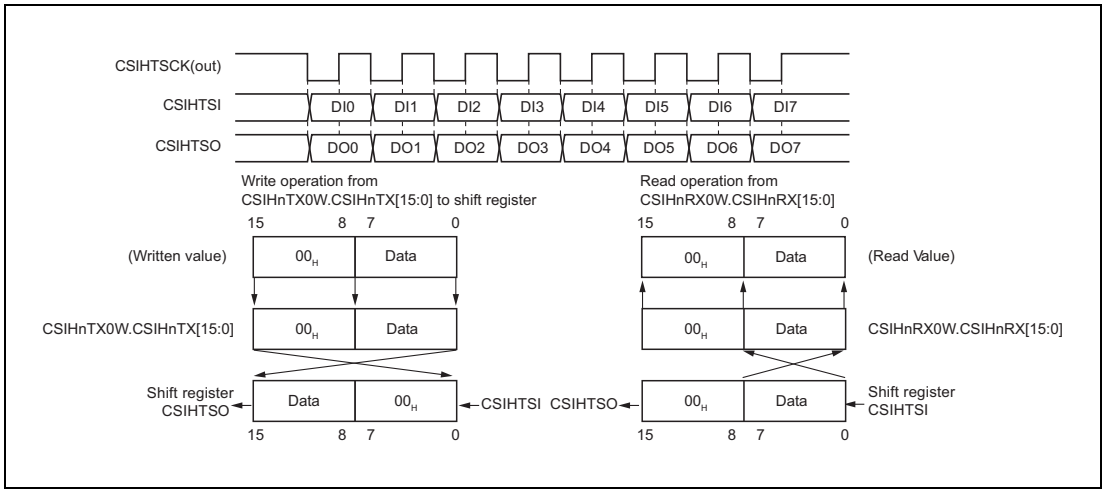


Figure 11.26 Serial Data Direction Select Function - LSB First (CSIHnDIRx = 1)

11.4.16 SS (Slave Select) Function

The SS function realizes communication between one master and multiple slaves.

In master mode, the master device outputs the slave select signal (CSIHTCSSx) to select a single slave. Communication by a device in slave mode is enabled when the slave input select signal (CSIHTSSI) is at the low level.

Refer to the **Section 11.4.8, Master/Slave Connections**, for an example of a connection using the SS function.

11.4.16.1 Communication Timing Using SS Function

The following figure illustrates the communication signal using the SS function and timings.

In slave mode, the data transfer configuration is determined by the CSIHnCFG0 register.

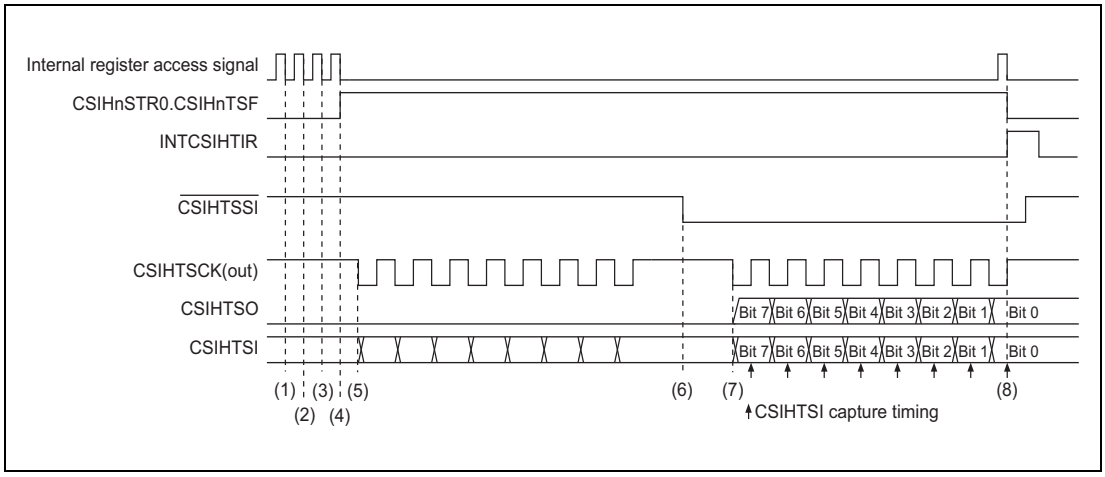


Figure 11.27 Tx/Rx Timing of Communication Using SS Function

(1) CSIH is put into slave mode by setting CSIHnCTL2.CSIHnPRS[2:0] = 111_B. CSIHnCFG0.CSIHnCKP0 and CSIHnCFG0.CSIHnDAP0 are 0.

(2) The data length is 8 bits (CSIHnCFG0.CSIHnDLS0[3:0] = 1000_B). The data direction is MSB first (CSIHnCFG0.CSIHnDIR0 = 0).

(3) The transmit/receive mode is set (CSIHnCTL0.CSIHnTXE = 1, CSIHnCTL0.CSIHnRXE = 1, CSIHnCTL0.CSIHnPWR = 1). Communication start is permitted.

(4) The transfer status flag CSIHnSTR0.CSIHnTSF is automatically set when transfer data is written to the CSIHnTX0W or CSIHnTX0H transmission register during direct access mode or FIFO mode.

(5) As long as signal $\overline{\text{CSIHTSSI}}$ is high, transmission/reception is not started, even if an external transmission clock CSIHTSCK is applied. Input at CSIHTSI is ignored.

(6) When $\overline{\text{CSIHTSSI}}$ falls to low level, indicating that CSIHTSO is enabled, transmission is enabled.

(7) Now, as soon as the external clock signal CSIHTSCK appears, the slave transmits data to CSIHTSO and simultaneously captures data from CSIHTSI.

(8) Interrupt INTCSIHTIR indicates when the reception is complete. The CSIHnRX0W/H register can be read.

11.4.16.2 CSIHTSSO Operation

CSIHnPWR	CSIHnTXE	CSIHnRXE	CSIHnSSE	CSIHTSSO
0	—	—	—	H
1	—	—	0	H
	0		1	H
	1		1	Reversed value of $\overline{\text{CSIHTSSI}}$ level

The CSIHTSSO pin is a signal to control the I/O function of the chip's SO pin in case of using the SS function.

The CSIHTSO pin is enabled when the CSIHTSSO pin is "High" (the chip's SO pin is being driven).

The CSIHTSO pin is disabled when the CSIHTSSO pin is "Low" (the chip's SO pin is not being driven).

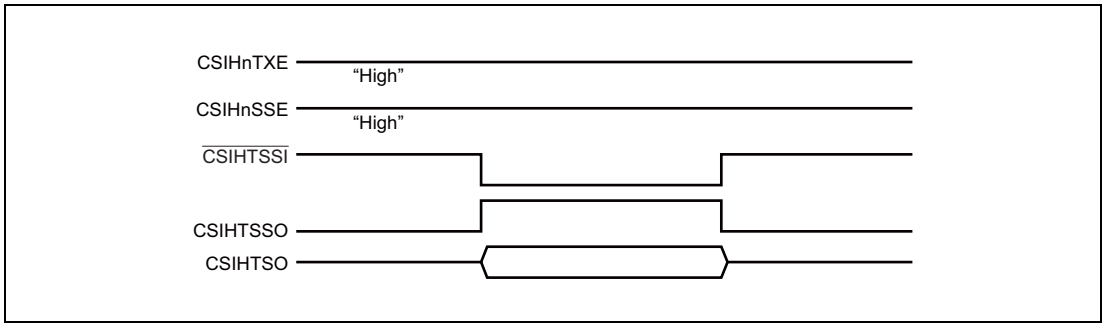


Figure 11.28 Operation of CSIHTSSO

CAUTION

If $\overline{\text{CSIHTSSI}}$ pin is changed during communication (CSIHnSTR0.CSIHnTSF = 1), current communication is not assured.

11.4.17 Handshake Function

CSIH features a handshake function to synchronize the master and the slave devices. This function can be enabled/disabled by the CSIHnCTL1.CSIHnHSE bit. For handshake, the signals CSIHTRYI and CSIHTRYO are used.

The busy timing depends on the data phase selection bit CSIHnCFGx.CSIHnDAPx.

11.4.17.1 Slave Mode

When CSIHnCTL1.CSIHnHSE = 1 and the slave is busy, the CSIHTRYO signal outputs low level. This can happen in two cases:

1. When the next data to be sent is not ready:
   When the slave is in transmit-only mode or transmit/receive mode (CSIHnCTL0.CSIHnTXE = 1) and is in the states listed below, the CSIHTRYO output indicates the busy state (is at the low level).

Table 11.40 Memory Mode and Slave Transfer State

Memory Mode	Transmission State of Slave
Direct access mode	No data for the next transfer
FIFO mode	No data for the next transfer (CSIHnSTR0.CSIHnEMF = 1)
Dual buffer mode	When CSIHnMCTL2.CSIHnBTST is not set to 1
Transmit-only buffer mode	

The example below is on the assumption of an eight-bit data length.

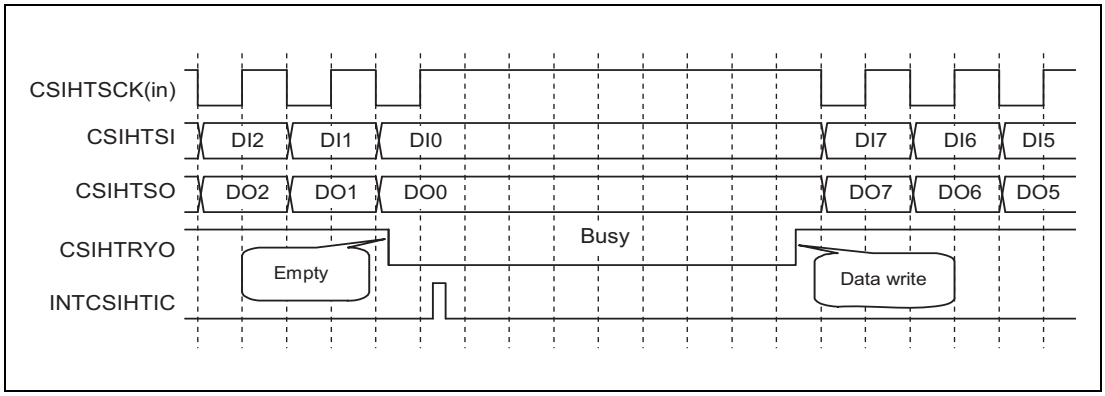


Figure 11.29 Busy Signal from the Slave (FIFO mode; CSIHnCFGx.CSIHnDAPx = 0)

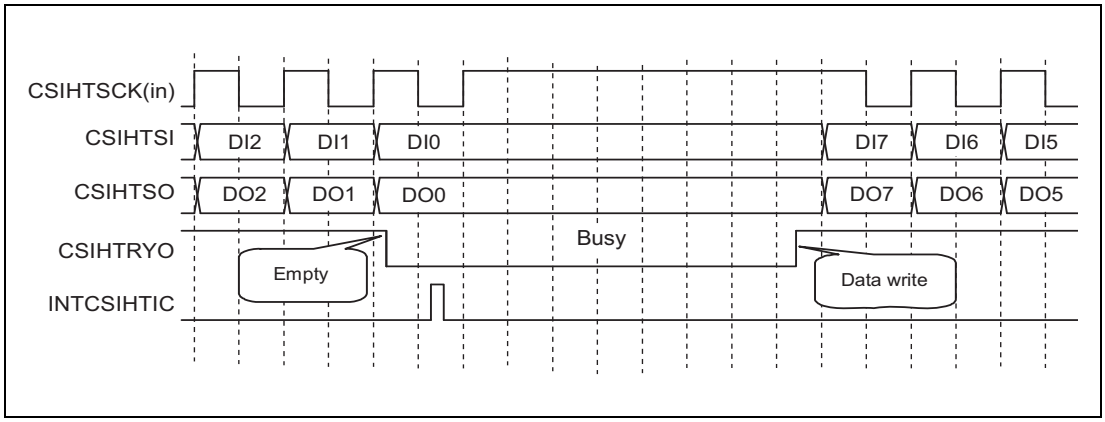


Figure 11.30 Busy Signal from the Slave (FIFO mode; CSIHnCFGx.CSIHnDAPx = 1)

2. When transmit register is full:
   When slave is set in receive-only mode or transmit/receive mode (CSIHnCTL0.CSIHnRXE = 1), and new data cannot be copied from a shift register to CSIHnRX0 (CSIHnRX0 is full) because the previously received data is still in the CSIHnRX0 register. When CSIHnCTL0.CSIHnRXE is 1 and is in the following states, CSIHTRYO outputs busy state (low level).

Table 11.41 Memory Mode and Slave Transfer State

Memory Mode	Transmission State of Slave
Direct access mode	When CSIHnRX0W or CSIHnRX0H is full
FIFO mode	Receive data remains in the buffer (when CSIHnSTR0.CSIHnFLF = 1)
Dual buffer mode	No applicable state
Transmit-only buffer mode	When CSIHnRX0W or CSIHnRX0H is full

The example below is on the assumption of an eight-bit data length.

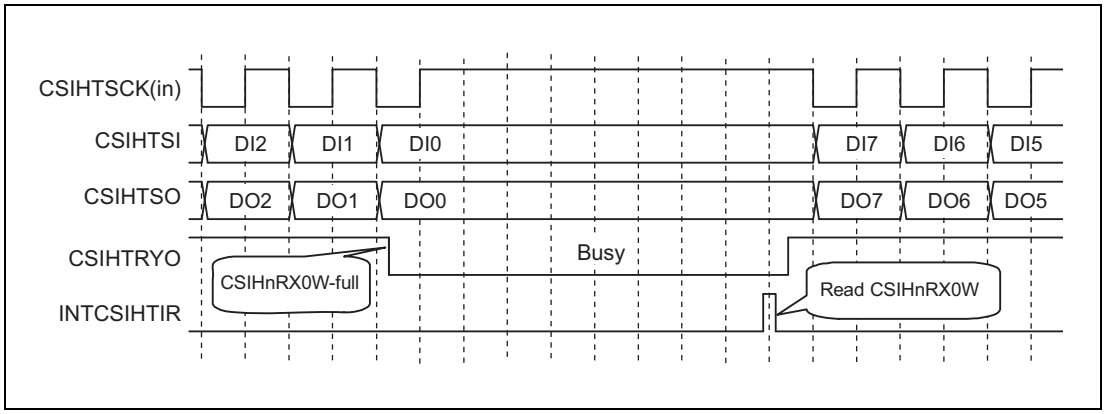


Figure 11.31 Busy Signal from the Slave (Direct access mode; CSIHnCFGx.CSIHnDAPx = 0)

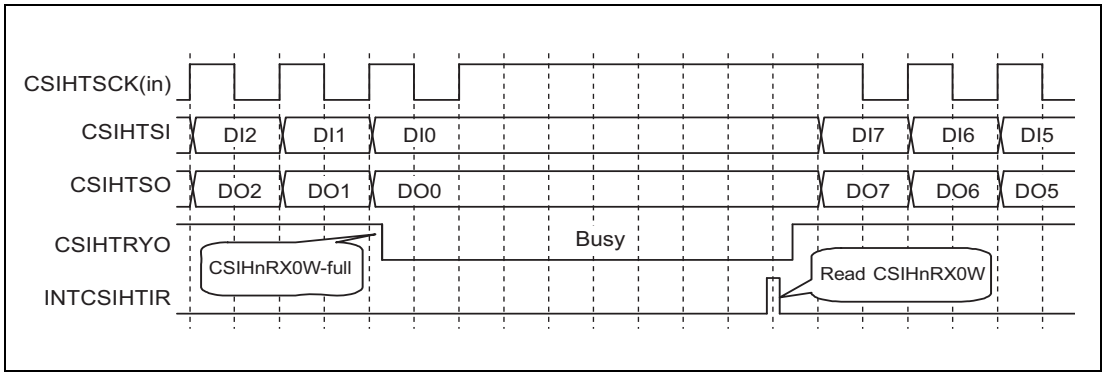


Figure 11.32 Busy Signal from the Slave (CSIHnCFGx.CSIHnDAPx = 1)

11.4.17.2 Master Mode

When the master detects CSIHTRYI = 0 while CSIHnCTL1.CSIHnHSE = 1, the following transfer is put on hold, and the master goes into wait status. It suspends the CSIHTSCK clock.

The CSIHTRYI level is checked at each half clock cycle of CSIHTSCK.

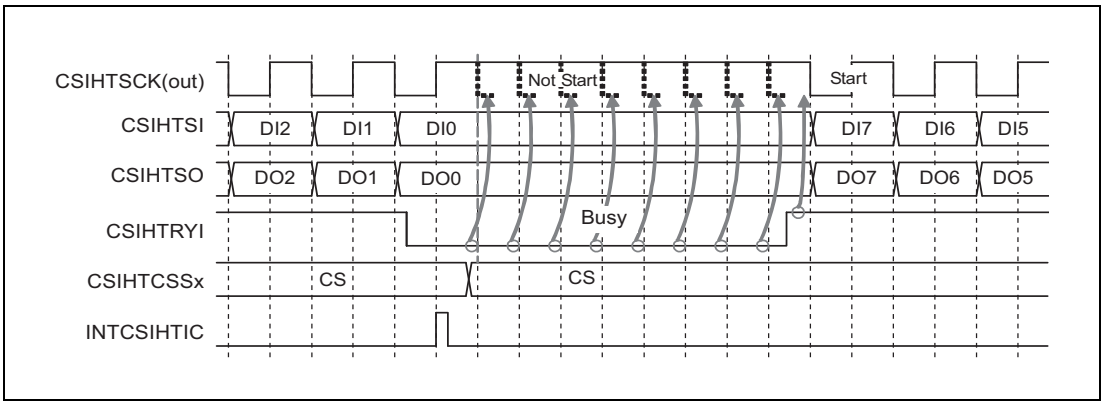


Figure 11.33 Master's Reaction on CSIHTSHSG (CSIHnCFGx.CSIHnDAPx = 0)

The CSIHTRYI signal must be pulled down by the slave before the next transfer starts. If this is done on the slave while data transfer is in progress, the serial clock from the master is suspended after the transfer is complete.

The master resumes the communication as soon as CSIHTRYI becomes high level (the slave is "ready").

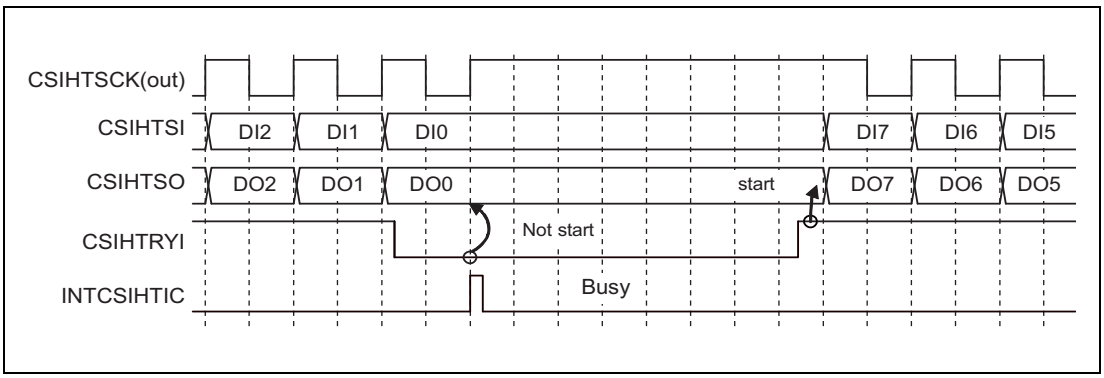


Figure 11.34 Master's Reaction on CSIHTRYI (CSIHnCFGx.CSIHnDAPx = 1)

CAUTIONS

1. **If multiple slaves are connected, the master must only detect the CSIHTRYI signal from the slave it has selected for communication.**
2. **Even when the CSIHTRYI pin of the master detects a CSIHTRYO signal from the slave during data transfer, the communication is not made to wait but continues until the data transfer is completed.**

11.4.18 Error Detection

CSIH can detect five error types:

- Data consistency error (transmission data)
- Parity error (received data)
- Overrun error (received data)
- Time-out error (in FIFO mode)
- Overflow error (in FIFO mode)

Check for parity, data consistency and time-out errors can be enabled/disabled individually.

If one of these errors is detected, the interrupt request, INTCSIHTIRE is generated and the corresponding flags are set.

11.4.18.1 Data Consistency Check

The purpose of the data consistency check is to ensure that the data physically sent as output signal is identical to the original data that was copied to the shift register.

The data consistency check can be enabled/disabled by the CSIHnCTL1.CSIHnDCS bit. When checking data consistency, always set the PIPCn.PIPCn_m to 1. It is not active if data transmission is disabled (CSIHnCTL0.CSIHnTXE = 0).

When the data consistency check is active, the data transferred from CSIHnTX0W or CSIHnTX0H to the shift register is copied to a separate register. In addition, the physical levels at CSIHTSO are read back via the CSIHTDCS signal into an own shift register.

After completion of the transmission, the sent data is compared with the original transmission data.

Mismatch is considered as a data consistency error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnDCE bit is set.

Additionally, CSIHnRX0W.CSIHnTDCE of data that contains the error is set.

The data consistency check function is illustrated in the following block diagram.

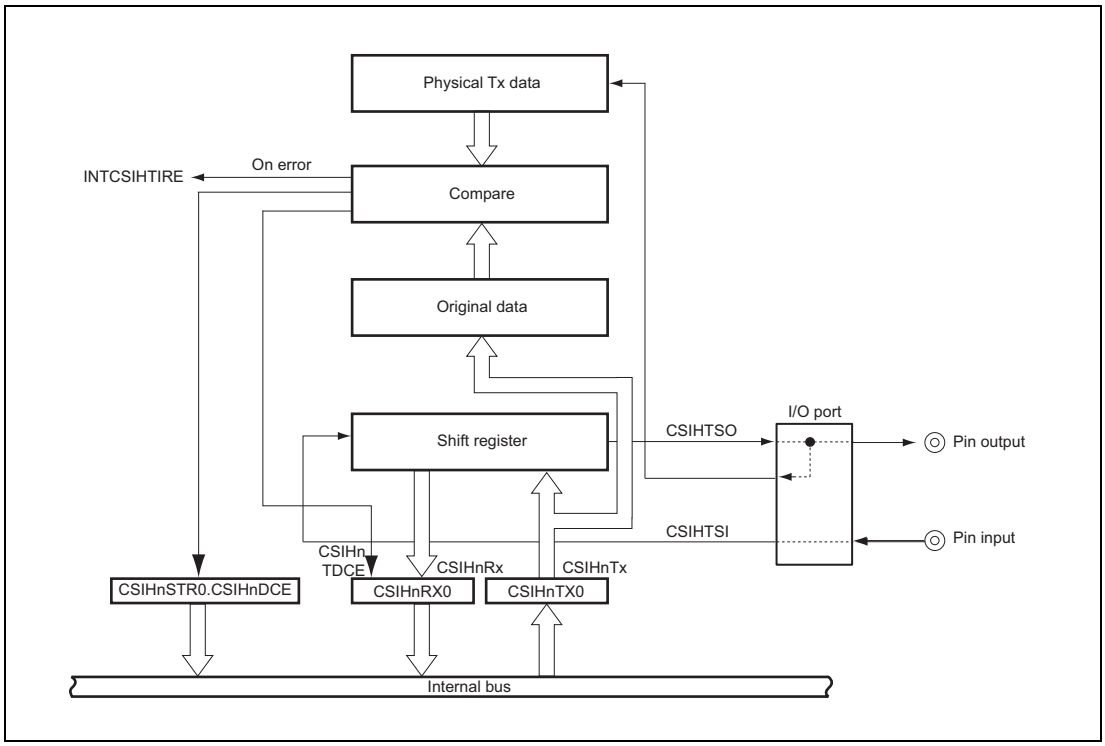


Figure 11.35 Data Consistency Check Functional Block Diagram

11.4.18.2 Parity Check

CSIH can append a parity bit to the last data bit (even if extended data length is used).

The use and type of parity is specified in CSIHnCFGx.CSIHnPSx[1:0].

Parity check is enabled if CSIHnCFGx.CSIHnPSx[1] = 1.

The parity bit is checked after a reception is complete. In case of parity error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnPE bit is set.

Additionally, CSIHnRX0W.CSIHnRPE of data that contains the error is set.

The figure below shows an example.

- Data length is 8 bits.
- The data to be transmitted is 05_H and 35_H.
- Data direction is LSB first.
- Parity type is odd.

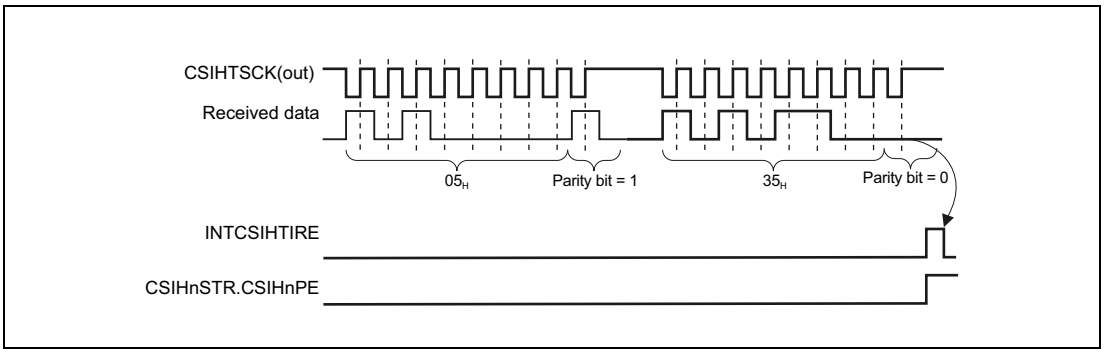


Figure 11.36 Parity Check Example

The parity bit of the first data is 1. There is no parity error, because the total number of ones (including the parity bit) is odd.

The parity bit of the second data is 0. This is detected as a parity error, because the total number of ones (including the parity bit) is even.

If the EDL (extended data length) function is used, the parity bit is added after the last bit of the data.

11.4.18.3 Time-out Error

Time-out errors can be checked only in slave FIFO mode.

This error occurs if neither of the following occurred within a certain period of time:

- Received data in FIFO is read
- FIFO receives data from CSIHTSI

The time is defined in CSIHnMCTL0.CSIHnTO[4:0] in multiples of 8 times the transmission clock, CSIHTSCK. A time-out error occurs when the specified time is exceeded (The time-out time is not detected when CSIHnMCTL0.CSIHnTO[4:0] = 00000_B).

A dedicated time-out counter measures the time between the last and the next read operation.

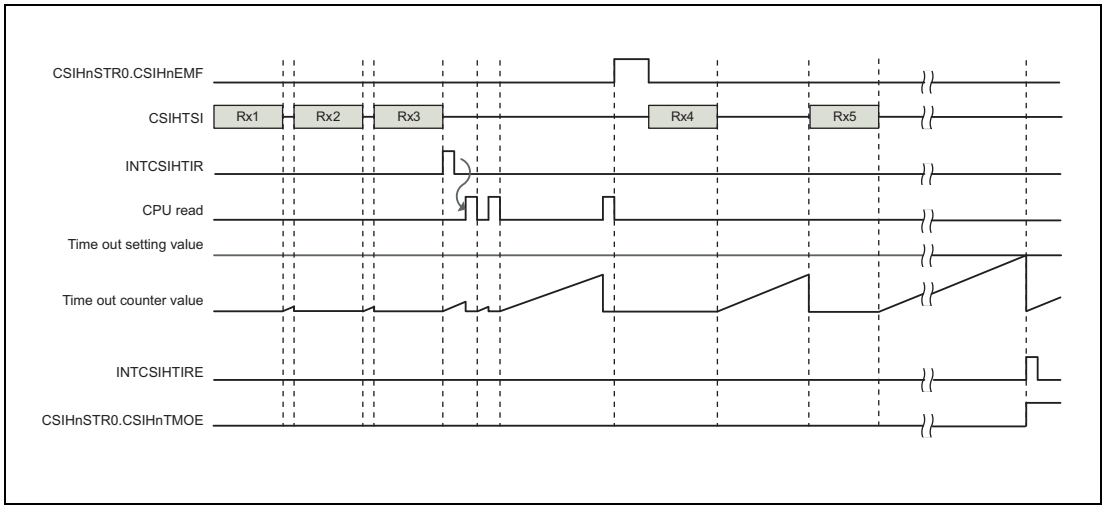


Figure 11.37 Time-out Check Functional Timing Diagram

The start timing of the time-out counter is as follows:

- When reception is completed
- When data read from the CPU completes
  (The counter does not start if the buffer is empty.)
- When a time-out error is detected

After a time-out error is detected, if data is still available in FIFO, the time-out counter restarts.

If the value set by bit CSIHnMCTL0.CSIHnTO[4:0] is reached again, the INTCSIHTIRE interrupt is output again.

The timeout counter continues to count until received data is read. To stop the counter, read all received data or set CSIHnSTCR0.CSIHnPCT to 1. Note that the pointer is cleared if you perform the latter.

The counter is reset at the following timing:

- Data is read once.
- New data is received.
- A timeout error is detected.
- The CSIHnSTCR0.CSIHnPCT bit is set to 1.

If a timeout error occurs, the following occurs:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnTMOE bit is set.

The dedicated time-out counter is set by the CSIHnCTL2.CSIHnPRS[2:0] and CSIHnBRSy.CSIHnBRS[11:0] bits. If the value of the CSIHnBRSy.CSIHnBRS[11:0] bits is left as 000_H, the dedicated time-out counter does not operate.

11.4.18.4 Overflow Error

An overflow error can happen in FIFO mode. It occurs when transmission data is written to the CSIHnTX0W register while the FIFO buffer is filled with received data.

Example

100 data have been transmitted. That means, the FIFO contains 100 received data. The application starts to read the received data.

While the read operation is in progress, the application begins to write another set of 50 transmission data to the FIFO. However, only 10 received data have been read up to now, 90 are still in the FIFO.

In this case, only 38 cells are available for new transmission data packets. When the CPU tries to write the 39th data, an overflow error happens.

This is illustrated in the following figure.

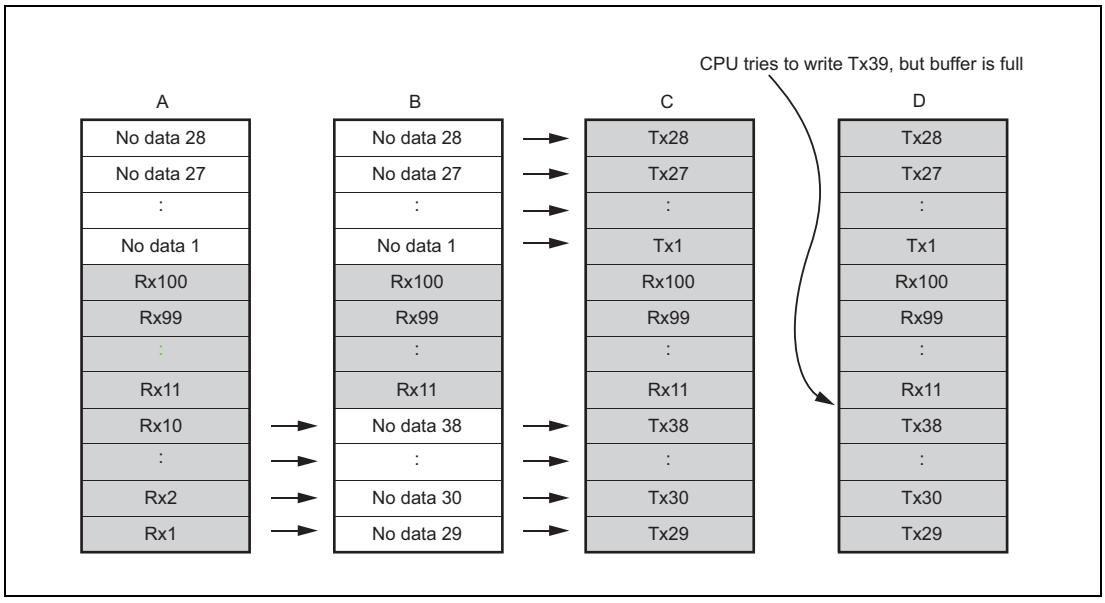


Figure 11.38 FIFO Overview

The data after 39 are discarded. The figure below shows the overflow timing.

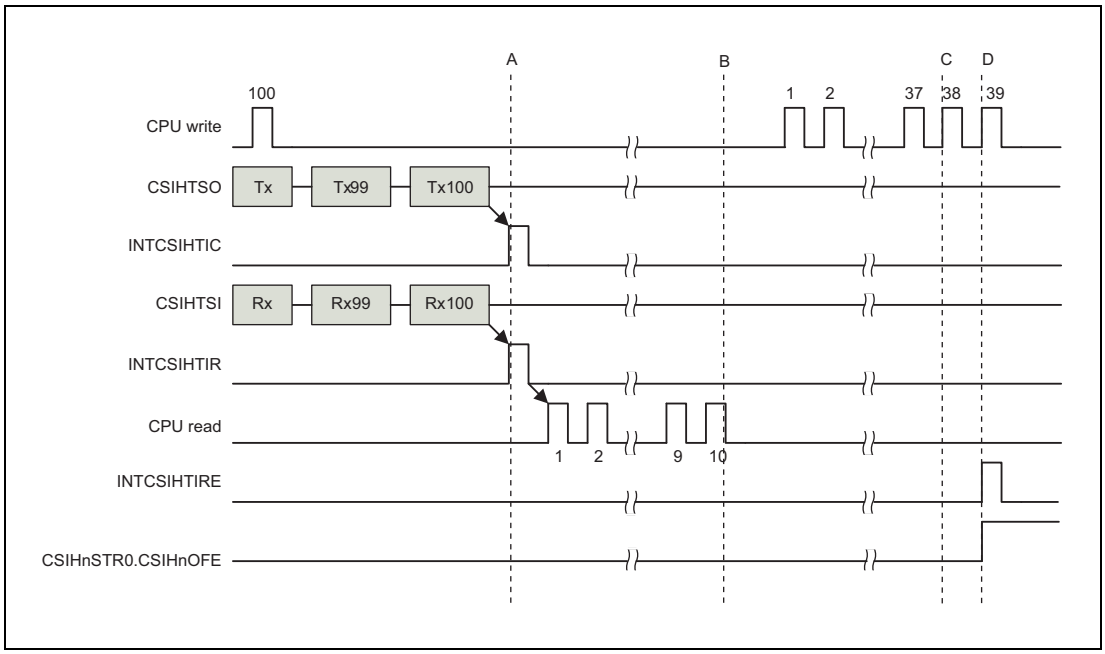


Figure 11.39 FIFO Overflow Timing

In case of overflow error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnOFE bit is set.

11.4.18.5 Overrun Error

An overrun error can happen in direct access, transmit-only buffer, and FIFO modes. It cannot happen in dual buffer mode. The overrun error is not generated if data reception is disabled (CSIHnCTL0.CSIHnRXE = 0).

There are two conditions for overrun errors.

Condition for errors 1

- In FIFO mode, while the number of received data is 0 and CPU reads the CSIHnRX0W/H register

Condition for errors 2

- In slave mode, when CSIHnCTL1.CSIHnHSE = 0 (handshake function disabled):
  - In direct access mode or transmit-only buffer mode, when reception is completed while the previous received data is remained in the CSIHnRX0W/H register.
  - In FIFO mode, when FIFO buffer completes receiving data in the full state.

(1) Direct Access/Transmit-only Buffer

In direct access and transmit-only buffer mode, this error occurs when newly received data cannot be transferred from the shift register to the reception register CSIHnRX0. This happens when CSIHnRX0 was not read and therefore contains previous reception data.

The following figure illustrates the function.

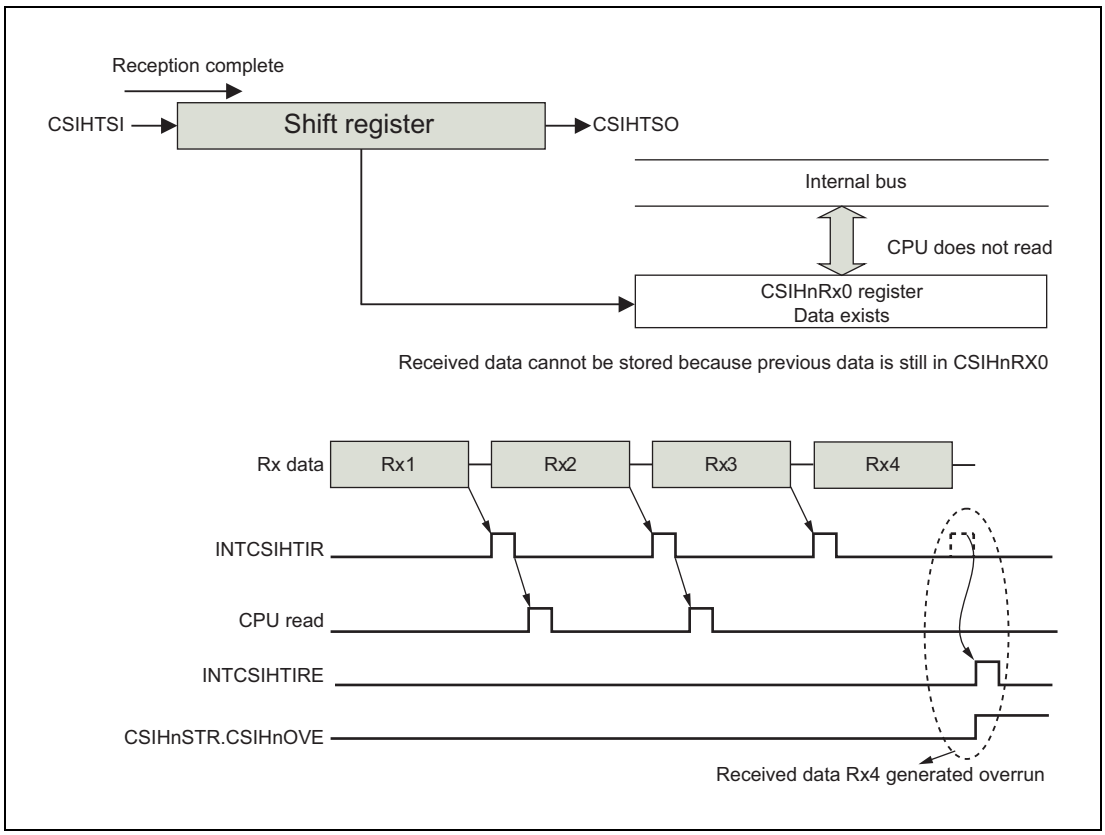


Figure 11.40 Overrun Error Detection in Direct Access and Transmit-only Buffer Mode

NOTE

An overrun error can be avoided in slave mode by using the handshake function.
When handshake is used in slave mode, the receiver (slave) signals to the transmitter (master) that it is busy. The transmitter then waits until the receiver has read its reception register and is ready again.

(2) FIFO Mode

In FIFO mode, this error occurs if:

1. Newly received data cannot be transferred from the shift register to the FIFO because the FIFO is full.

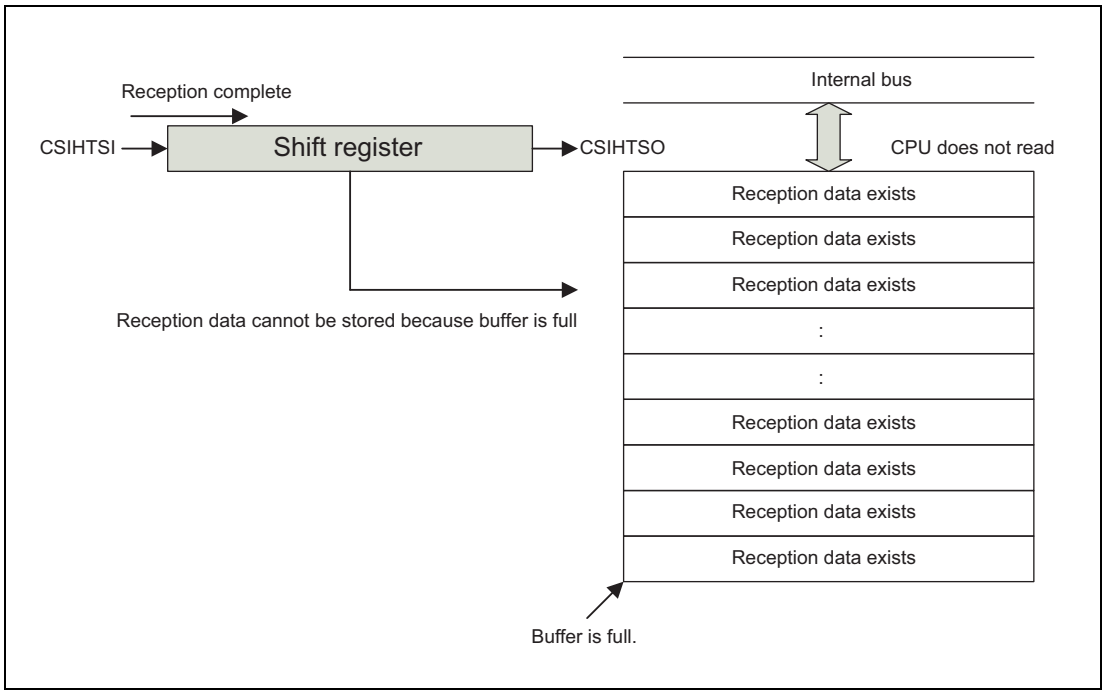


Figure 11.41 Overrun Error Detection in FIFO Mode (FIFO full)

NOTE

An overrun error can be avoided in slave mode by using the handshake function.
When handshake is used in slave mode, the receiver (slave) signals to the transmitter (master) that it is busy. The transmitter then waits until the receiver has read its reception register and is ready again.

2. The CPU attempts to read non existing reception data.

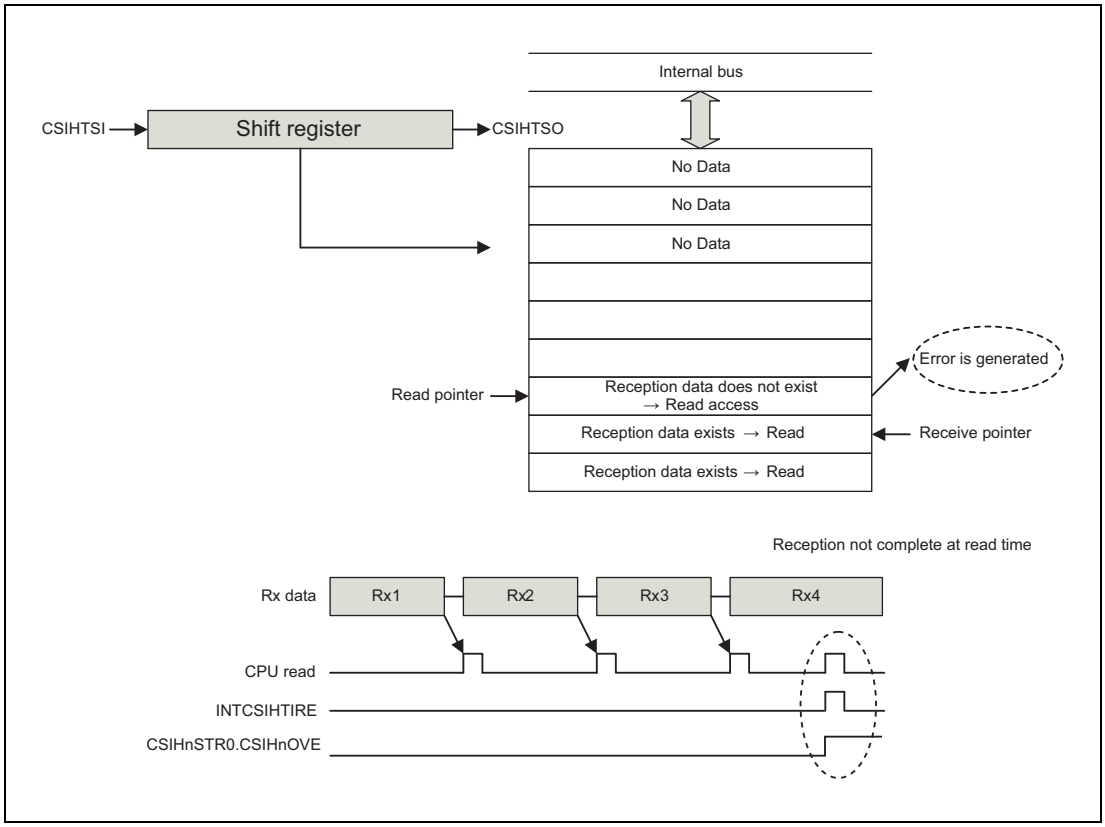


Figure 11.42 Overrun Error Detection in FIFO Mode (no data)

In case of overrun error:

- Interrupt INTCSIHTIRE is generated.
- The CSIHnSTR0.CSIHnOVE bit is set.
- Received data is overwritten and the communication continues.
  (When the CPU tries to read non-existent data, the CPU starts reading again after a wait until reception is completed.)

For details see **Section 11.4.17, Handshake Function**.

11.4.19 Loop-back Mode

Loop-back mode is a special mode for self-test. This feature is only available in master mode.

When this mode is active (CSIHnCTL1.CSIHnLBM = 1), CSIHTCSSx is fixed to the inactive level (the active level is defined by the CSIHnCTL1.CSIHnCSLx value). The transmit and receive signals are internally connected, as shown in the figures below. The signals CSIHTSCK, CSIHTSO, CSIHTSI, and CSIHTCSSx are disconnected from the ports. In addition, the CSIHTSO output level is fixed to low, and CSIHTSCK is set to reset level (High) regardless of the value of the CSIHnCFGx.CSIHnCKPx. The rest of CSIH works as in normal operation. Loop-back test has no effect on the device connected to.

In order to test CSIH, put it in loop-back mode and carry out normal transfer operations. Then check that the received data is the same as the transmitted data.

Table 11.42 Pin Output Level in Loop-back Mode

Pin Name	Output Level
CSIHTSCK (out)	High level
CSIHTCSS[x]	Inactive level
CSIHTSO	Low level (does not depend on the previous values)
Interrupt	Normal function
CSIHTRYO	Normal function (low level)

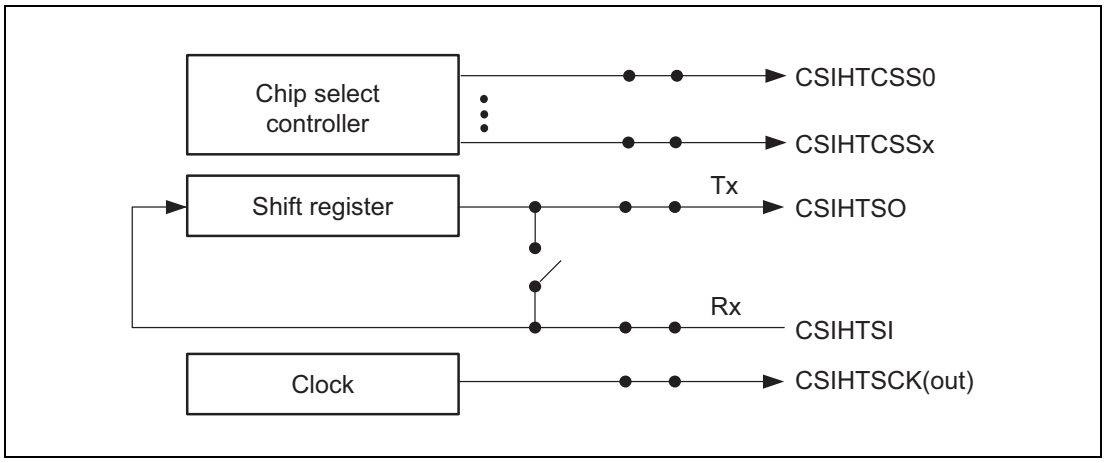


Figure 11.43 Normal Operation

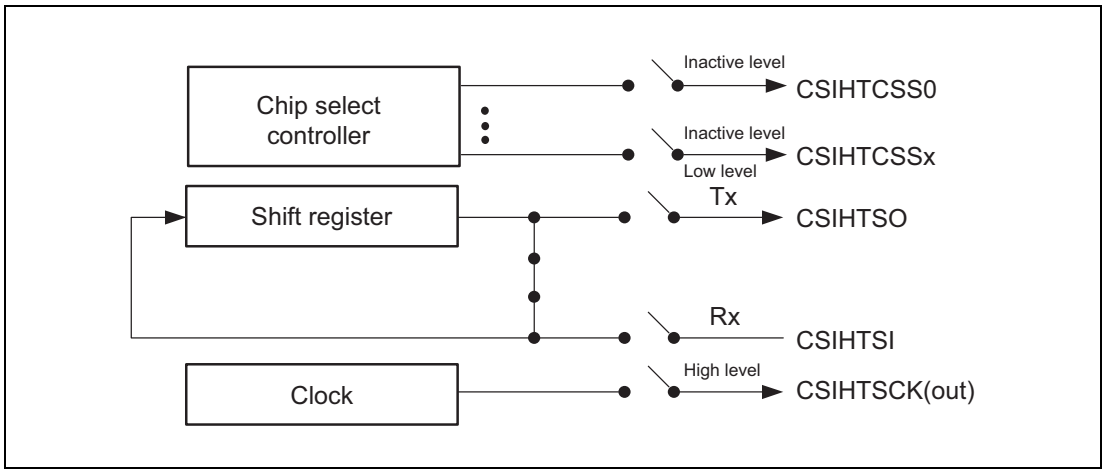


Figure 11.44 Loop-back Operation

11.4.20 CPU-controlled High Priority Communication Function

CSIH has a function to abort low priority communication to perform high priority communication if it receives a high-priority communication request from the CPU while low-priority communication is being used. This function supports transmit-only buffer mode as low priority communication and direct access mode as high-priority communication only.

To enable this function, CSIHnCTL1.CSIHnPHE and CSIHnCTL1.CSIHnJE must be set to 1.

The following figure illustrates CPU-controlled high-priority communication.

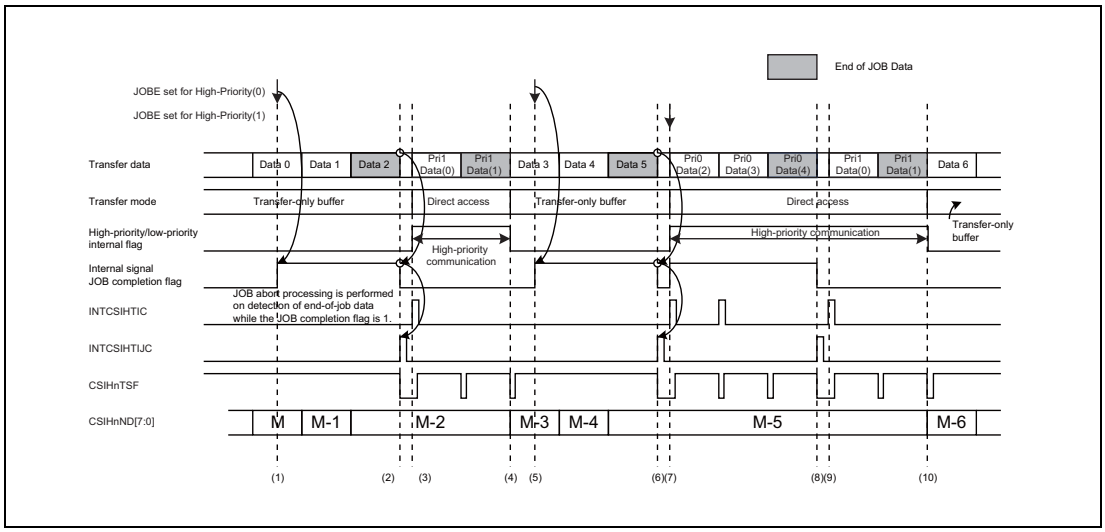


Figure 11.45 Example of CPU-Controlled High-Priority Communications, when CSIHnCTL1.CSIHnSLIT = 1

(1) By setting CSIHnCTL0.CSIHnJOBE = 1 during low-priority communication, start of high-priority communication following end-of-job data is notified, and the internal signal flag is set.

(2) When end-of-job data is detected, the current low-priority communication is aborted and the INTCSIHTIJC interrupt occurs. An internal signal, the JOB completion flag is cleared due to the abortion of communication, and memory mode is automatically switched to direct access mode for the subsequent high-priority communication.

(3) The CPU detects the interrupt and starts communication by writing the first transmission data of high-priority communication to CSIHnTX0W or CSIHnTX0H.

(4) When end-of-job data is detected, communication is aborted. At this time, because the internal signal, end-of-job flag is set to 0, the CSIH determines that the next communication is low-priority and switches memory mode to transmit-only buffer mode automatically, and then resumes the aborted low-priority communication.

(5) Same as (1) above.

(6) Same as (2) above.

(7) The CPU detects an interrupt and starts communication by writing the first transmission data of high-priority communication to CSIHnTX0W or CSIHnTX0H. The CPU sets CSIHnCTL0.CSIHnJOBE = 1 again to notify that the next communication is high-priority.

(8) When end-of-job data is detected, communication is aborted and the INTCSIHTIJC interrupt is generated. At this time, the CPU determines that the subsequent communication is high-priority because the internal signal JOB completion flag is 1, and waits for communication to start.

(9) Same as (3) above.

(10) Same as (4) above.

CAUTION

Memory mode is switched automatically when communication is changed from low priority to high priority (switch from transmit-only buffer mode to direct access mode) and from high priority to low priority (switch from direct access mode to transmit-only buffer mode).

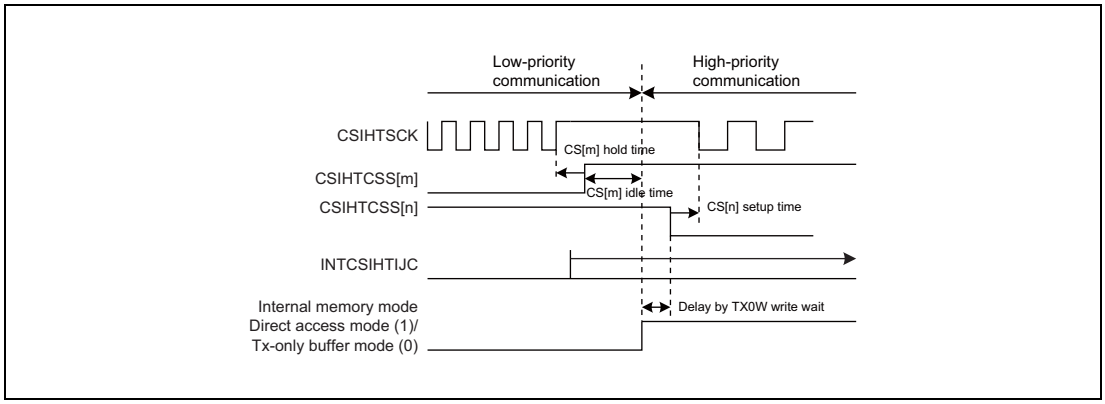


Figure 11.46 Transition from Low-priority Mode to High-priority Mode

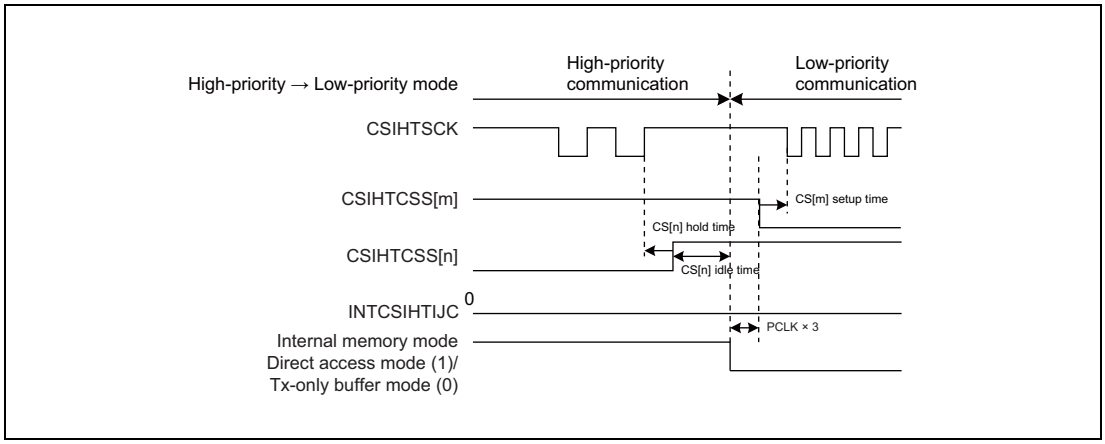


Figure 11.47 Transition from High-priority Mode to Low-priority Mode

Do not conduct write operation of communication data or CSIHnCTL0.CSIHnJOBE bit operation during setting prohibit period to switch low and high priority communication mode correctly.

CSIHnTX0W register write inhibited period:

- Period from when CSIHnJOBE bit is set for switching to high priority communication mode to when the INTCSIHTIJC interrupt is detected.
- Period from when the last data of high priority communication (End of JOB data) is written to when the CSIHnHPST state = 0 is detected.

CSIHnJOBE register write inhibited period:

- Period from when CSIHnJOBE bit is set for switching to high priority communication mode to when the INTCSIHTIJC interrupt is detected.

During high communication mode period, there is no setting prohibit period for CSIHnJOBE bit. It is possible to set CSIHnJOBE bit before writing communication data. For example, to communicate

multiple JOB data in high priority mode, it is possible to set CSIHnJOBE bit before writing the first communication data.

CAUTION

When CSIHnJOBE bit is set right before the last communication of high priority communication ends, different operations are required depending on the internal detection timing of CSIHnJOBE bit setting. When CSIHnJOBE bit setting is detected before the last bit communication, high priority communication mode continues.

When setting of the CSIHnJOBE bit is detected after the transfer of the last bit is completed, the mode temporarily returns to low priority communications. After detection of End of JOB data in low priority communications, the mode changes back to high priority communications.

11.4.21 Enforced Chip Select Idle Setting

This macro is able to insert an idle state between the two consecutive transfer data by the setting of CSIHnCFGx.CSIHnIDLx. Detail is as follows.

1. When CSIHnCFGx.CSIHnIDLx = 0:
   If a next CSIHTCSSx is the same as the previous one, an idle state is not inserted and an inter-data time is inserted.
   An idle state is always inserted if a next CSIHTCSSx is not different from the previous one.
2. When CSIHnCFGx.CSIHnIDLx = 1:
   An idle state is always inserted even if a next CSIHTCSSx is not different from the previous one.

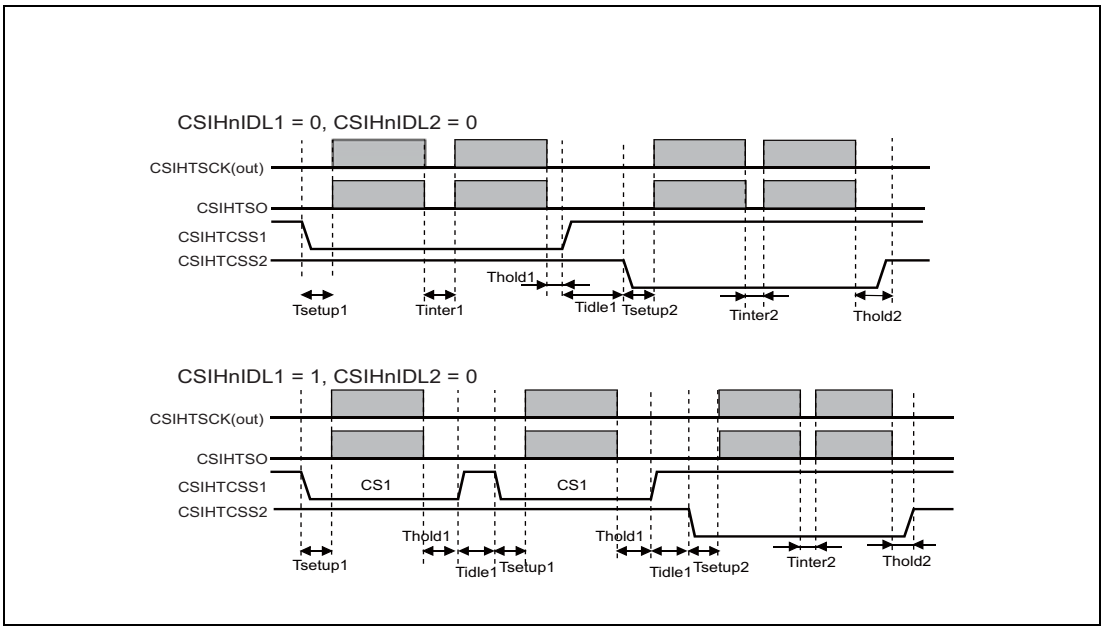


Figure 11.48 Enforced Chip Select Idle Example

CAUTION

If high priority communication function controlled by CPU is validated (CSIHnCTL1.CSIHnPHE = 1), when switch from low priority communication mode to high priority communication mode or switch from high priority communication mode to low priority communication mode, idle state is inserted regardless of the setting of CSIHnCFGx.CSIHnIDLx bit.

11.5 Procedures

The examples and procedures below are described according to the memory mode in the following order:

- Direct access mode
- Transmit-only buffer mode
- Dual buffer mode
- FIFO mode

11.5.1 Procedures in Direct Access Mode

Two examples for a master are provided, one with job mode disabled, and the other one with job mode enabled.

11.5.1.1 Transmission/Reception in Master Mode when Job Mode is Disabled

The procedures below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1)

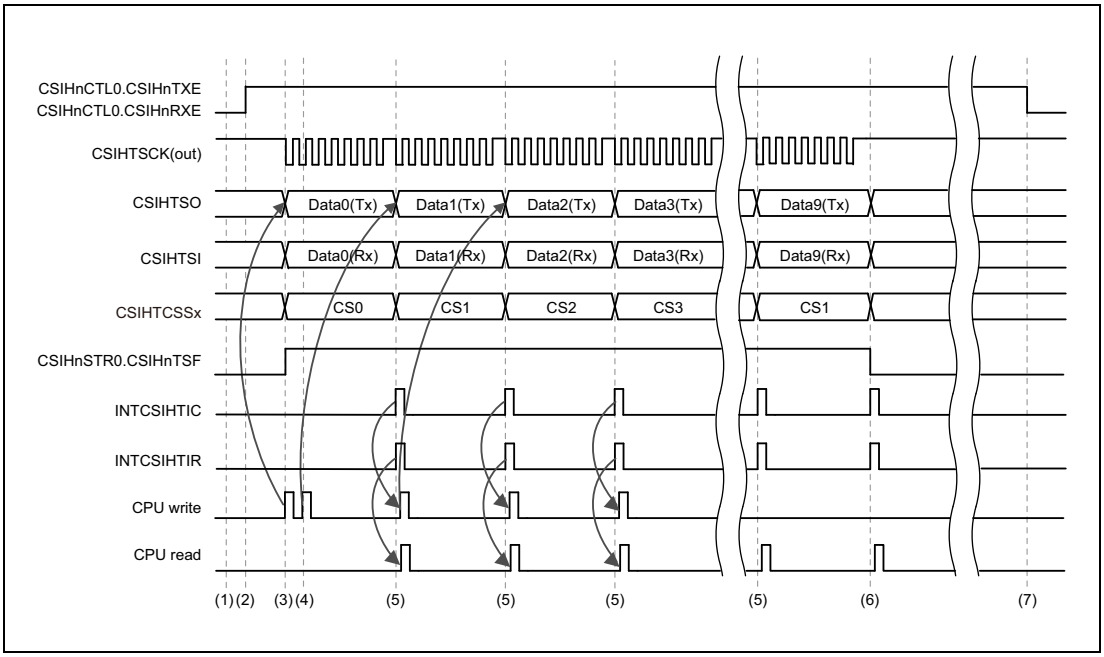


Figure 11.49 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 0

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits reception), and CSIHnMBS = 1 (selects direct access mode).
3. Write the first data to be sent to the transmission register, CSIHnTX0W. Within the same write operation, activate CS0. Transmission starts automatically when the first data becomes available.
4. Write the second data to CSIHnTX0W. If required, you can change the CS to address a different device. Writing the second data immediately after the first one avoids unnecessary delays between the data.
5. After every transmission/reception of a data the interrupts INTCSIHTIC and INTCSIHTIR are generated:
   - INTCSIHTIC indicates that the next data can be written to CSIHnTX0W.
   - INTCSIHTIR indicates that the reception register, CSIHnRX0 must be read.
6. No more write action is required after completion of data 8. Data 9 (the last data) has been written in advance.
   However, reception register CSIHnRX0 must be read after completion of writing data 8 and 9.
7. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.1.2 Transmission/Reception in Master Mode when Job Mode is Enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = $1000_{B)}$.
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1)
- Two jobs, each of them sends three data.

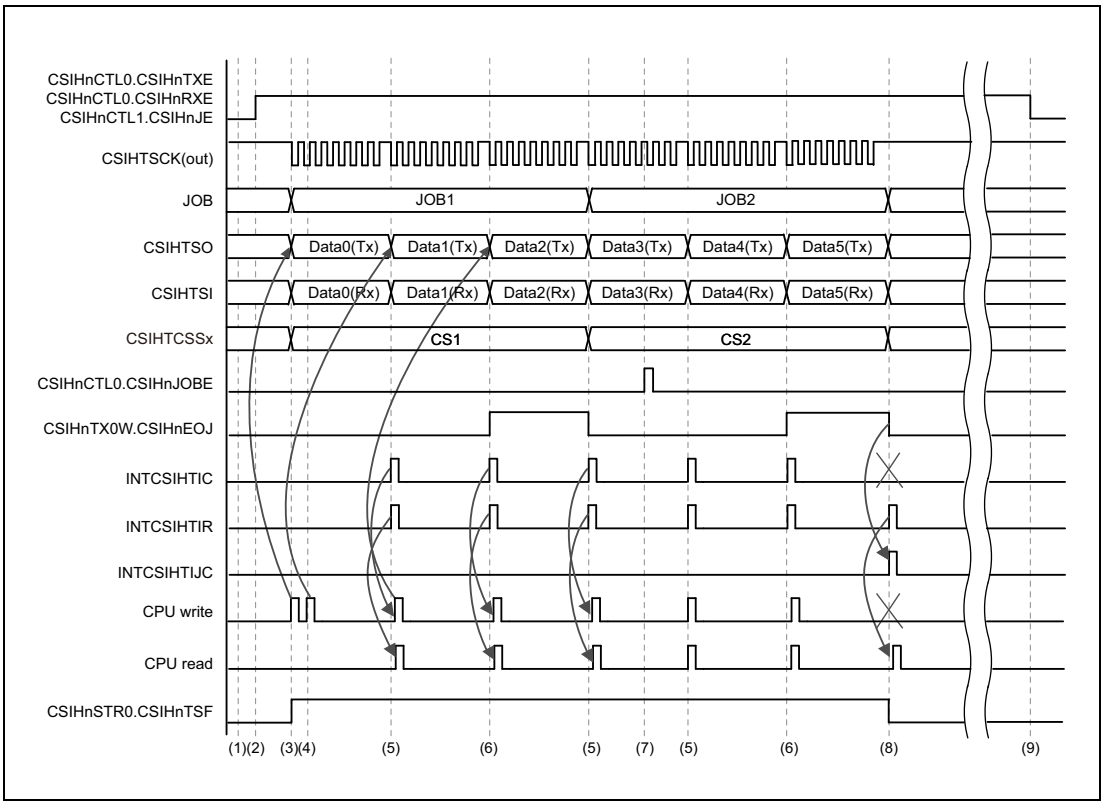


Figure 11.50 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 1

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS1 and CSIHTCSS2.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits the reception), and CSIHnMBS = 1 (selects direct access mode).
3. Write the first data to be sent to the transmission register CSIHnTX0W. Transmission starts automatically when the first data becomes available.
   The CSIHnSTR0.CSIHnTSF flag indicates that communication is in progress.
4. Write the second data to CSIHnTX0W. Writing the second data immediately after the first one avoids unnecessary delays between the data.
5. After every data transmission/reception, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated.
   - INTCSIHTIC indicates that the next data can be written to CSIHnTX0.
   - INTCSIHTIR indicates that the reception register, CSIHnRX0 must be read.
6. Setting CSIHnTX0W.CSIHnEOJ = 1 indicates that the last data of the current job is sent. After that, the next job may begin.
7. By setting CSIHnCTL0.CSIHnJOBE = 1, communication is forced to stop at the end of the current job (JOB2).
8. After the forced stop of communication, the interrupt request, INTCSIHTIC is replaced by INTCSIHTIJC. INTCSIHTIR is generated as usual.
   The interrupt request, INTCSIHTIJC indicates a forced stop of communication at the end of the current job.
   The interrupt request, INTCSIHTIC is not generated. Additionally, the transmission data available in the CSIHnTX0W register is not sent.
9. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.
   To start another transmission without stopping communication, perform steps 3 and later.

11.5.2 Procedures in Transmit-only Buffer Mode

Two examples for a master is provided, one with job mode disabled, and the other one with job mode enabled.

11.5.2.1 Transmission/Reception in Master Mode when Job Mode is Disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Transmit-only buffer mode (CSIHnCTL0.CSIHnMBS = 0,CSIHnMCTL0.CSIHnMMS[1:0] = 10_B)

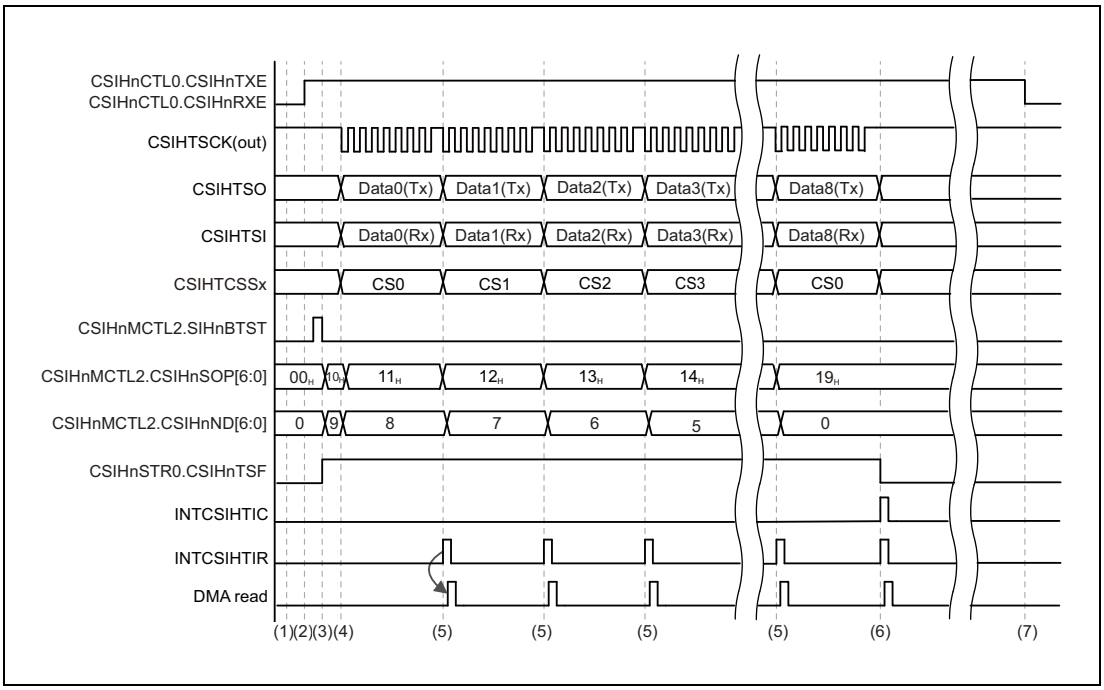


Figure 11.51 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The procedure of writing the data into the buffer is not described.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   CSIHnMCTL0.CSIHnMMS[1:0] is set to 10_B (transmit-only buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission) and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure the send pointer and the number of data by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission/reception is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data packet transmission.
5. After every data reception, the interrupt request, INTCSIHTIR is generated. INTCSIHTIR indicates necessity of reading reception register CSIHnRX0.
6. When all transmissions are complete, the interrupt request, INTCSIHTIC is generated.
7. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.2.2 Transmission/Reception in Master Mode when Job Mode is Enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- The number of data is 8 (CSIHnMCTL2.CSIHnND[7:0] = 08_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Transmit-only buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 10_B)

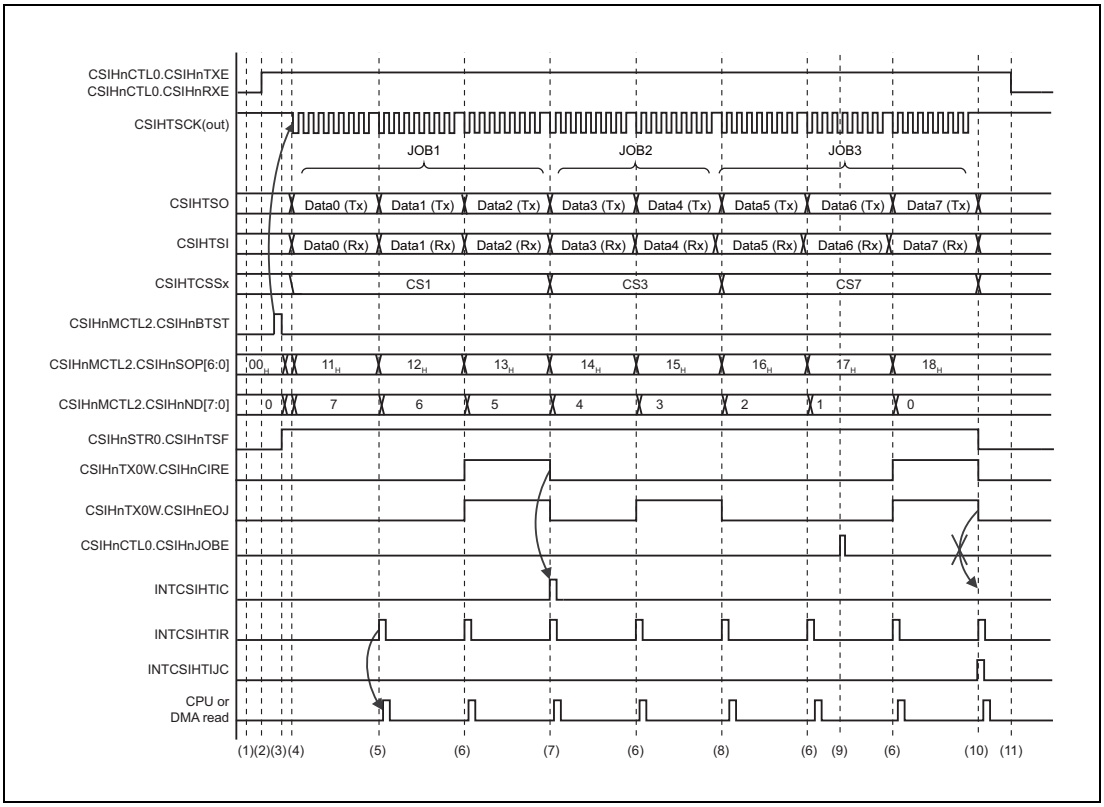


Figure 11.52 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 1

NOTE

The process of writing the data into the buffer is not described.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 10_B (transmit-only buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure the send pointer and the number of data by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission/reception is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. After every data reception, the interrupt request, INTCSIHTIR is generated. INTCSIHTIR indicates necessity of reading reception register CSIHnRX0.
6. The CSIHnTX0W.CSIHnEOJ = 1 setting indicates that the last data of the current job is sent.
7. The interrupt request INTCSIHTIC is generated. INTCSIHTIC indicates that the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 1.
8. The INTCSIHTIC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
9. By setting CSIHnCTL0.CSIHnJOBE = 1, communication is forced to stop at the end of JOB3.
10. After the forced stop of communication, interrupt requests INTCSIHTIJC and INTCSIHTIR are generated at the end of job3.
    The INTCSIHTIJC interrupt request indicates a forced stop of communication at the end of the current job.
    The INTCSIHTIC interrupt request is not generated because the INTCSIHTIJC interrupt request is generated instead of the INTCSIHTIC interrupt request. Additionally, the transmission data available in the CSIHnTX0 register is not sent.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3 Procedures in Dual Buffer Mode

Examples when job mode is enabled in master mode, disabled in master mode, and disabled in slave mode are provided below.

11.5.3.1 Transmission/Reception in Master Mode when Job Mode is Disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Default clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)

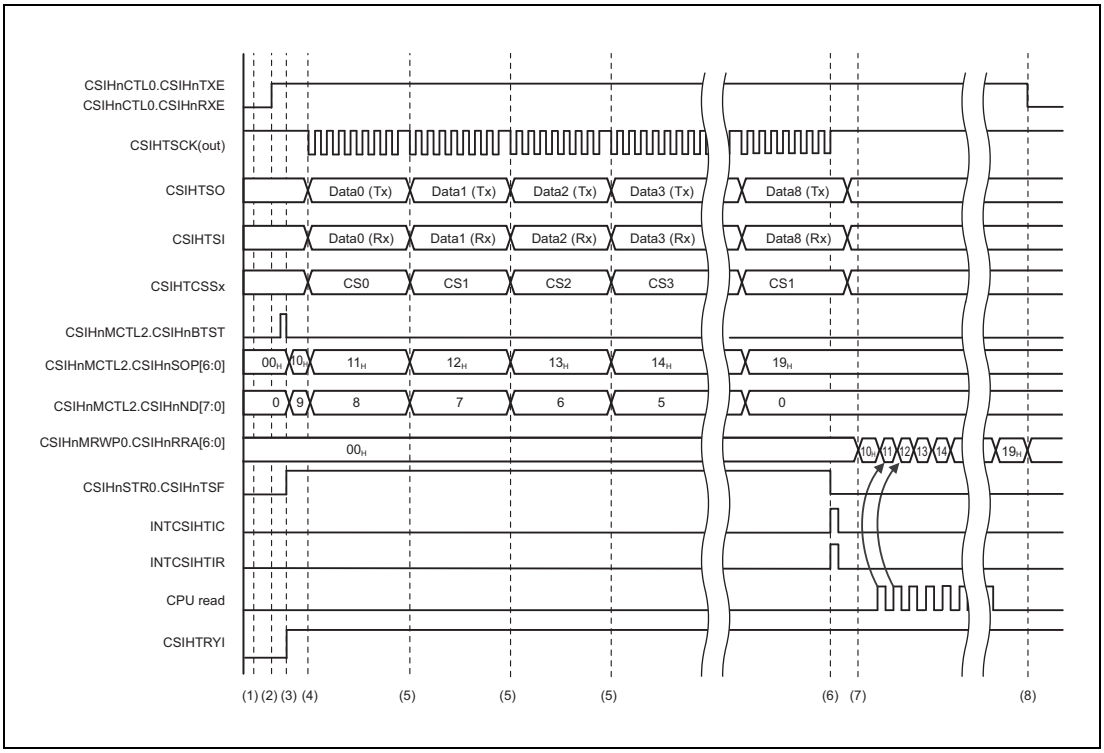


Figure 11.53 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The process of writing the data into the buffer is not described.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits the reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure communication by setting CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0]. Permit buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. This is repeated until the last data is transmitted/received.
   The interrupt requests, INTCSIHTIC and INTCSIHTIR are not generated.
6. When the last data is transmitted/received, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated.
   The CPU starts to read the received data from the Rx buffer.
7. The start address of the read access is specified in CSIHnMRWP0.CSIHnRRA[6:0].
   (10_H is set in CSIHnRRA[6:0] by software)
   These bits are incremented every time a data is read.
8. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3.2 Transmission/Reception in Master Mode when Job Mode is Enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- The number of data is 8 (CSIHnMCTL2.CSIHnND[7:0] = 08_H).
- The transfer start address is 00_H (CSIHnMCTL2.CSIHnSOP[6:0] = 00_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)

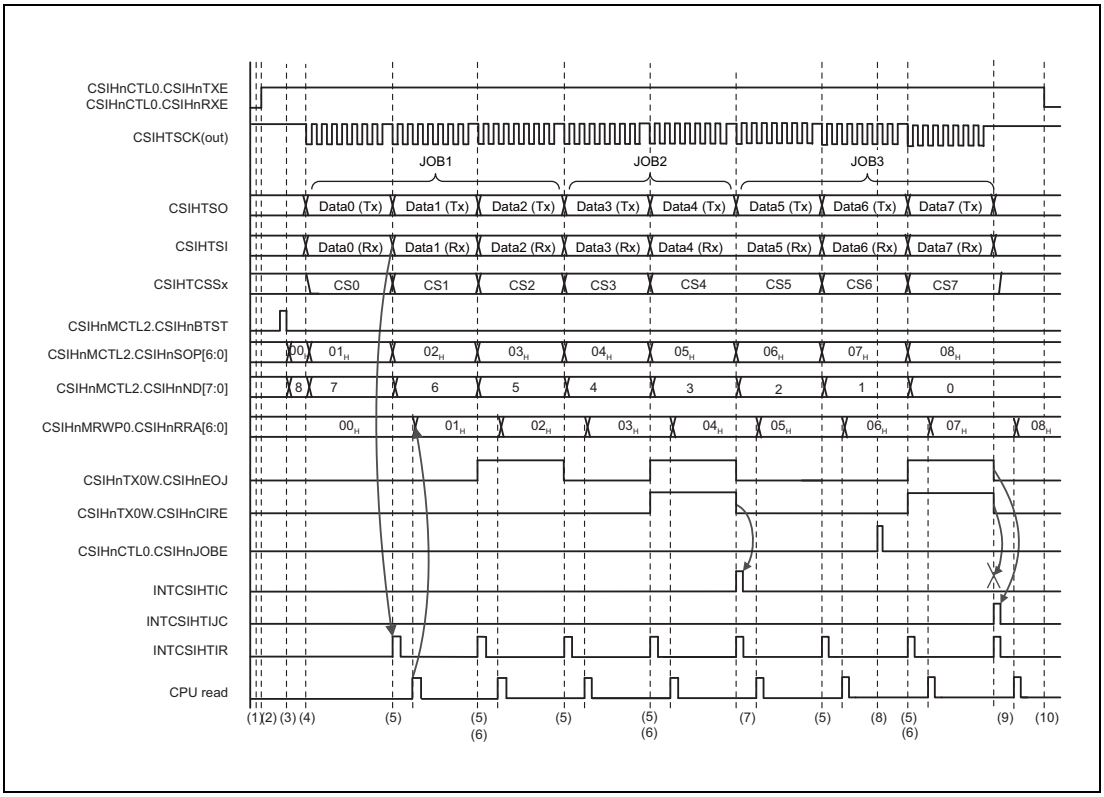


Figure 11.54 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 1

NOTE

The process of writing the data into the buffer is not described.

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Configure communication by setting the CSIHnMCTL2.CSIHnSOP[6:0] and CSIHnMCTL2.CSIHnND[7:0] bits. Start the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started. The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented, and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
   This is repeated until the last data is transmitted/received.
5. The INTCSIHTIR interrupt request is generated every time a data is received.
   The INTCSIHTIC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
6. CSIHnTX0W.CSIHnEOJ = 1 indicates that the last data of the current job is sent.
7. The INTCSIHTIC interrupt request is generated. INTCSIHTIC indicates that the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) is sent with CSIHnTX0W.CSIHnCIRE = 1.
8. CSIHnCTL0.CSIHnJOBE = 1 indicates that communication is forced to stop at the end of JOB3.
9. After the forced stop of communication, interrupt requests, INTCSIHTIJC and INTCSIHTIR are generated at the end of JOB3.
   The INTCSIHTIJC interrupt request indicates a forced stop of communication at the end of the current job.
   The INTCSIHTIC interrupt request is not generated because the INTCSIHTIJC interrupt request is generated instead of the INTCSIHTIC interrupt request. The transmit data available in CSIHnTX0 register is not transmitted this time.
10. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.3.3 Transmission/Reception in Slave Mode when Job Mode is Disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCTL1.CSIHnCKR = 0, CSIHnCFG0.CSIHnDAP0 = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- The number of data is 9 (CSIHnMCTL2.CSIHnND[7:0] = 09_H).
- The transfer start address is 10_H (CSIHnMCTL2.CSIHnSOP[6:0] = 10_H).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Dual buffer mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 01_B)
- Handshake function enable (CSIHnCTL1.CSIHnHSE = 1)

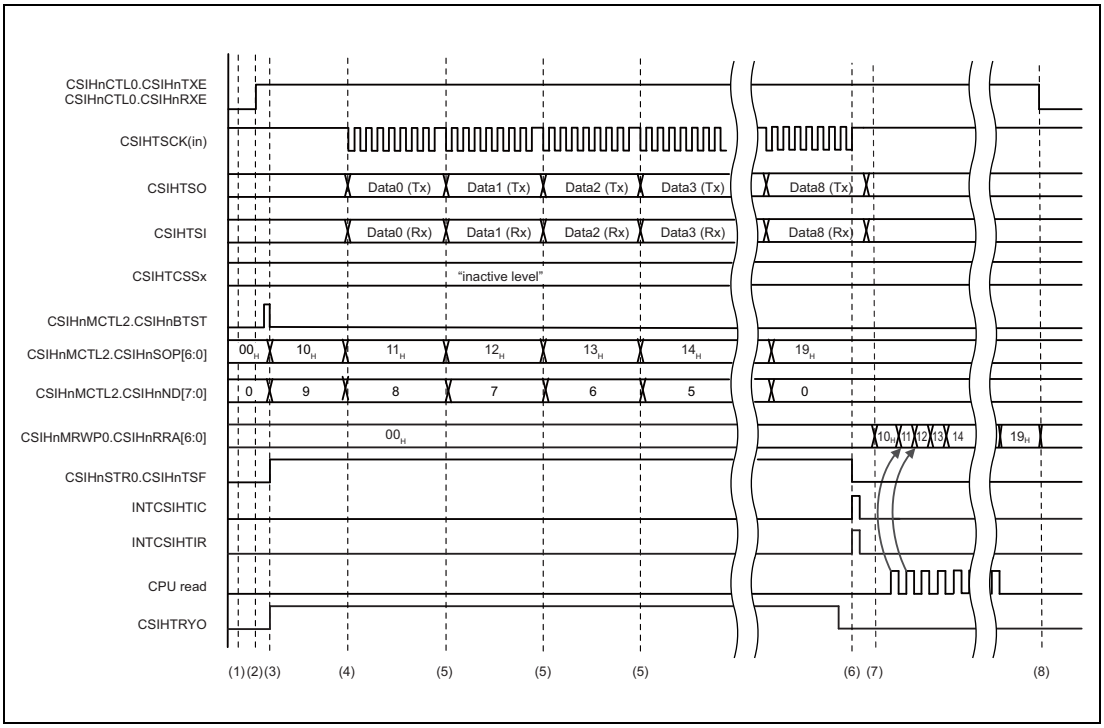


Figure 11.55 Slave in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0

NOTE

The process of writing the data into the buffer is not described.

Procedure:

1. Configure the communication protocol in the CSIHnCFG0 register.
   Specify the transfer and operating modes by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Configure the memory mode in CSIHnMCTL0.CSIHnMMS[1:0].
   Set CSIHnMCTL0.CSIHnMMS[1:0] = 01_B (dual buffer mode).
2. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
3. Specify the transfer start address by setting CSIHnMCTL2.CSIHnSOP[6:0] and the number of data by setting CSIHnMCTL2.CSIHnND[7:0]. Permit the buffer transfer by setting CSIHnMCTL2.CSIHnBTST.
4. Transmission is started when the input clock from the master is received.
   The CSIHnMCTL2.CSIHnSOP[6:0] bits are automatically incremented and the CSIHnMCTL2.CSIHnND[7:0] bits are decremented after each data transmission.
5. This is repeated until the last data is transmitted/received.
   The interrupt requests, INTCSIHTIC and INTCSIHTIR are not generated because transmission data is sent from the buffer, and received data is stored in the buffer.
6. When the last data is transmitted/received, the interrupt requests, INTCSIHTIC and INTCSIHTIR are generated.
   The CPU starts to read the received data that is stored in the receive buffer.
7. The start address of the read access is specified in CSIHnMRWP0.CSIHnRRA[6:0].
   (10_H is set in CSIHnRRA[6:0] by software)
   These bits are incremented every time a data is read.
8. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.4 Procedures in FIFO Mode

Two examples for a master is provided, one with job mode disabled, the other one with job mode enabled.

11.5.4.1 Transmission/Reception in Master Mode when Job Mode is Disabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- FIFO mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 00_B)

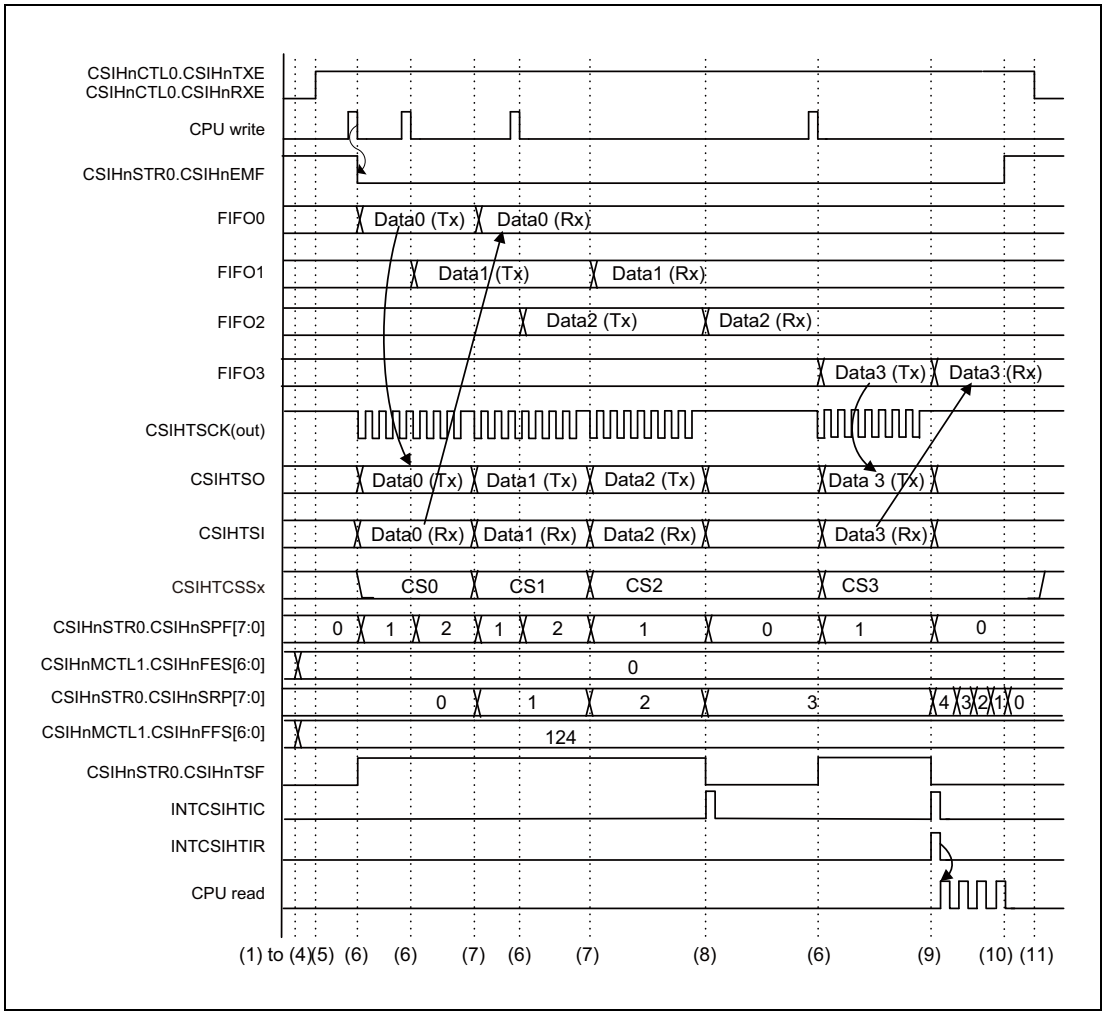


Figure 11.56 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 0

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register. Specify the job mode disable and master mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2. Specify the FIFO mode by CSIHnMCTL0.CSIHnMMS[1:0] = 00_B. This example uses chip select signals CSIHTCSS0 to CSIHTCSS3.
2. Set CSIHnSTCR0.CSIHnPCT = 1 to clear all buffer pointers.
3. Check that CSIHnSTR0.CSIHnFLF = 0, CSIHnSTR0.CSIHnEMF = 1, and CSIHnSTR0.CSIHnSPF[7:0] = 00_H.
4. The CSIHnMCTL1.CSIHnFES[6:0] bits specify the condition for the INTCSIHTIC interrupt output.
   The CSIHnFFS[6:0] bits in the same register specify the condition for the INTCSIHTIR interrupt.
5. Set CSIHnCTL0.CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
6. Write the first transmit data packet to the transmit register CSIHnTX0W. Transmission starts automatically when the first data becomes available.
7. The current transmission is completed.
   As the CSIHnFES[6:0] bits are not the same as the CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is not generated.
8. As the CSIHnFES[6:0] bits = CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is generated.
9. When CSIHnFFS[6:0] = 128-CSIHnSRP[7:0], the interrupt request INTCSIHTIR is generated.
   As CSIHnFES[6:0] = CSIHnSRF[7:0], the interrupt request INTCSIHTIC is generated.
   After the generation of the interrupt, the CPU starts reading received data that is stored in the receive buffer.
10. When the CPU completes reading the received data that is stored in the receive buffer, CSIHnSTR0.CSIHnEMF is set to 1 and the FIFO buffer will be empty.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.5.4.2 Transmission/Reception in Master Mode when Job Mode is Enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- The transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0)
- Normal clock and data phases (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- JOB1 consists of four data, JOB2 consists of three data, and JOB3 consists of five data packets.
- Normal INTCSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- FIFO mode (CSIHnCTL0.CSIHnMBS = 0, CSIHnMCTL0.CSIHnMMS[1:0] = 00_B)

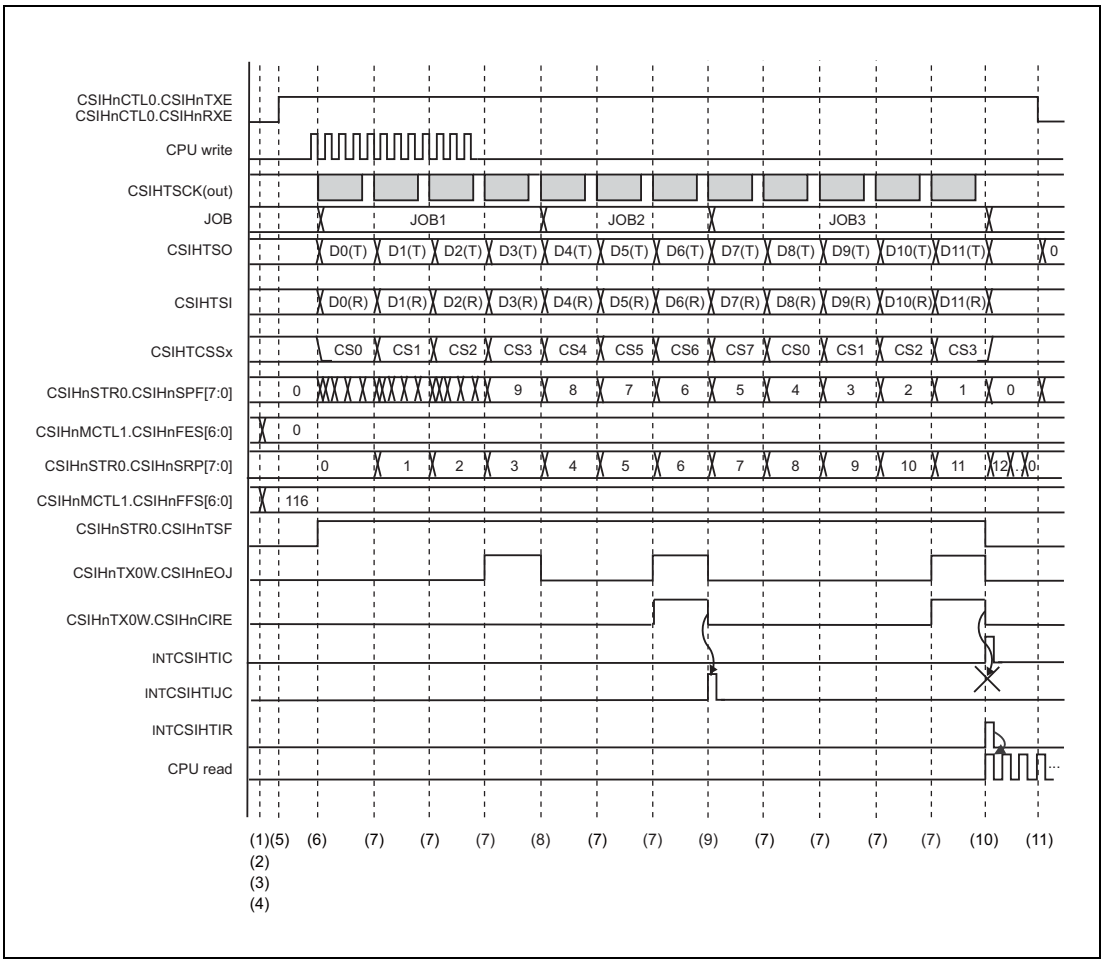


Figure 11.57 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 1

Procedure:

1. Configure the communication protocol in the CSIHnCFGx register.
   Disable the job mode and select the memory mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.
   Select the FIFO mode by setting CSIHnMCTL0.CSIHnMMS[1:0] = 00_B.
2. Set CSIHnSTCR0.CSIHnPCT to 1 to clear all buffer pointers.
3. Make sure CSIHnSTR0.CSIHnFLF is set to 0, CSIHnSTR0.CSIHnEMF is set to 1, and CSIHnSTR0.CSIHnSPF[7:0] is set to 00_H.
4. With CSIHnMCTL1.CSIHnFES[6:0], specify the conditions for generating the INTCSIHTIC interrupt request. With CSIHnMCTL1.CSIHnFFS[6:0], specify the conditions for generating the INTCSIHTIR interrupt request.
5. In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), and CSIHnRXE = 1 (permits reception). The CSIHnCTL0.CSIHnMBS bit must be cleared.
6. Write the first data to be sent to the CSIHnTX0W transmission register. Transmission starts automatically when the first data becomes available.
7. The current transmission is completed.
   As CSIHnFES[6:0] are not the same as CSIHnSPF[7:0], the interrupt request INTCSIHTIC is not generated.
8. The INTCSIHTIJC interrupt request is not generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 0.
9. The INTCSIHTIJC interrupt request is generated because the last data of the current job (CSIHnTX0W.CSIHnEOJ = 1) was sent with CSIHnTX0W.CSIHnCIRE = 1.
10. As the CSIHnFES[6:0] bits = the CSIHnSPF[7:0] bits, the interrupt request INTCSIHTIC is generated. As INTCSIHTIC is generated, INTCSIHTIJC is not generated.
    When CSIHnFFS[6:0] = 128-CSIHnSRP[7:0], the interrupt request INTCSIHTIR is generated. After the generation of the interrupt, the CPU starts reading received data that is stored in the receive buffer.
11. Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

11.6 List of Caution

Table 11.43 Notes on Setting Registers (1/3)

Register	Bit	Content
CSIHnCTL0	CSIHnPWR	If this bit is cleared during communication, ongoing communication is suspended. After that, the communication must be restarted.
CSIHnCTL0	CSIHnTXE CSIHnRXE	Do not modify any of these bits while CSIHnCTL0.CSIHnPWR=0. (These bits can be modified at the same time as the CSIHnCTL0.CSIHnPWR bit.) Do not modify these bits while CSIHnSTR0.CSIHnTSF = 1, because the specified operation is not guaranteed if ongoing communication is suspended.
CSIHnCTL0	CSIHnJOBE	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid when CSIHnCTL1.CSIHnJE = 1. Setting this bit is prohibited in slave mode.
CSIHnCTL0	CSIHnMBS	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. (This bit can be modified at the same time as the CSIHnCTL0.CSIHnPWR bit.) Modification of this bit is only allowed while CSIHnSTR0.CSIHnTSF = 0. Do not change the mode between FIFO mode and direct access mode while CSIHnCTL0.CSIHnPWR = 1. When the CPU-controlled high-priority communication function is enabled, the operation is the same as that in direct access mode regardless of the CSIHnMBS bit setting.
CSIHnCTL1	CSIHnCKR	Modification of this bit is only allowed while CSIHnCTL0.CSIHnPWR = 0. If CS is not used, use this bit instead of setting the CSIHnCFGx.CSIHnCKPx bit, and set the CSIHnCFGx.CSIHnCKPx bit to 0. This bit must be used in slave mode.
CSIHnCTL1	CSIHnSLIT CSIHnCSL[3:0] CSIHnEDLE CSIHnDCS CSIHnCSRI CSIHnHSE	Modification of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0.
CSIHnCTL1	CSIHnPHE CSIHnJE CSIHnLBM	Modification of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of this bit is prohibited in slave mode.
CSIHnCTL1	CSIHnSSE	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of this bit is prohibited in master mode.
CSIHnCTL1	CSIHnSIT	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid in master mode. In slave mode, no delay is generated.
CSIHnCTL2	CSIHnPRS[2:0]	Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting of the max baud rate is as follows. Master mode: PCLK/8 Slave mode: PCLK/16
CSIHnSTR0	CSIHnSRP[7:0] CSIHnSPF[7:0] CSIHnFLF CSIHnEMF CSIHnTSF	Writing these bits is prohibited, and only reading is permitted.
CSIHnSTR0	CSIHnTMOE CSIHnOFE CSIHnDCE CSIHnPE CSIHnOVE	Writing these bits is prohibited, and only reading is permitted. These bits are initialized when CSIHnCTL0.CSIHnPWR = 0 → 1 or CSIHnCTL0.CSIHnPWR = 1 → 0.
CSIHnSTCR0	CSIHnPCT	If this bit is set to1 during communication, ongoing communication is suspended.
CSIHnMCTL0	CSIHnMMS[1:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0 and CSIHnCTL0.CSIHnMBS = 0.

Table 11.43 Notes on Setting Registers (2/3)

Register	Bit	Content
CSIHnMCTL0	CSIHnTO[4:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. Set these bits to 0 in master mode. Set these bits to 0 in direct access, dual buffer, and transmit-only buffer mode.
CSIHnMCTL1	CSIHnFES[6:0] CSIHnFFS[6:0]	Write to these bits during communication is permitted.
CSIHnMCTL2	CSIHnBTST CSIHnND[7:0] CSIHnSOP[6:0]	Writing these bits is prohibited when CSIHnCTL0.CSIHnPWR = 0. Writing these bits is prohibited when CSIHnCTL0.CSIHnTXE = CSIHnCTL0.CSIHnRXE = 0. Writing these bits is prohibited when CSIHnSTR0.CSIHnTSF = 1. Writing these bits is prohibited in direct access or FIFO mode.
CSIHnMRWP0	CSIHnRRA[6:0]	Write to these bits during communication is permitted. Writing these bits is prohibited in direct access or FIFO mode. When writing is required, set "0000_H" to these bits in transmit-only buffer mode.
CSIHnMRWP0	CSIHnTRWA[6:0]	Write to these bits during communication is permitted. Writing these bits is prohibited in direct access or FIFO mode.
CSIHnCFGx	CSIHnBRSSx[1:0] CSIHnRCBx CSIHnIDLx CSIHnIDx[2:0] CSIHnHDx[3:0] CSIHnINx[3:0] CSIHnSPx[3:0]	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. These bits must be set to 0 in slave mode.
CSIHnCFGx	CSIHnPSx[1:0] CSIHnDLSx[3:0] CSIHnDIRx CSIHnDAPx	Modification of these bits is only allowed while CSIHnCTL0.CSIHnPWR = 0. In slave mode, the CSIHnCFG0 setting is used. Therefore, all the bits in CSIHnCFG1 to CSIHnCFG3 must be set to 0.
CSIHnCFGx	CSIHnCKPx	Modification of this bit is only allowed while CSIHnCTL0.CSIHnPWR = 0. As CSIHnCTL1.CSIHnCKR must be used in slave mode, set these bits to 0. If CS is not used, use the CSIHnCTL1.CSIHnCKR bit instead of these bits, and clear the bits to 0.
CSIHnTX0W	CSIHnEOJ CSIHnCIRE	This bit is only valid when CSIHnCTL1.CSIHnJE = 1. While CSIHnCTL1.CSIHnJE = 0, the value of this bit is ignored even if 1 is read. Set these bits to 0 in slave mode.
CSIHnTX0W	CSIHnEDL	This bit is only valid when CSIHnCTL1.CSIHnEDLE = 1. While CSIHnCTL1.CSIHnEDLE = 0, the value of this bit is ignored even if 1 is read.
CSIHnTX0W	CSIHnCS[3:0]	In master mode, setting these bits to F_H is prohibited. In slave mode, set these bits to E_H.
CSIHnTX0W CSIHnTX0H		Reading these bits during communication is prohibited in FIFO mode. While CSIHnCTL0.CSIHnPWR = 0 and FIFO mode, read/write access to these bits is prohibited. While CSIHnCTL0.CSIHnTXE = CSIHnCTL0.CSIHnRXE = 0, write access to these bits are prohibited in direct access mode.
CSIHnRX0W		These bits are initialized when CSIHnCTL0.CSIHnPWR = 0 → 1 or CSIHnCTL0.CSIHnPWR = 1 → 0. While CSIHnCTL0.CSIHnPWR = 0, reading and writing these bits is prohibited in FIFO mode. While CSIHnCTL0.CSIHnPWR = 0, reading and writing these bits is permitted in the mode(Transmit-only buffer, Dual buffer and Direct access modes) exited FIFO mode. While CSIHnCTL0.CSIHnPWR = 1, writing these bits is prohibited, and reading these bits is permitted. Writing is prohibited and only reading is permitted of these bits while CSIHnCTL0.CSIHnPWR = 1.

Table 11.43 Notes on Setting Registers (3/3)

Register	Bit	Content
CSIHnRX0H		These bits are initialized when CSIHnCTL0.CSIHnPWR = 0 → 1 or CSIHnCTL0.CSIHnPWR = 1 → 0. While CSIHnCTL0.CSIHnPWR = 0, reading and writing these bits is prohibited in FIFO mode. Writing is prohibited and only reading is permitted of these bits while CSIHnCTL0.CSIHnPWR =1. While CSIHnCTL0.CSIHnPWR = 1, writing these bits is prohibited in the FIFO mode and reading these bits is prohibited in the FIFO mode. In spite of CSIHnCTL0.CSIHnPWR value, writing is prohibited and only reading is permitted of these bits in the mode(Transmit-only buffer, Dual buffer and Direct access modes) exited FIFO mode.
CSIHnBRSy		Modification of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0.

Section 12 Serial Communication Interface 3 (SCI3)

This section contains a generic description of the serial communication interface (SCI3).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of SCI3.

12.1 Features of RH850/C1x SCI3

12.1.1 Units

This LSI has the following number of SCI3 units.

Table 12.1 Units

Product	RH850/C1x
Number of Units	3
Name	SCI3n (n = 0 to 2)

Table 12.2 Index

Index	Meaning
n	Throughout this section, the individual SCI3 units are identified by the index “n” (n = 0 to 2); for example, SCI3nRSR is the receive shift register.

12.1.2 Register Base Address

SCI3 base addresses are listed in the following table.

SCI3 register addresses are given as offsets from the base addresses in general.

Table 12.3 Register Base Address

Base Address Name	Base Address
<SCI30_base>	FFD9 0000_H
<SCI31_base>	FFD9 1000_H
<SCI32_base>	FFD9 2000_H

12.1.3 Clock Supply

SCI3 clock is listed in following table.

Table 12.4 Clock Supply

Unit Name	Clock for the Unit	Internal Clock Name
SCI3n	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)
	SCKn	SCInSCK pins (for operation with external clock signals)*1

Note 1. The SCI0SCK signal is not supported by RH850/C1M.

12.1.4 Interrupts and DMA

SCI3 interrupt requests are listed in the following table.

Table 12.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMAC Trigger Number	DTS Trigger Number
SCI30				
ERI	Receive error	203	—	—
RXI	Receive data full	204	107	112
TXI	Transmit data empty	205	108	113
TEI	Transmit complete	206	—	—
SCI31				
ERI	Receive error	207	—	—
RXI	Receive data full	208	109	114
TXI	Transmit data empty	209	110	115
TEI	Transmit complete	210	—	—
SCI32				
ERI	Receive error	211	—	—
RXI	Receive data full	212	111	116
TXI	Transmit data empty	213	112	117
TEI	Transmit complete	214	—	—

12.1.5 Reset Sources

SCI3 reset sources are listed in the following table. SCI3 is initialized by these reset sources.

Table 12.6 Reset Sources

Unit Name	Reset Source
SCI3n	Reset by any reset source

12.1.6 External Input/Output Signals

External input/output signals of SCI3 are listed in the following table.

Table 12.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
SCKn	Serial clock input/output signal	SCInSCK*[1]
RxDn	Receive data input signal	SCInRXD
TxDn	Transmit data output signal	SCInTXD

Note 1. The SCI0SCK signal is not supported by RH850/C1M.

12.2 Overview

12.2.1 Functional Overview

The SCI3 can handle two methods of serial communications: asynchronous and clock synchronous. Asynchronous serial data transfer can be handled with standard LSI chips for asynchronous communication such as Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). Asynchronous mode also provides for serial communications between multiple processors (multi-processor communications).

In clock synchronous transfer, the interface serves as the master in communications through the output of its internal clock signal for synchronization, or as the slave in communications, where transfer is synchronized with the signal from an external clock.

List of Functions

The table below summarizes available functions.

Function	Asynchronous Communications	Clock Synchronous Communications	
		Internal Clock (Master Mode)	External Clock (Slave Mode)
Full-duplex communication	Possible Independent transmitter unit and receiver unit are allow simultaneous transmission and reception. Both the transmitter and receiver have a double-buffered structure, enabling continuous data transmission and reception.		
SCKn pin	Possible of clock output	Clock output	Clock input
Transmit/receive clock source	Internal clock by the on-chip baud rate generator		External clock
LSB/MSB first	Selectable (except for 7-bit length data)	Selectable	
Interrupt and DMA/DTS transfer	• Transmission complete • Transmission data empty • Reception data full • Reception error DMA and DTS can be activated by a transmit data empty interrupt signal or reception data full interrupt signal.		
Bit rate modulation	Possible	Possible (except for setting of the maximum speed)	
Data length	Selectable: 7- or 8-bit	8-bit	
Stop bit length	Selectable: 1- or 2-bit	—	
Parity	Selectable: with even/odd parity or none	—	
Detection of reception errors	• Parity error • Overrun error • Framing error	• Overrun error	
Break detection	Detectable by reading a register when a framing error occurs	—	
Serial input data pin level check	Possible		
Multi-processor communication	Possible	—	
Double-speed operation	Possible	—	
Maximum bit rate	2.5 Mbps (PCLK/16)	5 Mbps (PCLK/8)	3.3 Mbps (PCLK/12)

12.2.2 Block Diagram

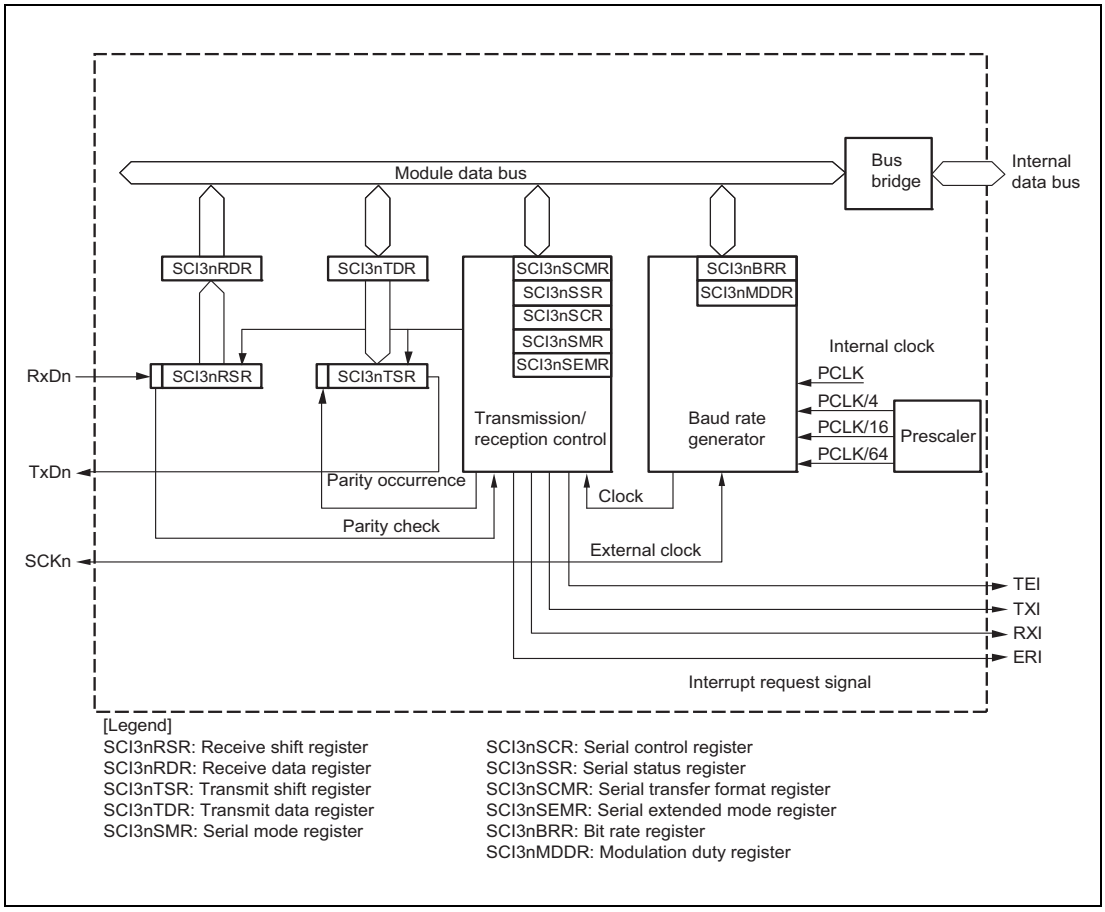


Figure 12.1 SCI3 Block Diagram

12.3 Registers

12.3.1 List of Registers

SCI3 registers are listed in the following table.

For information on <SCI3n_base>, see **Section 12.1.2, Register Base Address**.

Table 12.8 Register Configuration

Module Name	Register Name	Symbol	Address
SCI3n	Serial mode register	SCI3nSMR	<SCI3n_base>+ 00_H
SCI3n	Bit rate register / Modulation duty register*1	SCI3nBRR/ SCI3nMDDR	<SCI3n_base>+ 04_H
SCI3n	Serial control register	SCI3nSCR	<SCI3n_base>+ 08_H
SCI3n	Transmit data register	SCI3nTDR	<SCI3n_base>+ $0C_H$
SCI3n	Serial status register	SCI3nSSR	<SCI3n_base>+ 10_H
SCI3n	Receive data register	SCI3nRDR	<SCI3n_base>+ 14_H
SCI3n	Serial transfer format register	SCI3nSCMR	<SCI3n_base>+ 18_H
SCI3n	Serial extended mode register	SCI3nSEMR	<SCI3n_base>+ $1C_H$

Note 1. SCI3nBRR and SCI3nMDDR are assigned to the same address. The SCI3nSEMR.MDDRS bit is used to switch these registers.

12.3.2 SCI3nRDR — Receive Data Register

This register is used to store receive data. When one frame of data has been received, it is transferred from SCI3nRSR to this register allowing SCI3nRSR to receive the next data. SCI3nRSR and SCI3nRDR function as a double-buffer in this way, allowing continuous receive operations. Be sure to check that the SCI3nSSR.RDRF bit is set to 1 before reading SCI3nRDR. When the data length is 7 bits, receive data is stored in bit 0 to 6 and bit 7 is fixed to 0 regardless of the SCI3nSCMR.SINV bit.

Access: This register can be read in 8-bit units.

Address: <SCI3n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	SCI3nRDR							
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 12.9 SCI3nRDR Register Contents

Bit Position	Bit Name	Function
7 to 0	SCI3nRDR	Receive Data Register These bits store receive data.

12.3.3 SCI3nTDR — Transmit Data Register

This register is used to store transmit data. When SCI3nTSR empty is detected, the transmit data written to SCI3nTDR is transferred to SCI3nTSR and transmission starts. The double-buffered structure of SCI3nTDR and SCI3nTSR allows continuous serial transmission. If the next transmit data has been written to SCI3nTDR when one frame of data is sent, the transmit data is transferred to SCI3nTSR to continue transmission. Be sure to check that the SCI3nSSR.TDRE bit is set to 1 before writing transmit data to SCI3nTDR.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + $0C_H$

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	SCI3nTDR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.10 SCI3nTDR Register Contents

Bit Position	Bit Name	Function
7 to 0	SCI3nTDR	Transmit Data Register These bits store transmit data.

12.3.4 SCI3nSMR — Serial Mode Register

SCI3nSMR is a register used to select the transfer format and the clock source for the on-chip baud rate generator.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 00_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. Writable only when TE = RE = 0.

Table 12.11 SCI3nSMR Register Contents

Bit Position	Bit Name	Function
7	CM	Communications Mode 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	Character Length (Valid only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length. LSB-first is fixed and the MSB (bit 7) in SCI3nTDR is not transmitted in transmission. In clock synchronous mode, the data length is fixed to 8 bits.
5	PE	Parity Enable (Valid only in asynchronous mode) When this bit is set to 1, a parity bit is added to transmit data and the parity bit is checked in reception. Regardless of the setting of this bit, no parity bit is added or checked in the multi-processor format.
4	PM	Parity Mode (Valid only when PE = 1 in asynchronous mode) 0: Even parity 1: Odd parity When even parity is set, the value of the appended parity bit causes the total number of 1-valued bits in a transmitted or received character plus the parity bit to be even. Similarly, when odd parity is set, the value of the appended parity bit causes the total number of 1-valued bits in a transmitted or received character plus the parity bit to be odd.
3	STOP	Stop Bit Length (Valid only in asynchronous mode) 0: 1 stop bit for transmission 1: 2 stop bits for transmission In reception, only the first stop bit is checked regardless of the setting of this bit. If the second stop bit is 0, it is treated as the start bit of the next transmission frame.
2	MP	Multi-Processor Mode (Valid only in asynchronous mode) When this bit is set to 1, the multi-processor communication function is enabled. In multi-processor mode, settings of the PE and PM bits are invalid.
1, 0	CKS[1:0]	Clock Select 1, 0 These bits select the clock source for the on-chip baud rate generator. 00: PCLK clock (m = 0) 01: PCLK/4 clock (m = 1) 10: PCLK/16 clock (m = 2) 11: PCLK/64 clock (m = 3) For the relation between the setting of these bits and the baud rate, see **Section 12.3.9, SCI3nBRR — Bit Rate Register**. The character m is the decimal notation of the value of m in **Section 12.3.9, SCI3nBRR — Bit Rate Register**.

12.3.5 SCI3nSCR — Serial Control Register

SCI3nSCR is a register used for the transmission/reception control, interrupt control, and transmission/reception clock source selection listed below. For interrupt requests, see **Section 12.4.5, Interrupt Sources**.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W	R/W*2	R/W*2

Note 1. While the CM bit in SCI3nSMR is 1, a value of 1 can be written only when TE = 0 and RE = 0. After the TE or RE bit is set to 1, only 0 can be written to TE and RE. While the CM bit in SCI3nSMR is 0, writing is enabled at any timing.

Note 2. Writable only when TE = 0 and RE = 0. Also, writable at the same time when TE = 0 and RE = 0 are written.

Table 12.12 SCI3nSCR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TIE	Transmit Interrupt Enable When this bit is set to 1, TXI interrupt request is enabled. TXI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0, or clearing the TIE bit.
6	RIE	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled. RXI and ERI interrupt requests can be cancelled by reading 1 from the RDRF, FER, PER, or ORER flag and then clearing the flag to 0, or clearing the RIE bit.
5	TE	Transmit Enable When this bit is set to 1, transmission is enabled. In this state, serial transmission is started by writing transmit data to SCI3nTDR and clearing the TDRE flag in SCI3nSSR to 0. Be sure to set SCI3nSMR before setting the TE bit to 1 to determine the transmission format. When this bit is set to 0 to disable transmission, the TDRE flag in SCI3nSSR is fixed to 1.
4	RE	Receive Enable When this bit is set to 1, reception is enabled. In this state, serial reception is started by detecting a start bit in asynchronous mode or the input of synchronous clock in clock synchronous mode. Be sure to set SCI3nSMR before setting the RE bit to 1 to determine the reception format. Even if reception is disabled by clearing this bit, the RDRF, FER, PER, and ORER flags are not affected and the previous value is retained.
3	MPIE	Multiprocessor Interrupt Enable (Valid only when MP in SCI3nSMR = 1 in asynchronous mode) When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, this bit is automatically cleared to 0, and normal reception is resumed. For details, see **Section 12.4.2, Multi-Processor Communication Function**. When the receive data includes MPB = 0 in SCI3nSSR, the receive data is not transferred from SCI3nRSR to SCI3nRDR, a receive error is not detected, and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the receive data includes MPB = 1 in SCI3nSSR, the MPB bit in SCI3nSSR is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCI3nSCR is set to 1), and setting the FER and ORER flags to 1 is enabled.

Table 12.12 SCI3nSCR Register Contents (2/2)

Bit Position	Bit Name	Function
2	TEIE	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is enabled. TEI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0 to clear the TEND flag to 0, or clearing the TEIE bit.
1, 0	CKE[1:0]	Clock Enable 1, 0 These bits select the clock source and the SCInSCK pin function. For asynchronous mode 00: On-chip baud rate generator (The SCInSCK pin functions as an input/output port.) 01: On-chip baud rate generator (The clock with the same frequency as the bit rate is output from the SCInSCK pin.) 1X: Setting prohibited For clock synchronous mode 0X: Internal clock (The SCInSCK pin functions as a clock output pin.) 1X: External clock (The SCInSCK pin functions as a clock input pin.)

Note: X: Don't care

NOTE

When writing to any bit other than the MPIE bit of this register, use a store instruction such that the value of the MPIE bit becomes 0.

Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently set the MPIE bit to 1.

12.3.6 SCI3nSSR — Serial Status Register

SCI3nSSR consists of SCI3 status flags and transfer multi-processor bits.

The TDRE, RDRF, ORER, PER, and FER flags can only be cleared.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 10_H

Value after reset: 84_H

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Value after reset	1	0	0	0	0	1	0	0
R/W	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R/W

Note 1. Only 0 can be written to clear the flag.

Table 12.13 SCI3nSSR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TDRE	Transmit Data Register Empty Indicates whether or not transmit data exists in SCI3nTDR. [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the data in SCI3nTDR has been transferred to SCI3nTSR so that new data can be written to SCI3nTDR [Clearing condition] • Writing 0 to TDRE after reading TDRE = 1 When transmit data is written to SCI3nTDR while TE = 1
6	RDRF	Receive Data Register Full Indicates whether or not receive data exists in SCI3nRDR [Setting condition] • When reception finishes successfully and the receive data is transferred from SCI3nRSR to SCI3nRDR [Clearing conditions] • Writing 0 to RDRF after reading RDRF = 1 • When data is read from SCI3nRDR Even when the RE bit in SCI3nSCR is cleared, the RDRF flag is not affected and the previous value is retained. • Note that completing the reception of the next data with the RDRF flag set to 1 will cause an overrun error, resulting in the loss of the receive data.
5	ORER	Overrun Error Indicates that an overrun error has occurred during reception and the reception abended. [Setting condition] • When the next data is received while RDRF = 1 In SCI3nRDR, receive data prior to an overrun error is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be continued. In clock synchronous mode, serial transmission also cannot be continued. [Clearing condition] • Writing 0 to ORER after reading ORER = 1 Even when the RE bit in SCI3nSCR is cleared, the ORER flag is not affected and the previous value is retained.

Table 12.13 SCI3nSSR Register Contents (2/2)

Bit Position	Bit Name	Function
4	FER	Framing Error Indicates that a framing error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • When the stop bit is 0 In 2-stop-bit mode, only whether the first stop bit is 1 is checked but the second stop bit is not checked. Although the receive data when a framing error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the FER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to FER after reading FER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the FER flag is not affected and the previous value is retained.
3	PER	Parity Error Indicates that a parity error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • Although the receive data when a parity error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the PER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to PER after reading PER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the PER flag is not affected and the previous value is retained.
2	TEND	Transmit End [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the TDRE flag is 1 while the last bit of a transmit character is being transmitted [Clearing condition] • Writing 0 to the TDRE flag after reading TDRE = 1 • When transmit data is written to SCI3nTDR while TE = 1
1	MPB	Multi-processor Bit Holds the value of the multi-processor bit in the received frame.
0	MPBT	Multi-processor Bit Transfer Sets the value of the multi-processor bit to be added to the transmit frame.

12.3.7 SCI3nSCMR — Serial Transfer Format Register

SCI3nSCMR is a register to select transfer format for both asynchronous mode and clock synchronous mode.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 18_H

Value after reset: $F2_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SDIR	SINV	—	—
Value after reset	1	1	1	1	0	0	1	0
R/W	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 12.14 SCI3nSCMR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 1. When writing, always write the value after reset.
3	SDIR	Serial Data Transfer Direction (Valid in asynchronous mode and clock synchronous mode) This bit is used to select the direction of serial/parallel conversion. 0: Transfer with LSB-first 1: Transfer with MSB-first This bit is valid only when the transfer format is 8-bit data. For 7-bit data, LSB-first transfer is used.
2	SINV	Serial Data Invert (Valid in asynchronous mode and clock synchronous mode) This bit is used to invert the transfer data logic level. The SINV bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SCI3nSMR. 0: SCI3nTDR data is transmitted as it is, and receive data is stored in SCI3nRDR as it is. 1: SCI3nTDR data is inverted and transmitted, and receive data is inverted and stored in SCI3nRDR.
1	—	Reserved These bits are always read as 1. When writing, always write the value after reset.
0	—	Reserved These bits are always read as 0. When writing, always write the value after reset.

12.3.8 SCI3nSEMR — Serial Extended Mode Register

SCI3nSEMR is a register to select a 1-bit period.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + $1C_H$

Value after reset: 04_H

Bit	7	6	5	4	3	2	1	0
	BRME	MDDRS	—	—	ABCS	RXDMON	—	—
Value after reset	0	0	0	0	0	1	0	0
R/W	R/W*1	R/W*1	R	R	R/W*1	R	R	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 12.15 SCI3nSEMR Register Contents

Bit Position	Bit Name	Function
7	BRME	Bit Rate Modulation Enable When this bit is set to 1, the bit rate modulation function is enabled.
6	MDDRS	Modulation Duty Register Select This bit is used to select an accessible register. 0: SCI3nBRR is accessible. 1: SCI3nMDDR is accessible.
5, 4	—	Reserved These bits are always read as 0.
3	ABCS	Asynchronous Reference Clock Select (Valid only in asynchronous mode) This bit is used to select the reference clock for 1-bit period. 0: Operates on the reference clock with a frequency of 16 times the transfer rate. 1: Operates on the reference clock with a frequency of 8 times the transfer rate (double-speed operation).
2	RXDMON	Serial Input Data Monitor This bit indicates the SCInRxD pin state. 0: SCInRxD pin state is the low level. 1: SCInRxD pin state is the high level.
1, 0	—	Reserved These bits are always read as 0.

12.3.9 SCI3nBRR — Bit Rate Register

SCI3nBRR is an 8-bit register to adjust the bit rate. Since each SCI3 channel has an independent baud rate generator, different bit rates can be set for each channel. **Table 12.17** shows the relationship between the setting (N) in SCI3nBRR and the bit rate (B) in normal asynchronous mode and clock synchronous mode. The value after reset of SCI3nBRR is FFH. SCI3nBRR is allocated at the same address as SCI3nMDDR and is selected when MDDRS in SCI3nSEMR is 0. This register is writable only when TE = 0 and RE = 0.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 04_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	BRR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.16 SCI3nBRR Register Contents

Bit Position	Bit Name	Function
7 to 0	BRR	Baud rate generator setting (0 ≤ N ≤ 255)

Table 12.17 Relationship between Setting N in SCI3nBRR and Bit Rate B

Mode	ABCS Setting	Bit Rate	Mean Error
Asynchronous	0	$B = \frac{PCLK \times 10^6}{64 \times 2^{2m-1} \times (N + 1)}$	$Error\ (\%) = \{ \frac{PCLK \times 10^6}{B \times 64 \times 2^{2m-1} \times (N + 1)} - 1 \} \times 100$
	1	$B = \frac{PCLK \times 10^6}{32 \times 2^{2m-1} \times (N + 1)}$	$Error\ (\%) = \{ \frac{PCLK \times 10^6}{B \times 32 \times 2^{2m-1} \times (N + 1)} - 1 \} \times 100$
Clock synchronous	—	$B = \frac{PCLK \times 10^6}{8 \times 2^{2m-1} \times (N + 1)}$	

Note: B: Bit rate (bps)
N: SCI3nBRR setting for baud rate generator (0 ≤ N ≤ 255)
PCLK: Operating frequency (MHz)
m: Determined by the SCI3nSMR setting shown in the following table.

SCI3nSMR Setting		
CKS1	**CKS0**	**m**
0	0	0
0	1	1
1	0	2
1	1	3

Table 12.18 lists sample N settings of the SCI3nBRR register in asynchronous mode. **Table 12.19** lists the maximum settable bit rates.

Table 12.18 Examples of BRR Settings for Bit Rates (Asynchronous Mode)

Bit Rate (bps)	Operating Frequency PCLK = 40 (MHz)							
	SCI3nSEMR.ABCS = 0				SCI3nSEMR.ABCS = 1			
	m	N	Actual Bit Rate (bps)	Error (%)	m	N	Actual Bit Rate (bps)	Error (%)
110	3	177	109.73	-0.25	—	—	—	—
150	3	129	150.24	0.16	3	255	152.59	1.73
300	3	64	300.48	0.16	3	129	300.48	0.16
600	2	129	600.96	0.16	3	64	600.96	0.16
1200	2	64	1201.92	0.16	2	129	1201.92	0.16
2400	1	129	2403.85	0.16	2	64	2403.85	0.16
4800	1	64	4807.69	0.16	1	129	4807.69	0.16
9600	0	129	9615.38	0.16	1	64	9615.38	0.16
19200	0	64	19230.77	0.16	0	129	19230.77	0.16
31250	0	39	31250.00	0.00	0	79	31250.00	0.00
38400	0	32	37878.79	-1.36	0	64	38461.54	0.16

Table 12.19 Maximum Bit Rate (Asynchronous Mode)

PCLK (MHz)	Setting			Maximum Bit Rate (bps)
	ABCS Setting	m	N	
40	0	0	0	1250000
	1	0	0	2500000

Table 12.20 lists sample N settings of the SCI3nBRR register in clock synchronous mode.

Table 12.20 Examples of Bit Rate Settings for Clock Synchronous Mode

Bit Rate (bps)	Operating Frequency PCLK = 40 (MHz)		
	m	N	Actual Bit Rate (bps)
1k	3	155	1001.60
2.5k	3	62	2480.16
5k	2	124	5000.00
10k	2	62	9920.63
25k	1	99	25000.00
50k	1	49	50000.00
100k	0	99	100000.00
250k	0	39	250000.00
500k	0	19	500000.00
1M	0	9	1000000.00
2M	0	4	2000000.00
2.5M	0	3	2500000.00
5M	0	1	5000000.00

Table 12.21 Maximum Bit Rate when Internal Clock Is Output (Clock Synchronous Mode)

PCLK (MHz)	m	N	Maximum Bit Rate (bps)
40	0	1	5000000.00

12.3.10 SCI3nMDDR — Modulation Duty Register

SCI3nMDDR is a register to correct the bit rate adjusted by SCI3nBRR. The value after reset of SCI3nMDDR is FFH. When the BRME bit in SCI3nSEMR is set to 1, the bit rate generated by the on-chip baud rate generator is corrected to SCI3nMDDR/256 on average. **Table 12.23** shows the relationship between the SCI3nMDDR setting and the bit rate B. SCI3nMDDR is allocated at the same address as SCI3nBRR and is selected when MDDRS in SCI3nSEMR is 1. This register is writable only when TE = 0 and RE = 0. Bit 7 is fixed to 1.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 04_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	MDDR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.22 SCI3nMDDR Register Contents

Bit Position	Bit Name	Function
7 to 0	MDDR	Baud rate generator setting (128 ≤ MDDR ≤ 255)

Table 12.23 Relationship between SCI3nMDDR Setting and Bit Rate B when Bit Rate Modulation Function Is Used

Mode	ABCS Setting	Bit Rate	Mean Error
Asynchronous	0	$B = \frac{PCLK \times 10^6}{64 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \{\frac{PCLK \times 10^6}{B \times 64 \times 2^{2m-1} \times (256/MDDR) \times (N+1)} - 1\} \times 100$
	1	$B = \frac{PCLK \times 10^6}{32 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \{\frac{PCLK \times 10^6}{B \times 32 \times 2^{2m-1} \times (256/MDDR) \times (N+1)} - 1\} \times 100$
Clock synchronous	—	$B = \frac{PCLK \times 10^6}{8 \times 2^{2m-1} \times (256/MDDR) \times (N+1)}$	

Note: B: Bit rate (bps)
N: SCI3nBRR setting of baud rate generator (0 ≤ N ≤ 255)
PCLK: Operating frequency (MHz)
m: See **Table 12.17, Relationship between Setting N in SCI3nBRR and Bit Rate B**.
MDDR: SCI3nMDDR setting (128 ≤ SCI3nMDDR ≤ 255)

12.4 Functions

12.4.1 Operation in Asynchronous Mode

Figure 12.2 shows the general format for asynchronous serial communications. One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level). In asynchronous serial communications, the communication line is usually held in the mark state (high level). The SCI3 monitors the communication line, and when the SCI3 detects the space state (low level), it recognizes a start bit and starts serial communications. Inside the SCI3, the transmitter and the receiver are independent, enabling full-duplex communications. Both the transmitter and the receiver have a double-buffered structure so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

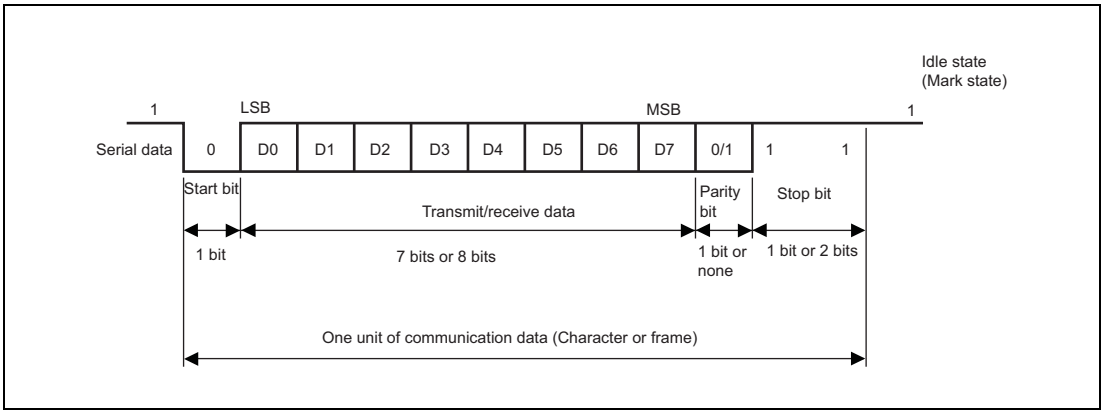


Figure 12.2 Data Format in Asynchronous Serial Communications (Example with 8-Bit Data, Parity, Two Stop Bits)

12.4.1.1 Transmission/Reception Format

Table 12.24 lists settable transmission/reception formats in asynchronous mode. Any of 12 transmission/reception formats can be selected according to the SCI3nSMR setting. For details of the multi-processor bit, see **Section 12.4.2, Multi-Processor Communication Function**.

Table 12.24 Serial Transmission/Reception Formats (Asynchronous Mode)

SMR Setting				Serial Transmission/Reception Format and Frame Length
CHR	PE	MP	STOP	1 2 3 4 5 6 7 8 9 10 11 12
0	0	0	0	S 8-bit data STOP
0	0	0	1	S 8-bit data STOP STOP
0	1	0	0	S 8-bit data P STOP
0	1	0	1	S 8-bit data P STOP STOP
1	0	0	0	S 7-bit data STOP
1	0	0	1	S 7-bit data STOP STOP
1	1	0	0	S 7-bit data P STOP
1	1	0	1	S 7-bit data P STOP STOP
0	—	1	0	S 8-bit data MPB STOP
0	—	1	1	S 8-bit data MPB STOP STOP
1	—	1	0	S 7-bit data MPB STOP
1	—	1	1	S 7-bit data MPB STOP STOP

Note: S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

12.4.1.2 Receive Data Sampling Timing and Reception Margin

In asynchronous mode, the SCI3 operates on a reference clock with a frequency of 16 times (8 times for the double-speed mode) the bit rate. In reception, the SCI3 samples the falling edge of the beginning of the start bit (low level) using the reference clock and performs internal synchronization. As shown in **Figure 12.3**, data is latched at the middle of each bit by sampling receive data at the rising edge of the eighth pulse (fourth pulse for the double-speed mode) of the reference clock.

Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\ [\%] \quad \cdots \text{formula (1)}$$

M: Reception margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SCI3nSEMR = 0, N = 8 when ABCS = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency deviation

Assuming that F (absolute value of clock frequency deviation) = 0, D (duty cycle of clock) = 0.5, and N = 16 in formula (1), the reception margin is obtained by the formula below.

$M = \{0.5 - 1/(2 \times 16)\} \times 100\ [\%] = 46.875\%$

However, this is only the calculated value, and a margin of 20% to 30% should be allowed in system design.

When the bit rate modulation function is used, the reference clock frequency is corrected on average.

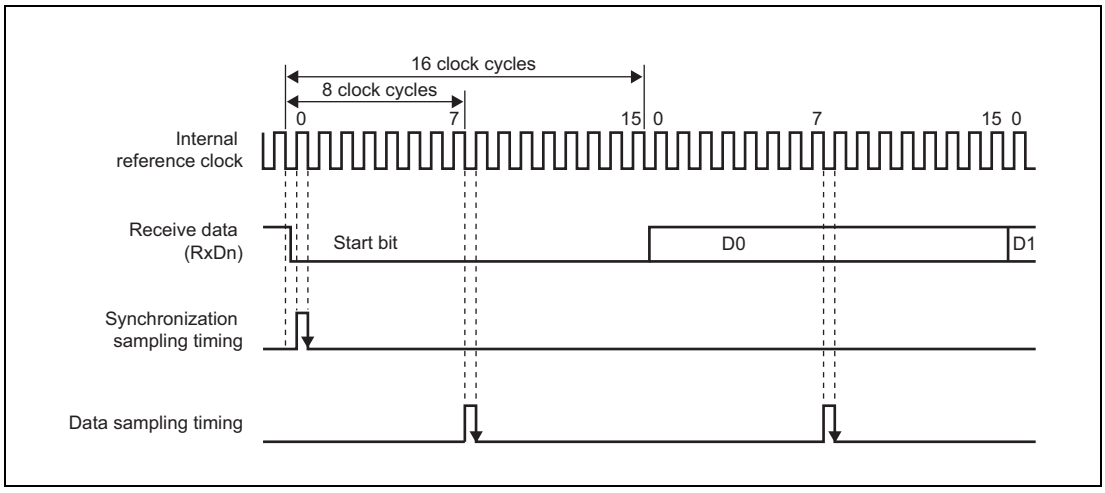


Figure 12.3 Receive Data Sampling Timing in Asynchronous Mode

12.4.1.3 Clock

An internal clock generated by the on-chip baud rate generator can be selected as the SCI3's transmission/reception clock according to the settings of the CM bit in SCI3nSMR and the CKE1 and CKE0 bits in SCI3nSCR. When the SCI3 is operated on an internal clock, the clock can be output from the SCInSCK pin.

For details of clock synchronous mode, see **Section 12.4.3, Operation in Clock Synchronous Mode**.

In asynchronous mode, the frequency of the clock output is equal to the bit rate and the phase is such that the rising edge of the clock comes at the center of the transmit data, as shown in **Figure 12.4**.

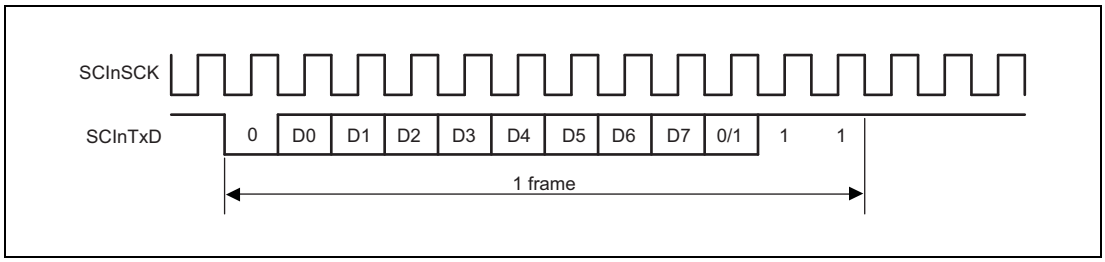


Figure 12.4 Phase Relationship between Output Clock and Transmit Data (Asynchronous Mode)

12.4.1.4 Double-Speed Operation

In addition to the operation described in **Section 12.4.1.3, Clock**, double-speed operation is enabled by the setting of the ABCS bit in SCI3nSEMR.

In double-speed operation, the same operation on a clock with a frequency of 16 times the bit rate in normal operation can be conducted on a clock with a frequency of 8 times the bit rate, meaning that the SCI3 operates on a double transfer rate using the same reference clock.

12.4.1.5 SCI3 Initialization (Asynchronous Mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR and then initialize the SCI3 according to the sample flowchart in **Figure 12.5**. Before changing the operating mode or transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

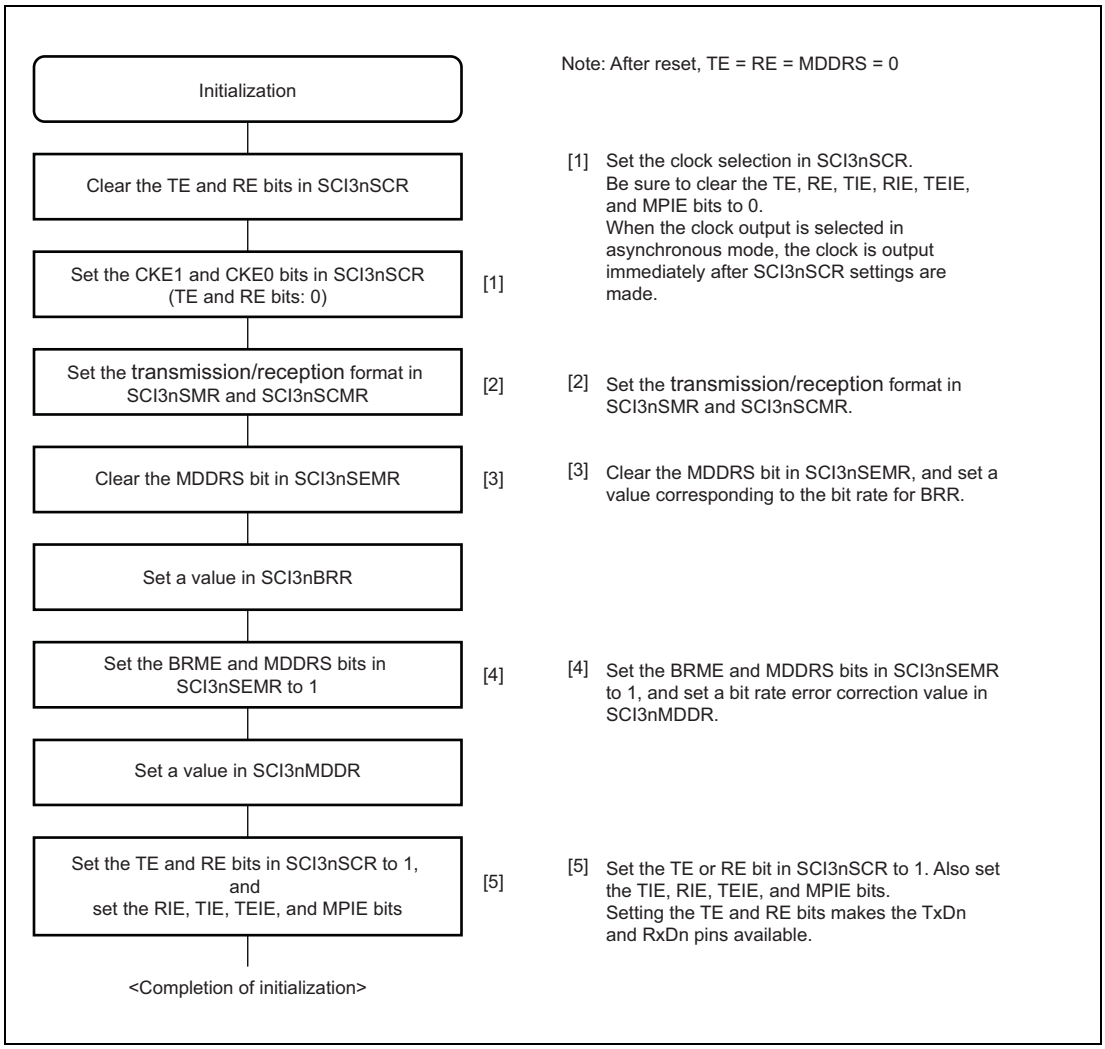


Figure 12.5 Sample Flowchart for SCI3 Initialization

12.4.1.6 Serial Data Transmission (Asynchronous Mode)

Figure 12.6 shows an example of operation for data transmission in asynchronous mode. In data transmission, the SCI3 operates as described below.

(1) When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR. When writing transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.

(2) Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR and the TDRE flag is set to 1. If the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before transmission of the previously transferred data is completed. When using a TEI interrupt request, write the last transmit data to SCI3nTDR and then clear the TIE bit to 0 and set the TEIE bit to 1.

(3) Data is sent from the SCInTxD pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.

(4) The TDRE flag is checked when the stop bit is output.

(5) When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR. After the stop bit has been sent, transmission of the next frame starts.

(6) When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1, the stop bit is sent, and then 1 is output to enter the mark state. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated.

Figure 12.7 shows a sample flowchart for data transmission. **Figure 12.8** shows a sample flowchart for stopping the SCI3 after data transmission.

Supplementary note on operation when data transmission in asynchronous mode is enabled:

When the TE bit is set to 1, the high level (preamble) is output for a frame.

When transmit data is written to SCI3nTDR while the preamble is output, the data is transferred from SCI3nTDR to SCI3nTSR following the completion of the preamble output.

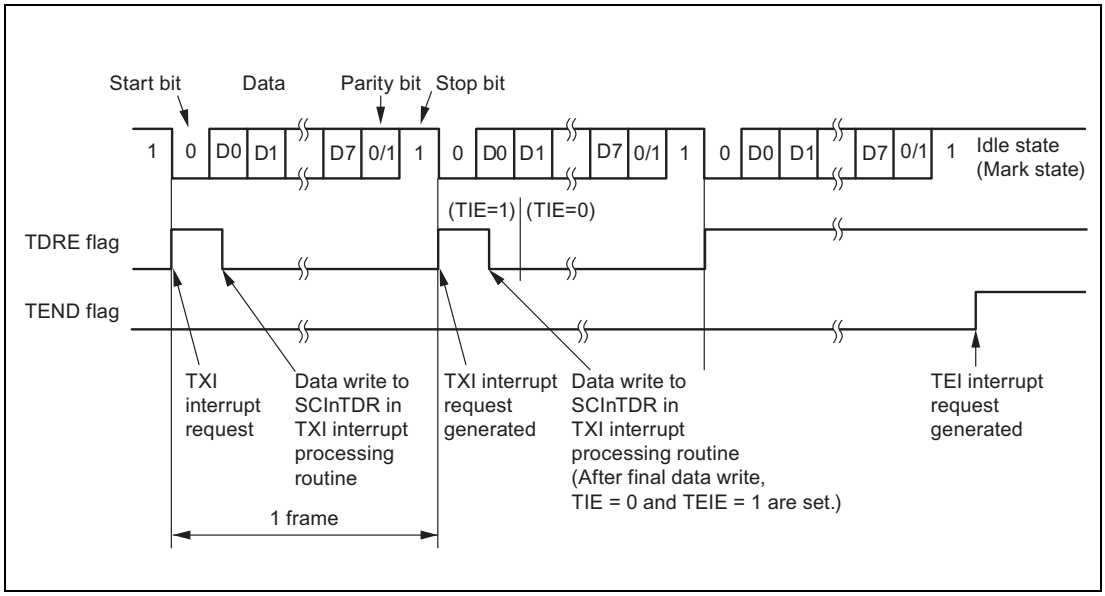


Figure 12.6 Example of Operation for Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

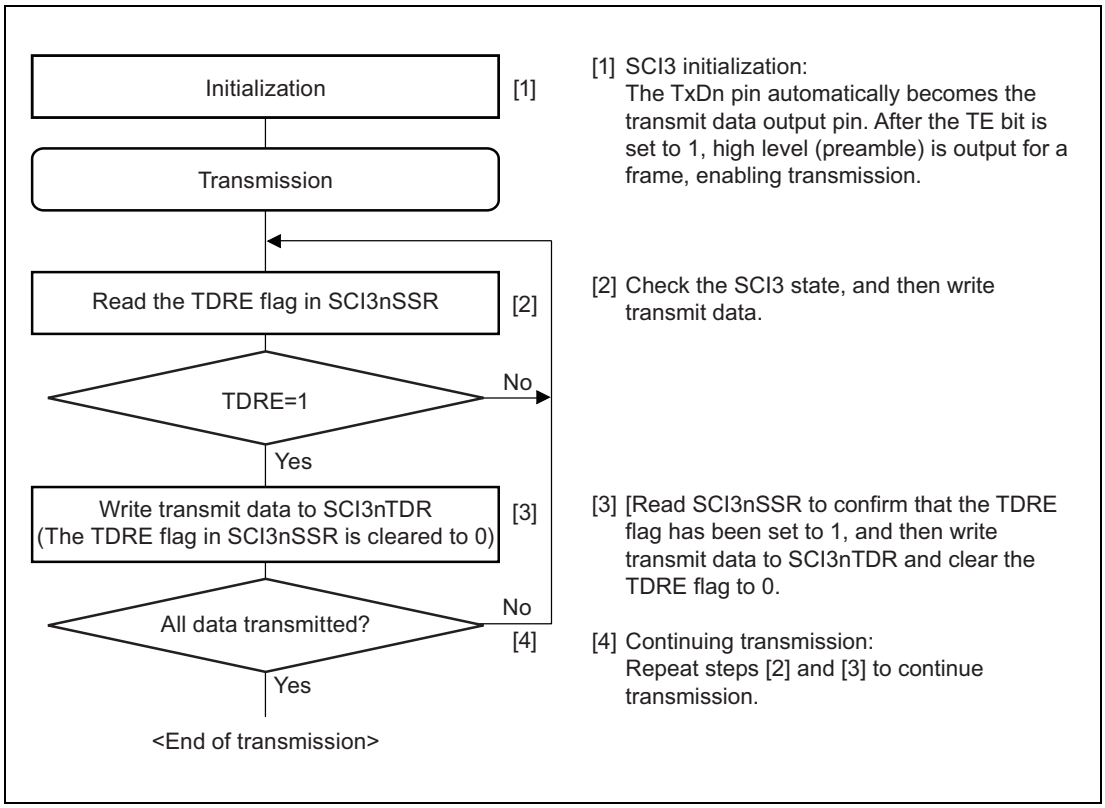


Figure 12.7 Example of Serial Transmission Flowchart

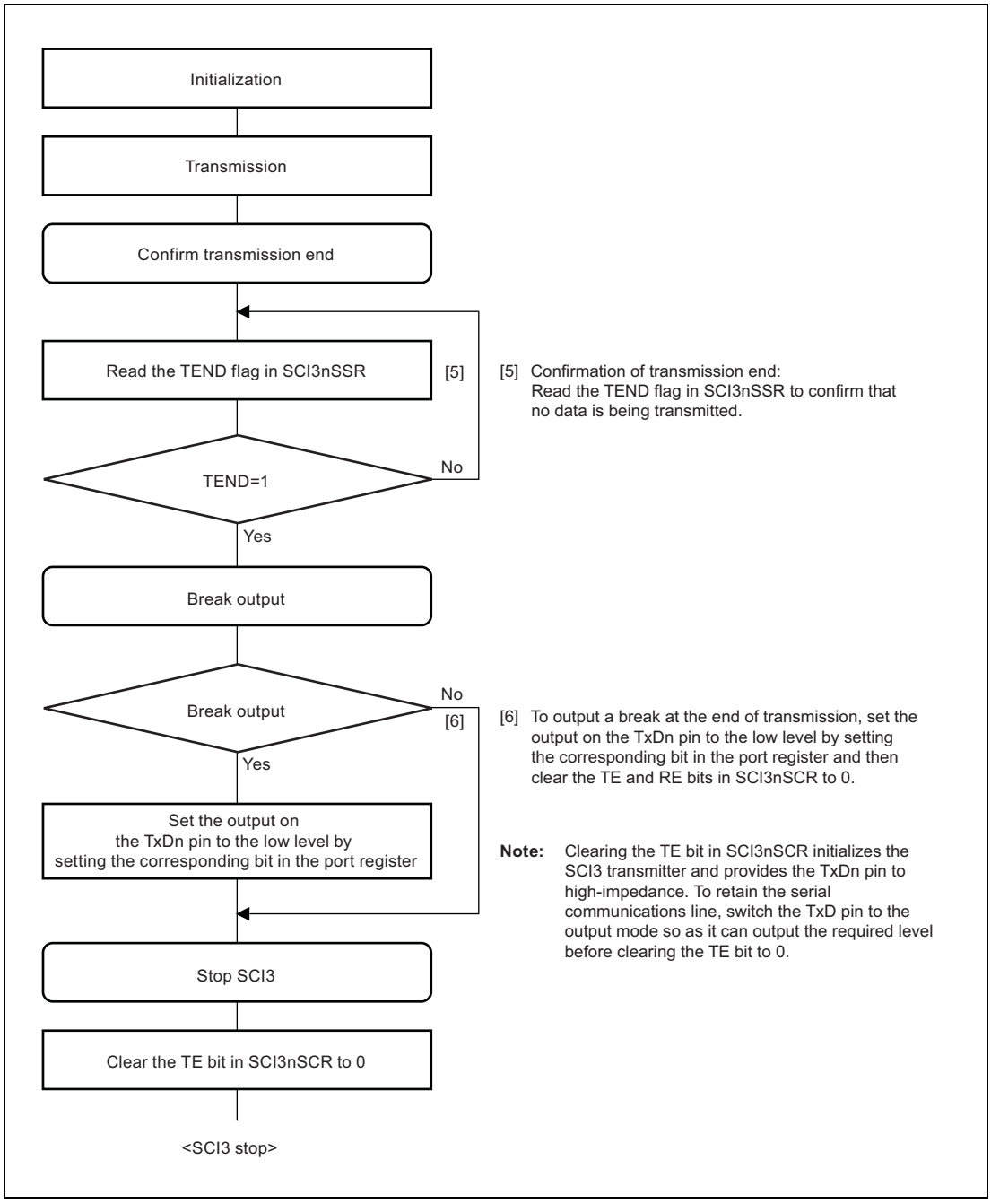


Figure 12.8 Example Flowchart for Stopping the SCI3 after Serial Transmission

12.4.1.7 Serial Data Reception (Asynchronous Mode)

Figure 12.9 shows an example of the operation for data reception in asynchronous mode. In data reception, the SCI3 operates as described below.

(1) When the SCI3 monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in SCI3nRSR, and checks the parity bit and the stop bit.

(2) When an overrun error occurs (the next data has been received with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.

(3) When a parity error is detected, the PER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.

(4) When a framing error (when the stop bit is 0) is detected, the FER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.

(5) When reception finishes successfully, the RDRF flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next receive data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

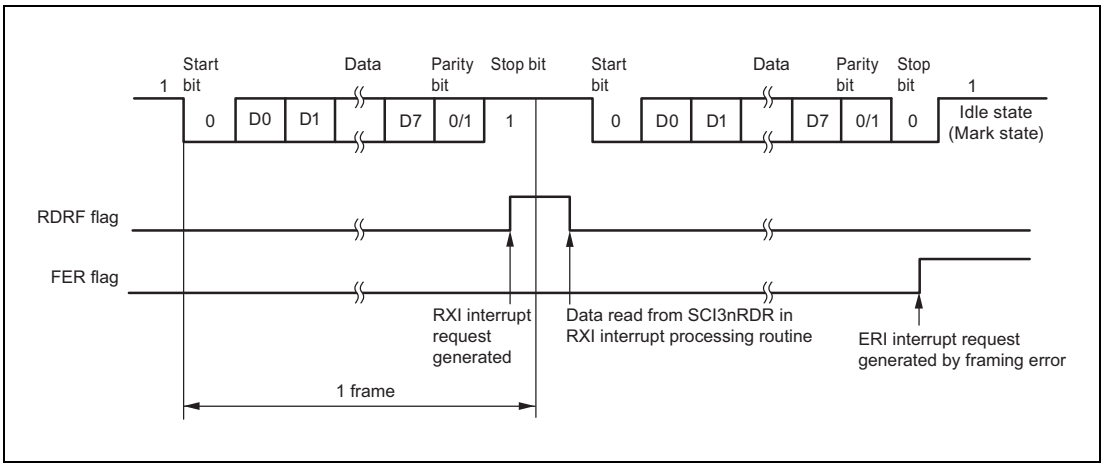


Figure 12.9 Example of Operation for Reception in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

Table 12.25 lists the states of the SCI3nSSR status flags and receive data handling when a receive error is detected. When a receive error is detected, the RDRF flag retains the status before receiving the data. Subsequent data reception is disabled while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF flags before continuing data reception. **Figure 12.10** shows a sample flowchart for data reception.

Table 12.25 SCI3nSSR Status Flags and Receive Data Handling

SCI3nSSR Status Flags					
RDRF*1	ORER	FER	PER	Receive Data	Receive Status
1	0	0	0	Transferred to SCI3nRDR	Successful reception
0	0	1	0	Transferred to SCI3nRDR	Framing error
0	0	0	1	Transferred to SCI3nRDR	Parity error
0	0	1	1	Transferred to SCI3nRDR	Framing error + parity error
1*	1	0	0	Lost	Overrun error
1*	1	1	0	Lost	Overrun error + framing error
1*	1	0	1	Lost	Overrun error + parity error
1*	1	1	1	Lost	Overrun error + framing error + parity error

Note 1. In the case of an overrun error, the RDRF flag retains the state before the data reception.

Note: A sign “+” indicates that two or more receive states occur simultaneously in a single reception operation.

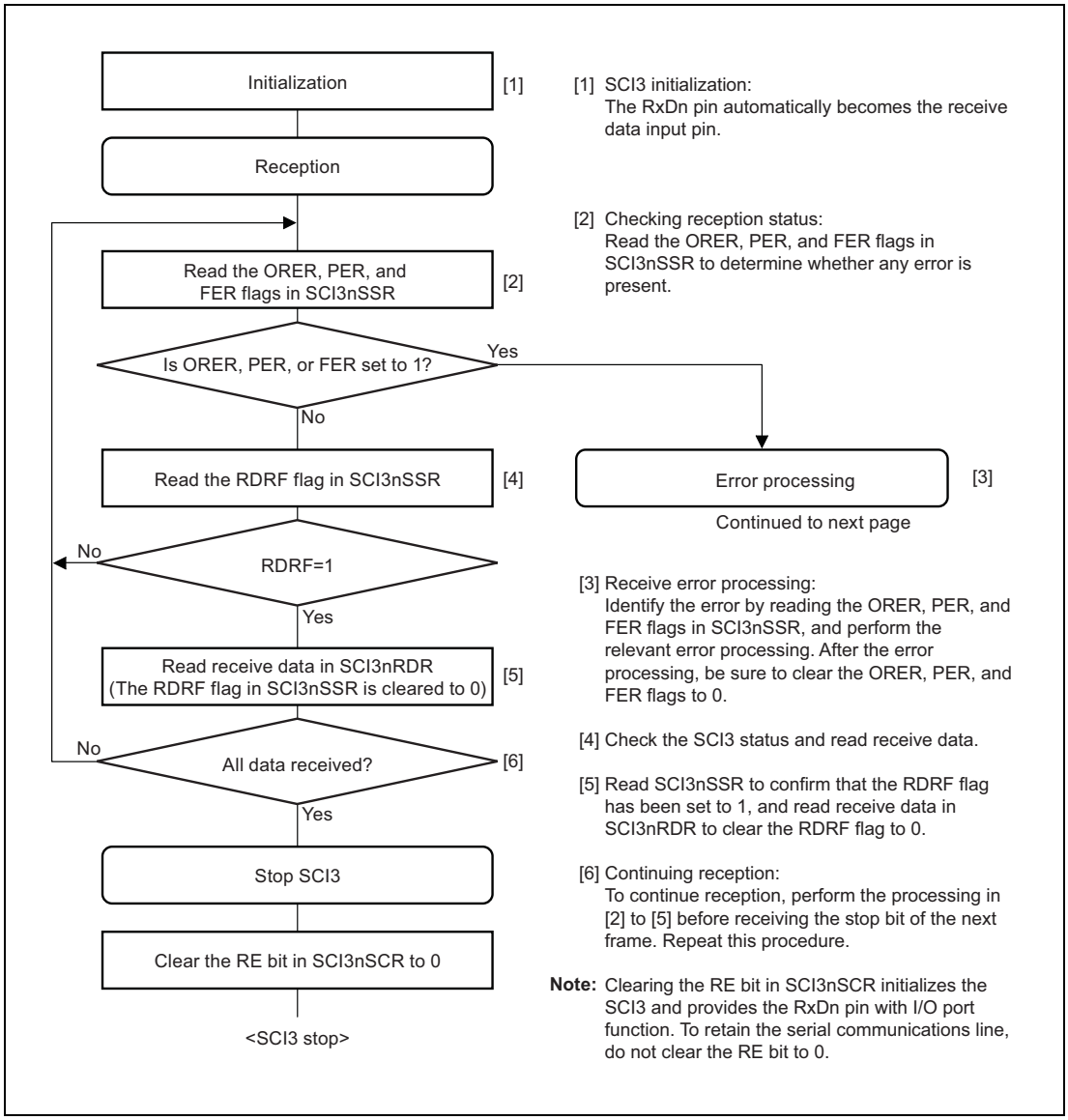


Figure 12.10 Example of Serial Reception Flowchart (1)

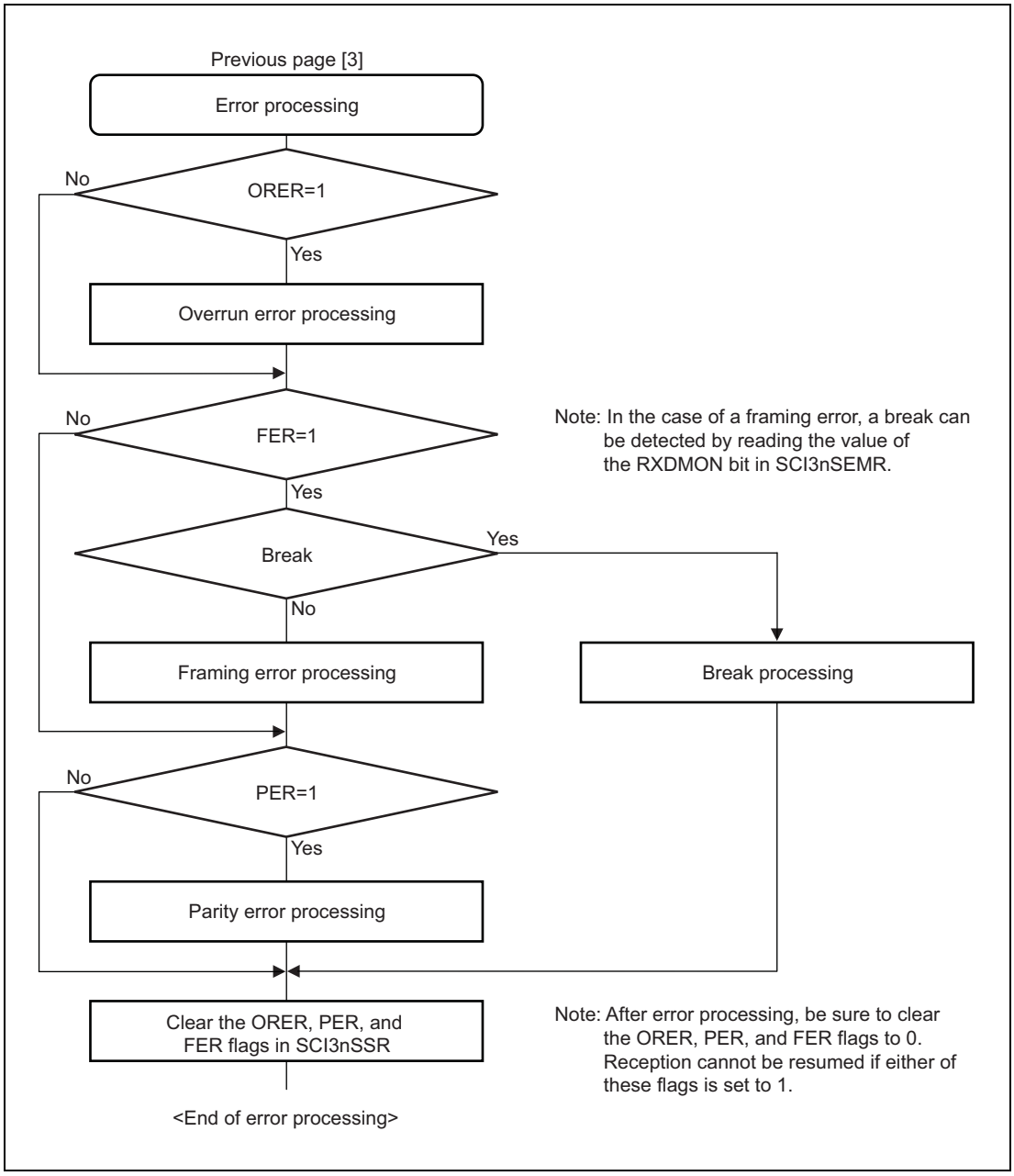


Figure 12.11 Example of Serial Reception Flowchart (2)

12.4.2 Multi-Processor Communication Function

12.4.2.1 Overview and Sample Connection

Using the multi-processor communication function allows data transmission and reception by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish the ID transmission cycle from the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle. When the multi-processor bit is set to 0, it indicates the data transmission cycle. **Figure 12.12** shows an example of communication between processors by using the multi-processor format. First, a transmitting station sends communication data in which the multi-processor bit (= 1) is added to the ID code of the receiving station. Next, the transmitting station sends communication data in which the multi-processor bit (= 0) is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself. When these IDs match, the receiving station receives communication data that is subsequently transmitted. If these IDs do not match, the receiving station skips communication data until it receives communication data in which the multi-processor bit is set to 1.

To support this function, the SCI3 provides the MPIE bit in SCI3nSCR. When the MPIE bit is set to 1, transfer of receive data from SCI3nRSR to SCI3nRDR, detection of a receive error, and setting the RDRF, FER, and ORER flags in SCI3nSSR are disabled until data in which the multi-processor bit is set to 1 is received. Upon receiving a character in which the multi-processor bit is set to 1, the MPB bit in SCI3nSSR is set to 1 and the MPIE bit is automatically cleared to 0, thus returning to a normal reception operation. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. While the MPIE bit is cleared to 0, reception operation is conducted regardless of the multi-processor bit value. The multi-processor bit is stored in the MPB bit in SCI3nSSR.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock that is used for multi-processor communications is also the same as the clock used in the normal asynchronous mode.

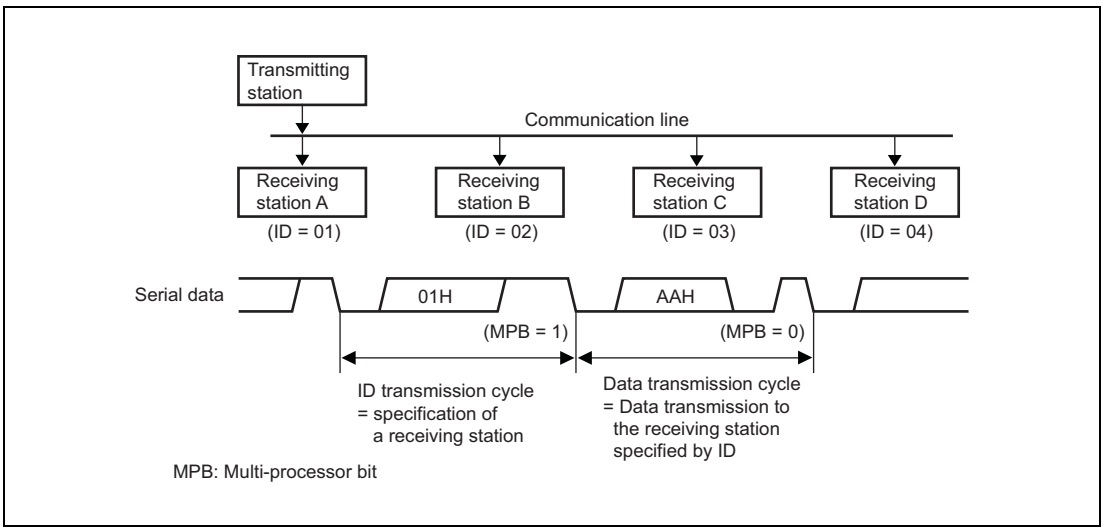


Figure 12.12 Example of Communication Using the Multi-Processor Format (Example of Transmission of Data AAH to Receiving Station A)

12.4.2.2 Multi-Processor Serial Data Transmission

Figure 12.13 shows a sample flowchart of multi-processor data processing. In the ID transmission cycle, send the ID with the MPBT bit in SCI3nSSR set to 1. In the data transmission cycle, send data with the MPBT bit in SCI3nSSR cleared to 0. Other operations are the same as operations in asynchronous mode.

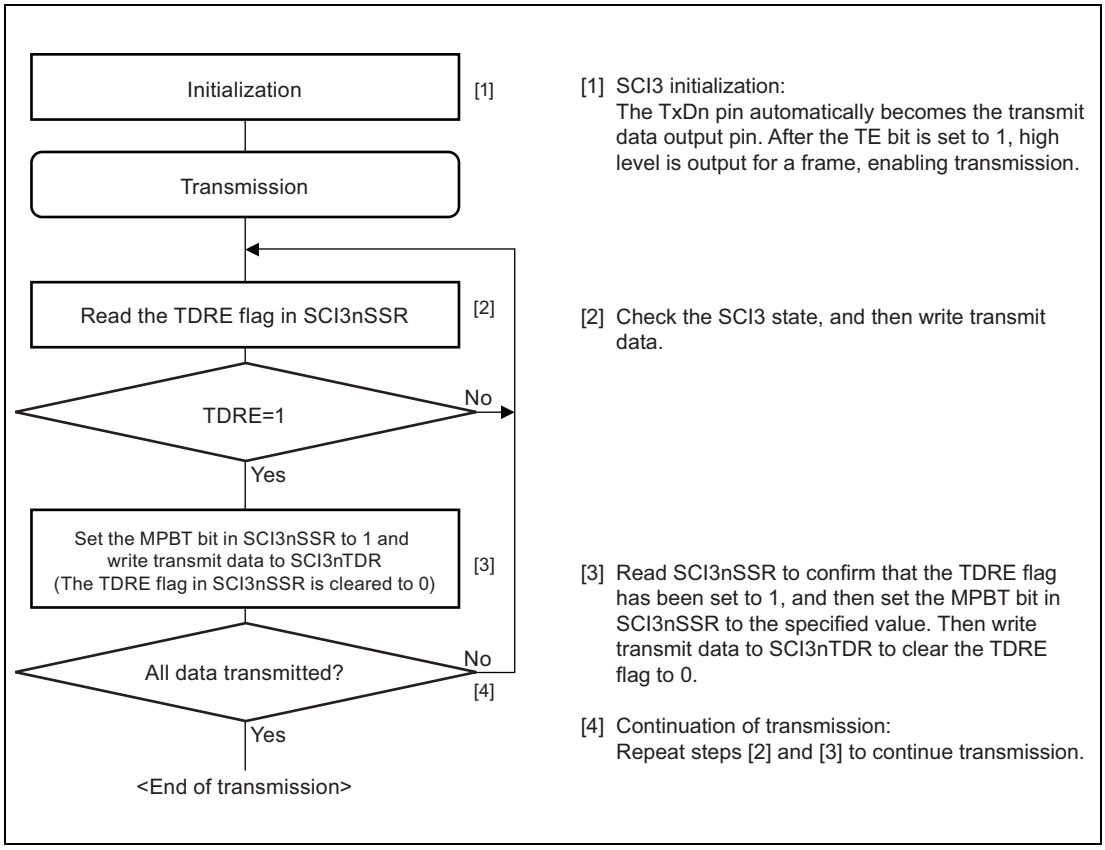


Figure 12.13 Example of Multi-Processor Serial Transmission Flowchart

12.4.2.3 Multi-Processor Serial Data Reception

Figure 12.15 shows sample flowcharts of multi-processor data reception. When the MPIE bit in SCI3nSCR is set to 1, reading the communication data is skipped until communication data in which the multi-processor bit is set to 1 is received. When communication data in which the multi-processor bit is set to 1 is received, the receive data is transferred to SCI3nRDR. At this time, an RXI interrupt request is generated. Other operations are the same as operations in asynchronous mode. **Figure 12.14** shows an example of operation for reception.

CAUTION

Do not write data to SCI3nSCR when communication data in which the multi-processor bit is set to 1 is received. The MPIE bit may not become the desired state.

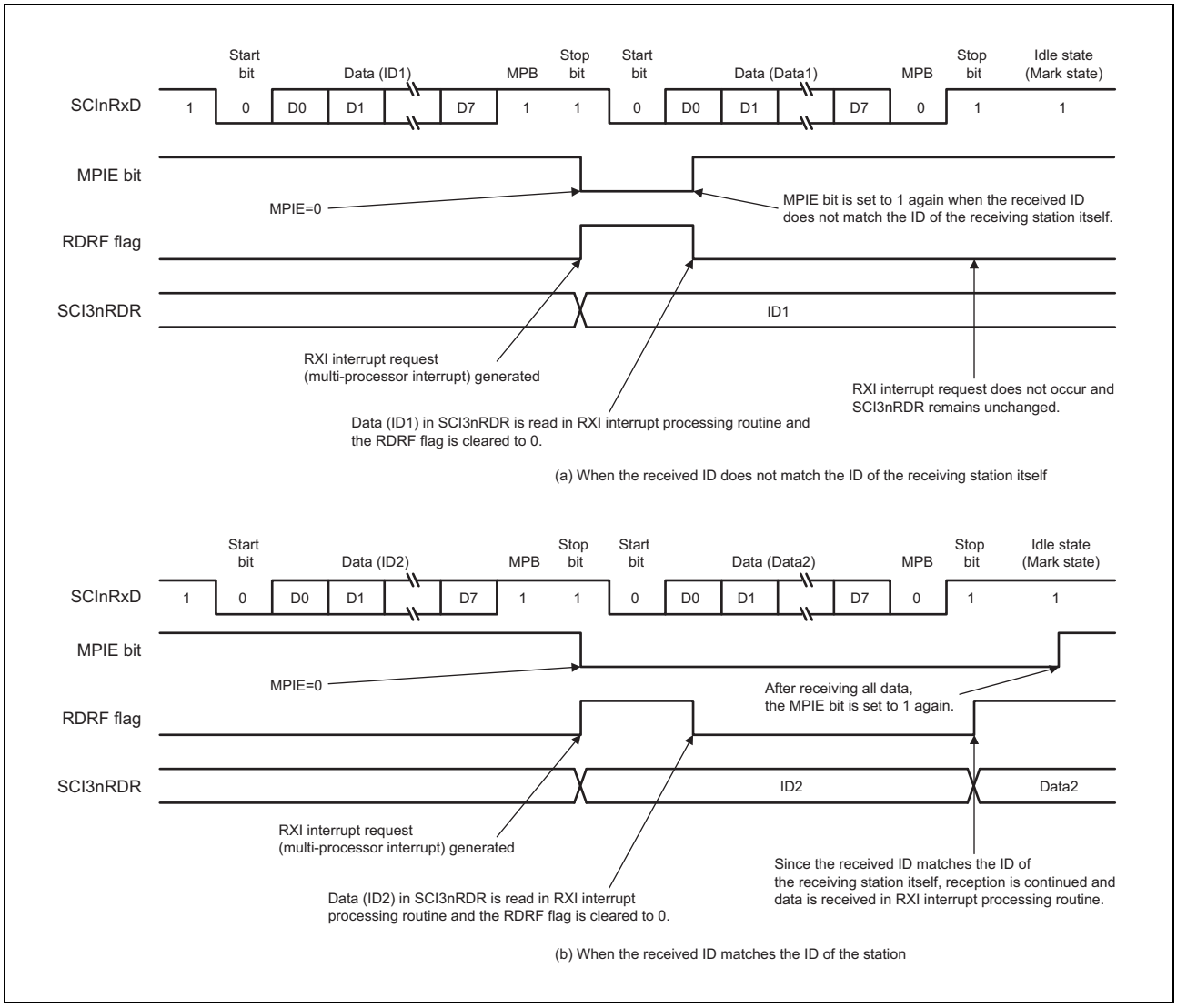


Figure 12.14 Example of SCI3 Reception (8-Bit Data, Multi-Processor Bit, One Stop Bit)

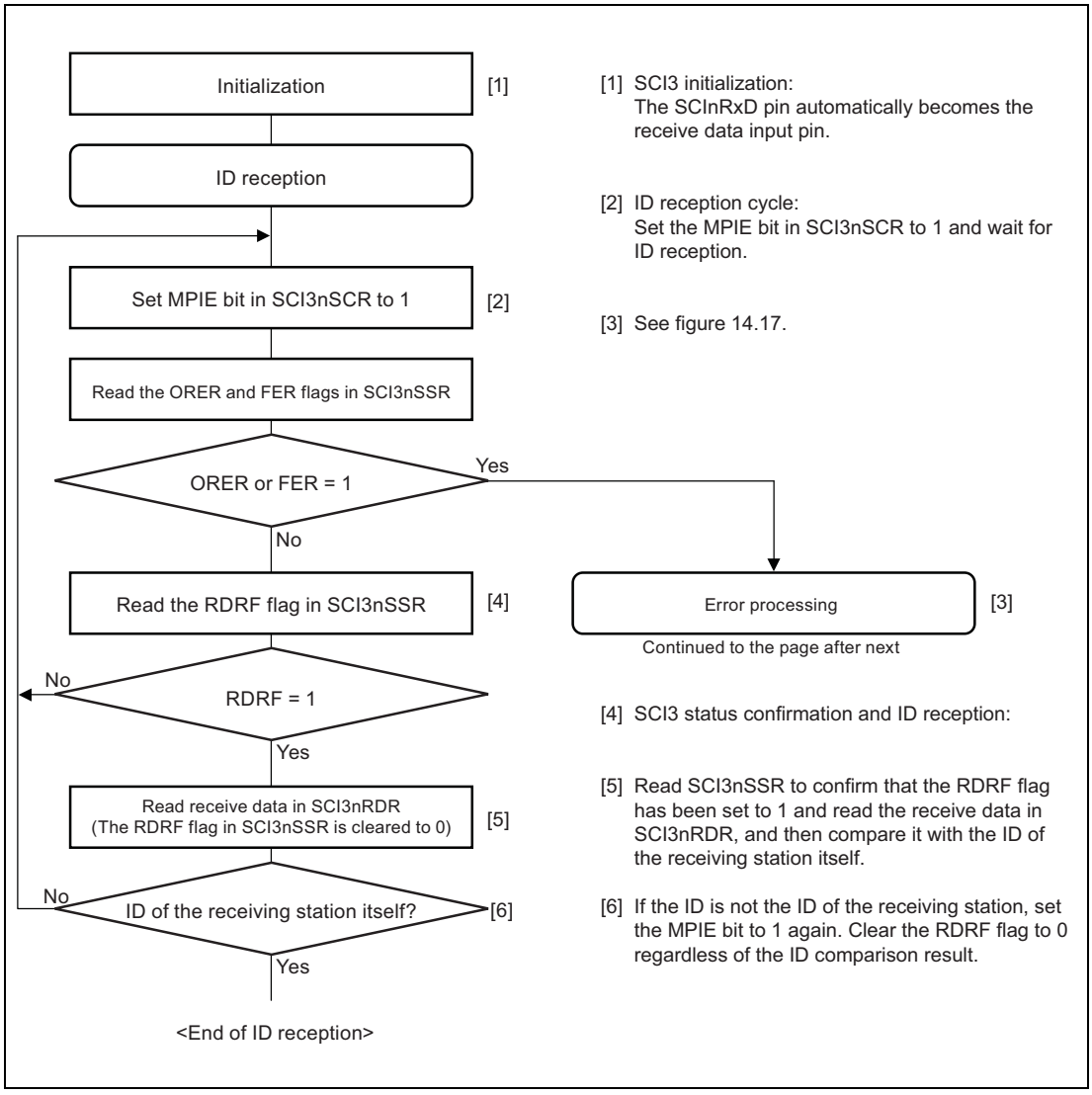


Figure 12.15 Example of Multi-Processor Serial Reception Flowchart (1)

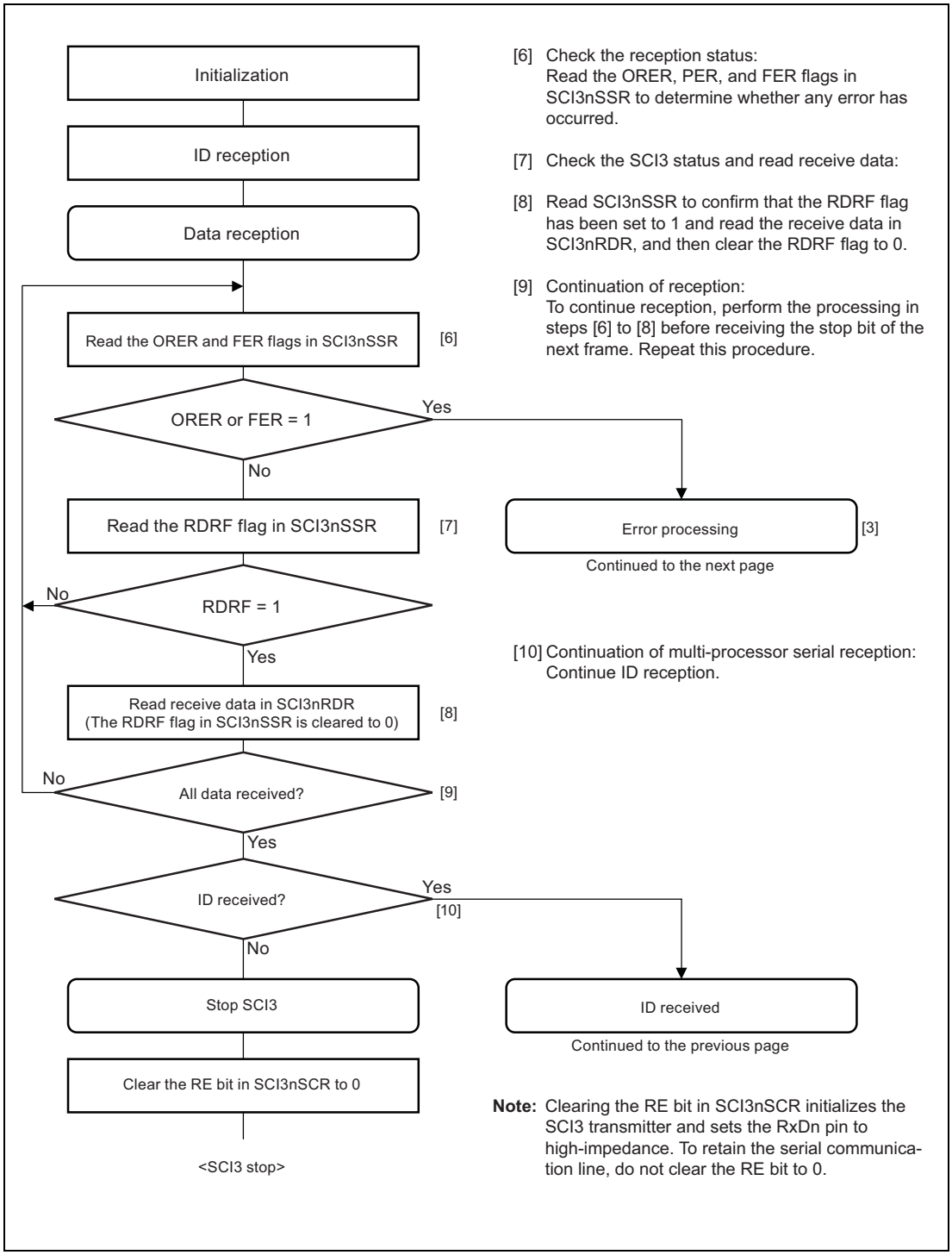


Figure 12.16 Example of Multi-Processor Serial Reception Flowchart (2)

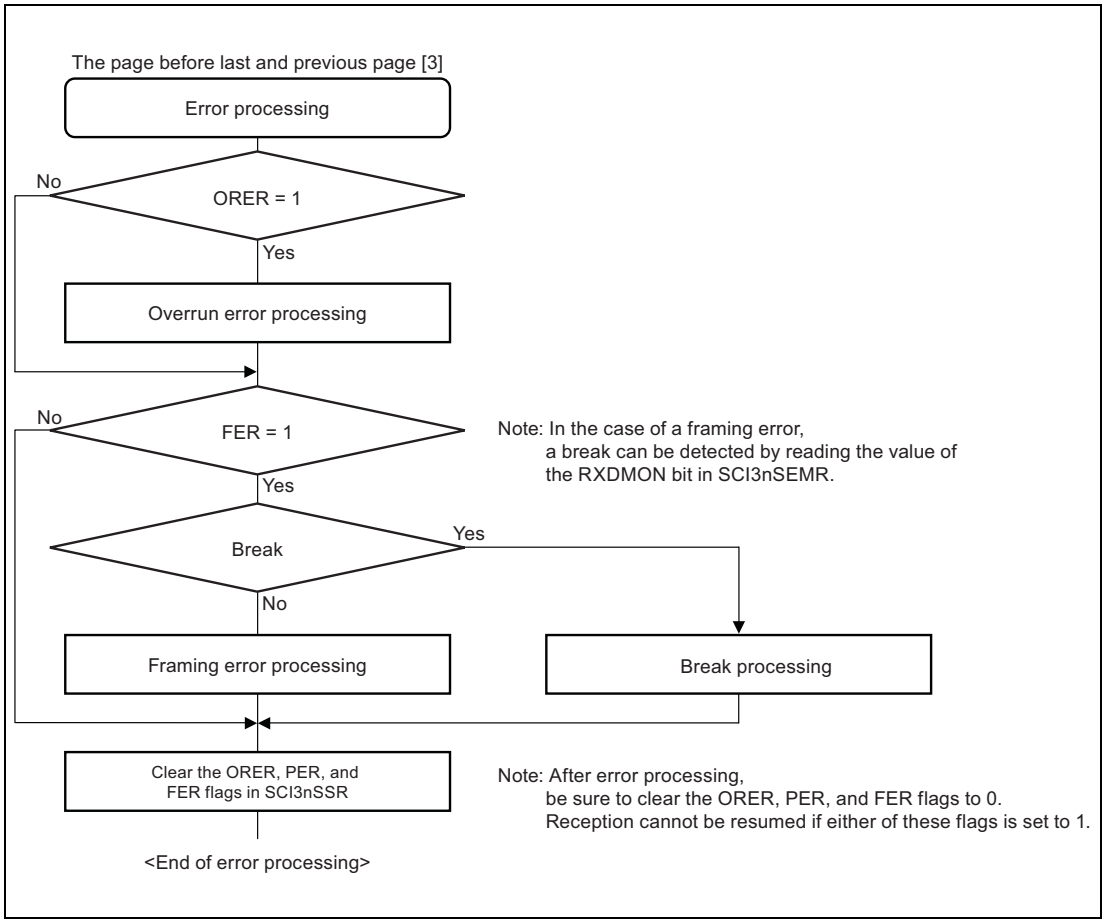


Figure 12.17 Example of Multi-Processor Serial Reception Flowchart (3)

12.4.3 Operation in Clock Synchronous Mode

Figure 12.18 shows the data format for clock synchronous serial data communication. In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In data transmission with the synchronization clock output, the SCI3 outputs data from one falling edge to the next falling edge of the synchronization clock. In data transmission with the synchronization clock input, the SCI3 outputs the first data (bit 0) after starting transfer immediately after clearing the TDRE bit in SCI3nSSR to 0, and then outputs the next bit data after 2 to 3 PCLK clock cycles from the falling edge of the synchronization clock. In data reception, the SCI3 receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last-bit output state. In clock synchronous mode, neither parity bit nor multi-processor bit can be added. The transmitter and the receiver are independent in the SCI3, enabling full-duplex communication by using a common clock. Both the transmitter and the receiver have a double-buffered structure so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.

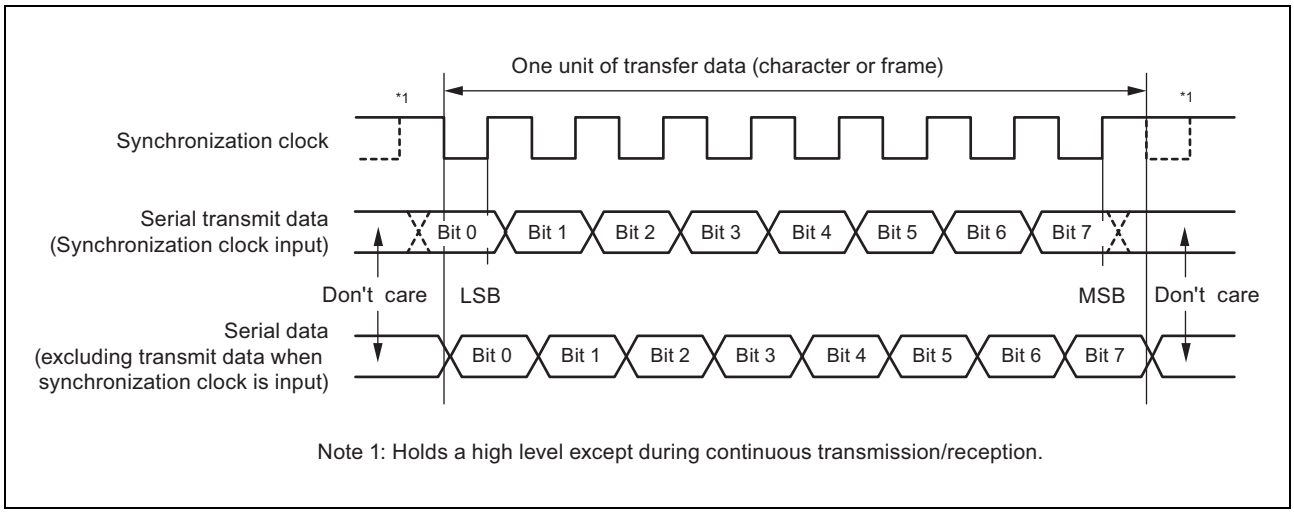


Figure 12.18 Data Format in Clock Synchronous Mode (LSB-First)

12.4.3.1 Clock

An internal clock generated by the on-chip baud rate generator or an external synchronous clock that is input from the SCInSCK pin can be selected by the setting of the CKE1 and CKE0 bits in SCI3nSCR. When operating the SCI3 on the internal clock, a synchronous clock is output from the SCInSCK pin. Eight pulses of the synchronous clock are output during transfer of one character, and the clock is held high while no data is transferred.

12.4.3.2 SCI3 Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR to 0 and then initialize the SCI3 according to the sample flowchart in **Figure 12.19**. To switch the operation between transmission, reception, and transmission/reception, clear the TE and RE bits to 0 and then set these bits to the desired value. Before changing the transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

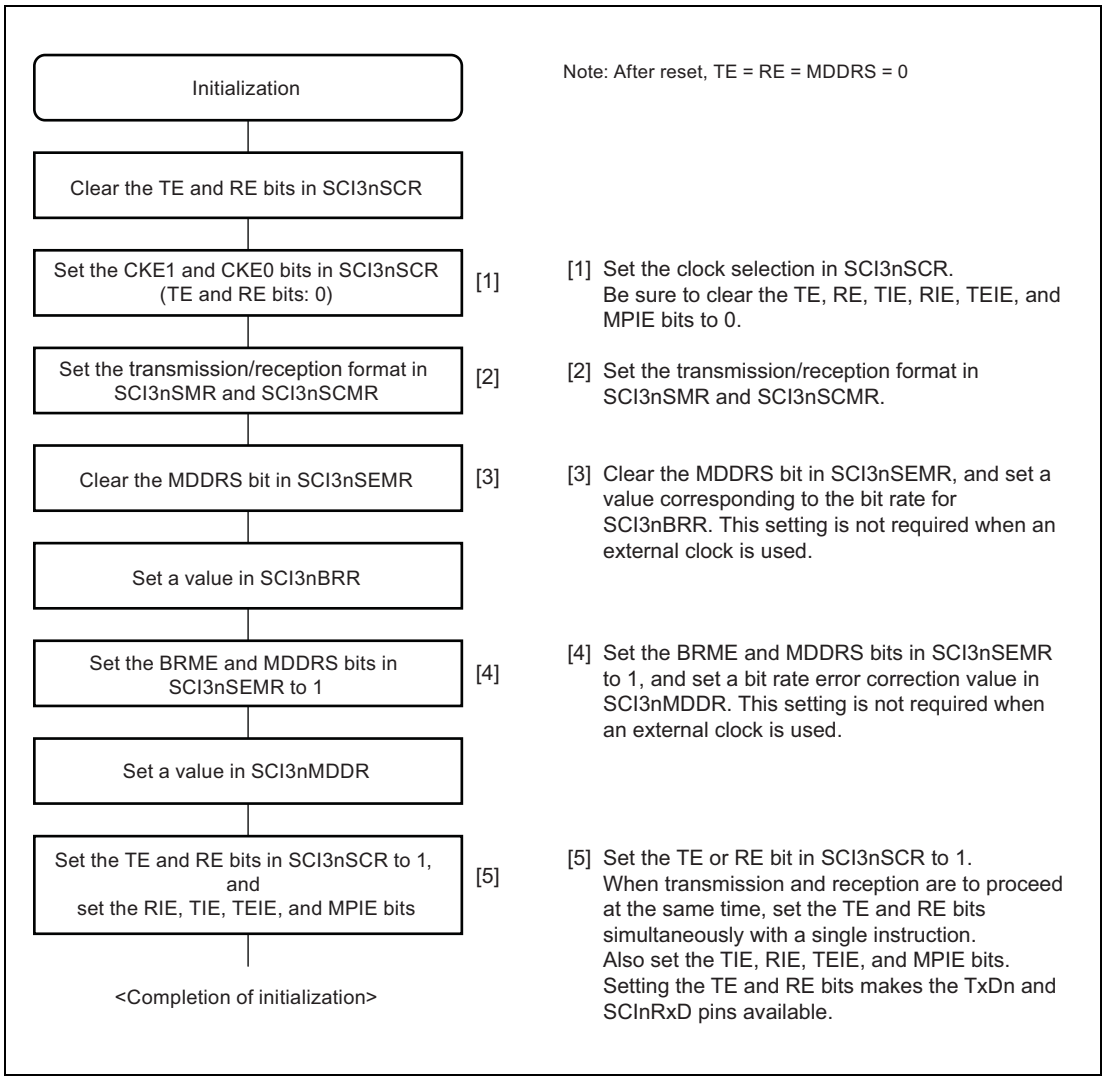


Figure 12.19 Example of SCI3 Initialization Flowchart

12.4.3.3 Serial Data Transmission (Clock Synchronous Mode)

Figure 12.20 shows an example of the operation for serial transmission in clock synchronous mode. In serial data transmission, the SCI3 operates as described below.

(1) When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR, starting output of the first bit when the synchronization clock is input. To write transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.

(2) Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR, and the TDRE flag is set to 1. When the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before the previously transferred data has been transmitted. When a TEI interrupt request is used, the TIE bit is cleared to 0 after the last transmit data has been written to SCI3nTDR, and the TEIE bit is set to 1.

(3) 8-bit data is output from the SCInTxD pin in synchronization with the output clock (when clock output mode has been specified) or in synchronization with the input clock (when external clock has been specified).

(4) The SCI3 checks for the TDRE flag at the time of the last bit output.

(5) When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR, and serial transmission of the next frame starts.

(6) When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1 and the SCInSCK pin retains the output state of the last bit. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated. The SCInSCK pin is held high.

Figure 12.21 shows a sample flowchart of serial data transmission. Also, **Figure 12.22** shows a sample flowchart for stopping the SCI3 after data transmission. Transmission will not start even if the TDRE flag is cleared while a receive error flag (ORER) is set to 1. Be sure to clear the receive error flags to 0 before starting transmission. Note that clearing the RE bit to 0 does not clear the receive error flags.

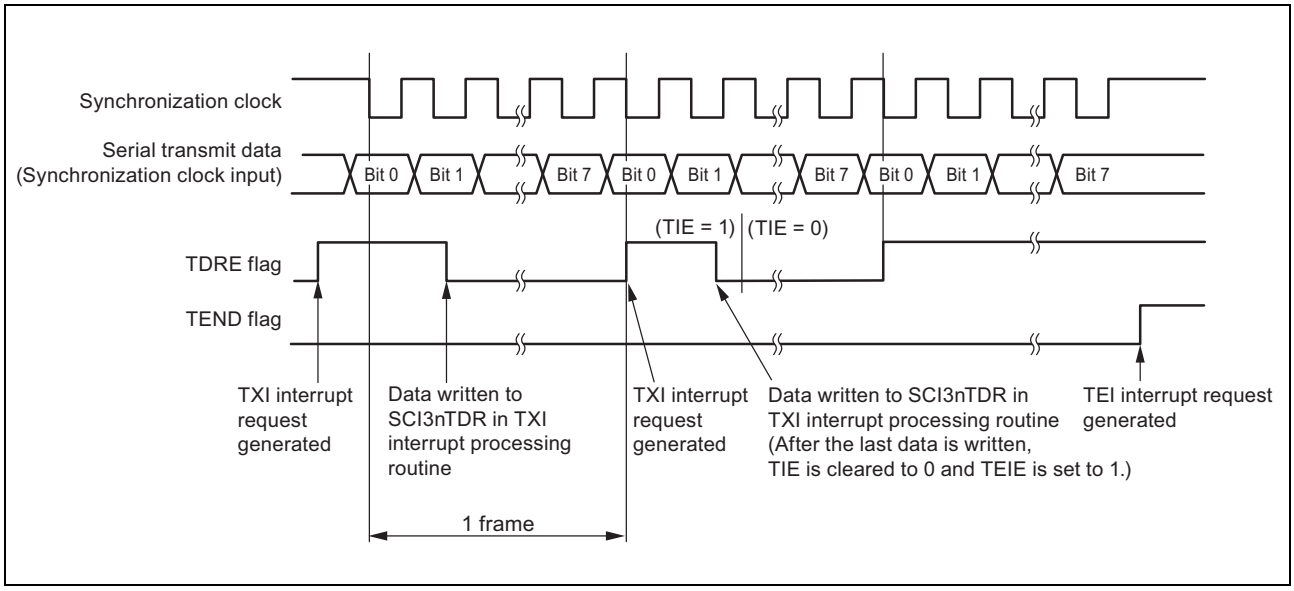


Figure 12.20 Example of Operation for Transmission in Clock Synchronous Mode

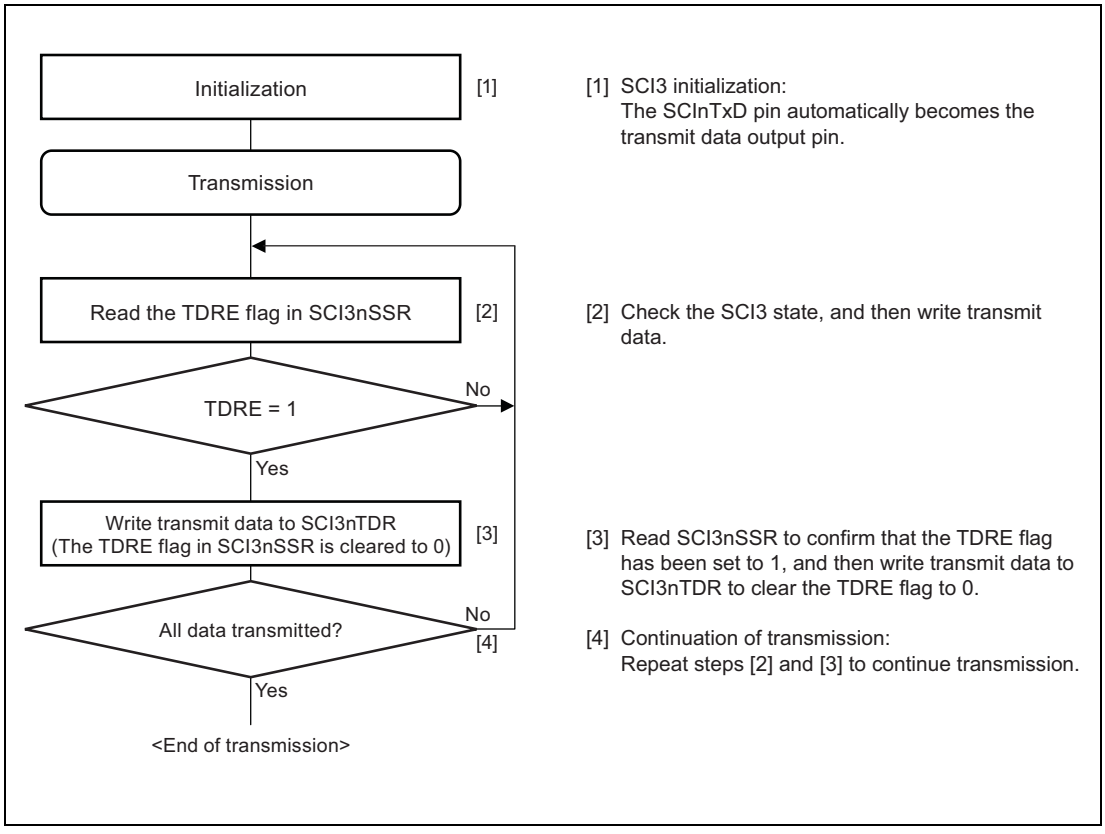


Figure 12.21 Example of Serial Transmission Flowchart

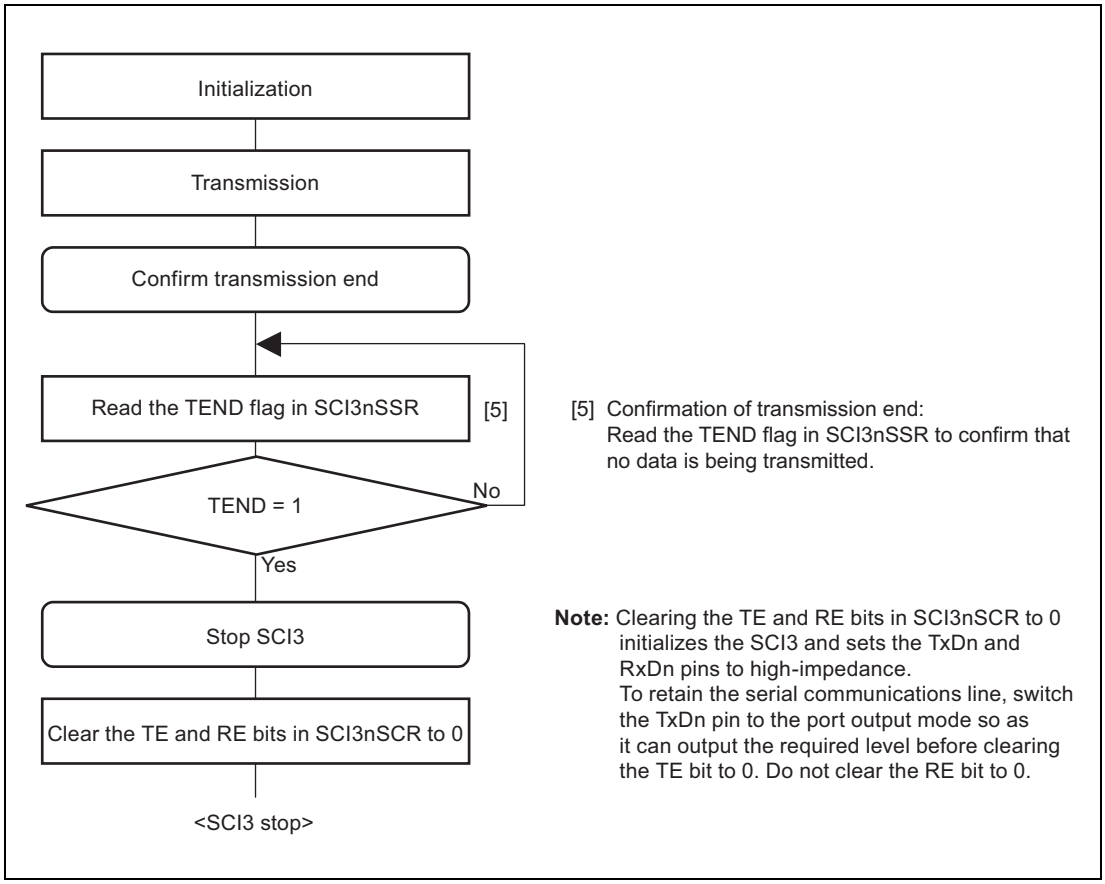


Figure 12.22 Example Flowchart for Stopping the SCI3 after Serial Transmission

12.4.3.4 Serial Data Reception (Clock Synchronous Mode)

Figure 12.23 shows an example of SCI3 operation for serial reception in clock synchronous mode. In serial data reception, the SCI3 operates as described below.

(1) The SCI3 performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores receive data in SCI3nRSR.

(2) When an overrun error occurs (the reception of the next data is completed with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.

(3) When data has been successfully received, the RDRF flag in SCI3nSSR is set to 1 and the receive data is transferred to SCI3nRDR. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

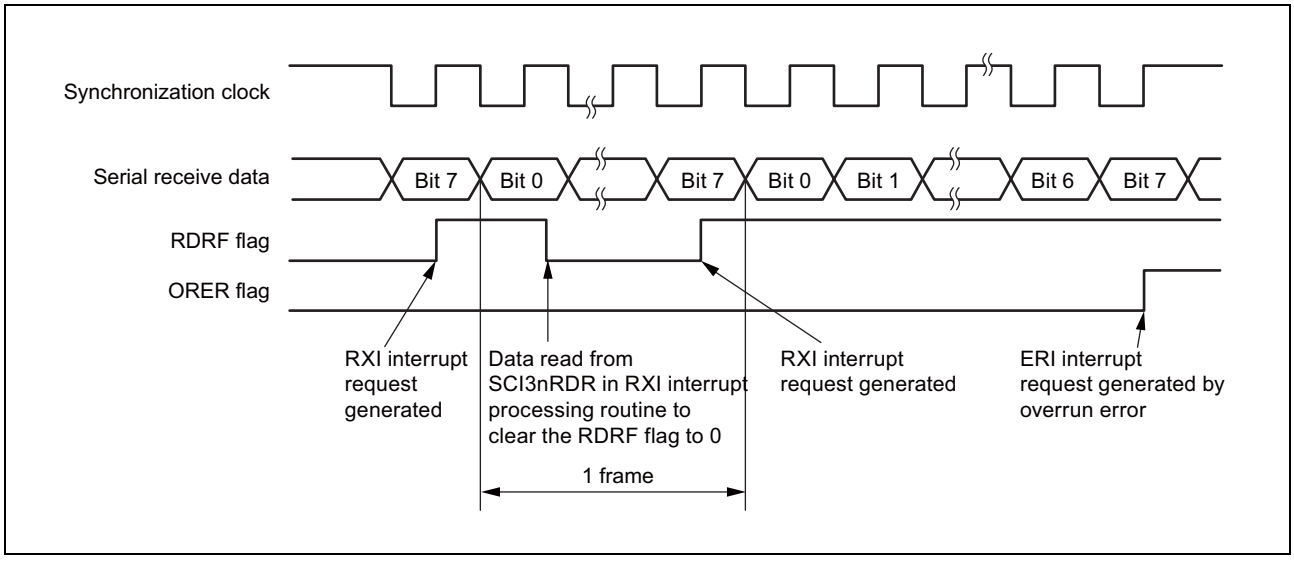


Figure 12.23 Example of SCI3 Operation for Reception

Subsequent transmission and reception are disabled with a receive error flag set to 1. Therefore, be sure to clear the ORER, FER, PER, and RDRF flags before continuing reception. **Figure 12.24** shows an example of flowchart for data reception.

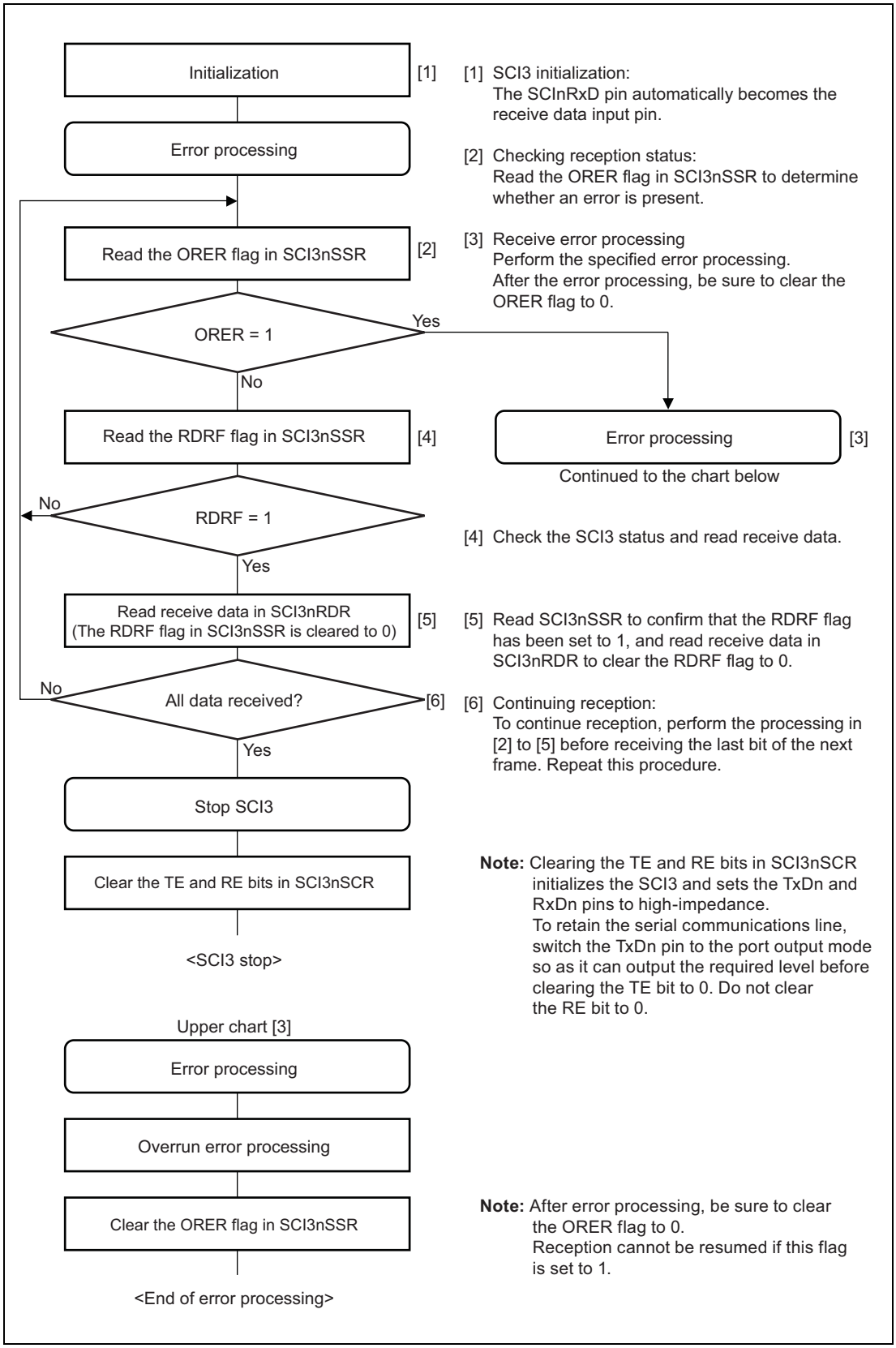


Figure 12.24 Example of Serial Reception Flowchart

12.4.3.5 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 12.25 shows a sample flowchart for simultaneous data transmit and receive operations. After the SCI3 is initialized, perform the following procedure for simultaneous data transmit and receive operations.

(1) To change transmit mode to simultaneous transmit and receive mode, check that the SCI3 has finished transmission and the TDRE and TEND flags are set to 1. Then clear the TE bit to 0 and then set the TE and RE bits to 1 with a single instruction.

(2) To change receive mode to simultaneous transmit and receive mode, check that the SCI3 has finished reception and clear the RE bit to 0. Then check that the RDRF and error flags (ORER, FER, and PER) are cleared to 0 and then set the TE and RE bits to 1 with a single instruction.

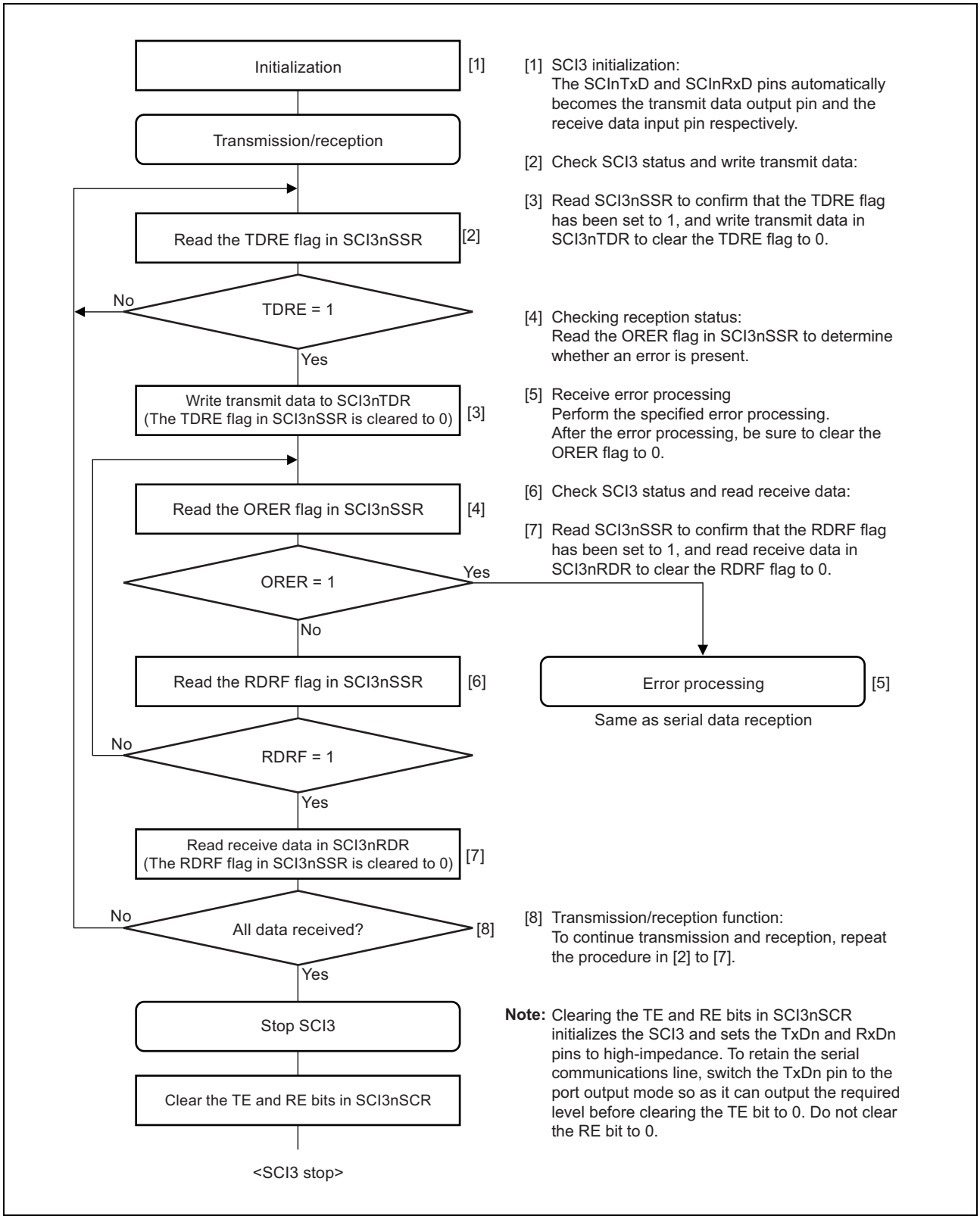


Figure 12.25 Example of Simultaneous Serial Transmission and Reception Flowchart

12.4.4 Bit Rate Modulation Function

The bit rate modulation function corrects a bit rate by averagely enabling the internal clocks specified by the CKS1 and CKS0 bits in SCI3nSMR for the number specified by SCI3nMDDR out of 256 clocks.

Figure 12.26 shows an example of asynchronous mode in which PCLK clock is selected with the CKS1 and CKS0 bits, SCI3nBRR is set to 0, and SCI3nMDDR is set to 160. In this example, the reference clock cycle is corrected to 256/160 on average and the bit rate is corrected to 160/256. Note that the pulse widths of the internal reference clock expand or contract for the amount of the selected internal clocks because there is a deviation in the enabling of the internal clocks.

Do not use this function with the maximum speed settings (CKS1 and CKS0 bits = 0 in SCI3nSMR, CKE1 bit = 0 in SCI3nSCR, and SCI3nBRR = 0) for clock synchronous mode.

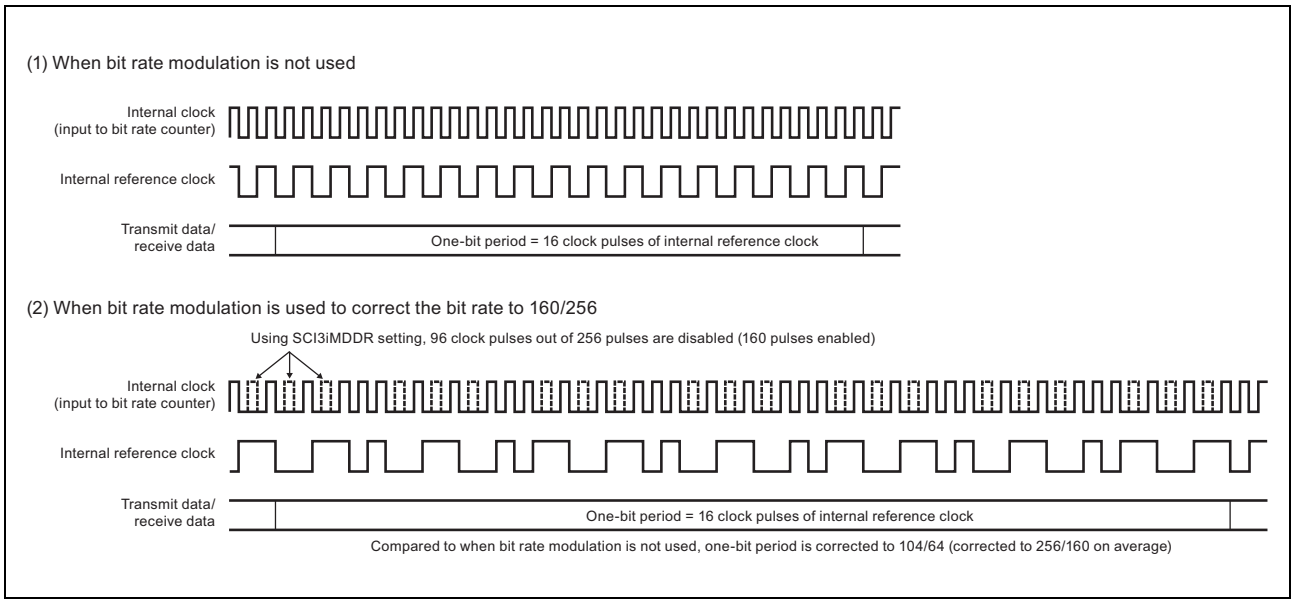


Figure 12.26 Example of Internal Reference Clock when the Bit Rate Modulation Function is Used

12.4.5 Interrupt Sources

Table 12.26 lists interrupt sources. Each interrupt source outputs an individual interrupt request signal. These interrupt sources can be enabled independently with the enable bit in SCI3nSCR.

A TXI interrupt request is generated when the TDRE flag in SCI3nSSR is set to 1. A TEI interrupt request is generated when the TEND flag in SCI3nSSR is set to 1. A TXI interrupt request can activate the DMAC to handle data transfer. The TDRE flag is automatically cleared to 0 when data is transferred using the DMAC.

CAUTION

The TDRE and TEND flags cannot be cleared to 0 while the TE bit in SCI3nSCR is 0.

Since the TEND flag is the level interrupt request flag for a TEI interrupt, do not set the TEIE bit in SCI3nSCR to 1 while the TE bit is 0.

An RXI interrupt request is generated when the RDRF flag in SCI3nSSR is set to 1. An ERI interrupt is generated when either of the ORER, PER, and FER flags in SCI3nSSR is set to 1. An RXI interrupt request can activate the DMAC to handle data transfer. The RDRF flag is automatically cleared to 0 when data is transferred using the DMAC.

A TEI interrupt request is generated when the TEND flag is set to 1 with the TEIE bit set to 1.

CAUTION

If a TEI interrupt request and a TXI interrupt request are generated at the same time, the TXI interrupt request is accepted first. Note that clearing the TDRE flag to 0 at this time in the TXI interrupt processing routine also clears the TEND flag to 0 automatically, disabling the branch to the TEI interrupt processing routine.

Table 12.26 SCI3 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	DTS Activation
ERI	Receive error	ORER, FER, PER	Not possible	Not possible
RXI	Receive data full	RDRF	Possible	Possible
TXI	Transmit data empty	TDRE	Possible	Possible
TEI	Transmit end	TEND	Not possible	Not possible

12.5 Notes

12.5.1 Break Detection and Processing

A break can be detected by reading the RXDMON bit in SCI3nSEMR when detecting a framing error. Since all inputs from the SCInRxD pin are 0 in the break state, the FER flag is set to 1 and the PER flag may also be set to 1. The SCI3 also continues data reception after it receives a break. For this reason, note that the FER flag is set to 1 again even after the FER flag is cleared to 0.

12.5.2 Mark State and Break Output

While the TE bit is 0 (transmission/reception disabled), the SCInTxD pin can output any level by switching the SCInTxD pin to a general output port. This allows the SCInTxD pin to be the mark state or break output during data transmission.

12.5.3 Receive Error Flags and Transmit Operations in Clock Synchronous Mode

During the clock synchronous simultaneous data transmit/receive operation, transmission cannot be started when a receive error flag (ORER) is set to 1, even if the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note that the receive error flags cannot be cleared to 0 even by clearing the RE bit to 0.

12.5.4 Relationship between Writing to SCI3nTDR and the TDRE Flag

The TDRE flag in SCI3nSSR is a status flag indicating that transmit data in SCI3nTDR has been transferred to SCI3nTSR. The TDRE flag is set to 1 when the SCI3 transfers data from SCI3nTDR to SCI3nTSR

Data can be written to SCI3nTDR regardless of the status of the TDRE flag. However, writing new data to SCI3nTDR while the TDRE flag is 0 will make the data stored in SCI3nTDR lost because it has not been transferred to SCI3nTSR. Therefore, make sure to check that the TDRE flag is set to 1 before writing transmit data to SCI3nTDR.

12.5.5 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

When using an external synchronization clock, clear the TDRE flag to 0 and then input a transmission clock (**Figure 12.27**). Also in continuous transmission mode, clear the TDRE flag to 0 and then input a transmission clock for the next frame.

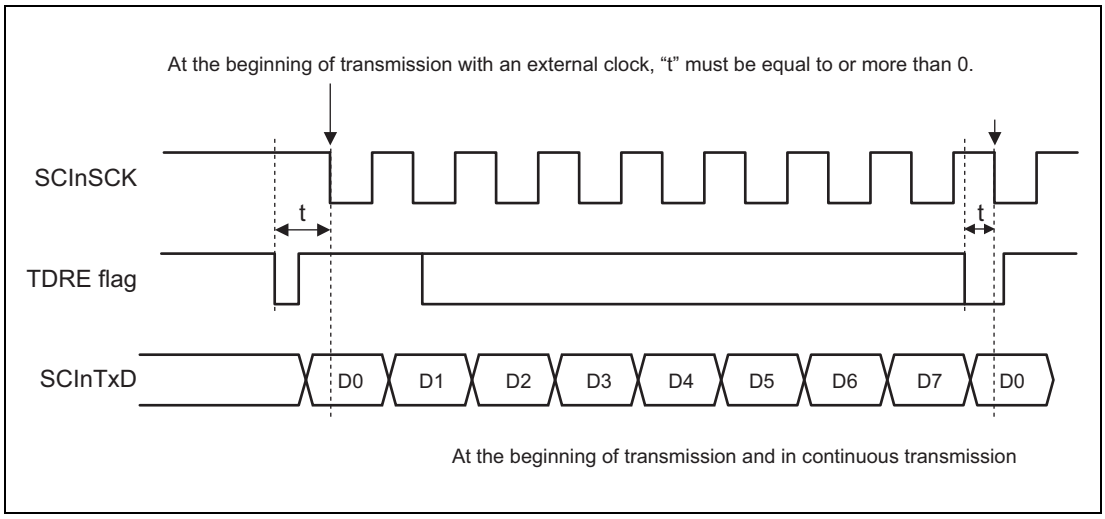


Figure 12.27 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

12.5.6 External Clock Input in Clock Synchronous Mode

As for inputting the external clock signal to the SCInSCK pin in clock synchronous mode, see **Section 35.3.6, SCI/FLSCI Timing**.

Section 13 LIN Master Interface (RLIN2)

This section contains a generic description of the LIN master interface (RLIN2).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of RLIN2.

13.1 Overview of RH850/C1x RLIN2

13.1.1 Units and Channels

This microcontroller has the following number of RLIN2 units and channels.

Table 13.1 Units

Product	RH850/C1H	RH850/C1M
Units	3	—
Name	RLIN21n (n = 0 to 2)	—

Table 13.2 Unit Configurations and Channels

Unit Name RLIN21n	Channels per Unit	Unit Channel Number	Channel Name RLIN2m
RLIN210	1	0	RLIN20
RLIN211	1	0	RLIN21
RLIN212	1	0	RLIN22

Table 13.3 Index

Index	Meaning
n	Throughout this section, the individual RLIN2 units are identified by the index “n” (n = 0 to 2).
m	Throughout this section, the individual channels are identified by the index “m” (m = 0 to 2).
i	Throughout this section, the individual channels of units that configure RLIN2 are identified by the index “i” (i = 0).
b	Throughout this section, the individual data buffers implemented in RLIN2 are identified by the index “b” (b = 1 to 8).

For example, RLN21nGLWBR are the LIN wake-up baud rate select registers, which are the global registers of RLIN2. RLN21nmLiMD are the LIN mode registers, which are the channel registers.

13.1.2 Register Base Addresses

RLIN2 base addresses are listed in the following table.

RLIN2 register addresses are given as offsets from the base addresses in general.

Table 13.4 Register Base Addresses

Base Address Name	Base Address
<RLIN210_base>	FFCE 0000_H
<RLIN211_base>	FFCE 0020_H
<RLIN212_base>	FFCE 0040_H

13.1.3 Clock Supply

RLIN2 clock is listed in following table.

Table 13.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RLIN21n	PCLK	CLK_LSB (low-speed peripheral clock)
	clkc	CLKC_LSB (unmodulated low-speed peripheral clock)

13.1.4 Interrupt Requests

RLIN2 interrupt requests are listed in the following table.

Table 13.6 Interrupt Requests

Unit Interrupt Name	Outline	Interrupt Number
INTRLIN0	Transmit complete interrupt/ receive complete interrupt/ error detection interrupt (RLIN210 interrupt)	200
INTRLIN1	Transmit complete interrupt/ receive complete interrupt/ error detection interrupt (RLIN211 interrupt)	201
INTRLIN2	Transmit complete interrupt/ receive complete interrupt/ error detection interrupt (RLIN212 interrupt)	202

13.1.5 Reset Sources

RLIN2 reset sources are listed in the following table. RLIN2 is initialized by these reset sources.

Table 13.7 Reset Sources

Unit Name	Reset Source
RLIN21n	Any reset source

13.1.6 External Input/Output Signals

External input/output signals of RLIN2 are listed below.

Table 13.8 I/O signals of RLIN2n

Unit Signal Name	Outline	Alternative Port Pin Signal
RLIN210		
RLIN20RX	RLIN210 receive data input	RLIN20RX
RLIN20TX	RLIN210 transmit data output	RLIN20TX
RLIN211		
RLIN21RX	RLIN211 receive data input	RLIN21RX
RLIN21TX	RLIN211 transmit data output	RLIN21TX
RLIN212		
RLIN22RX	RLIN212 receive data input	RLIN22RX
RLIN22TX	RLIN212 transmit data output	RLIN22TX

13.2 Function

13.2.1 Functional Overview

The LIN Master Interface is a hardware LIN communication controller that complies to LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, and SAEJ2602 (SEP 2005) and automatically performs frame communication and error determination.

Table 13.9 gives the LIN Master Interface specifications and **Figure 13.1** shows a block diagram of the LIN Master Interface.

Table 13.9 LIN Master Interface Specifications

Item		Specifications
Channel count		3 channels
LIN communication function	Protocol	LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, and SAEJ2602 (SEP 2005)
	Variable frame structure	• Break transmission width: 13 to 28 Tbits • Break delimiter transmission width: 1 to 4 Tbits • Inter-byte space (header): 0 to 7 Tbits (space between Sync field and ID field)*1 • Response space: 0 to 7 Tbits*1 • Inter-byte space: 0 to 3 Tbits (space between data bytes in response area) • Transmission wake-up: 1 to 16 Tbits
	Checksum	• Automatic operation for both transmission and reception • Classic or enhanced selectable (for each frame)
	Response field data byte count	Variable from 0 to 8 bytes
	Frame communication modes	• Mode in which header transmission and response transmission/reception is started with a single transmission start request • Mode in which header transmission and response transmission are started with separate transmission start requests (frame separate mode)
	Wake-up transmission and reception	LIN wake-up mode provided • Wake-up transmission (1 to 16 Tbits) • Wake-up reception Low-level width of input signals measured
	Status	• Successful frame/wake-up transmission • Successful header transmission • Successful frame/wake-up reception*2 • Successful data 1 reception • Error detection • Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, LIN self-test mode)
	Error status	• Bit error • Checksum error • Frame timeout error • Physical bus error • Framing error
	Baud rate selection	Baud rate conforming to the LIN specifications generated using baud rate generator
	Test mode	Self-test mode for user evaluation
	Interrupt function	• Successful frame/wake-up transmission • Successful frame/wake-up reception*2 • Error detection These three logical OR of these three operations events are the interrupt sources (INTRLINm) for each channel.

Note 1. Since the same register is used for setting, the inter-byte space (header) = response space.
Note 2. For wake-up reception, the low level width of the input signal is indicated.

13.2.2 Block Diagram

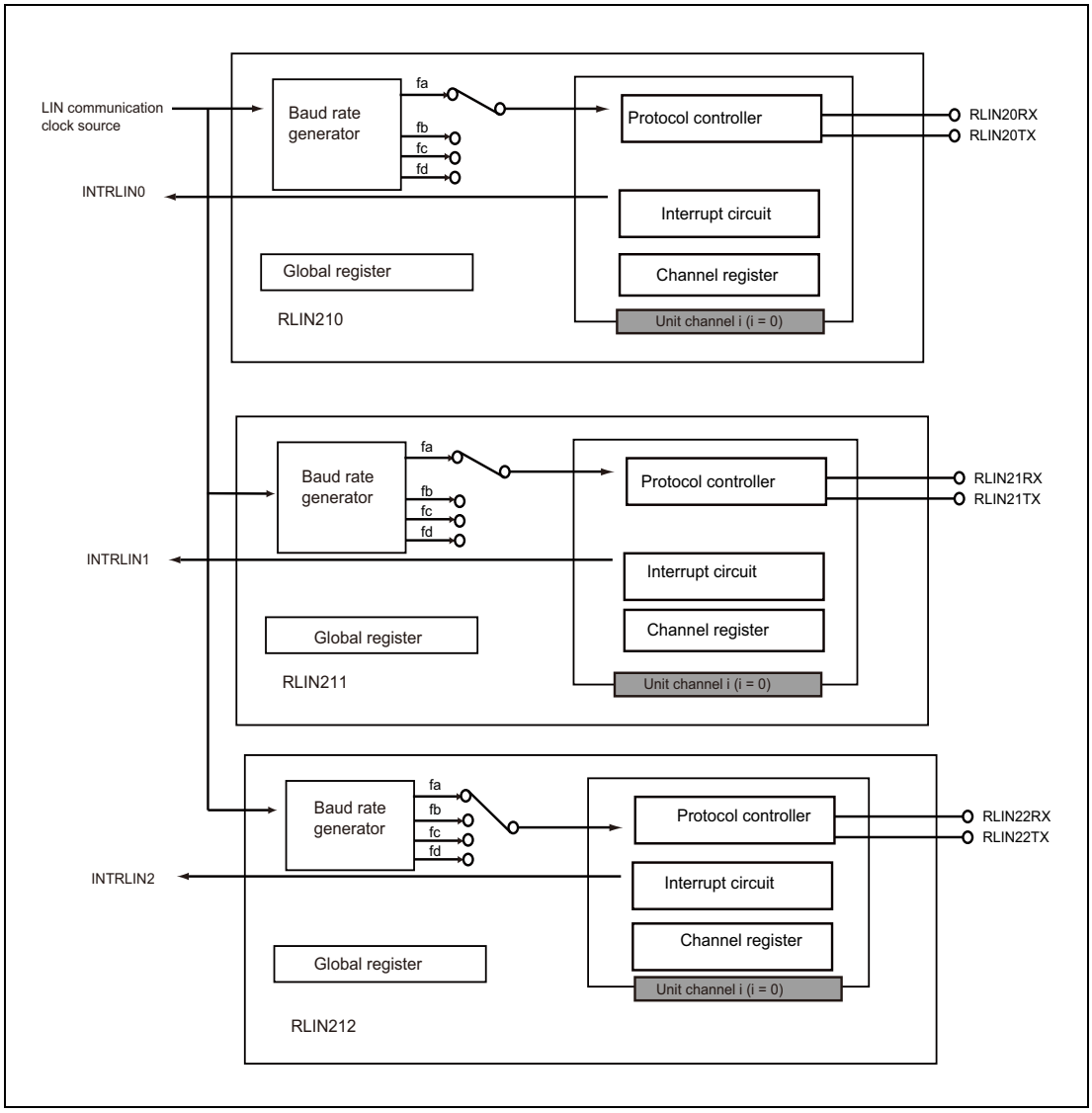


Figure 13.1 LIN Master Interface Block Diagram

13.3 Registers

The registers of the LIN master interface are configured with global registers and channel registers. As the global registers are allocated for each unit, they can be set respectively in units. As the channel registers are allocated for each channel, they can be controlled respectively in channels.

13.3.1 List of Registers

RLIN2 registers are listed in the following table.

For information on <RLIN21n_base>, see **Section 13.1.2, Register Base Addresses**.

Table 13.10 LIN Master Interface Registers Overview

Module Name	Register Name	Shortcut	Address
Global registers			
RLN21n	LIN wake-up baud rate select register	RLN21nGLWBR	<RLIN21n_base> + 01_H
RLN21n	LIN baud rate prescaler 0 register	RLN21nGLBRP0	<RLIN21n_base> + 02_H
RLN21n	LIN baud rate prescaler 1 register	RLN21nGLBRP1	<RLIN21n_base> + 03_H
RLN21n	LIN self test control register	RLN21nGLSTC	<RLIN21n_base> + 04_H
Channel registers			
RLN21nm	LIN mode register	RLN21nmLiMD	<RLIN21n_base> + 08_H
RLN21nm	LIN break field configuration register	RLN21nmLiBFC	<RLIN21n_base> + 09_H
RLN21nm	LIN space configuration register	RLN21nmLiSC	<RLIN21n_base> + $0A_H$
RLN21nm	LIN wake-up configuration register	RLN21nmLiWUP	<RLIN21n_base> + $0B_H$
RLN21nm	LIN interrupt enable register	RLN21nmLiIE	<RLIN21n_base> + $0C_H$
RLN21nm	LIN error detection enable register	RLN21nmLiEDE	<RLIN21n_base> + $0D_H$
RLN21nm	LIN control register	RLN21nmLiCUC	<RLIN21n_base> + $0E_H$
RLN21nm	LIN transmission control register	RLN21nmLiTRC	<RLIN21n_base> + 10_H
RLN21nm	LIN mode status register	RLN21nmLiMST	<RLIN21n_base> + 11_H
RLN21nm	LIN status register	RLN21nmLiST	<RLIN21n_base> + 12_H
RLN21nm	LIN error status register	RLN21nmLiEST	<RLIN21n_base> + 13_H
RLN21nm	LIN data field configuration register	RLN21nmLiDFC	<RLIN21n_base> + 14_H
RLN21nm	LIN ID buffer register	RLN21nmLiIDB	<RLIN21n_base> + 15_H
RLN21nm	LIN check sum buffer register	RLN21nmLiCBR	<RLIN21n_base> + 16_H
RLN21nm	LIN data buffer 1 register	RLN21nmLiDBR1	<RLIN21n_base> + 18_H
RLN21nm	LIN data buffer 2 register	RLN21nmLiDBR2	<RLIN21n_base> + 19_H
RLN21nm	LIN data buffer 3 register	RLN21nmLiDBR3	<RLIN21n_base> + $1A_H$
RLN21nm	LIN data buffer 4 register	RLN21nmLiDBR4	<RLIN21n_base> + $1B_H$
RLN21nm	LIN data buffer 5 register	RLN21nmLiDBR5	<RLIN21n_base> + $1C_H$
RLN21nm	LIN data buffer 6 register	RLN21nmLiDBR6	<RLIN21n_base> + $1D_H$
RLN21nm	LIN data buffer 7 register	RLN21nmLiDBR7	<RLIN21n_base> + $1E_H$
RLN21nm	LIN data buffer 8 register	RLN21nmLiDBR8	<RLIN21n_base> + $1F_H$

NOTE

When writing to a register not used, write the value after reset.

13.3.2 Global Registers

13.3.2.1 RLN21nGLWBR — LIN Wake-up Baud Rate Select Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 01_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LWBR0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 13.11 RLN21nGLWBR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	The write value should be the value after reset.
0	LWBR0	Wake-up Baud Rate Select 0: In LIN wake-up mode, the clock specified in the RLN21nmLiMD.LCKS bit is used. (for LIN1.3) 1: In LIN wake-up mode, the clock fa is used regardless of the setting in the RLN21nmLiMD.LCKS bit. (for LIN2.x)

Set the RLN21nGLWBR register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

LWBR0 Bit (Wake-Up Baud Rate Select)

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN21nGLWBR register to 0. This allows the 2.5-Tbit or longer low-level width of the input signal to be measured.

When LIN Specification Package Revision 2.0 or 2.1 is used, set the LWBR0 bit to 1.

With this setting, fa is selected as the LIN system clock (fLIN) during LIN wake-up mode regardless of the setting of the LCKS bit in the RLN21nmLiMD register (the LCKS bit is not changed) and the 2.5-Tbit or longer low-level width of the input signal to be measured.

Setting the baud rate to 19200 bps while fa is selected allows the 130 µs or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the LCKS bit in the RLN21nmLiMD register.

13.3.2.2 RLN21nGLBRP0 — LIN Baud Rate Prescaler 0 Register

Access: This register can be read/written in 8-bit units

Address: <RLIN21n_base> + 02_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP0[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.12 RLN21nGLBRP0 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP0[7:0]	Assuming that the value set in this register is N (0 to 255), the baud rate prescaler 0 divides the frequency of the peripheral function clock by N + 1. Setting Range: 00_H to FF_H

Set the RLN21nGLBRP0 register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

The value set in this register is used to control the frequency of baud rate clock source fa, fb, and fc.

Assuming that the value set in this register is N, baud rate prescaler 0 divides the LIN communication clock source by N+1.

13.3.2.3 RLN21nGLBRP1 — LIN Baud Rate Prescaler 1 Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 03_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP1[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.13 RLN21nGLBRP1 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP1[7:0]	Assuming that the value set in this register is M (0 to 255), the baud rate prescaler 1 divides the frequency of the peripheral function clock by M+1. Setting Range: 00_H to FF_H

Set the RLN21nGLBRP1 register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

The value set in this register is used to control the frequency of baud rate clock source fd.

Assuming that the value set in this register is M, baud rate prescaler 1 divides the LIN communication clock source by M+1.

13.3.2.4 RLN21nGLSTC — LIN Self-Test Control Register

The RLN21nGLSTC register is used to unlock protection in LIN self-test mode.

Set the RLN21nGLSTC register while the RLN21nmLiMST.OMM0 bits for all channels in the given unit are 0 (the channels are in LIN reset mode).

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSTC[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Table 13.14 RLN21nGLSTC Register Contents

Bit Position	Bit Name	Function
7 to 0	LSTC[7:0]	LIN Self-Test Mode Setting 00_H: The module is not in LIN self-test mode*2 01_H: The module is in LIN self-test mode.*3 Writing $A7_H$, 58_H, and 01_H successively to the RLN21nGLSTC register places the module into LIN self-test mode. Any other settings are prohibited. The read value after the setting is undefined.

Note 1. Writing is ignored when all the channels in the same unit are not LIN reset mode.

Note 2. See **Section 13.15.4**, for ending LIN self-test mode. When LIN self-test mode is ended, 00_H is read.

Note 3. See **Section 13.15.1**, for shifting to LIN self-test mode. After transition to LIN self-test mode, 01_H is read.

13.3.3 Channel Registers

13.3.3.1 RLN21nmLiMD — LIN Mode Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiMD: <RLIN21n_base> + 08_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	LCKS[1:0]		—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R

Table 13.15 RLN21nmLiMD Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	The write value should be the value after reset.
3, 2	LCKS[1:0]	LIN System Clock Select b3 b2 0 0: fa (Clock generated by baud rate prescaler 0) 0 1: fb (1/2 clock generated by baud rate prescaler 0) 1 0: fc (1/8 clock generated by baud rate prescaler 0) 1 1: fd (1/2 clock generated by baud rate prescaler 1)
1, 0	Reserved	These bits are always read as 0. The write value should always be 0.

Set the RLN21nmLiMD register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

LCKS[1:0] Bits (LIN System Clock Select)

The LCKS bits select the clock to be input to the protocol controller.

With 00_B set, the protocol controller is provided with fa (clock generated by baud rate prescaler 0).

With 01_B set, the protocol controller is provided with fb (1/2 clock generated by baud rate prescaler 0).

With 10_B set, the protocol controller is provided with fc (1/8 clock generated by baud rate prescaler 0).

With 11_B set, the protocol controller is provided with fd (1/2 clock generated by baud rate prescaler 1).

When the LWBR0 bit in the RLN21nGLWBR is 1 (when LIN 2.0 or 2.1 is used) and the RLN21nmLiMST register is 01h (LIN wake-up mode), regardless of the setting of the LWBR0 bit, fa is input to the protocol controller (LCKS bit is not changed).

13.3.3.2 RLN21nmLiBFC — LIN Break Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiBFC: <RLIN21n_base> + 09_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	BDT[1:0]		BLT[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.16 RLN21nmLiBFC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5, 4	BDT[1:0]	Break Delimiter (High Level) Width Select b5 b4 0 0: 1 Tbit 0 1: 2 Tbits 1 0: 3 Tbits 1 1: 4 Tbits
3 to 0	BLT[3:0]	Transmit Break (Low Level) Width Select b3 b0 0 0 0 0: 13 Tbits 0 0 0 1: 14 Tbits 0 0 1 0: 15 Tbits : 1 1 1 0: 27 Tbits 1 1 1 1: 28 Tbits

Set the RLN21nmLiBFC register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

Some combinations of the set values result in the length of a frame exceeding the timeout time. Set the appropriate values in this register.

BDT[1:0] Bits (Transmission Break Delimiter High Level Width Setting)

This bit is used to set the break high level width of transmission frame header.
1 Tbit to 4 Tbits can be set.

BLT[3:0] Bits (Transmission Break Low Level Width Setting)

This BLT bits set the break low level width of transmission frame header.
13 Tbits to 28 Tbits can be set.

13.3.3.3 RLN21nmLiSC — LIN Space Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiSC: <RLIN21n_base> + $0A_H$ + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	IBHS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 13.17 RLN21nmLiSC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5, 4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbits 0 1: 1 Tbit 1 0: 2 Tbits 1 1: 3 Tbits
3	Reserved	The write value should be the value after reset.
2 to 0	IBHS[2:0]	Inter-Byte Space (Header)/Response Space Select b2 b0 0 0 0: 0 Tbits 0 0 1: 1 Tbit 0 1 0: 2 Tbits 0 1 1: 3 Tbits 1 0 0: 4 Tbits 1 0 1: 5 Tbits 1 1 0: 6 Tbits 1 1 1: 7 Tbits

Set the RLN21nmLiSC register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

Some combinations of the set values result in the length of a frame or a response exceeding the timeout time. Set the appropriate values in this register.

IBS[1:0] Bits (Inter-Byte Space Select)

The IBS bits set the width of the inter-byte space of the transmission frame response field.
0 to 3 Tbits can be set.
These bits are enabled only during response transmission; these are disabled during response reception.

IBHS[2:0] Bits (Inter-Byte Space (Header)/Response Space Select)

The IBHS bits set the width of the inter-byte space (header) of the transmission frame header field and the response space.
0 to 7 Tbits can be set.
The response space setting is enabled only during response transmission; setting is disabled during response reception.
The inter-byte space (header) is equal to the response space.

13.3.3.4 RLN21nmLiWUP — LIN Wake-up Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiWUP: <RLIN21n_base> + $0B_H$ + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	WUTL[3:0]				—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 13.18 RLN21nmLiWUP Register Contents

Bit Position	Bit Name	Function
7 to 4	WUTL[3:0]	Wake-up Transmission Low level Width Select b7 b4 0 0 0 0: 1 Tbit 0 0 0 1: 2 Tbits 0 0 1 0: 3 Tbits 0 0 1 1: 4 Tbits : 1 1 0 0: 13 Tbits 1 1 0 1: 14 Tbits 1 1 1 0: 15 Tbits 1 1 1 1: 16 Tbits
3 to 0	Reserved	The write value should be the value after reset.

Set the RLN21nmLiWUP register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

WUTL[3:0] Bits (Wake-Up Transmission Low Level Width Select)

The WUTL bits set the low level width of the wake-up signal transmission.
1 Tbit to 16 Tbits can be set.

When the LWBR0 bit in the RLN21nGLWBR is 1 (when LIN 2.0 or 2.1 is used), regardless of the setting of the LCKS bit in the RLN21nmLiMD register, fa is selected as the LIN system clock (fLIN) (LCKS bit is not changed).

13.3.3.5 RLN21nmLiIE — LIN Interrupt Enable Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiIE: <RLIN21n_base> + $0C_H$ + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ERRIE	FRCIE	FTCIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 13.19 RLN21nmLiIE Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	The write value should be the value after reset.
2	ERRIE	Error Detection Interrupt Enable 0: Disables error detection interrupt. 1: Enables error detection interrupt.
1	FRCIE	Successful Frame/Wake-up Reception Interrupt Enable 0: Disables successful frame/wake-up reception interrupt. 1: Enables successful frame/wake-up reception interrupt.
0	FTCIE	Successful Frame/Wake-up Transmission Interrupt Enable 0: Disables successful frame/wake-up transmission interrupt. 1: Enables successful frame/wake-up transmission interrupt.

Set the RLN21nmLiIE register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

ERRIE Bit (Error Detection Interrupt Enable)

The ERRIE bit enables or disables interrupt generation upon detection of an error.
With 0 set, the interrupt request is not generated when the ERR flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the ERR flag in the RLN21nmLiST register is set to 1.
Interrupt sources can be the bit error, physical bus error, frame timeout error, framing error, and checksum error.
Detection of the bit error, physical bus error, frame/response timeout error, and framing error can be enabled or disabled using the RLN21nmLiEDE register.

FRCIE Bit (Successful Frame/Wake-Up Reception Interrupt Enable)

The FRCIE bit enables or disables interrupt generation upon successful reception of a frame or a wake-up signal (counting of low level width of the input signal).
With 0 set, the interrupt request is not generated when the FRC flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the FRC flag in the RLN21nmLiST register is set to 1.

FTCIE Bit (Successful Frame/Wake-Up Transmission Interrupt Enable)

The FTCIE bit enables or disables interrupt generation upon successful transmission of a frame or a wake-up signal.
With 0 set, the interrupt request is not generated when the FTC flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the FTC flag in the RLN21nmLiST register is set to 1.

13.3.3.6 RLN21nmLiEDE — LIN Error Detection Enable Register

Access: This register can be read/written in 8-bit units

Address: RLN21nmLiEDE: <RLIN21n_base> + $0D_H$ + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FERE	FTERE	PBERE	BERE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 13.20 RLN21nmLiEDE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	The write value should be the value after reset.
3	FERE	Framing Error Detection Enable 0: Disables framing error detection. 1: Enables framing error detection.
2	FTERE	Frame Timeout Error Detection Enable 0: Disables frame timeout error detection. 1: Enables frame timeout error detection.
1	PBERE	Physical Bus Error Detection Enable 0: Disables physical bus error detection. 1: Enables physical bus error detection.
0	BERE	Bit Error Detection Enable 0: Disables bit error detection. 1: Enables bit error detection.

Set the RLN21nmLiEDE register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

FERE Bit (Framing Error Detection Enable)

The FERE bit enables or disables detection of the framing error.
With 0 set, the framing error is not detected.
With 1 set, the framing error is detected.
When this bit is set to 1, the detection result is indicated in the FER flag in the RLN21nmLiEST register.
For details of the framing error, see **Section 13.14, Error Status**.

FTERE Bit (Frame Timeout Error Detection Enable)

The FTERE bit enables or disables detection of the frame timeout error.
With 0 set, the frame timeout error is not detected.
With 1 set, the frame timeout error is detected.
When this bit is set to 1, the detection result is indicated in the FTER flag in the RLN21nmLiEST register.
Do not use the timeout error if response data of 9 bytes or more is to be transmitted or received.
For details of the frame timeout error, see **Section 13.14, Error Status**.

PBERE Bit (Physical Bus Error Detection Enable)

The PBERE bit enables or disables detection of the physical bus error.
With 0 set, the physical bus error is not detected.
With 1 set, the physical bus error is detected.
When this bit is set to 1, the detection result is indicated in the PBER flag in the RLN21nmLiEST register.
For details of the physical bus error, see **Section 13.14, Error Status**.

BERE Bit (Bit Error Detection Enable)

The BERE bit enables or disables detection of the bit error.
With 0 set, the bit error is not detected.
With 1 set, the bit error is detected.
When this bit is set to 1, the detection result is indicated in the BER flag in the RLN21nmLiEST register.
For details of the bit error, see **Section 13.14, Error Status**.

13.3.3.7 RLN21nmLiCUC — LIN Control Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiCUC: <RLIN21n_base> + $0E_H$ + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OM1	OM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.21 RLN21nmLiCUC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The write value should be the value after reset.
1	OM1	LIN Mode Select 0: LIN wake-up mode. 1: LIN operation mode.
0	OM0	LIN Reset 0: LIN reset mode. 1: Release from LIN reset mode.

Set the RLN21nmLiCUC register to 01_H to cause a transition to LIN wake-up mode after release from LIN reset mode, and set the register to 03_H to cause a transition to LIN operation mode.

In LIN self-test mode, set the RLN21nmLiCUC register to 03_H after a transition to LIN self-test mode is completed.

After a value is written to this register, confirm that the value written is actually indicated in the RLN21nmLiMST register before writing another value.

OM1 Bit (LIN Mode Select)

The OM1 bit selects the specific LIN operation mode (either LIN wake-up mode or LIN operation mode) after release from LIN reset mode.
Setting this bit to 0 causes a transition to LIN wake-up mode.
Setting this bit to 1 causes a transition to LIN operation mode.
This bit is valid only when the OMM0 bit in the RLN21nmLiMST register is 1.
Writing a value to this bit is disabled while the FTS bit in the RLN21nmLiTRC register is 1.

OM0 Bit (LIN Reset)

The OM0 bit selects either a transition to LIN reset mode or release from LIN reset mode.
Setting this bit to 0 causes a transition to LIN reset mode.
Setting this bit to 1 leads to release from LIN reset mode.

13.3.3.8 RLN21nmLiTRC — LIN Transmission Control Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiTRC: <RLIN21n_base> + 10_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RTS	FTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 13.22 RLN21nmLiTRC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The write value should be the value after reset.
1	RTS	Response Transmission Start 0: Response transmission is stopped in frame separate mode. 1: Response transmission is started in frame separate mode.
0	FTS	Frame Transmission/Wake-up Transmission/Reception Start 0: Frame Transmission/wake-up transmission /reception is stopped. 1: Frame Transmission/wake-up transmission reception is started.

RTS Bit (Response Transmission Start)

Set the RTS bit to 1 in frame separate mode after header transmission is started (FTS bit is 1) and response transmission data is ready. Once set, this bit is automatically cleared to 0 upon completion of frame communication or transition to LIN reset mode.
Only 1 can be written to this bit; 0 cannot be written.
To write 1 to this bit, write 02_H to the RLN21nmLiTRC register using the store instruction.
Writing a value to this bit is disabled when the OMM0 bit is 0 (in LIN reset mode).
Writing a value to this bit is disabled when the FTS bit is 0 (frame transmission or wake-up transmission/reception is halted).

FTS Bit (Frame Transmission/Wake-Up Transmission/reception Start)

Set the FTS bit to 1 to start frame/wake-up transmission.
Also set this bit to 1 to allow wake-up reception (counting of the low level width of the input signal).
Only 1 can be written to this bit; 0 cannot be written.
Writing a value to this bit is disabled when the OMM0 bit is 0 (in LIN reset mode). This bit is set to 0 upon completion of frame or wake-up communication and transition to LIN reset mode.

13.3.3.9 RLN21nmLiMST — LIN Mode Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiMST: <RLIN21n_base> + 11_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OMM1	OMM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 13.23 RLN21nmLiMST Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is returned.
1	OMM1	LIN Mode Status Monitor 0: The LIN master interface is in LIN wake-up mode. 1: The LIN master interface is in LIN operation mode.
0	OMM0	LIN Reset Status Monitor 0: The LIN master interface is in LIN reset mode. 1: The LIN master interface is not in LIN reset mode.

OMM1 Bit (LIN Mode Status Monitor)

The OMM1 bit indicate the current operating mode.

The value of this bit is invalid while the OMM0 bit is 0_B (LIN reset mode).

OMM0 Bit (LIN Reset Status Monitor)

The OMM0 bit indicates the current operating mode.

13.3.3.10 RLN21nmLiST — LIN Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiST: <RLIN21n_base> + 12_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W

Table 13.24 RLN21nmLiST Register Contents

Bit Position	Bit Name	Function
7	HTRC	Successful Header Transmission Flag 0: Header transmission has not been completed. 1: Header transmission has been completed.
6	D1RC	Successful Data 1 Reception Flag 0: Data 1 Reception has not been completed. 1: Data 1 Reception has been completed.
5, 4	Reserved	The write value should be the value after reset.
3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
2	Reserved	The write value should be the value after reset.
1	FRC	Successful Frame/Wake-up Reception Flag 0: Frame or wake-up reception has not been completed. 1: Frame or wake-up reception has been completed.
0	FTC	Successful Frame/Wake-up Transmission Flag 0: Frame or wake-up transmission has not been completed. 1: Frame or wake-up transmission has been completed.

The RLN21nmLiST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication.

In LIN reset mode, the register retains 00_H.

Writing to this register is prohibited while the FTS bit in the RLN21nmLiTRC register is 1 (frame transmit or wake-up transmission/reception is started)

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

HTRC Flag (Successful Header Transmission Flag)

Only 0 can be written to the HTRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The HTRC flag is set to 1 upon completion of header transmission, but no interrupt request is generated. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

D1RC Flag (Successful Data 1 Reception Flag)

Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt request is not generated. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

ERR Flag (Error Detection Flag)

The ERR flag is set to 1 upon detection of an error. Here, an interrupt request is generated if the ERRIE bit in the RLN21nmLiIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the CSER, FER, FTER, PBER, and BER flags in the RLN21nmLiEST register in LIN operation mode or LIN wake-up mode. This clears the ERR flag to 0.

FRC Flag (Successful Frame/Wake-Up Reception Flag)

Only 0 can be written to the FRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FRC flag is set to 1 upon completion of frame or wake-up reception. Here, an interrupt request is generated if the FRCIE bit in the RLN21nmLiIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

FTC Flag (Successful Frame/Wake-Up Transmission Flag)

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTC flag is set to 1 upon completion of frame or wake-up transmission. Here, an interrupt request is generated if the FTCIE bit in the RLN21nmLiIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

13.3.3.11 RLN21nmLiEST — LIN Error Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiEST: <RLIN21n_base> + 13_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	CSER	—	FER	FTER	PBER	BER
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Table 13.25 RLN21nmLiEST Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5	CSER	Checksum Error Flag 0: Checksum error has not been detected. 1: Checksum error has been detected.
4	Reserved	The write value should be the value after reset.
3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
2	FTER	Frame Timeout Error Flag 0: Frame timeout error has not been detected. 1: Frame timeout error has been detected.
1	PBER	Physical Bus Error Flag 0: Physical bus error has not been detected. 1: Physical bus error has been detected.
0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN21nmLiEST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication.

In LIN reset mode, the register retains 00_H.

When the FTS bit in the RLN21nmLiTRC register is 1 (frame transmission or wake-up transmission/ reception is started), do not write a value to this register.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

CSER Flag (Checksum Error Flag)

Only 0 can be written to the CSER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The CSER flag is set to 1 upon checksum error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

FER Flag (Framing Error Flag)

Only 0 can be written to the FER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FER flag is set to 1 upon framing error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

FTER Flag (Frame Timeout Error Flag)

Only 0 can be written to the FTER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTER flag is set to 1 upon frame timeout error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

PBER Flag (Physical Bus Error Flag)

Only 0 can be written to the PBER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The PBER flag is set to 1 upon physical bus error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

BER Flag (Bit Error Flag)

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The BER flag is set to 1 upon bit error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

13.3.3.12 RLN21nmLiDFC — LIN Data Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiDFC: <RLIN21n_base> + 14_H + i × 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	FSM	CSM	RFT	RFDL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.26 RLN21nmLiDFC Register Contents

Bit Position	Bit Name	Function
7	Reserved	The write value should be the value after reset.
6	FSM	Frame Separate Mode Select 0: Frame separate mode is not set. 1: Frame separate mode is set.
5	CSM	Checksum Select 0: Classic checksum mode 1: Enhanced checksum mode
4	RFT	Response Field Communication Direction Select 0: Reception 1: Transmission
3 to 0	RFDL[3:0]	Response Field Length Select b3 b0 0 0 0 0: 0 byte (+ checksum) 0 0 0 1: 1 byte (+ checksum) 0 0 1 0: 2 bytes (+ checksum) : 0 1 1 1: 7 bytes (+ checksum) 1 0 0 0: 8 bytes (+ checksum) Settings other than the above are prohibited.

Set the RLN21nmLiDFC register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

FSM Bit (Frame Separate Mode Select)

The FSM bit sets the response transmission mode.

With 0 set, frame separate mode is not selected. In this case, after header transmission is started (the FTS bit in the RLN21nmLiTRC register is 1), response is transmitted/received without the RTS bit in the RLN21nmLiTRC register being set.

With 1 set, frame separate mode is selected. In this case, when the RTS bit in the RLN21nmLiTRC register is set to 1 during header transmission, response is transmitted after the header transmission is completed.

For response reception (the RFT bit is 0), set the FSM bit to 0.
When causing a transition to LIN self-test mode, set this bit to 0 before transition.
For details of frame separate mode, see **Section 13.11.1, Transmission of LIN frames**.

CSM Bit (Checksum Select)

The CSM bit sets the checksum mode.
With 0 set, classic checksum mode is selected.
With 1 set, enhanced checksum mode is selected.
When the frame timeout error is used (the FTERE bit in the RLN21nmLiEDE register is 1), the specific timeout time depends on the setting of this bit. For details of the bit error, see **Section 13.14, Error Status**.

RFT Bit (Response Field Communication Direction Select)

The RFT bits set the direction of the response field communication.
With 0 set, reception is performed in the response field. In LIN wake-up mode, wake-up reception is performed (low level width of the input signal is counted).
With 1 set, transmission is performed in the response field. In LIN wake-up mode, wake-up transmission is performed.

RFDL[3:0] Bits (Response Field Length Select)

The RFDL bits set the length of the response field data.
The data length can be 0 to 8 bytes excluding the checksum size.

13.3.3.13 RLN21nmLiIDB — LIN ID Buffer Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiIDB: <RLIN21n_base> + 15_H + i × 20_H

Value after reset: XX_H

Bit	7	6	5	4	3	2	1	0
	IDP1	IDP0	ID[5:0]					
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.27 RLN21nmLiIDB Register Contents

Bit Position	Bit Name	Function
7	IDP1	Parity Setting (P1) Sets the parity bits (P1) to be transmitted in the ID field.
6	IDP0	Parity Setting (P0) Sets the parity bits (P0) to be transmitted in the ID field.
5 to 0	ID[5:0]	ID Setting Sets the 6-bit ID value to be transmitted in the ID field.

Set the RLN21nmLiIDB register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

In LIN self-test mode, this register operates as follows:

- When RFT bit is 1 (transmission):

  The inverse of the value transmitted can be read from the register. The value to be transmitted can be written to the register before communication.

- When RFT bit is 0 (reception):

  The inverse of the value received can be read from the register. The value to be received can be written to the register before communication.

IDP Bits (Parity Setting)

The IDP bits set the parity bits (P0 and P1) to be transmitted in the ID field of the LIN frame. IDP0 is P0 and IDP1 is P1. Since parity is not automatically calculated, set the calculation result. Note that if the erroneous result is set, it is transmitted as is.

ID Bits (ID Setting)

The ID bit sets the 6-bit ID value to be transmitted in the ID field of the LIN frame.

13.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register

Access: This register can be read/written in 8-bit units.
However, this register can be read/written in 8-bit units in LIN self-test mode.

Address: RLN21nmLiCBR: <RLIN21n_base> + 16_H + i × 20_H

Value after reset: XX_H

Bit	7	6	5	4	3	2	1	0
	CKSM[7:0]							
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.28 RLN21nmLiCBR Register Contents

Bit Position	Bit Name	Function
7 to 0	CKSM[7:0]	Holds the checksum value transmitted or received.

In LIN operation mode, this register operates as follows:

- When the RFT bit in the RLN21nmLiDFC register is 1 (transmission):
  The value transmitted can be read from the register. Read the value after transmission is completed. Read the value after transmission is completed.
  Writing to this register is invalid.
- When the RFT bit in the RLN21nmLiDFC register is 0 (reception):
  The value received can be read from the register. Read the value after reception is completed.
  Writing to this register is invalid.

In LIN self-test mode, this register operates as follows:

- When the RFT bit in the RLN21nmLiDFC register is 1 (transmission):
  The inverse of the value received can be read from the register after frame transmission/reception completion (after loop-back). Writing to this register is invalid.
- When the RFT bit in the RLN21nmLiDFC register is 0 (reception):
  The value to be received should be written to the register before communication. The inverse of the value received can be read from the register after frame transmission/reception is completed (after loop-back).

Set the RLN21nmLiCBR register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

13.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiDBR1: <RLIN21n_base> + 18_H + i × 20_H, RLN21nmLiDBR2: <RLIN21n_base> + 19_H + i × 20_H, RLN21nmLiDBR3: <RLIN21n_base> + $1A_H$ + i × 20_H, RLN21nmLiDBR4: <RLIN21n_base> + $1B_H$ + i × 20_H, RLN21nmLiDBR5: <RLIN21n_base> + $1C_H$ + i × 20_H, RLN21nmLiDBR6: <RLIN21n_base> + $1D_H$ + i × 20_H, RLN21nmLiDBR7: <RLIN21n_base> + $1E_H$ + i × 20_H, RLN21nmLiDBR8: <RLIN21n_base> + $1F_H$ + i × 20_H

Value after reset: XX_H

Bit	7	6	5	4	3	2	1	0
	LDB[7:0]							
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.29 RLN21nmLiDBRm (m = 1 to 8) Register Contents

Bit Position	Bit Name	Function
7 to 0	LDB[7:0]	Sets the data to be transmitted or allows the received data to be read. Setting Range: 00_H to FF_H

- For response transmission:

  The RLN21nmLiDBRm registers set the data to be transmitted in the response field.
  Use these registers with the following settings.

  – RFT in RLN21nmLiDFC register is 1 (transmission)
  – FSM in RLN21nmLiDFC register is 0 (not frame separate mode)
  – FTS bit in RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/ reception is halted)

  or

  – RFT in RLN21nmLiDFC register is 1 (transmission)
  – FSM in RLN21nmLiDFC register is 1 (frame separate mode)
  – RTS in RLN21nmLiTRC register is 0 (response transmission is halted)

- For response reception:

  The RLN21nmLiDBRm registers hold the data received in the response field.
  The received data is overwritten. If an error is detected, the data up to the byte in which the error was detected are stored in the register.

  Do not read these registers when the FTS bit is 1 (frame transmission or wake-up transmission/ reception is started)

In LIN self-test mode, these registers operate as follows:

- When the RFT bit in the RLN21nmLiDFC register is 1 (transmission):

  The inverse of the value transmitted can be read from the registers. The value to be transmitted can be written to the registers before communication.

- When the RFT bit in the RLN21nmLiDFC register is 0 (reception):

  The inverse of the value received can be read from the register. The value to be received can be written to the registers before communication.

13.4 Interrupt Sources

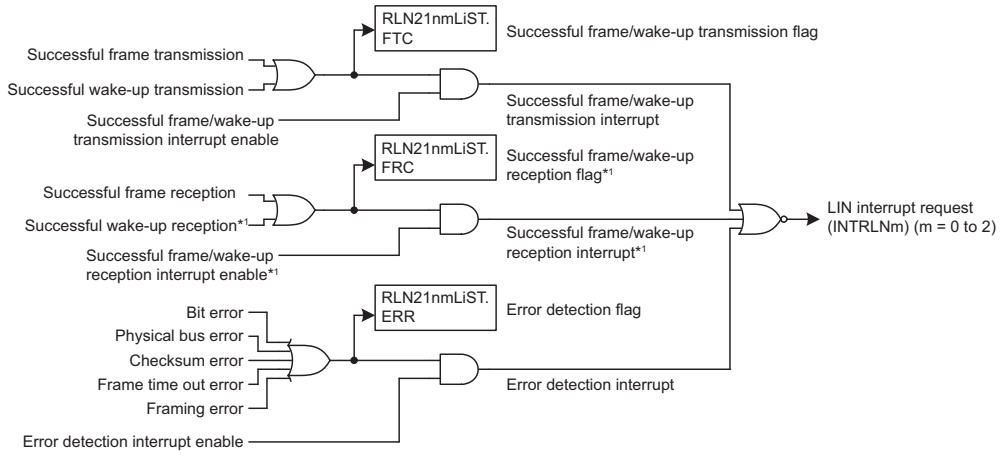
The LIN interrupts are interrupt requests generated by the LIN master interface.

There are three interrupt factors for each channel; frame/wake-up transmit completion, frame/wake-up receive completion, and error detection.

The interrupt request from three interrupt states, frame/wake-up transmit completion, frame/wake-up receive completion, and error detection, is ORed to be one interrupt request “LIN interrupt”.

The respective interrupt request is output when the corresponding flag in the RLN21nmLiST register is set to 1 while the corresponding bit in the RLN21nmLiIE register is 1 (interrupt enabled). However, if an interrupt is requested when the corresponding flag in the RLN21nmLiST register has been set to 1, it is ignored. Therefore, clear the corresponding flag to 0 to enable the interrupt.

Figure 13.2 shows a block diagram of the LIN interrupt.



Note 1. The wake-up receive signal indicates the count of input signal low width.

Figure 13.2 LIN Interrupt Block Diagram

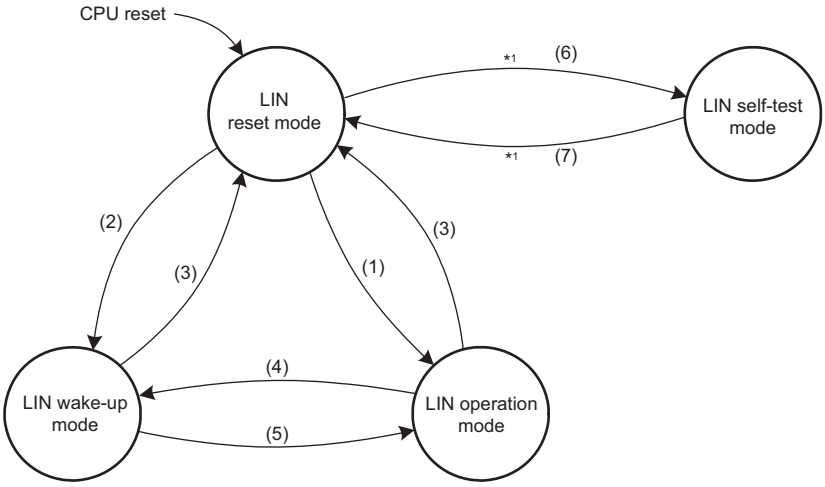
13.5 Modes

The LIN Master Interface provides the following four modes, depending upon the specific function to be performed:

- LIN reset mode
- LIN operation mode
- LIN wake-up mode
- LIN self-test mode

The transition of the operating modes except LIN self-test mode is controlled independently for respective channels.

Figure 13.3 shows mode transitions. **Table 13.30** describes mode transition conditions. **Table 13.31** lists operations available in each mode.



Note 1. See **Section 13.15, LIN Self-Test Mode** for the details.

Figure 13.3 Mode Transitions

Table 13.30 Transition Condition of Each Mode

	Mode Transition			Transition Condition
1	LIN reset mode	→	LIN operation mode	RLN21nmLiCUC.OM1,OM0 = 11_B
2	LIN reset mode	→	LIN wake-up mode	RLN21nmLiCUC.OM1,OM0 = 01_B
3	LIN wake-up mode LIN operation mode	→	LIN reset mode	RLN21nmLiCUC.OM0 = 0_B
4	LIN operation mode	→	LIN wake-up mode	RLN21nmLiCUC.OM1,OM0 = 01_B
5	LIN wake-up mode	→	LIN operation mode	RLN21nmLiCUC.OM1,OM0 = 11_B
6	LIN reset mode	→	LIN self-test mode	See **Section 13.15, LIN Self-Test Mode**.
7	LIN self-test mode	→	LIN reset mode	See **Section 13.15, LIN Self-Test Mode**.

Table 13.31 Operations Available in Each Mode

Lin Operation Mode	Lin Wake-up Mode	Lin Self-test Mode
Header transmission Response transmission Response reception Error detection	Wake-up transmission Wake-up reception Error detection	Self test

Whether a transition has been made to LIN reset mode, the LIN mode, or the UART mode can be verified by reading the OMM1 and OMM0 bits in the RLN21nmLiMST register.

For a description of the LIN self-test mode, see **Section 13.15, LIN Self-Test Mode**.

13.6 LIN Reset Mode

Setting the OM0 bit in the RLN21nmLiCUC register to 0_B (LIN reset mode) causes a transition to LIN reset mode. The change to LIN reset mode can be verified by determining that the OMM0 bit in the RLN21nmLiMST register has been set to 0_B (LIN reset mode). In this mode, the LIN communication stops.

From LIN reset mode, transitions to LIN operation mode, LIN wake-up mode, and LIN self-test mode can be made.

When the mode changes to LIN reset mode, the following registers are initialized to their reset values, and as long as LIN reset mode is in effect, they retain their value after resets.

- RLN21nmLiTRC register
- RLN21nmLiST register
- RLN21nmLiEST register

The following registers retain their previous values even when a transition to LIN reset mode is made:

- RLN21nGLWBR register
- RLN21nGLBRP0 register
- RLN21nGLBRP1 register
- RLN21nmLiMD register
- RLN21nmLiBFC register
- RLN21nmLiSC register
- RLN21nmLiWUP register
- RLN21nmLiIE register
- RLN21nmLiEDE register
- RLN21nmLiDFC register
- RLN21nmLiIDB register
- RLN21nmLiCBR register
- RLN21nmLiDBRb register

13.7 LIN Operation Mode

In LIN operation mode, frame processing (header transmission, header reception, response transmission, response reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN21nmLiCUC register to 11_B changes the mode to LIN operation mode, changing the OMM1 and OMM0 bits in the RLN21nmLiMST register to 11_B. Communication settings should be performed after the RLN21nmLiMST register has become 11_B.

13.8 LIN Wake-up Mode

In LIN wake-up mode, wake-up signal processing (wake-up transmission, wake-up reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN21nmLiCUC register to 01_B changes the mode to LIN wake-up mode, changing the OMM1 and OMM0 bits in the RLN21nmLiMST register to 01_B. Communication settings should be performed after the RLN21nmLiMST register has become 01_B.

13.9 Header Transmission/Response Transmission/ Response Reception

13.9.1 Header Transmission

Figure 13.4 shows the operation of the LIN Master Interface (LIN master mode) in header transmission. **Table 13.32** provides processing in header transmission.

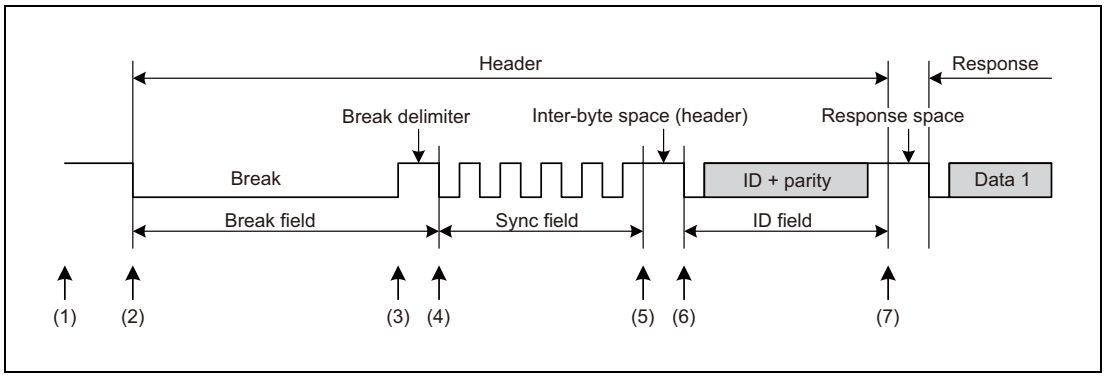


Figure 13.4 Operation in Header Transmission

Table 13.32 Processing in Header Transmission

	Software Processing	LIN Master Interface Processing
(1)	• Sets a baud rate • Enables interrupt • Enables error detection • Sets frame configuration parameters • Changes LIN operation mode • Sets information on the frame to be transmitted (ID, parity, data length, response direction, checksum method, and transmission data)	Waits for the setting of the FTS bit in the RLN21nmLiTRC register to 1 by software (idle).
(2)	Sets the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started)	Transmits a break.
(3)	Waits for an interrupt request	Transmits a break delimiter.
(4)		Transmits a sync field (55_H).
(5)		Transmits an inter-byte space (header).
(6)		Transmits an ID field.
(7)		Sets a successful header transmission flag.

Note: For information about error detection, see **Section 13.14, Error Status**.

13.9.2 Response Transmission

Figure 13.5 shows the operation of the LIN Master Interface (LIN master mode) in response transmission. **Table 13.33** provides processing in response transmission.

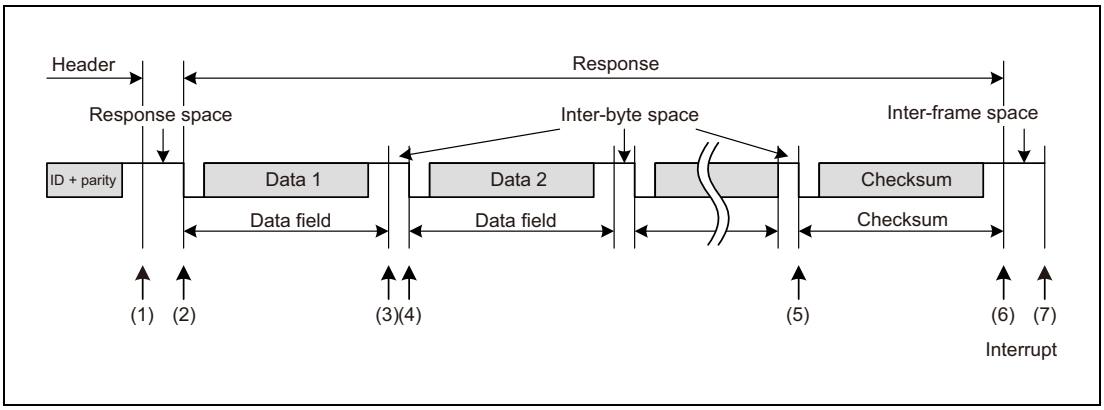


Figure 13.5 Operation in Response Transmission

Table 13.33 Processing in Response Transmission

	Software Processing	LIN Master Interface Processing
(1)	(When in frame separate mode) • Sets the RTS bit in the RLN21nmLiTRC register to 1 (response transmission started) (When not in frame separate mode) • Waits for an interrupt request	(When in frame separate mode) • Waits for the setting of the RTS bit in the RLN21nmLiTRC register to 1 by software. (During this time, 1 is output.) • When the bit is set to 1, sends a response space. (When not in frame separate mode) • Sends a response space.
(2)	Waits for an interrupt request	Transmits the data 1.
(3)		Transmits an inter-byte space.
(4)		• Transmits the data 2. • Transmits an inter-byte space • Transmits the data 3. • Transmits an inter-byte space (Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN21nmLiDFC register. : :
(5)		Transmits the checksum.
(6)		• Sets a successful frame/wake-up transmission flag. • Sets the FTS bit in the RLN21nmLiTRC register to 0 (frame transmission or wake-up transmission/reception stopped) (When in frame separate mode). • Sets the RTS bit in the RLN21nmLiTRC register to 0 (response transmission stopped).
(7)	• Processing after communication Checks the RLN21nmLiST register, and clears flags.	Idle

Note: For information about error detection, see **Section 13.14, Error Status**.

13.9.3 Response Reception

Figure 13.6 shows the operation of the LIN Master Interface (LIN master mode) on response reception. **Table 13.34** provides processing in response reception.

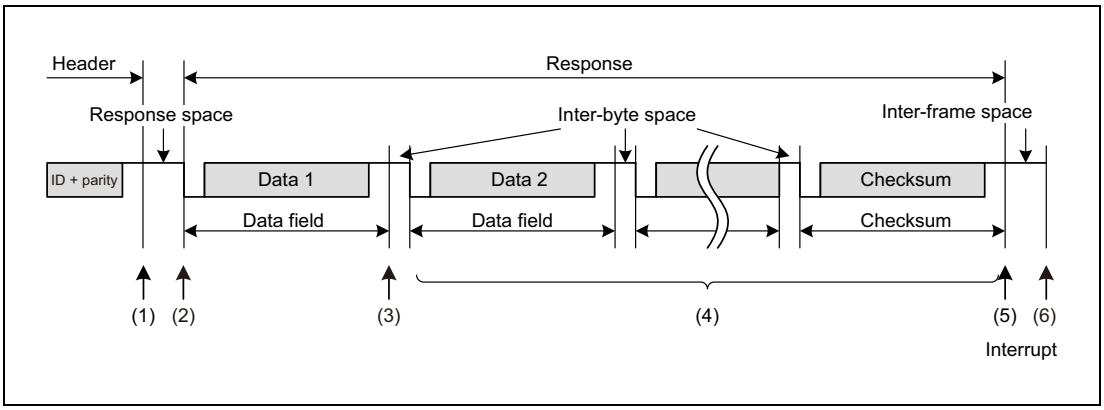


Figure 13.6 Operation in Response Reception

Table 13.34 Processing in Response Reception

	Software Processing	LIN Master Interface Processing
(1)	Waits for an interrupt request (no processing).	Waits for detection of a start bit.
(2)	Waits for an interrupt request.	Receives the data 1 when the start bit is detected.
(3)		Sets the successful data 1 reception flag.
(4)		• Receives the data 2 when the start bit is detected. • Receives the data 3 when the start bit is detected. Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN21nmLiDFC register. : : • Receives the checksum when the start bit is detected.
(5)		• Determines the checksum. • Sets the successful frame/wake-up reception flag. • Sets the FTS bit in the RLN21nmLiTRC register to 0 (frame transmission or wake-up transmission/reception stopped).
(6)	• Processing after communication Reads the received data. Checks the RLN21nmLiST register, and clears flags.	Idle

NOTE

For information about error detection, see **Section 13.14, Error Status**.

13.10 Data Transmission/Reception

13.10.1 Data Transmission

One bit of data is transmitted per 1 Tbit.

The data that is transmitted returns to the reception data input pin via the LIN transceiver. The received data and the transmitted data is compared bit by bit, and the results are stored in the BER flag in the RLN21nmLiEST register (see **Section 13.14, Error Status**).

In LIN Master interface, the sampling point for received data, 1 Tbit is generated to be 16fLIN, and thus is at the 13th clock cycle (81.25% position).

Figure 13.7 shows an example of data transmission timing.

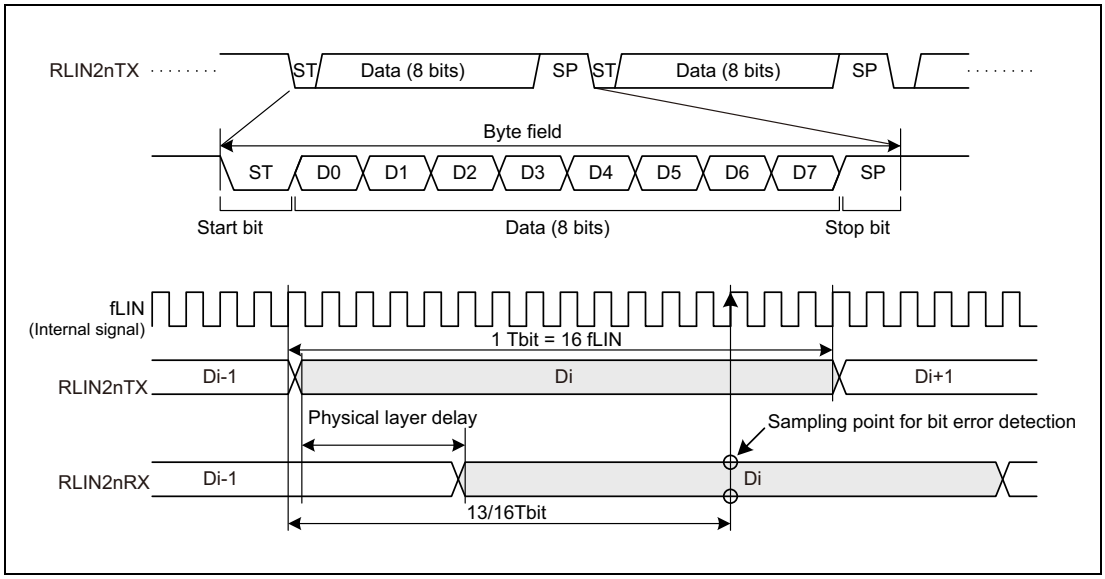


Figure 13.7 Example of Data Transmission Timing

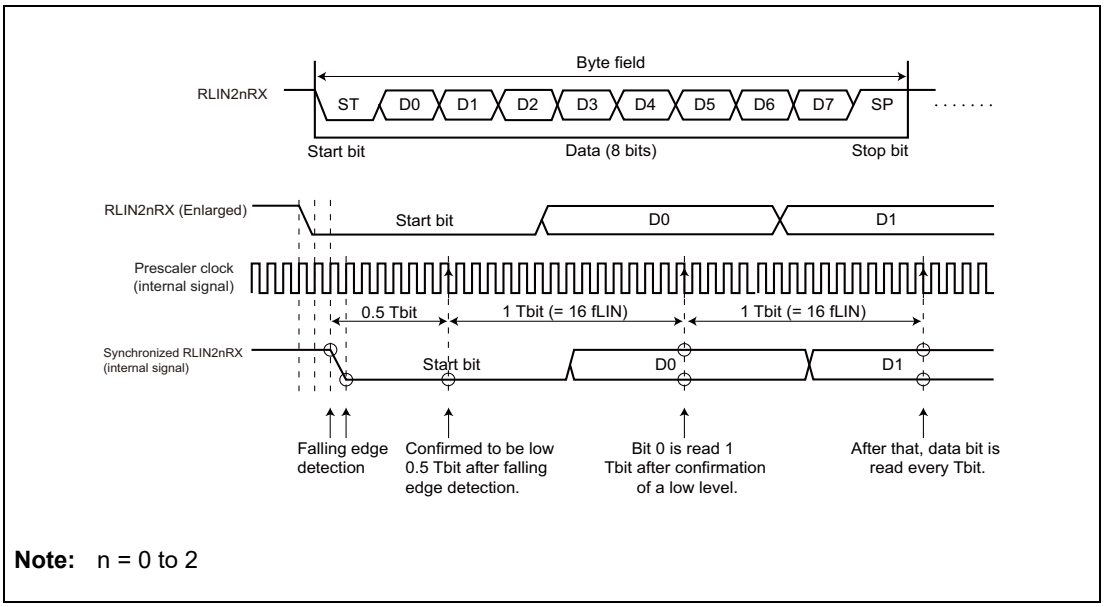
13.10.2 Data Reception

Data reception is performed by using the synchronized RLIN2nRX signal (an internal signal) that is the input from the RLIN2nRX pin synchronized with the LIN system clock (fLIN).

The byte field is synchronized at the falling edge of the start bit for the synchronized RLIN2nRX signal. After the falling edge is detected, sampling is performed again 0.5 Tbit later, and the falling edge is recognized as a start bit if the synchronized RLIN2nRX signal is low level. The falling edge is not recognized as a start bit if the RLIN2nRX signal after the clearing of the resetting is low-level-fixed or if a high level is detected on re-sampling.

After the start bit is detected, the system samples 1 bit per Tbit.

Figure 13.8 shows an example of data reception timing.



Note: n = 0 to 2

Figure 13.8 Example of Data Reception Timing

13.11 Transmission/Reception Data Buffering

This section explains the buffer processing that takes place when the LIN Master Interface sends or receives data continuously.

13.11.1 Transmission of LIN frames

For an 8-byte transmission, the contents stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR8 are sequentially transmitted to data areas 1 to 8 of the LIN frame. In the case of a 4-bytes transmission, the contents stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR4 are transmitted to data areas 1 to 4 of the LIN frame, but the contents of registers RLN21nmLiDBR5 to RLN21nmLiDBR8 are not transmitted. The transmitted checksum data is stored in the RLN21nmLiCBR register.

Figure 13.9 depicts the LIN transmission processing and the required buffer.

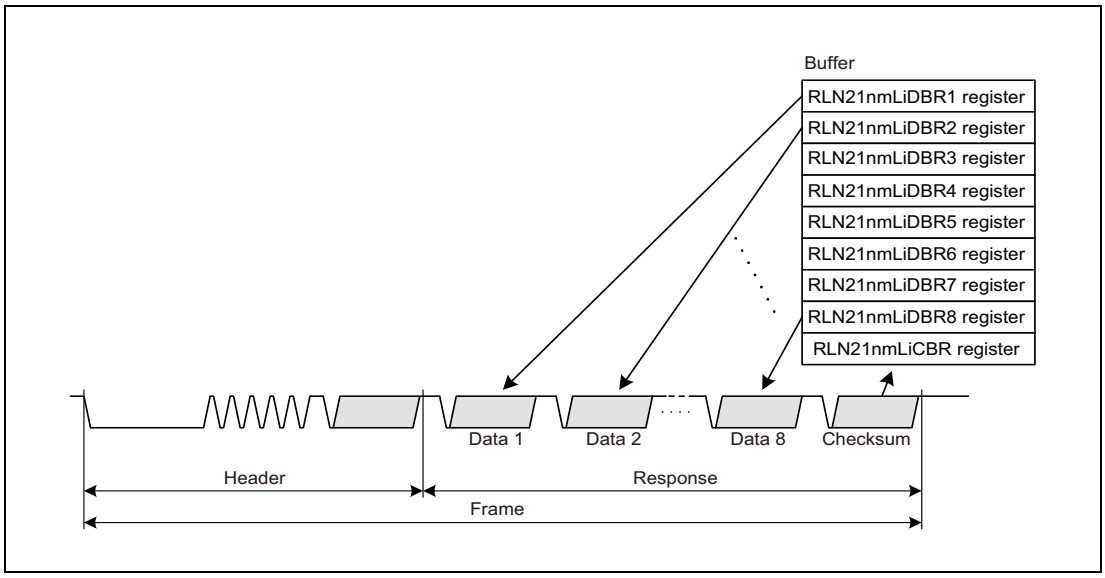


Figure 13.9 LIN Transmission Processing and Required Buffer

(1) Frame Separate Mode

Setting the FSM bit in the RLN21nmLiDFC register to 1 turns on the frame separate mode.

In frame separate mode, a header and a response are transmitted when prompted by separate transmission start requests.

When the transmission of a header is finished, the HTRC flag in the RLN21nmLiST register turns 1 (successful header transmission).

13.11.2 Reception of LIN Frames

For an 8-byte reception, the contents of data areas 1 to 8 of the LIN frame is stored in registers RLN21nmLiDBR1 to RLN21nmLDBR8, respectively, upon receipt of a stop bit. In the case of a 4-byte reception, the contents of data areas 1 to 4 of the LIN frame are stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR4, respectively; however, no data is stored in registers RLN21nmLiDBR5 to RLN21nmLiDBR8. Also, the received checksum data is stored in the RLN21nmLiCBR register.

Figure 13.10 depicts the LIN reception processing and the required buffer.

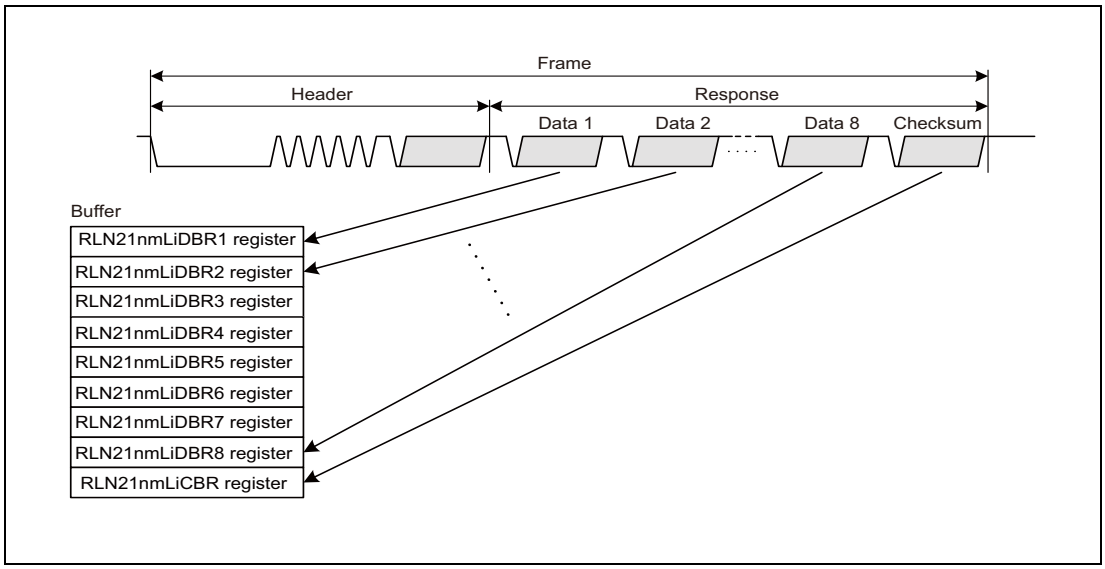


Figure 13.10 LIN Reception Processing and Required Buffer

(1) Reception of Data 1

When the reception of the first byte of data is finished, the D1RC flag in the RLN21nmLiST register turns 1 (successful data 1 reception).

13.12 Wake-up Transmission/Reception

The wake-up transmission/reception can be used in LIN wake-up mode.

13.12.1 Wake-up Transmission

In LIN wake-up mode, setting the RFT bit in the RLN21nmLiDFC register to 1 (transmission) and the FTS bit in the RLN21nmLiTRC register to 1 (header transmission or wake-up transmission/reception started) causes a wake-up signal to be output from the output pin. The low level width of the wake-up signal should be set using the WUTL[3:0] bits in the RLN21nmLiWUP register. However, if the value of the RLN21nGLWBR.LWBR0 bit is 1 (for LIN2.x use), the LIN system clock (f_{LIN}) becomes low level width at fa regardless of the setting of the RLN21nmLiMD.LCKS bit. By setting the baud rate to 19200 bps (while fa is selected) and the RLN21nmLiWUP.WUTL[3:0] bits to 0100_B (5 Tbits), 260 µs low width can be output in LIN wake-up mode regardless of the setting of the RLN21nmLiMD.LCKS bit.

If a wake-up low is output without any error, the FTC flag in the RLN21nmLiST register turns 1 (successful frame or wake-up transmission); when the FTCIE bit in the RLN21nmLiIE register is 1 (successful frame/wake-up transmission interrupt enabled), an interrupt request is generated.

If an error is detected, wake-up transmission is suspended and the error flag (the PBER flag or the BER flag in the RLN21nmLiEST register) for the error detected is set to 1 (detection of the physical bus error or bit error).

Figure 13.11 shows the wake-up transmission timing.

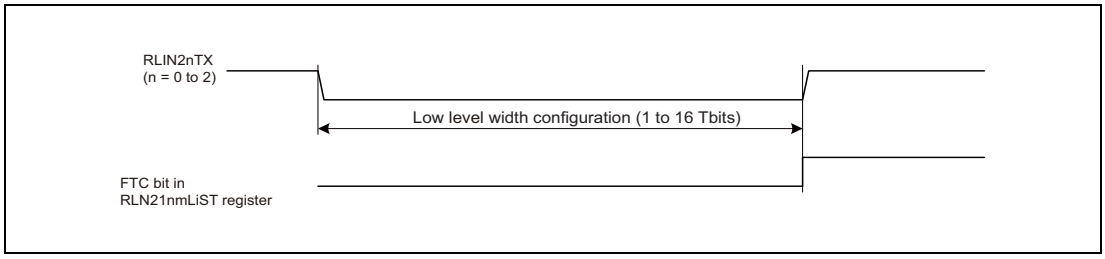


Figure 13.11 Wake-up Transmission Timing

13.12.2 Wake-up Reception

The detection of a wake-up signal involves the use of an input signal low level width count function.

The input signal low level width count function measures the low level width of the input signal to the RLIN2nRX pin, using the same sampling point as data reception. This allows the 2.5-Tbit or longer low-level width of the input signal of fLIN to be measured.

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN21nmGLWBR register to 0. When LIN Specification Package Revision 2.0, 2.1 is used, set the RLN21nmGLWBR register to 1.

When the LWBR0 bit is set to 1, regardless of the setting of the LCKS bit in the RLN21nmLiMD register, fa is selected as the LIN system clock (fLIN) (the LCKS bit is not changed).

Setting the baud rate to 19200 bps while fa is selected allows the 130 µs or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the RLN21nmLiMD register.

When using this function, in LIN wake-up mode set the RFT bit in the RLN21nmLiDFC register to 0 (reception), and the FTS bit in the RLN21nmLiTRC register to 1 (header transmission or wake-up transmission/reception a started).

When the low level width to be measured is reached, the FRC flag in the RLN21nmLiST register turns 1 (successful frame/wake-up reception). If the FRCIE bit in the RLN21nmLiIE register is 1 (successful frame or wake-up reception interrupt enabled), an interrupt request is generated.

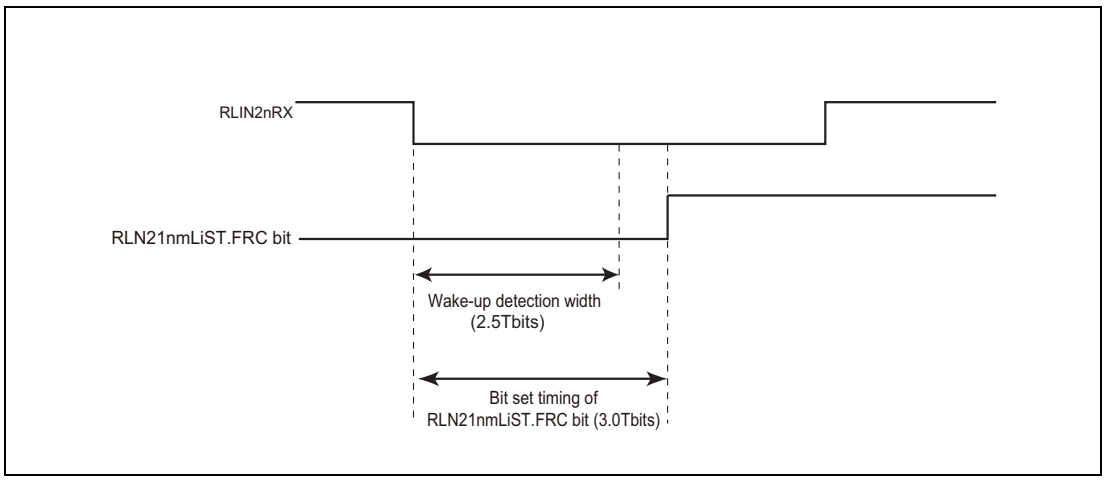


Figure 13.12 Input Signal Low level Count Function

At the transmission of wake-up, low-level width of the input signal is not counted.

13.12.3 Wake-up Collision

If the master node and the slave node transmit wake-up signals simultaneously, a collision will occur on the LIN bus, though a collision of wake-up signals is not detected in the LIN module.

13.13 Status

During LIN mode operation, the LIN Master Interface can detect seven types of status.

The three statuses, successful frame/wake-up transmission, successful frame/wake-up reception, error detection, scan generate interrupt requests.

Table 13.35 shows the types of status available.

Table 13.35 Types of Status

Status	Status Set Condition	Status Clear Condition	Operation Mode Capable Of Status Detection	Corresponding Bit	Interrupt
Reset	After the OM0 bit in the RLN21nmLiCUC register is set to not-LIN–reset-mode, if actually the LIN Master Interface is cleared from LIN reset mode.	After the OM0 bit in the RLN21nmLiCUC register is set to LIN reset mode, if actually the LIN Master Interface enters LIN reset mode.	All modes	OMM0 bit in RLN21nmLiMST register	—
Operation mode	After the OM1 bit in the RLN21nmLiCUC register is set to LIN operation mode, if actually the LIN Master Interface enters LIN operation mode.	After the OM1 bit in the RLN21nmLiCUC register is set to LIN wake-up mode, if actually the LIN Master Interface enters LIN wake-up mode.	• LIN operation mode • LIN wake-up mode	OMM1 bit in RLN21nmLiMST register	—
Frame/wake-up transmission end	When a frame (header transmission + response transmission), a wake-up signal, or a data group is transmitted successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FTC flag in RLN21nmLiST register	√
Frame/wake-up reception end	When a frame (header transmission + response reception), a wake-up signal, or a data group is received successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FRC flag in RLN21nmLiST register	√
Error detection	If any of the PRER flag, CSER flag, FER flag, FTER flag, PBER flag, and BER flags in the RLN21nmLiEST register turns 1 (error detected).	• When another communication is started • When cleared by software *1 • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	ERR flag in RLN21nmLiST register	√
Data 1 reception end	The RFT bit in the RLN21nmLiDFC register is 0 (reception) and the first byte of the response field is received.*2	• When another communication is started • When cleared by software • After transition to LIN reset mode	LIN operation mode	D1RC flag in RLN21nmLiST register	—
Header reception end	When a header field is transmitted successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	LIN operation mode	HTRC flag in RLN21nmLiST register	—

Note 1. In LIN operation mode, the ERR flag in the RLN21nmLiST register is cleared to 0 by writing 0 to the CSER flag, FER flag, FER flag, FTER flag, PBER flag or BER flags in the RLN21nmLiEST register.

Note 2. Not detected when the RFDL [3:0] bits in the RLN21nmLiDFC register are 0000_B (0-byte + checksum).

13.14 Error Status

13.14.1 Types of Error Status

The LIN Master Interface can detect five types of error status in LIN master mode. The condition of these error statuses can be checked by means of the corresponding bits in the RLN21nmLiEST register.

All error statuses represent interrupt events.

Table 13.36 shows the types of error status.

Table 13.36 Types of Error Status

Status	Error Detection Condition	Operation Mode Capable Of Error Detection	Comm-unication	Enable/ Disable Detection	Corresponding Bit
Bit error	The transmitted data and the data on the LIN bus monitored by the receive pin do not match *1	• LIN operation mode • LIN wake-up mode	Suspended	√	BER flag in RLN21nmLiEST register
Physical bus error	• LIN bus is detected to be high level when sending a break • LIN bus is detected to be low level when sending a break delimiter • LIN bus is detected to be high level when sending a wake-up	• LIN operation mode • LIN wake-up mode	Suspended	√	PBER flag in RLN21nmLiEST register
Timeout error	A frame or response transmission/reception does not terminate within a given time*2	LIN operation mode	Suspended	√	FTER flag in RLN21nmLiEST register
Framing error	In response field reception, a stop bit of each data byte is low level	LIN operation mode	Suspended	√	FER flag in RLN21nmLiEST register
Checksum error	In response field reception, the result of checksum test gives an error	LIN operation mode	—	—	CSER flag in RLN21nmLiEST register

Note 1. If a bit error is detected, processing is stopped after a stop bit is sent. If a bit error is detected in a non-data area, such as an inter-byte space, the transmission is suspended immediately after the bit which had the error is sent. If a bit error is detected during the transmission of a wake-up, the transmission of the wake-up is suspended after the bit which had the error is sent.

Note 2. The timeout time depends on the response field data length (the RFDL 3:0] bits in the RLN21nmLiDFC register) and the checksum selection (the CSM bit in the RLN21nmLiDFC register), and can be calculated from the following formula:
On classic selection (when the CSM bit in RLN21nmLiDFC is 0): Timeout time = 49 + (number of data bytes + 1) × 14 [Tbit]
On enhanced selection (when the CSM bit in RLN21nmLiDFC is 1): Timeout time = 48 + (number of data bytes + 1) × 14 [Tbit]

The aforementioned timeout time is a time greater than the TFRAME_MAX of LIN Specification Package Revision 1.3 on classic selection, or the TFRAME_MAX of LIN Specification Package Revision 2.0 or 2.1 on enhanced selection.

Target Time Area for LIN Error Detection.

The error status is cleared when the next communication is started, by software, or at a transition to LIN reset mode.

13.14.2 Target Time Area for Error Detection

Figure 13.13 shows the time domain in which the LIN Master Interface performs monitoring for error detection.

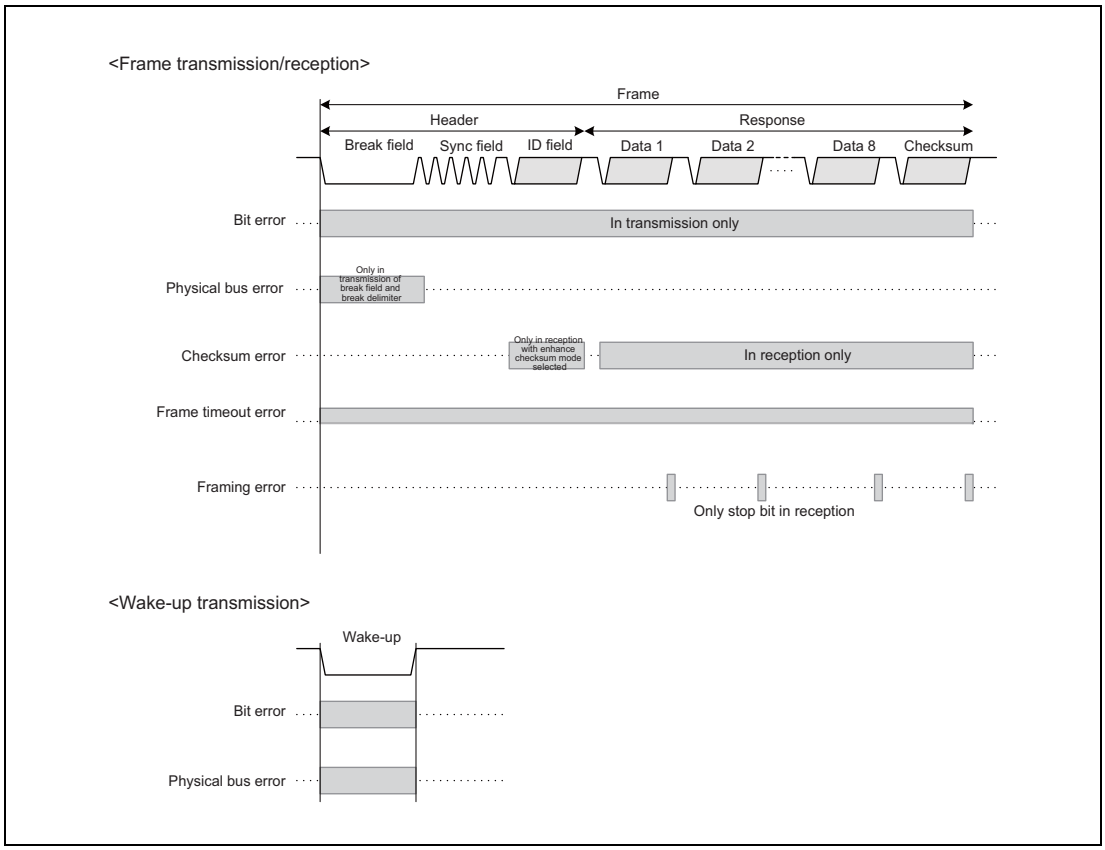


Figure 13.13 Target Time Area for Error Detection

13.15 LIN Self-Test Mode

The LIN Master Interface provides LIN self-test mode. When LIN self-test mode is turned on, RLIN2nTX and RLIN2nRX are disconnected from the external pin, and the internal RLIN2nTX and RLIN2nRX are connected. Thus, the frame transmitted from RLIN2nTX is returned to the internal RLIN2nRX (loop back).

The functions of the LIN self-test mode operate in the following conditions:

- LIN self-test mode (transmission): header transmission and response transmission
- LIN self-test mode (reception): header reception and response reception

In LIN self-test mode, the operation is at the fastest baud rate, regardless of the setting of the baud rate generator. Regardless of the setting of the baud rate related registers, the baud rate operates at the LIN communication clock source/16 [bps].

In LIN self-test mode, the following functions are not supported:

- LIN Wake-up function
- Frame separate mode

Do not use above functions.

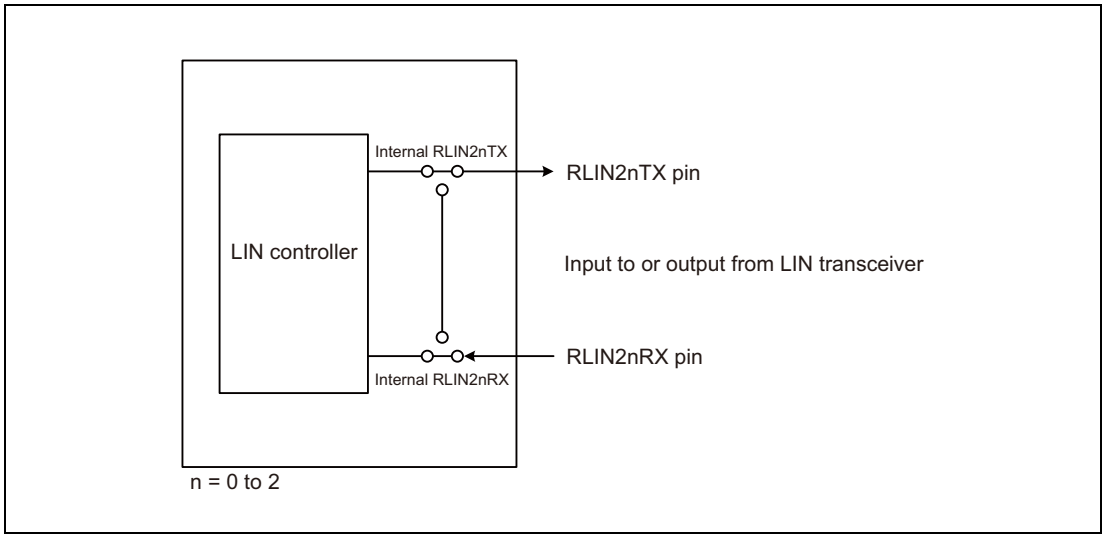


Figure 13.14 Connection in LIN Reset Mode, LIN Wake-Up Mode, and LIN Operation Mode

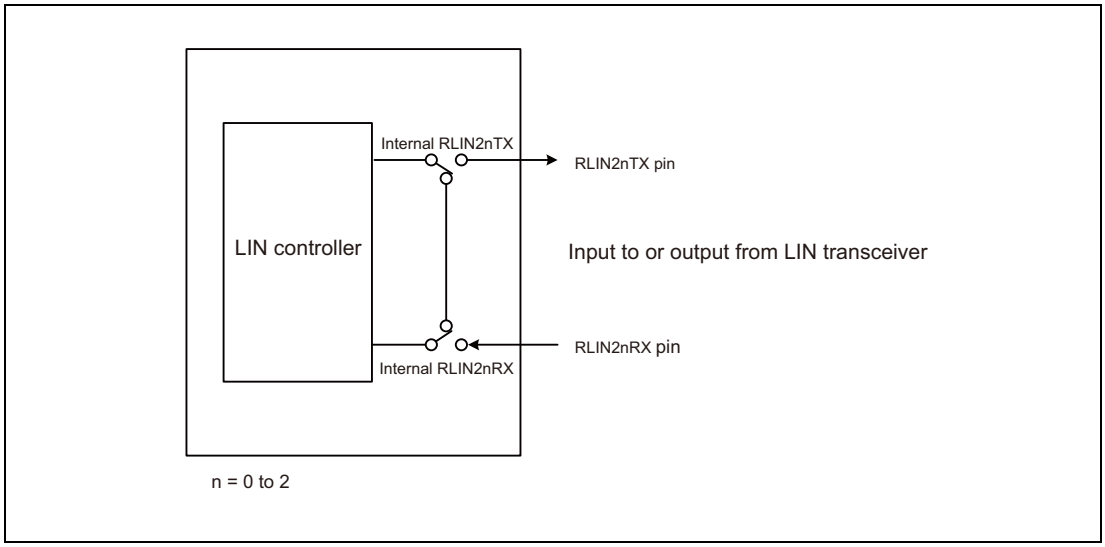


Figure 13.15 Connection in LIN Self-Test Mode

13.15.1 Change to LIN Self-Test Mode

Writing to the RLN21nGLSTC register enables LIN self-test mode.

The RLN21nGLSTC register becoming 01_H indicates that the mode is transit to LIN self-test mode.

When changing to LIN self-test mode, be sure to execute a specific sequence. In that sequence, information must be written three times consecutively to the LIN self-test control register, as follows:

- Change to LIN reset mode regarding all channels in the unit.
  Set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode).
  Read the OMM0 bit in the RLN21nmLiMST register; verify that it is 0 (LIN reset mode).
- 1st write: RLN21nGLSTC register = $1010\ 0111_B$ ($A7_H$)
- 2nd write: RLN21nGLSTC register = $0101\ 1000_B$ (58_H)
- 3rd write: RLN21nGLSTC register = $0000\ 0001_B$ (01_H)
- Verify the transition to LIN self-test mode
  Read the RLN21nGLSTC register; verify that it is 01_H (LIN self-test mode).

If the key of the first write ($A7_H$) is written twice by mistake, the transition to LIN self-test mode is suspended. The above sequence should be retried from the step of first write. In addition, if a write to another LIN-related register in the same unit is performed during transition to LIN self-test mode (three consecutive write operations to the RLN21nmLiSTC register), the transition is also suspended.

13.15.2 Transmission in LIN Self-Test Mode

To execute a self-test on transmission, perform the procedure below:

- Set the baud rate related registers
  RLN21nGLBRP0 register = xxxx $xxxx_B$*1
  RLN21nGLBRP1 register = xxxx $xxxx_B$*1
  RLN21nmLiMD register = 0000 $xx00_B$*1
- Set interrupt enable register and error enable related registers
  RLN21nmLiIE register = 0000 $0xxx_B$*2
  RLN21nmLiEDE register = 0000 $xxxx_B$
- Set the break field and space related registers.
  RLN21nmLiBFC register = 00xx $xxxx_B$
  RLN21nmLiSC register = 00xx $0xxx_B$
- Release from LIN reset mode
  Write 11_B to the OM1 and OM0 bits in the RLN21nmLiCUC register, and check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are 11_B.
- Set the transmit frame related registers.
  RLN21nmLiDFC register = 00x1 $xxxx_B$
  RLN21nmLiIDB register = xxxx $xxxx_B$
  RLN21nmLiDBR1 to RLN21nmLiDBR8 registers = xxxx $xxxx_B$
- Header transmission → response transmission started
  Set the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started).
  The LIN self-test mode (transmission) is executed. In this mode, interrupt are generated, and status and error status are also updated. The checksum is automatically calculated by the LIN master interface.
  To suspend the LIN self-test mode (transmission) while it is running, set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode), which causes a transition to the LIN reset mode.
- When the transmission is completed, the inverse of the looped-back frame data is stored in the RLN21nmLiIDB, RLN21nmLiDBRb, and RLN21nmLiCBR registers (the data is inverted before being stored because the transmitted value should be compared with the looped-back value). Then, the FTS bit in the RLN21nmLiTRC register is cleared.
- If the transmission fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN21nmLiTRC register is cleared.

NOTE

x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LCKS bit in the RLN21nGLBRP0 register, RLN21nGLBRP1 register, and RLN21nmLiMD register. Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

13.15.3 Reception in LIN Self-Test Mode

To execute a self-test on LIN master transmission, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN21nGLBRP0 register = xxxx $xxxx_B$*1
  RLN21nGLBRP1 register = xxxx $xxxx_B$*1
  RLN21nmLiMD register = 0000 $xx00_B$*1
- Set the interrupt enable and error enable related registers.
  RLN21nmLiIE register = 0000 $0xxx_B$*2
  RLN21nmLiEDE register = 0000 $xxxx_B$
- Set the break field and space related registers.
  RLN21nmLiBFC register = 00xx $xxxx_B$
  RLN21nmLiSC register = 00xx $0xxx_B$*1
- Release from LIN reset mode
  Write 11_B to the OM1 and OM0 bits in the RLN21nmLiCUC register, and check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are 11_B.
- Set the receive frame related registers.
  RLN21nmLiDFC register = 00x0 $xxxx_B$
  RLN21nmLiIDB register = xxxx $xxxx_B$
  RLN21nmLiDBR1 to RLN21nmLiDBR8 registers = xxxx $xxxx_B$
  RLN21nmLiCBR register = xxxx $xxxx_B$
  Since the checksum value to be transmitted is not automatically calculated, set the calculation value to the RLN21nmLiCBR register. If an incorrect checksum is set at this time, the checksum error can be tested.
- Header transmission → response reception started
  Set the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started).
  The LIN self-test mode (reception) is executed. In this mode, interrupt are generated, and status and error status are also updated.
  To suspend the LIN self-test mode (reception) while it is running, set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode), which causes a transition to the LIN reset mode.
- When the reception is completed, the inverse of the looped-back frame data is stored in the RLN21nmLiIDB, RLN21nmLiDBRb, and RLN21nmLiCBR registers (the data is inverted before being stored because the set value should be compared with the looped-back value). Then, the FTS bit in the RLN21nmLiTRC register is cleared.
- If the reception fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN21nmLiTRC register is cleared.

NOTE

x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LCKS bit in the RLN21nGLBRP0 register, the RLN21nGLBRP1 register, and the RLN21nmLiMD register, and the IBS bit and IBHS bit (response space only) in the RLN21nmLiSC register.
Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

13.15.4 Terminating LIN Self-Test Mode

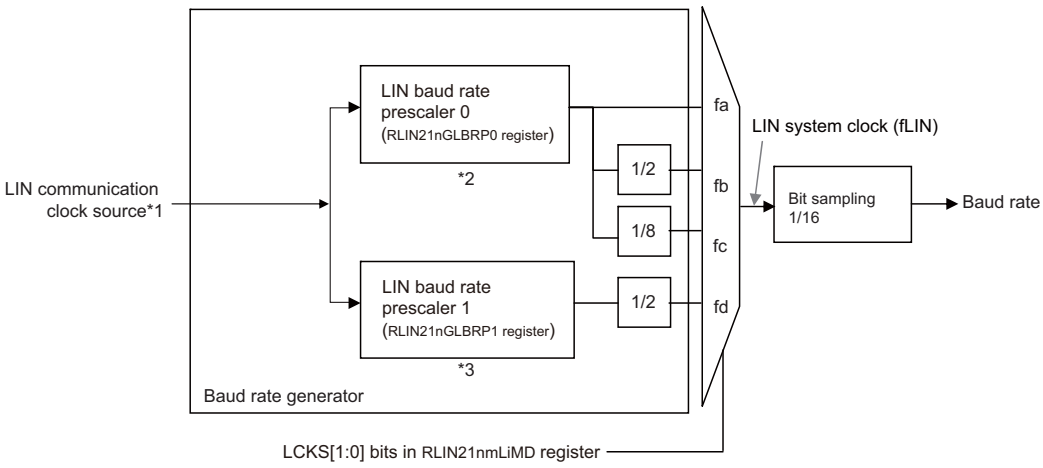
To terminate LIN self-test mode, perform the procedure below:

- All the channels in the unit make a transition to LIN reset mode
  Write 0 to the OM0 bit in the RLN21nmLiCUC register to make a transition to LIN reset mode. However, if the OMM1 and OMM0 bits in the RLN21nmLiMST register are not "11_B" in any channels of the unit after the transition to LIN self-test mode, write "11_B" to the OM1 and OM0 bits in the RLN21nmLiCUC register in any one channel. Check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are set to "11_B", and then make a transition to LIN reset mode.
- Verify release from LIN self-test mode.
  Read the RLN21nGLSTC register; confirm that it is not 00_H (not in LIN self-test)
- Verify the transition to LIN reset mode.
  Read the OMM0 bit in the RLN21nmLiMST register; verify that it is 0 (LIN reset mode).

13.16 Baud Rate Generator

The LIN system clock (fLIN) is the clock that is made by dividing the LIN communication clock source by the baud rate generator, and the bit rate is made by dividing that clock by 16. The inverse of this bit rate is the bit time (Tbit).

Figure 13.16 shows a block diagram of baud rate generation.



Note 1. For the LIN communication clock source, see **Section 10, Clock Controller**
Note 2. According to the RLN21nGLBRP0 register, the clock frequency is divided by N + 1 (N = 0 to 255).
Note 3. According to the RLIN21nGLBPR1 register, the clock frequency is divided by M + 1 (M = 0 to 255).

Figure 13.16 Block Diagram of Baud Rate Generation in LIN Master Mode

Set the LIN communication clock source to the range from 4 MHz to 40 MHz.

By setting the RLN21nGLBRP0 register so that fa is 307200 Hz (= 19200 × 16), the resulting bit rates are fa = 19200 × 16, fb = 9600 × 16 and fc = 2400 × 16. These bit rates are frequency-divided by 16 in the bit timing generator, enabling bit rates of 19200 bps, 9600 bps and 2400 bps to be generated. Also, by setting the RLN21nGLBRP1 register so that fd is 166672 Hz (= 10417 × 16), the resulting bit rate is fd = 10417 × 16. This bit rate is frequency-divided by 16 in the bit timing generator, enabling 10417 bps to be generated.

Table 13.37 shows examples of baud rate (19200, 10417, 9600, and 2400 bps) generation for each LIN communication clock source frequency, and also the corresponding errors.

Table 13.37 Examples of Baud Rate (19200, 10417, 9600, and 2400 bps) Generation

LIN Communication Clock Source	Baud Rate Generator 0 (N+1) Frequency-divided	Baud Rate Generator 1 (M+1) Frequency-divided	System Clock	Baud Rate	Error
40 MHz	130	—	fa	19230.77	+0.16%
	—	120	fd	10416.67	-0.003%
	130	—	fb	9615.38	+0.16%
	130	—	fc	2403.85	+0.16%

Section 14 CAN Interface (RS-CAN)

14.1 Features of RH850/C1x RS-CAN

14.1.1 Number of Units

This microcontroller has the following number of RS-CAN units.

Table 14.1 Units of RS-CAN

Product	RH850/C1x
Number of units	1
Name	RS-CANn (n = 0)

Table 14.2 Index

Index	Meaning
n	Throughout this section, the individual RS-CAN units are identified by the index “n” (n = 0): for example, RSCANnGCFG is the global control register of the RS-CAN0 unit.
m	The number of channels of RS-CAN is identified by the index “m” (m = 0 to 3): for example, RSCAN0CmSTS is the channel m status register.
j	The individual registers associated with receive rule table are identified by the index “j” (j = 0 to 15): for example, RSCAN0GAFLIDj is the receive rule ID register.
k	The individual transmit/receive FIFO buffers are identified by the index “k” (k = 0 to 11): for example, RSCAN0CFCCk is the transmit/receive FIFO buffer configuration/control register.
x	The individual receive FIFOs are identified by the index “x” (x = 0 to 7): for example, RSCAN0RFSTSx is the receive buffer status register.
q	The individual receive buffers are identified by the index “q” (q = 0 to 63): for example, RSCAN0RMIDq is the receive buffer ID register.
p	The individual transmit buffers are identified by the index “p” (p = 0 to 63): for example, RSCAN0TMCp is the transmit buffer control register.
r	The individual RAM tests for CAN are identified by the index “r” (r = 0 to 63): for example, RSCAN0RPGACCr is the RAM test page access register.
y	The registers not covered above are indicated by the index “y” (y = 0, 1): for example, RSCAN0RMNDy is a receive buffer new data register.

The RS-CAN has four channels.

Table 14.3 Channels of RS-CAN

Unit Name	Number of Channels	Channel Name
RS-CAN0	4	CANm (m = 0 to 3)

14.1.2 Register Base Address

RS-CAN base addresses are listed in the following table.

RS-CAN register addresses are given as offsets from the base addresses.

Table 14.4 Register Base Address

Base Address Name	Base Address
<RSCAN0_base>	FFD0 0000_H

14.1.3 Clock Supply

The RS-CAN clock supply is shown in the following table.

Table 14.5 Clock Supply

Unit Name	Clock for the Unit	Internal Clock Signal
RS-CAN0	pclk	CLK_HSB (high-speed peripheral clock)
	clkc	CLKC_LSB (unmodulated low-speed peripheral clock)
	clk_xincan	Main OSC

14.1.4 Interrupts

RS-CAN interrupt requests are listed in the following table.

Table 14.6 RS-CAN Interrupt Requests

Name	Function	Interrupt Number
Global interrupts	Receive FIFOm interrupt	186
	Global error interrupt	187
Channel CAN0 interrupts	CAN0 transmit/receive FIFO reception complete interrupt (in receive mode or gateway mode)	188
	CAN0 error interrupt	189
	CAN0 transmit interrupt	190
Channel CAN1 interrupts	CAN1 transmit/receive FIFO reception complete interrupt (in receive mode or gateway mode)	191
	CAN1 error interrupt	192
	CAN1 transmit interrupt	193
Channel CAN2 interrupts	CAN2 transmit/receive FIFO reception complete interrupt (in receive mode or gateway mode)	194
	CAN2 error interrupt	195
	CAN2 transmit interrupt	196
Channel CAN3 interrupts	CAN3 transmit/receive FIFO reception complete interrupt (in receive mode or gateway mode)	197
	CAN3 error interrupt	198
	CAN3 transmit interrupt	199

14.1.5 Reset Sources

RS-CAN reset sources are listed in the following table. RS-CAN is initialized by these reset sources.

Table 14.7 Reset Sources

Unit Name	Reset Source
RS-CAN0	All reset sources

14.1.6 External Input/Output Signals

External input/output signals of RS-CAN are listed below.

Table 14.8 External Input/Output Signals

Pin Name	Description	Alternative Port Pin Signal
Rxm (m = 0 to 3)	Receive data input pins of the CAN communication function	CAN0RX, CAN1RX, CAN2RX, and CAN3RX
Txm (m = 0 to 3)	Transmit data output pins of the CAN communication function	CAN0TX, CAN1TX, CAN2TX, and CAN3TX

14.2 Overview

14.2.1 Functional Overview

The RH850/C1x incorporates one unit of the CAN interface (RS-CAN) which consists of four channels (CAN0 to CAN3) of the CAN controller conforming to the ISO11898-1 specifications. **Table 14.9** shows the RCAN module specifications.

Figure 14.1 shows the RS-CAN module block diagram.

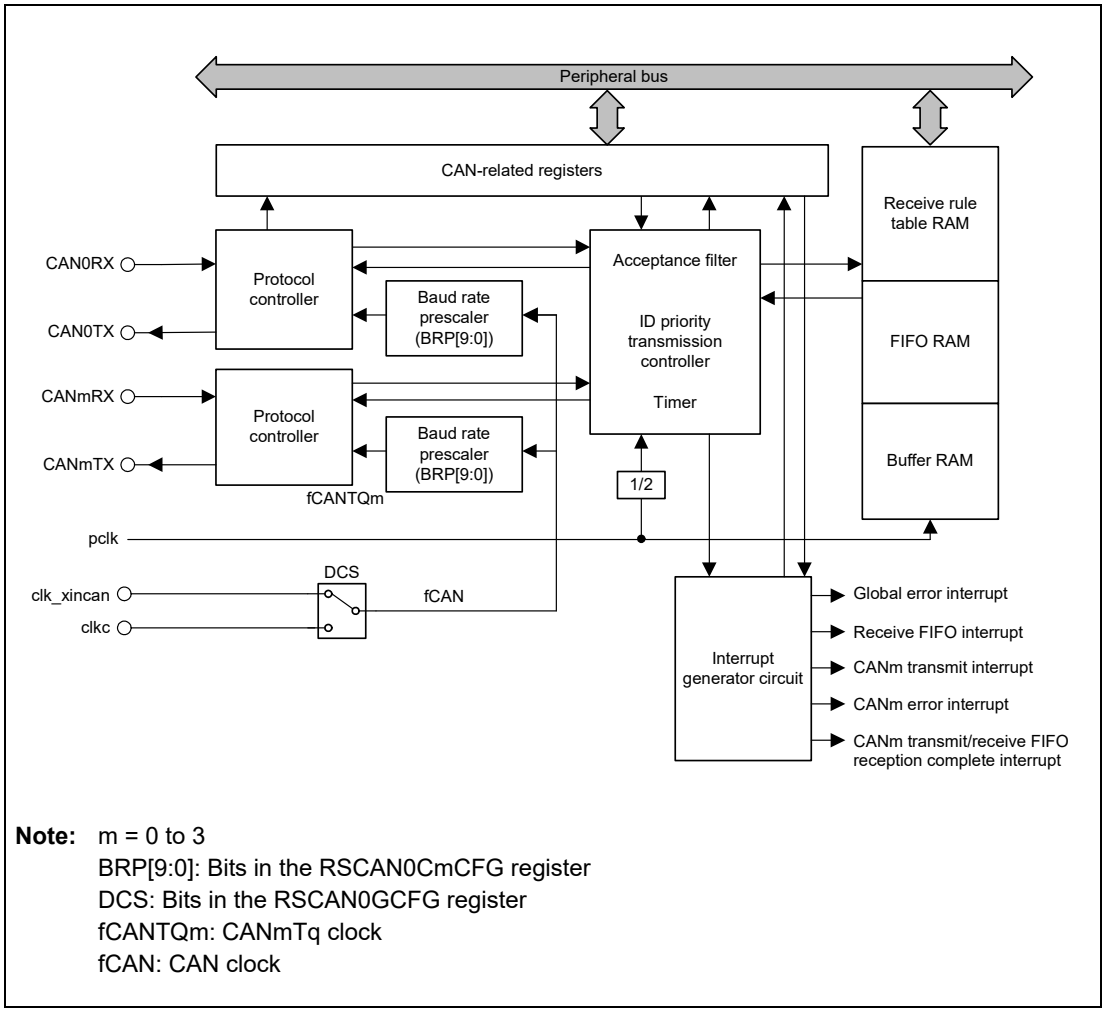
Table 14.9 RS-CAN Module Specifications (1/2)

Item	Specification
Number of channels	4
Protocol	ISO11898-1 compliant
Communication speed	• Up to 1 Mbps $\text{Communication speed (CANn bit time clock)} = \frac{1}{\text{CANm bit time}}$ CANm bit time = CANmTq × Tq count per bit $\text{CANmTq} = \frac{\text{(BRP[9:0] bits in the RCAN0mCFG register + 1)}}{\text{fCAN}}$ m = 0 to 3 Tq: Time quantum fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCAN0GCFG register)
Buffer	320 buffers in total • Individual buffers: 64 buffers (16 buffers × 4 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocatable) • Shared buffers: 256 buffers for all channels Receive buffer: 0 to 64 buffers Receive FIFO buffer: 8 FIFO buffers (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each)
Reception function	• Receives data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be received. • Sets interrupt enable/disable for each FIFO. • Mirror function (CAN mode receives its own transmitted message.) • Timestamp function (to record message reception time as a 16-bit timer value)
Reception filter function	• Selects receive messages according to 256 receive rules. • Sets the number of receive rules (0 to 128) for each channel. • Acceptance filter processing: Sets ID and mask for each receive rule. • DLC filter processing: Sets DLC check value for each acceptance rule.
Receive message transfer function	• Routing function to transfer receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with a message in a receive buffer and FIFO buffer.
Transmission function	• Transmits data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. • Selects ID priority transmission or transmit buffer number priority transmission. • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function

Table 14.9 RS-CAN Module Specifications (2/2)

Item	Specification
Interval transmission function	Transmit messages at intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmitted messages.
Gateway function	A received message is automatically routed to a different channel.
Bus off recovery mode selection	Selects the method for returning from bus off state. • ISO11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Transition to channel standby mode by program request • Transition to the error-active state by program request
Error status monitoring	• Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). • Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) • Reads the error counter. • Monitors DLC errors.
Interrupt source	14 sources • Global Interrupts (2 sources) Receive FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) CANm transmit interrupt (m = 0 to 3) – CANm transmission complete interrupt – CANm transmission abort interrupt – CANm transmit/receive FIFO transmission complete interrupt (in transmit mode, gateway mode) – CANm transmit history interrupt – CANm transmit queue interrupt CANm transmit/receive FIFO reception complete interrupt (in receive mode, gateway mode) CANm error interrupt CANm wakeup is generated using the external INT interrupt.
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CAN module.
CAN clock source	Clkc or clk_xincan selectable
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • RAM test (read/write test) • Inter-channel communication test

14.2.2 Block Diagram



Note: m = 0 to 3
BRP[9:0]: Bits in the RSCAN0CmCFG register
DCS: Bits in the RSCAN0GCFG register
fCANTQm: CANmTq clock
fCAN: CAN clock

Figure 14.1 RS-CAN Module Block Diagram

14.3 Registers

14.3.1 List of Registers

RS-CAN registers are listed in the following table.

See **Section 14.1.2, Register Base Address** for <RSCAN0_base>.

Table 14.10 Registers (1/23)

Module	Register	Symbol	Address
RS-CAN0	Channel 0 configuration register	RSCAN0C0CFG	<RSCAN0_base> + 0000_H
RS-CAN0	Channel 0 control register	RSCAN0C0CTR	<RSCAN0_base> + 0004_H
RS-CAN0	Channel 0 status register	RSCAN0C0STS	<RSCAN0_base> + 0008_H
RS-CAN0	Channel 0 error flag register	RSCAN0C0ERFL	<RSCAN0_base> + $000C_H$
RS-CAN0	Channel 1 configuration register	RSCAN0C1CFG	<RSCAN0_base> + 0010_H
RS-CAN0	Channel 1 control register	RSCAN0C1CTR	<RSCAN0_base> + 0014_H
RS-CAN0	Channel 1 status register	RSCAN0C1STS	<RSCAN0_base> + 0018_H
RS-CAN0	Channel 1 error flag register	RSCAN0C1ERFL	<RSCAN0_base> + $001C_H$
RS-CAN0	Channel 2 configuration register	RSCAN0C2CFG	<RSCAN0_base> + 0020_H
RS-CAN0	Channel 2 control register	RSCAN0C2CTR	<RSCAN0_base> + 0024_H
RS-CAN0	Channel 2 status register	RSCAN0C2STS	<RSCAN0_base> + 0028_H
RS-CAN0	Channel 2 error flag register	RSCAN0C2ERFL	<RSCAN0_base> + $002C_H$
RS-CAN0	Channel 3 configuration register	RSCAN0C3CFG	<RSCAN0_base> + 0030_H
RS-CAN0	Channel 3 control register	RSCAN0C3CTR	<RSCAN0_base> + 0034_H
RS-CAN0	Channel 3 status register	RSCAN0C3STS	<RSCAN0_base> + 0038_H
RS-CAN0	Channel 3 error flag register	RSCAN0C3ERFL	<RSCAN0_base> + $003C_H$
RS-CAN0	Global configuration register	RSCAN0GCFG	<RSCAN0_base> + 0084_H
RS-CAN0	Global control register	RSCAN0GCTR	<RSCAN0_base> + 0088_H
RS-CAN0	Global status register	RSCAN0GSTS	<RSCAN0_base> + $008C_H$
RS-CAN0	Global error flag register	RSCAN0GERFL	<RSCAN0_base> + 0090_H
RS-CAN0	Global timestamp counter register	RSCAN0GTSC	<RSCAN0_base> + 0094_H
RS-CAN0	Receive rule entry control register	RSCAN0GAFLECTR	<RSCAN0_base> + 0098_H
RS-CAN0	Receive rule configuration register 0	RSCAN0GAFLCFG0	<RSCAN0_base> + $009C_H$
RS-CAN0	Receive buffer number register	RSCAN0RMNB	<RSCAN0_base> + $00A4_H$
RS-CAN0	Receive buffer new data register 0	RSCAN0RMND0	<RSCAN0_base> + $00A8_H$
RS-CAN0	Receive buffer new data register 1	RSCAN0RMND1	<RSCAN0_base> + $00AC_H$
RS-CAN0	Receive FIFO buffer configuration and control register 0	RSCAN0RFCC0	<RSCAN0_base> + $00B8_H$
RS-CAN0	Receive FIFO buffer configuration and control register 1	RSCAN0RFCC1	<RSCAN0_base> + $00BC_H$
RS-CAN0	Receive FIFO buffer configuration and control register 2	RSCAN0RFCC2	<RSCAN0_base> + $00C0_H$
RS-CAN0	Receive FIFO buffer configuration and control register 3	RSCAN0RFCC3	<RSCAN0_base> + $00C4_H$
RS-CAN0	Receive FIFO buffer configuration and control register 4	RSCAN0RFCC4	<RSCAN0_base> + $00C8_H$
RS-CAN0	Receive FIFO buffer configuration and control register 5	RSCAN0RFCC5	<RSCAN0_base> + $00CC_H$
RS-CAN0	Receive FIFO buffer configuration and control register 6	RSCAN0RFCC6	<RSCAN0_base> + $00D0_H$
RS-CAN0	Receive FIFO buffer configuration and control register 7	RSCAN0RFCC7	<RSCAN0_base> + $00D4_H$
RS-CAN0	Receive FIFO buffer status register 0	RSCAN0RFSTS0	<RSCAN0_base> + $00D8_H$
RS-CAN0	Receive FIFO buffer status register 1	RSCAN0RFSTS1	<RSCAN0_base> + $00DC_H$
RS-CAN0	Receive FIFO buffer status register 2	RSCAN0RFSTS2	<RSCAN0_base> + $00E0_H$
RS-CAN0	Receive FIFO buffer status register 3	RSCAN0RFSTS3	<RSCAN0_base> + $00E4_H$

Table 14.10 Registers (2/23)

Module	Register	Symbol	Address
RS-CAN0	Receive FIFO buffer status register 4	RSCAN0RFSTS4	<RSCAN0_base> + $00E8_H$
RS-CAN0	Receive FIFO buffer status register 5	RSCAN0RFSTS5	<RSCAN0_base> + $00EC_H$
RS-CAN0	Receive FIFO buffer status register 6	RSCAN0RFSTS6	<RSCAN0_base> + $00F0_H$
RS-CAN0	Receive FIFO buffer status register 7	RSCAN0RFSTS7	<RSCAN0_base> + $00F4_H$
RS-CAN0	Receive FIFO buffer pointer control register 0	RSCAN0RFPCTR0	<RSCAN0_base> + $00F8_H$
RS-CAN0	Receive FIFO buffer pointer control register 1	RSCAN0RFPCTR1	<RSCAN0_base> + $00FC_H$
RS-CAN0	Receive FIFO buffer pointer control register 2	RSCAN0RFPCTR2	<RSCAN0_base> + 0100_H
RS-CAN0	Receive FIFO buffer pointer control register 3	RSCAN0RFPCTR3	<RSCAN0_base> + 0104_H
RS-CAN0	Receive FIFO buffer pointer control register 4	RSCAN0RFPCTR4	<RSCAN0_base> + 0108_H
RS-CAN0	Receive FIFO buffer pointer control register 5	RSCAN0RFPCTR5	<RSCAN0_base> + $010C_H$
RS-CAN0	Receive FIFO buffer pointer control register 6	RSCAN0RFPCTR6	<RSCAN0_base> + 0110_H
RS-CAN0	Receive FIFO buffer pointer control register 7	RSCAN0RFPCTR7	<RSCAN0_base> + 0114_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 0	RSCAN0CFCC0	<RSCAN0_base> + 0118_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 1	RSCAN0CFCC1	<RSCAN0_base> + $011C_H$
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 2	RSCAN0CFCC2	<RSCAN0_base> + 0120_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 3	RSCAN0CFCC3	<RSCAN0_base> + 0124_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 4	RSCAN0CFCC4	<RSCAN0_base> + 0128_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 5	RSCAN0CFCC5	<RSCAN0_base> + $012C_H$
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 6	RSCAN0CFCC6	<RSCAN0_base> + 0130_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 7	RSCAN0CFCC7	<RSCAN0_base> + 0134_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 8	RSCAN0CFCC8	<RSCAN0_base> + 0138_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 9	RSCAN0CFCC9	<RSCAN0_base> + $013C_H$
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 10	RSCAN0CFCC10	<RSCAN0_base> + 0140_H
RS-CAN0	Transmit/receive FIFO buffer configuration and control register 11	RSCAN0CFCC11	<RSCAN0_base> + 0144_H
RS-CAN0	Transmit/receive FIFO buffer status register 0	RSCAN0CFSTS0	<RSCAN0_base> + 0178_H
RS-CAN0	Transmit/receive FIFO buffer status register 1	RSCAN0CFSTS1	<RSCAN0_base> + $017C_H$
RS-CAN0	Transmit/receive FIFO buffer status register 2	RSCAN0CFSTS2	<RSCAN0_base> + 0180_H
RS-CAN0	Transmit/receive FIFO buffer status register 3	RSCAN0CFSTS3	<RSCAN0_base> + 0184_H
RS-CAN0	Transmit/receive FIFO buffer status register 4	RSCAN0CFSTS4	<RSCAN0_base> + 0188_H
RS-CAN0	Transmit/receive FIFO buffer status register 5	RSCAN0CFSTS5	<RSCAN0_base> + $018C_H$
RS-CAN0	Transmit/receive FIFO buffer status register 6	RSCAN0CFSTS6	<RSCAN0_base> + 0190_H
RS-CAN0	Transmit/receive FIFO buffer status register 7	RSCAN0CFSTS7	<RSCAN0_base> + 0194_H
RS-CAN0	Transmit/receive FIFO buffer status register 8	RSCAN0CFSTS8	<RSCAN0_base> + 0198_H
RS-CAN0	Transmit/receive FIFO buffer status register 9	RSCAN0CFSTS9	<RSCAN0_base> + $019C_H$
RS-CAN0	Transmit/receive FIFO buffer status register 10	RSCAN0CFSTS10	<RSCAN0_base> + $01A0_H$
RS-CAN0	Transmit/receive FIFO buffer status register 11	RSCAN0CFSTS11	<RSCAN0_base> + $01A4_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 0	RSCAN0CFPCTR0	<RSCAN0_base> + $01D8_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 1	RSCAN0CFPCTR1	<RSCAN0_base> + $01DC_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 2	RSCAN0CFPCTR2	<RSCAN0_base> + $01E0_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 3	RSCAN0CFPCTR3	<RSCAN0_base> + $01E4_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 4	RSCAN0CFPCTR4	<RSCAN0_base> + $01E8_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 5	RSCAN0CFPCTR5	<RSCAN0_base> + $01EC_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 6	RSCAN0CFPCTR6	<RSCAN0_base> + $01F0_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 7	RSCAN0CFPCTR7	<RSCAN0_base> + $01F4_H$

Table 14.10 Registers (3/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit/receive FIFO buffer pointer control register 8	RSCAN0CFPCTR8	<RSCAN0_base> + $01F8_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 9	RSCAN0CFPCTR9	<RSCAN0_base> + $01FC_H$
RS-CAN0	Transmit/receive FIFO buffer pointer control register 10	RSCAN0CFPCTR10	<RSCAN0_base> + 0200_H
RS-CAN0	Transmit/receive FIFO buffer pointer control register 11	RSCAN0CFPCTR11	<RSCAN0_base> + 0204_H
RS-CAN0	FIFO empty status register	RSCAN0FESTS	<RSCAN0_base> + 0238_H
RS-CAN0	FIFO full status register	RSCAN0FFSTS	<RSCAN0_base> + $023C_H$
RS-CAN0	FIFO message lost status register	RSCAN0FMSTS	<RSCAN0_base> + 0240_H
RS-CAN0	Receive FIFO buffer interrupt flag status register	RSCAN0RFISTS	<RSCAN0_base> + 0244_H
RS-CAN0	Transmit/receive FIFO buffer RX interrupt flag status register	RSCAN0CFRISTS	<RSCAN0_base> + 0248_H
RS-CAN0	Transmit/receive FIFO buffer TX interrupt flag status register	RSCAN0CFTISTS	<RSCAN0_base> + $024C_H$
RS-CAN0	Transmit buffer control register 0	RSCAN0TMC0	<RSCAN0_base> + 0250_H
RS-CAN0	Transmit buffer control register 1	RSCAN0TMC1	<RSCAN0_base> + 0251_H
RS-CAN0	Transmit buffer control register 2	RSCAN0TMC2	<RSCAN0_base> + 0252_H
RS-CAN0	Transmit buffer control register 3	RSCAN0TMC3	<RSCAN0_base> + 0253_H
RS-CAN0	Transmit buffer control register 4	RSCAN0TMC4	<RSCAN0_base> + 0254_H
RS-CAN0	Transmit buffer control register 5	RSCAN0TMC5	<RSCAN0_base> + 0255_H
RS-CAN0	Transmit buffer control register 6	RSCAN0TMC6	<RSCAN0_base> + 0256_H
RS-CAN0	Transmit buffer control register 7	RSCAN0TMC7	<RSCAN0_base> + 0257_H
RS-CAN0	Transmit buffer control register 8	RSCAN0TMC8	<RSCAN0_base> + 0258_H
RS-CAN0	Transmit buffer control register 9	RSCAN0TMC9	<RSCAN0_base> + 0259_H
RS-CAN0	Transmit buffer control register 10	RSCAN0TMC10	<RSCAN0_base> + $025A_H$
RS-CAN0	Transmit buffer control register 11	RSCAN0TMC11	<RSCAN0_base> + $025B_H$
RS-CAN0	Transmit buffer control register 12	RSCAN0TMC12	<RSCAN0_base> + $025C_H$
RS-CAN0	Transmit buffer control register 13	RSCAN0TMC13	<RSCAN0_base> + $025D_H$
RS-CAN0	Transmit buffer control register 14	RSCAN0TMC14	<RSCAN0_base> + $025E_H$
RS-CAN0	Transmit buffer control register 15	RSCAN0TMC15	<RSCAN0_base> + $025F_H$
RS-CAN0	Transmit buffer control register 16	RSCAN0TMC16	<RSCAN0_base> + 0260_H
RS-CAN0	Transmit buffer control register 17	RSCAN0TMC17	<RSCAN0_base> + 0261_H
RS-CAN0	Transmit buffer control register 18	RSCAN0TMC18	<RSCAN0_base> + 0262_H
RS-CAN0	Transmit buffer control register 19	RSCAN0TMC19	<RSCAN0_base> + 0263_H
RS-CAN0	Transmit buffer control register 20	RSCAN0TMC20	<RSCAN0_base> + 0264_H
RS-CAN0	Transmit buffer control register 21	RSCAN0TMC21	<RSCAN0_base> + 0265_H
RS-CAN0	Transmit buffer control register 22	RSCAN0TMC22	<RSCAN0_base> + 0266_H
RS-CAN0	Transmit buffer control register 23	RSCAN0TMC23	<RSCAN0_base> + 0267_H
RS-CAN0	Transmit buffer control register 24	RSCAN0TMC24	<RSCAN0_base> + 0268_H
RS-CAN0	Transmit buffer control register 25	RSCAN0TMC25	<RSCAN0_base> + 0269_H
RS-CAN0	Transmit buffer control register 26	RSCAN0TMC26	<RSCAN0_base> + $026A_H$
RS-CAN0	Transmit buffer control register 27	RSCAN0TMC27	<RSCAN0_base> + $026B_H$
RS-CAN0	Transmit buffer control register 28	RSCAN0TMC28	<RSCAN0_base> + $026C_H$
RS-CAN0	Transmit buffer control register 29	RSCAN0TMC29	<RSCAN0_base> + $026D_H$
RS-CAN0	Transmit buffer control register 30	RSCAN0TMC30	<RSCAN0_base> + $026E_H$
RS-CAN0	Transmit buffer control register 31	RSCAN0TMC31	<RSCAN0_base> + $026F_H$
RS-CAN0	Transmit buffer control register 32	RSCAN0TMC32	<RSCAN0_base> + 0270_H
RS-CAN0	Transmit buffer control register 33	RSCAN0TMC33	<RSCAN0_base> + 0271_H

Table 14.10 Registers (4/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer control register 34	RSCAN0TMC34	<RSCAN0_base> + 0272_H
RS-CAN0	Transmit buffer control register 35	RSCAN0TMC35	<RSCAN0_base> + 0273_H
RS-CAN0	Transmit buffer control register 36	RSCAN0TMC36	<RSCAN0_base> + 0274_H
RS-CAN0	Transmit buffer control register 37	RSCAN0TMC37	<RSCAN0_base> + 0275_H
RS-CAN0	Transmit buffer control register 38	RSCAN0TMC38	<RSCAN0_base> + 0276_H
RS-CAN0	Transmit buffer control register 39	RSCAN0TMC39	<RSCAN0_base> + 0277_H
RS-CAN0	Transmit buffer control register 40	RSCAN0TMC40	<RSCAN0_base> + 0278_H
RS-CAN0	Transmit buffer control register 41	RSCAN0TMC41	<RSCAN0_base> + 0279_H
RS-CAN0	Transmit buffer control register 42	RSCAN0TMC42	<RSCAN0_base> + $027A_H$
RS-CAN0	Transmit buffer control register 43	RSCAN0TMC43	<RSCAN0_base> + $027B_H$
RS-CAN0	Transmit buffer control register 44	RSCAN0TMC44	<RSCAN0_base> + $027C_H$
RS-CAN0	Transmit buffer control register 45	RSCAN0TMC45	<RSCAN0_base> + $027D_H$
RS-CAN0	Transmit buffer control register 46	RSCAN0TMC46	<RSCAN0_base> + $027E_H$
RS-CAN0	Transmit buffer control register 47	RSCAN0TMC47	<RSCAN0_base> + $027F_H$
RS-CAN0	Transmit buffer control register 48	RSCAN0TMC48	<RSCAN0_base> + 0280_H
RS-CAN0	Transmit buffer control register 49	RSCAN0TMC49	<RSCAN0_base> + 0281_H
RS-CAN0	Transmit buffer control register 50	RSCAN0TMC50	<RSCAN0_base> + 0282_H
RS-CAN0	Transmit buffer control register 51	RSCAN0TMC51	<RSCAN0_base> + 0283_H
RS-CAN0	Transmit buffer control register 52	RSCAN0TMC52	<RSCAN0_base> + 0284_H
RS-CAN0	Transmit buffer control register 53	RSCAN0TMC53	<RSCAN0_base> + 0285_H
RS-CAN0	Transmit buffer control register 54	RSCAN0TMC54	<RSCAN0_base> + 0286_H
RS-CAN0	Transmit buffer control register 55	RSCAN0TMC55	<RSCAN0_base> + 0287_H
RS-CAN0	Transmit buffer control register 56	RSCAN0TMC56	<RSCAN0_base> + 0288_H
RS-CAN0	Transmit buffer control register 57	RSCAN0TMC57	<RSCAN0_base> + 0289_H
RS-CAN0	Transmit buffer control register 58	RSCAN0TMC58	<RSCAN0_base> + $028A_H$
RS-CAN0	Transmit buffer control register 59	RSCAN0TMC59	<RSCAN0_base> + $028B_H$
RS-CAN0	Transmit buffer control register 60	RSCAN0TMC60	<RSCAN0_base> + $028C_H$
RS-CAN0	Transmit buffer control register 61	RSCAN0TMC61	<RSCAN0_base> + $028D_H$
RS-CAN0	Transmit buffer control register 62	RSCAN0TMC62	<RSCAN0_base> + $028E_H$
RS-CAN0	Transmit buffer control register 63	RSCAN0TMC63	<RSCAN0_base> + $028F_H$
RS-CAN0	Transmit buffer status register 0	RSCAN0TMSTS0	<RSCAN0_base> + $02D0_H$
RS-CAN0	Transmit buffer status register 1	RSCAN0TMSTS1	<RSCAN0_base> + $02D1_H$
RS-CAN0	Transmit buffer status register 2	RSCAN0TMSTS2	<RSCAN0_base> + $02D2_H$
RS-CAN0	Transmit buffer status register 3	RSCAN0TMSTS3	<RSCAN0_base> + $02D3_H$
RS-CAN0	Transmit buffer status register 4	RSCAN0TMSTS4	<RSCAN0_base> + $02D4_H$
RS-CAN0	Transmit buffer status register 5	RSCAN0TMSTS5	<RSCAN0_base> + $02D5_H$
RS-CAN0	Transmit buffer status register 6	RSCAN0TMSTS6	<RSCAN0_base> + $02D6_H$
RS-CAN0	Transmit buffer status register 7	RSCAN0TMSTS7	<RSCAN0_base> + $02D7_H$
RS-CAN0	Transmit buffer status register 8	RSCAN0TMSTS8	<RSCAN0_base> + $02D8_H$
RS-CAN0	Transmit buffer status register 9	RSCAN0TMSTS9	<RSCAN0_base> + $02D9_H$
RS-CAN0	Transmit buffer status register 10	RSCAN0TMSTS10	<RSCAN0_base> + $02DA_H$
RS-CAN0	Transmit buffer status register 11	RSCAN0TMSTS11	<RSCAN0_base> + $02DB_H$
RS-CAN0	Transmit buffer status register 12	RSCAN0TMSTS12	<RSCAN0_base> + $02DC_H$
RS-CAN0	Transmit buffer status register 13	RSCAN0TMSTS13	<RSCAN0_base> + $02DD_H$

Table 14.10 Registers (5/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer status register 14	RSCAN0TMSTS14	<RSCAN0_base> + $02DE_H$
RS-CAN0	Transmit buffer status register 15	RSCAN0TMSTS15	<RSCAN0_base> + $02DF_H$
RS-CAN0	Transmit buffer status register 16	RSCAN0TMSTS16	<RSCAN0_base> + $02E0_H$
RS-CAN0	Transmit buffer status register 17	RSCAN0TMSTS17	<RSCAN0_base> + $02E1_H$
RS-CAN0	Transmit buffer status register 18	RSCAN0TMSTS18	<RSCAN0_base> + $02E2_H$
RS-CAN0	Transmit buffer status register 19	RSCAN0TMSTS19	<RSCAN0_base> + $02E3_H$
RS-CAN0	Transmit buffer status register 20	RSCAN0TMSTS20	<RSCAN0_base> + $02E4_H$
RS-CAN0	Transmit buffer status register 21	RSCAN0TMSTS21	<RSCAN0_base> + $02E5_H$
RS-CAN0	Transmit buffer status register 22	RSCAN0TMSTS22	<RSCAN0_base> + $02E6_H$
RS-CAN0	Transmit buffer status register 23	RSCAN0TMSTS23	<RSCAN0_base> + $02E7_H$
RS-CAN0	Transmit buffer status register 24	RSCAN0TMSTS24	<RSCAN0_base> + $02E8_H$
RS-CAN0	Transmit buffer status register 25	RSCAN0TMSTS25	<RSCAN0_base> + $02E9_H$
RS-CAN0	Transmit buffer status register 26	RSCAN0TMSTS26	<RSCAN0_base> + $02EA_H$
RS-CAN0	Transmit buffer status register 27	RSCAN0TMSTS27	<RSCAN0_base> + $02EB_H$
RS-CAN0	Transmit buffer status register 28	RSCAN0TMSTS28	<RSCAN0_base> + $02EC_H$
RS-CAN0	Transmit buffer status register 29	RSCAN0TMSTS29	<RSCAN0_base> + $02ED_H$
RS-CAN0	Transmit buffer status register 30	RSCAN0TMSTS30	<RSCAN0_base> + $02EE_H$
RS-CAN0	Transmit buffer status register 31	RSCAN0TMSTS31	<RSCAN0_base> + $02EF_H$
RS-CAN0	Transmit buffer status register 32	RSCAN0TMSTS32	<RSCAN0_base> + $02F0_H$
RS-CAN0	Transmit buffer status register 33	RSCAN0TMSTS33	<RSCAN0_base> + $02F1_H$
RS-CAN0	Transmit buffer status register 34	RSCAN0TMSTS34	<RSCAN0_base> + $02F2_H$
RS-CAN0	Transmit buffer status register 35	RSCAN0TMSTS35	<RSCAN0_base> + $02F3_H$
RS-CAN0	Transmit buffer status register 36	RSCAN0TMSTS36	<RSCAN0_base> + $02F4_H$
RS-CAN0	Transmit buffer status register 37	RSCAN0TMSTS37	<RSCAN0_base> + $02F5_H$
RS-CAN0	Transmit buffer status register 38	RSCAN0TMSTS38	<RSCAN0_base> + $02F6_H$
RS-CAN0	Transmit buffer status register 39	RSCAN0TMSTS39	<RSCAN0_base> + $02F7_H$
RS-CAN0	Transmit buffer status register 40	RSCAN0TMSTS40	<RSCAN0_base> + $02F8_H$
RS-CAN0	Transmit buffer status register 41	RSCAN0TMSTS41	<RSCAN0_base> + $02F9_H$
RS-CAN0	Transmit buffer status register 42	RSCAN0TMSTS42	<RSCAN0_base> + $02FA_H$
RS-CAN0	Transmit buffer status register 43	RSCAN0TMSTS43	<RSCAN0_base> + $02FB_H$
RS-CAN0	Transmit buffer status register 44	RSCAN0TMSTS44	<RSCAN0_base> + $02FC_H$
RS-CAN0	Transmit buffer status register 45	RSCAN0TMSTS45	<RSCAN0_base> + $02FD_H$
RS-CAN0	Transmit buffer status register 46	RSCAN0TMSTS46	<RSCAN0_base> + $02FE_H$
RS-CAN0	Transmit buffer status register 47	RSCAN0TMSTS47	<RSCAN0_base> + $02FF_H$
RS-CAN0	Transmit buffer status register 48	RSCAN0TMSTS48	<RSCAN0_base> + 0300_H
RS-CAN0	Transmit buffer status register 49	RSCAN0TMSTS49	<RSCAN0_base> + 0301_H
RS-CAN0	Transmit buffer status register 50	RSCAN0TMSTS50	<RSCAN0_base> + 0302_H
RS-CAN0	Transmit buffer status register 51	RSCAN0TMSTS51	<RSCAN0_base> + 0303_H
RS-CAN0	Transmit buffer status register 52	RSCAN0TMSTS52	<RSCAN0_base> + 0304_H
RS-CAN0	Transmit buffer status register 53	RSCAN0TMSTS53	<RSCAN0_base> + 0305_H
RS-CAN0	Transmit buffer status register 54	RSCAN0TMSTS54	<RSCAN0_base> + 0306_H
RS-CAN0	Transmit buffer status register 55	RSCAN0TMSTS55	<RSCAN0_base> + 0307_H
RS-CAN0	Transmit buffer status register 56	RSCAN0TMSTS56	<RSCAN0_base> + 0308_H
RS-CAN0	Transmit buffer status register 57	RSCAN0TMSTS57	<RSCAN0_base> + 0309_H

Table 14.10 Registers (6/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer status register 58	RSCAN0TMSTS58	<RSCAN0_base> + $030A_H$
RS-CAN0	Transmit buffer status register 59	RSCAN0TMSTS59	<RSCAN0_base> + $030B_H$
RS-CAN0	Transmit buffer status register 60	RSCAN0TMSTS60	<RSCAN0_base> + $030C_H$
RS-CAN0	Transmit buffer status register 61	RSCAN0TMSTS61	<RSCAN0_base> + $030D_H$
RS-CAN0	Transmit buffer status register 62	RSCAN0TMSTS62	<RSCAN0_base> + $030E_H$
RS-CAN0	Transmit buffer status register 63	RSCAN0TMSTS63	<RSCAN0_base> + $030F_H$
RS-CAN0	Transmit buffer transmission request status register 0	RSCAN0TMTRSTS0	<RSCAN0_base> + 0350_H
RS-CAN0	Transmit buffer transmission request status register 1	RSCAN0TMTRSTS1	<RSCAN0_base> + 0354_H
RS-CAN0	Transmit buffer transmission abort request status register 0	RSCAN0TMTARSTS0	<RSCAN0_base> + 0360_H
RS-CAN0	Transmit buffer transmission abort request status register 1	RSCAN0TMTARSTS1	<RSCAN0_base> + 0364_H
RS-CAN0	Transmit buffer transmission complete status register 0	RSCAN0TMTCSTS0	<RSCAN0_base> + 0370_H
RS-CAN0	Transmit buffer transmission complete status register 1	RSCAN0TMTCSTS1	<RSCAN0_base> + 0374_H
RS-CAN0	Transmit buffer transmission abort status register 0	RSCAN0TMTASTS0	<RSCAN0_base> + 0380_H
RS-CAN0	Transmit buffer transmission abort status register 1	RSCAN0TMTASTS1	<RSCAN0_base> + 0384_H
RS-CAN0	Transmit buffer interrupt enable configuration register 0	RSCAN0TMIEC0	<RSCAN0_base> + 0390_H
RS-CAN0	Transmit buffer interrupt enable configuration register 1	RSCAN0TMIEC1	<RSCAN0_base> + 0394_H
RS-CAN0	Transmit queue configuration and control register 0	RSCAN0TXQCC0	<RSCAN0_base> + $03A0_H$
RS-CAN0	Transmit queue configuration and control register 1	RSCAN0TXQCC1	<RSCAN0_base> + $03A4_H$
RS-CAN0	Transmit queue configuration and control register 2	RSCAN0TXQCC2	<RSCAN0_base> + $03A8_H$
RS-CAN0	Transmit queue configuration and control register 3	RSCAN0TXQCC3	<RSCAN0_base> + $03AC_H$
RS-CAN0	Transmit queue status register 0	RSCAN0TXQSTS0	<RSCAN0_base> + $03C0_H$
RS-CAN0	Transmit queue status register 1	RSCAN0TXQSTS1	<RSCAN0_base> + $03C4_H$
RS-CAN0	Transmit queue status register 2	RSCAN0TXQSTS2	<RSCAN0_base> + $03C8_H$
RS-CAN0	Transmit queue status register 3	RSCAN0TXQSTS3	<RSCAN0_base> + $03CC_H$
RS-CAN0	Transmit queue pointer control register 0	RSCAN0TXQPCTR0	<RSCAN0_base> + $03E0_H$
RS-CAN0	Transmit queue pointer control register 1	RSCAN0TXQPCTR1	<RSCAN0_base> + $03E4_H$
RS-CAN0	Transmit queue pointer control register 2	RSCAN0TXQPCTR2	<RSCAN0_base> + $03E8_H$
RS-CAN0	Transmit queue pointer control register 3	RSCAN0TXQPCTR3	<RSCAN0_base> + $03EC_H$
RS-CAN0	Transmit history configuration and control register 0	RSCAN0THLCC0	<RSCAN0_base> + 0400_H
RS-CAN0	Transmit history configuration and control register 1	RSCAN0THLCC1	<RSCAN0_base> + 0404_H
RS-CAN0	Transmit history configuration and control register 2	RSCAN0THLCC2	<RSCAN0_base> + 0408_H
RS-CAN0	Transmit history configuration and control register 3	RSCAN0THLCC3	<RSCAN0_base> + $040C_H$
RS-CAN0	Transmit history status register 0	RSCAN0THLSTS0	<RSCAN0_base> + 0420_H
RS-CAN0	Transmit history status register 1	RSCAN0THLSTS1	<RSCAN0_base> + 0424_H
RS-CAN0	Transmit history status register 2	RSCAN0THLSTS2	<RSCAN0_base> + 0428_H
RS-CAN0	Transmit history status register 3	RSCAN0THLSTS3	<RSCAN0_base> + $042C_H$
RS-CAN0	Transmit history pointer control register 0	RSCAN0THLPCTR0	<RSCAN0_base> + 0440_H
RS-CAN0	Transmit history pointer control register 1	RSCAN0THLPCTR1	<RSCAN0_base> + 0444_H
RS-CAN0	Transmit history pointer control register 2	RSCAN0THLPCTR2	<RSCAN0_base> + 0448_H
RS-CAN0	Transmit history pointer control register 3	RSCAN0THLPCTR3	<RSCAN0_base> + $044C_H$
RS-CAN0	Global TX interrupt status register 0	RSCAN0GTINTSTS0	<RSCAN0_base> + 0460_H
RS-CAN0	Global test configuration register	RSCAN0GTSTCFG	<RSCAN0_base> + 0468_H
RS-CAN0	Global test control register	RSCAN0GTSTCTR	<RSCAN0_base> + $046C_H$
RS-CAN0	Global lock key register	RSCAN0GLOCKK	<RSCAN0_base> + $047C_H$

Table 14.10 Registers (7/23)

Module	Register	Symbol	Address
RS-CAN0	Receive rule ID register 0	RSCAN0GAFLID0	<RSCAN0_base> + 0500_H
RS-CAN0	Receive rule mask register 0	RSCAN0GAFLM0	<RSCAN0_base> + 0504_H
RS-CAN0	Receive rule pointer 0 register 0	RSCAN0GAFLP00	<RSCAN0_base> + 0508_H
RS-CAN0	Receive rule pointer 1 register 0	RSCAN0GAFLP10	<RSCAN0_base> + $050C_H$
RS-CAN0	Receive rule ID register 1	RSCAN0GAFLID1	<RSCAN0_base> + 0510_H
RS-CAN0	Receive rule mask register 1	RSCAN0GAFLM1	<RSCAN0_base> + 0514_H
RS-CAN0	Receive rule pointer 0 register 1	RSCAN0GAFLP01	<RSCAN0_base> + 0518_H
RS-CAN0	Receive rule pointer 1 register 1	RSCAN0GAFLP11	<RSCAN0_base> + $051C_H$
RS-CAN0	Receive rule ID register 2	RSCAN0GAFLID2	<RSCAN0_base> + 0520_H
RS-CAN0	Receive rule mask register 2	RSCAN0GAFLM2	<RSCAN0_base> + 0524_H
RS-CAN0	Receive rule pointer 0 register 2	RSCAN0GAFLP02	<RSCAN0_base> + 0528_H
RS-CAN0	Receive rule pointer 1 register 2	RSCAN0GAFLP12	<RSCAN0_base> + $052C_H$
RS-CAN0	Receive rule ID register 3	RSCAN0GAFLID3	<RSCAN0_base> + 0530_H
RS-CAN0	Receive rule mask register 3	RSCAN0GAFLM3	<RSCAN0_base> + 0534_H
RS-CAN0	Receive rule pointer 0 register 3	RSCAN0GAFLP03	<RSCAN0_base> + 0538_H
RS-CAN0	Receive rule pointer 1 register 3	RSCAN0GAFLP13	<RSCAN0_base> + $053C_H$
RS-CAN0	Receive rule ID register 4	RSCAN0GAFLID4	<RSCAN0_base> + 0540_H
RS-CAN0	Receive rule mask register 4	RSCAN0GAFLM4	<RSCAN0_base> + 0544_H
RS-CAN0	Receive rule pointer 0 register 4	RSCAN0GAFLP04	<RSCAN0_base> + 0548_H
RS-CAN0	Receive rule pointer 1 register 4	RSCAN0GAFLP14	<RSCAN0_base> + $054C_H$
RS-CAN0	Receive rule ID register 5	RSCAN0GAFLID5	<RSCAN0_base> + 0550_H
RS-CAN0	Receive rule mask register 5	RSCAN0GAFLM5	<RSCAN0_base> + 0554_H
RS-CAN0	Receive rule pointer 0 register 5	RSCAN0GAFLP05	<RSCAN0_base> + 0558_H
RS-CAN0	Receive rule pointer 1 register 5	RSCAN0GAFLP15	<RSCAN0_base> + $055C_H$
RS-CAN0	Receive rule ID register 6	RSCAN0GAFLID6	<RSCAN0_base> + 0560_H
RS-CAN0	Receive rule mask register 6	RSCAN0GAFLM6	<RSCAN0_base> + 0564_H
RS-CAN0	Receive rule pointer 0 register 6	RSCAN0GAFLP06	<RSCAN0_base> + 0568_H
RS-CAN0	Receive rule pointer 1 register 6	RSCAN0GAFLP16	<RSCAN0_base> + $056C_H$
RS-CAN0	Receive rule ID register 7	RSCAN0GAFLID7	<RSCAN0_base> + 0570_H
RS-CAN0	Receive rule mask register 7	RSCAN0GAFLM7	<RSCAN0_base> + 0574_H
RS-CAN0	Receive rule pointer 0 register 7	RSCAN0GAFLP07	<RSCAN0_base> + 0578_H
RS-CAN0	Receive rule pointer 1 register 7	RSCAN0GAFLP17	<RSCAN0_base> + $057C_H$
RS-CAN0	Receive rule ID register 8	RSCAN0GAFLID8	<RSCAN0_base> + 0580_H
RS-CAN0	Receive rule mask register 8	RSCAN0GAFLM8	<RSCAN0_base> + 0584_H
RS-CAN0	Receive rule pointer 0 register 8	RSCAN0GAFLP08	<RSCAN0_base> + 0588_H
RS-CAN0	Receive rule pointer 1 register 8	RSCAN0GAFLP18	<RSCAN0_base> + $058C_H$
RS-CAN0	Receive rule ID register 9	RSCAN0GAFLID9	<RSCAN0_base> + 0590_H
RS-CAN0	Receive rule mask register 9	RSCAN0GAFLM9	<RSCAN0_base> + 0594_H
RS-CAN0	Receive rule pointer 0 register 9	RSCAN0GAFLP09	<RSCAN0_base> + 0598_H
RS-CAN0	Receive rule pointer 1 register 9	RSCAN0GAFLP19	<RSCAN0_base> + $059C_H$
RS-CAN0	Receive rule ID register 10	RSCAN0GAFLID10	<RSCAN0_base> + $05A0_H$
RS-CAN0	Receive rule mask register 10	RSCAN0GAFLM10	<RSCAN0_base> + $05A4_H$
RS-CAN0	Receive rule pointer 0 register 10	RSCAN0GAFLP010	<RSCAN0_base> + $05A8_H$
RS-CAN0	Receive rule pointer 1 register 10	RSCAN0GAFLP110	<RSCAN0_base> + $05AC_H$

Table 14.10 Registers (8/23)

Module	Register	Symbol	Address
RS-CAN0	Receive rule ID register 11	RSCAN0GAFLID11	<RSCAN0_base> + $05B0_H$
RS-CAN0	Receive rule mask register 11	RSCAN0GAFLM11	<RSCAN0_base> + $05B4_H$
RS-CAN0	Receive rule pointer 0 register 11	RSCAN0GAFLP011	<RSCAN0_base> + $05B8_H$
RS-CAN0	Receive rule pointer 1 register 11	RSCAN0GAFLP111	<RSCAN0_base> + $05BC_H$
RS-CAN0	Receive rule ID register 12	RSCAN0GAFLID12	<RSCAN0_base> + $05C0_H$
RS-CAN0	Receive rule mask register 12	RSCAN0GAFLM12	<RSCAN0_base> + $05C4_H$
RS-CAN0	Receive rule pointer 0 register 12	RSCAN0GAFLP012	<RSCAN0_base> + $05C8_H$
RS-CAN0	Receive rule pointer 1 register 12	RSCAN0GAFLP112	<RSCAN0_base> + $05CC_H$
RS-CAN0	Receive rule ID register 13	RSCAN0GAFLID13	<RSCAN0_base> + $05D0_H$
RS-CAN0	Receive rule mask register 13	RSCAN0GAFLM13	<RSCAN0_base> + $05D4_H$
RS-CAN0	Receive rule pointer 0 register 13	RSCAN0GAFLP013	<RSCAN0_base> + $05D8_H$
RS-CAN0	Receive rule pointer 1 register 13	RSCAN0GAFLP113	<RSCAN0_base> + $05DC_H$
RS-CAN0	Receive rule ID register 14	RSCAN0GAFLID14	<RSCAN0_base> + $05E0_H$
RS-CAN0	Receive rule mask register 14	RSCAN0GAFLM14	<RSCAN0_base> + $05E4_H$
RS-CAN0	Receive rule pointer 0 register 14	RSCAN0GAFLP014	<RSCAN0_base> + $05E8_H$
RS-CAN0	Receive rule pointer 1 register 14	RSCAN0GAFLP114	<RSCAN0_base> + $05EC_H$
RS-CAN0	Receive rule ID register 15	RSCAN0GAFLID15	<RSCAN0_base> + $05F0_H$
RS-CAN0	Receive rule mask register 15	RSCAN0GAFLM15	<RSCAN0_base> + $05F4_H$
RS-CAN0	Receive rule pointer 0 register 15	RSCAN0GAFLP015	<RSCAN0_base> + $05F8_H$
RS-CAN0	Receive rule pointer 1 register 15	RSCAN0GAFLP115	<RSCAN0_base> + $05FC_H$
RS-CAN0	Receive buffer ID register 0	RSCAN0RMID0	<RSCAN0_base> + 0600_H
RS-CAN0	Receive buffer pointer register 0	RSCAN0RMPTR0	<RSCAN0_base> + 0604_H
RS-CAN0	Receive buffer data field 0 register 0	RSCAN0RMDF00	<RSCAN0_base> + 0608_H
RS-CAN0	Receive buffer data field 1 register 0	RSCAN0RMDF10	<RSCAN0_base> + $060C_H$
RS-CAN0	Receive buffer ID register 1	RSCAN0RMID1	<RSCAN0_base> + 0610_H
RS-CAN0	Receive buffer pointer register 1	RSCAN0RMPTR1	<RSCAN0_base> + 0614_H
RS-CAN0	Receive buffer data field 0 register 1	RSCAN0RMDF01	<RSCAN0_base> + 0618_H
RS-CAN0	Receive buffer data field 1 register 1	RSCAN0RMDF11	<RSCAN0_base> + $061C_H$
RS-CAN0	Receive buffer ID register 2	RSCAN0RMID2	<RSCAN0_base> + 0620_H
RS-CAN0	Receive buffer pointer register 2	RSCAN0RMPTR2	<RSCAN0_base> + 0624_H
RS-CAN0	Receive buffer data field 0 register 2	RSCAN0RMDF02	<RSCAN0_base> + 0628_H
RS-CAN0	Receive buffer data field 1 register 2	RSCAN0RMDF12	<RSCAN0_base> + $062C_H$
RS-CAN0	Receive buffer ID register 3	RSCAN0RMID3	<RSCAN0_base> + 0630_H
RS-CAN0	Receive buffer pointer register 3	RSCAN0RMPTR3	<RSCAN0_base> + 0634_H
RS-CAN0	Receive buffer data field 0 register 3	RSCAN0RMDF03	<RSCAN0_base> + 0638_H
RS-CAN0	Receive buffer data field 1 register 3	RSCAN0RMDF13	<RSCAN0_base> + $063C_H$
RS-CAN0	Receive buffer ID register 4	RSCAN0RMID4	<RSCAN0_base> + 0640_H
RS-CAN0	Receive buffer pointer register 4	RSCAN0RMPTR4	<RSCAN0_base> + 0644_H
RS-CAN0	Receive buffer data field 0 register 4	RSCAN0RMDF04	<RSCAN0_base> + 0648_H
RS-CAN0	Receive buffer data field 1 register 4	RSCAN0RMDF14	<RSCAN0_base> + $064C_H$
RS-CAN0	Receive buffer ID register 5	RSCAN0RMID5	<RSCAN0_base> + 0650_H
RS-CAN0	Receive buffer pointer register 5	RSCAN0RMPTR5	<RSCAN0_base> + 0654_H
RS-CAN0	Receive buffer data field 0 register 5	RSCAN0RMDF05	<RSCAN0_base> + 0658_H
RS-CAN0	Receive buffer data field 1 register 5	RSCAN0RMDF15	<RSCAN0_base> + $065C_H$

Table 14.10 Registers (9/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 6	RSCAN0RMID6	<RSCAN0_base> + 0660_H
RS-CAN0	Receive buffer pointer register 6	RSCAN0RMPTR6	<RSCAN0_base> + 0664_H
RS-CAN0	Receive buffer data field 0 register 6	RSCAN0RMDF06	<RSCAN0_base> + 0668_H
RS-CAN0	Receive buffer data field 1 register 6	RSCAN0RMDF16	<RSCAN0_base> + $066C_H$
RS-CAN0	Receive buffer ID register 7	RSCAN0RMID7	<RSCAN0_base> + 0670_H
RS-CAN0	Receive buffer pointer register 7	RSCAN0RMPTR7	<RSCAN0_base> + 0674_H
RS-CAN0	Receive buffer data field 0 register 7	RSCAN0RMDF07	<RSCAN0_base> + 0678_H
RS-CAN0	Receive buffer data field 1 register 7	RSCAN0RMDF17	<RSCAN0_base> + $067C_H$
RS-CAN0	Receive buffer ID register 8	RSCAN0RMID8	<RSCAN0_base> + 0680_H
RS-CAN0	Receive buffer pointer register 8	RSCAN0RMPTR8	<RSCAN0_base> + 0684_H
RS-CAN0	Receive buffer data field 0 register 8	RSCAN0RMDF08	<RSCAN0_base> + 0688_H
RS-CAN0	Receive buffer data field 1 register 8	RSCAN0RMDF18	<RSCAN0_base> + $068C_H$
RS-CAN0	Receive buffer ID register 9	RSCAN0RMID9	<RSCAN0_base> + 0690_H
RS-CAN0	Receive buffer pointer register 9	RSCAN0RMPTR9	<RSCAN0_base> + 0694_H
RS-CAN0	Receive buffer data field 0 register 9	RSCAN0RMDF09	<RSCAN0_base> + 0698_H
RS-CAN0	Receive buffer data field 1 register 9	RSCAN0RMDF19	<RSCAN0_base> + $069C_H$
RS-CAN0	Receive buffer ID register 10	RSCAN0RMID10	<RSCAN0_base> + $06A0_H$
RS-CAN0	Receive buffer pointer register 10	RSCAN0RMPTR10	<RSCAN0_base> + $06A4_H$
RS-CAN0	Receive buffer data field 0 register 10	RSCAN0RMDF010	<RSCAN0_base> + $06A8_H$
RS-CAN0	Receive buffer data field 1 register 10	RSCAN0RMDF110	<RSCAN0_base> + $06AC_H$
RS-CAN0	Receive buffer ID register 11	RSCAN0RMID11	<RSCAN0_base> + $06B0_H$
RS-CAN0	Receive buffer pointer register 11	RSCAN0RMPTR11	<RSCAN0_base> + $06B4_H$
RS-CAN0	Receive buffer data field 0 register 11	RSCAN0RMDF011	<RSCAN0_base> + $06B8_H$
RS-CAN0	Receive buffer data field 1 register 11	RSCAN0RMDF111	<RSCAN0_base> + $06BC_H$
RS-CAN0	Receive buffer ID register 12	RSCAN0RMID12	<RSCAN0_base> + $06C0_H$
RS-CAN0	Receive buffer pointer register 12	RSCAN0RMPTR12	<RSCAN0_base> + $06C4_H$
RS-CAN0	Receive buffer data field 0 register 12	RSCAN0RMDF012	<RSCAN0_base> + $06C8_H$
RS-CAN0	Receive buffer data field 1 register 12	RSCAN0RMDF112	<RSCAN0_base> + $06CC_H$
RS-CAN0	Receive buffer ID register 13	RSCAN0RMID13	<RSCAN0_base> + $06D0_H$
RS-CAN0	Receive buffer pointer register 13	RSCAN0RMPTR13	<RSCAN0_base> + $06D4_H$
RS-CAN0	Receive buffer data field 0 register 13	RSCAN0RMDF013	<RSCAN0_base> + $06D8_H$
RS-CAN0	Receive buffer data field 1 register 13	RSCAN0RMDF113	<RSCAN0_base> + $06DC_H$
RS-CAN0	Receive buffer ID register 14	RSCAN0RMID14	<RSCAN0_base> + $06E0_H$
RS-CAN0	Receive buffer pointer register 14	RSCAN0RMPTR14	<RSCAN0_base> + $06E4_H$
RS-CAN0	Receive buffer data field 0 register 14	RSCAN0RMDF014	<RSCAN0_base> + $06E8_H$
RS-CAN0	Receive buffer data field 1 register 14	RSCAN0RMDF114	<RSCAN0_base> + $06EC_H$
RS-CAN0	Receive buffer ID register 15	RSCAN0RMID15	<RSCAN0_base> + $06F0_H$
RS-CAN0	Receive buffer pointer register 15	RSCAN0RMPTR15	<RSCAN0_base> + $06F4_H$
RS-CAN0	Receive buffer data field 0 register 15	RSCAN0RMDF015	<RSCAN0_base> + $06F8_H$
RS-CAN0	Receive buffer data field 1 register 15	RSCAN0RMDF115	<RSCAN0_base> + $06FC_H$
RS-CAN0	Receive buffer ID register 16	RSCAN0RMID16	<RSCAN0_base> + 0700_H
RS-CAN0	Receive buffer pointer register 16	RSCAN0RMPTR16	<RSCAN0_base> + 0704_H
RS-CAN0	Receive buffer data field 0 register 16	RSCAN0RMDF016	<RSCAN0_base> + 0708_H
RS-CAN0	Receive buffer data field 1 register16	RSCAN0RMDF116	<RSCAN0_base> + $070C_H$

Table 14.10 Registers (10/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 17	RSCAN0RMID17	<RSCAN0_base> + 0710_H
RS-CAN0	Receive buffer pointer register 17	RSCAN0RMPTR17	<RSCAN0_base> + 0714_H
RS-CAN0	Receive buffer data field 0 register 17	RSCAN0RMDF017	<RSCAN0_base> + 0718_H
RS-CAN0	Receive buffer data field 1 register 17	RSCAN0RMDF117	<RSCAN0_base> + $071C_H$
RS-CAN0	Receive buffer ID register 18	RSCAN0RMID18	<RSCAN0_base> + 0720_H
RS-CAN0	Receive buffer pointer register 18	RSCAN0RMPTR18	<RSCAN0_base> + 0724_H
RS-CAN0	Receive buffer data field 0 register 18	RSCAN0RMDF018	<RSCAN0_base> + 0728_H
RS-CAN0	Receive buffer data field 1 register 18	RSCAN0RMDF118	<RSCAN0_base> + $072C_H$
RS-CAN0	Receive buffer ID register 19	RSCAN0RMID19	<RSCAN0_base> + 0730_H
RS-CAN0	Receive buffer pointer register 19	RSCAN0RMPTR19	<RSCAN0_base> + 0734_H
RS-CAN0	Receive buffer data field 0 register 19	RSCAN0MDF019	<RSCAN0_base> + 0738_H
RS-CAN0	Receive buffer data field 1 register 19	RSCAN0RMDF119	<RSCAN0_base> + $073C_H$
RS-CAN0	Receive buffer ID register 20	RSCAN0RMID20	<RSCAN0_base> + 0740_H
RS-CAN0	Receive buffer pointer register 20	RSCAN0RMPTR20	<RSCAN0_base> + 0744_H
RS-CAN0	Receive buffer data field 0 register 20	RSCAN0RMDF020	<RSCAN0_base> + 0748_H
RS-CAN0	Receive buffer data field 1 register 20	RSCAN0RMDF120	<RSCAN0_base> + $074C_H$
RS-CAN0	Receive buffer ID register 21	RSCAN0RMID21	<RSCAN0_base> + 0750_H
RS-CAN0	Receive buffer pointer register 21	RSCAN0RMPTR21	<RSCAN0_base> + 0754_H
RS-CAN0	Receive buffer data field 0 register 21	RSCAN0RMDF021	<RSCAN0_base> + 0758_H
RS-CAN0	Receive buffer data field 1 register 21	RSCAN0RMDF121	<RSCAN0_base> + $075C_H$
RS-CAN0	Receive buffer ID register 22	RSCAN0RMID22	<RSCAN0_base> + 0760_H
RS-CAN0	Receive buffer pointer register 22	RSCAN0RMPTR22	<RSCAN0_base> + 0764_H
RS-CAN0	Receive buffer data field 0 register 22	RSCAN0RMDF022	<RSCAN0_base> + 0768_H
RS-CAN0	Receive buffer data field 1 register 22	RSCAN0RMDF122	<RSCAN0_base> + $076C_H$
RS-CAN0	Receive buffer ID register 23	RSCAN0RMID23	<RSCAN0_base> + 0770_H
RS-CAN0	Receive buffer pointer register 23	RSCAN0RMPTR23	<RSCAN0_base> + 0774_H
RS-CAN0	Receive buffer data field 0 register 23	RSCAN0RMDF023	<RSCAN0_base> + 0778_H
RS-CAN0	Receive buffer data field 1 register 23	RSCAN0RMDF123	<RSCAN0_base> + $077C_H$
RS-CAN0	Receive buffer ID register 24	RSCAN0RMID24	<RSCAN0_base> + 0780_H
RS-CAN0	Receive buffer pointer register 24	RSCAN0RMPTR24	<RSCAN0_base> + 0784_H
RS-CAN0	Receive buffer data field 0 register 24	RSCAN0RMDF024	<RSCAN0_base> + 0788_H
RS-CAN0	Receive buffer data field 1 register 24	RSCAN0RMDF124	<RSCAN0_base> + $078C_H$
RS-CAN0	Receive buffer ID register 25	RSCAN0RMID25	<RSCAN0_base> + 0790_H
RS-CAN0	Receive buffer pointer register 25	RSCAN0RMPTR25	<RSCAN0_base> + 0794_H
RS-CAN0	Receive buffer data field 0 register 25	RSCAN0RMDF025	<RSCAN0_base> + 0798_H
RS-CAN0	Receive buffer data field 1 register 25	RSCAN0RMDF125	<RSCAN0_base> + $079C_H$
RS-CAN0	Receive buffer ID register 26	RSCAN0RMID26	<RSCAN0_base> + $07A0_H$
RS-CAN0	Receive buffer pointer register 26	RSCAN0RMPTR26	<RSCAN0_base> + $07A4_H$
RS-CAN0	Receive buffer data field 0 register 26	RSCAN0RMDF026	<RSCAN0_base> + $07A8_H$
RS-CAN0	Receive buffer data field 1 register 26	RSCAN0RMDF126	<RSCAN0_base> + $07AC_H$
RS-CAN0	Receive buffer ID register 27	RSCAN0RMID27	<RSCAN0_base> + $07B0_H$
RS-CAN0	Receive buffer pointer register 27	RSCAN0RMPTR27	<RSCAN0_base> + $07B4_H$
RS-CAN0	Receive buffer data field 0 register 27	RSCAN0RMDF027	<RSCAN0_base> + $07B8_H$
RS-CAN0	Receive buffer data field 1 register 27	RSCAN0RMDF127	<RSCAN0_base> + $07BC_H$

Table 14.10 Registers (11/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 28	RSCAN0RMID28	<RSCAN0_base> + $07C0_H$
RS-CAN0	Receive buffer pointer register 28	RSCAN0RMPTR28	<RSCAN0_base> + $07C4_H$
RS-CAN0	Receive buffer data field 0 register 28	RSCAN0RMDF028	<RSCAN0_base> + $07C8_H$
RS-CAN0	Receive buffer data field 1 register 28	RSCAN0RMDF128	<RSCAN0_base> + $07CC_H$
RS-CAN0	Receive buffer ID register 29	RSCAN0RMID29	<RSCAN0_base> + $07D0_H$
RS-CAN0	Receive buffer pointer register 29	RSCAN0RMPTR29	<RSCAN0_base> + $07D4_H$
RS-CAN0	Receive buffer data field 0 register 29	RSCAN0RMDF029	<RSCAN0_base> + $07D8_H$
RS-CAN0	Receive buffer data field 1 register 29	RSCAN0RMDF129	<RSCAN0_base> + $07DC_H$
RS-CAN0	Receive buffer ID register 30	RSCAN0RMID30	<RSCAN0_base> + $07E0_H$
RS-CAN0	Receive buffer pointer register 30	RSCAN0RMPTR30	<RSCAN0_base> + $07E4_H$
RS-CAN0	Receive buffer data field 0 register 30	RSCAN0RMDF030	<RSCAN0_base> + $07E8_H$
RS-CAN0	Receive buffer data field 1 register 30	RSCAN0RMDF130	<RSCAN0_base> + $07EC_H$
RS-CAN0	Receive buffer ID register 31	RSCAN0RMID31	<RSCAN0_base> + $07F0_H$
RS-CAN0	Receive buffer pointer register 31	RSCAN0RMPTR31	<RSCAN0_base> + $07F4_H$
RS-CAN0	Receive buffer data field 0 register 31	RSCAN0RMDF031	<RSCAN0_base> + $07F8_H$
RS-CAN0	Receive buffer data field 1 register 31	RSCAN0RMDF131	<RSCAN0_base> + $07FC_H$
RS-CAN0	Receive buffer ID register 32	RSCAN0RMID32	<RSCAN0_base> + 0800_H
RS-CAN0	Receive buffer pointer register 32	RSCAN0RMPTR32	<RSCAN0_base> + 0804_H
RS-CAN0	Receive buffer data field 0 register 32	RSCAN0RMDF032	<RSCAN0_base> + 0808_H
RS-CAN0	Receive buffer data field 1 register 32	RSCAN0RMDF132	<RSCAN0_base> + $080C_H$
RS-CAN0	Receive buffer ID register 33	RSCAN0RMID33	<RSCAN0_base> + 0810_H
RS-CAN0	Receive buffer pointer register 33	RSCAN0RMPTR33	<RSCAN0_base> + 0814_H
RS-CAN0	Receive buffer data field 0 register 33	RSCAN0RMDF033	<RSCAN0_base> + 0818_H
RS-CAN0	Receive buffer data field 1 register 33	RSCAN0RMDF133	<RSCAN0_base> + $081C_H$
RS-CAN0	Receive buffer ID register 34	RSCAN0RMID34	<RSCAN0_base> + 0820_H
RS-CAN0	Receive buffer pointer register 34	RSCAN0RMPTR34	<RSCAN0_base> + 0824_H
RS-CAN0	Receive buffer data field 0 register 34	RSCAN0RMDF034	<RSCAN0_base> + 0828_H
RS-CAN0	Receive buffer data field 1 register 34	RSCAN0RMDF134	<RSCAN0_base> + $082C_H$
RS-CAN0	Receive buffer ID register 35	RSCAN0RMID35	<RSCAN0_base> + 0830_H
RS-CAN0	Receive buffer pointer register 35	RSCAN0RMPTR35	<RSCAN0_base> + 0834_H
RS-CAN0	Receive buffer data field 0 register 35	RSCAN0RMDF035	<RSCAN0_base> + 0838_H
RS-CAN0	Receive buffer data field 1 register 35	RSCAN0RMDF135	<RSCAN0_base> + $083C_H$
RS-CAN0	Receive buffer ID register 36	RSCAN0RMID36	<RSCAN0_base> + 0840_H
RS-CAN0	Receive buffer pointer register 36	RSCAN0RMPTR36	<RSCAN0_base> + 0844_H
RS-CAN0	Receive buffer data field 0 register 36	RSCAN0RMDF036	<RSCAN0_base> + 0848_H
RS-CAN0	Receive buffer data field 1 register 36	RSCAN0RMDF136	<RSCAN0_base> + $084C_H$
RS-CAN0	Receive buffer ID register 37	RSCAN0RMID37	<RSCAN0_base> + 0850_H
RS-CAN0	Receive buffer pointer register 37	RSCAN0RMPTR37	<RSCAN0_base> + 0854_H
RS-CAN0	Receive buffer data field 0 register 37	RSCAN0RMDF037	<RSCAN0_base> + 0858_H
RS-CAN0	Receive buffer data field 1 register 37	RSCAN0RMDF137	<RSCAN0_base> + $085C_H$
RS-CAN0	Receive buffer ID register 38	RSCAN0RMID38	<RSCAN0_base> + 0860_H
RS-CAN0	Receive buffer pointer register 38	RSCAN0RMPTR38	<RSCAN0_base> + 0864_H
RS-CAN0	Receive buffer data field 0 register 38	RSCAN0RMDF038	<RSCAN0_base> + 0868_H
RS-CAN0	Receive buffer data field 1 register 38	RSCAN0RMDF138	<RSCAN0_base> + $086C_H$

Table 14.10 Registers (12/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 39	RSCAN0RMID39	<RSCAN0_base> + 0870_H
RS-CAN0	Receive buffer pointer register 39	RSCAN0RMPTR39	<RSCAN0_base> + 0874_H
RS-CAN0	Receive buffer data field 0 register 39	RSCAN0RMDF039	<RSCAN0_base> + 0878_H
RS-CAN0	Receive buffer data field 1 register 39	RSCAN0RMDF139	<RSCAN0_base> + $087C_H$
RS-CAN0	Receive buffer ID register 40	RSCAN0RMID40	<RSCAN0_base> + 0880_H
RS-CAN0	Receive buffer pointer register 40	RSCAN0RMPTR40	<RSCAN0_base> + 0884_H
RS-CAN0	Receive buffer data field 0 register 40	RSCAN0RMDF040	<RSCAN0_base> + 0888_H
RS-CAN0	Receive buffer data field 1 register 40	RSCAN0RMDF140	<RSCAN0_base> + $088C_H$
RS-CAN0	Receive buffer ID register 41	RSCAN0RMID41	<RSCAN0_base> + 0890_H
RS-CAN0	Receive buffer pointer register 41	RSCAN0RMPTR41	<RSCAN0_base> + 0894_H
RS-CAN0	Receive buffer data field 0 register 41	RSCAN0RMDF041	<RSCAN0_base> + 0898_H
RS-CAN0	Receive buffer data field 1 register 41	RSCAN0RMDF141	<RSCAN0_base> + $089C_H$
RS-CAN0	Receive buffer ID register 42	RSCAN0RMID42	<RSCAN0_base> + $08A0_H$
RS-CAN0	Receive buffer pointer register 42	RSCAN0RMPTR42	<RSCAN0_base> + $08A4_H$
RS-CAN0	Receive buffer data field 0 register 42	RSCAN0RMDF042	<RSCAN0_base> + $08A8_H$
RS-CAN0	Receive buffer data field 1 register 42	RSCAN0RMDF142	<RSCAN0_base> + $08AC_H$
RS-CAN0	Receive buffer ID register 43	RSCAN0RMID43	<RSCAN0_base> + $08B0_H$
RS-CAN0	Receive buffer pointer register 43	RSCAN0RMPTR43	<RSCAN0_base> + $08B4_H$
RS-CAN0	Receive buffer data field 0 register 43	RSCAN0RMDF043	<RSCAN0_base> + $08B8_H$
RS-CAN0	Receive buffer data field 1 register 43	RSCAN0RMDF143	<RSCAN0_base> + $08BC_H$
RS-CAN0	Receive buffer ID register 44	RSCAN0RMID44	<RSCAN0_base> + $08C0_H$
RS-CAN0	Receive buffer pointer register 44	RSCAN0RMPTR44	<RSCAN0_base> + $08C4_H$
RS-CAN0	Receive buffer data field 0 register 44	RSCAN0RMDF044	<RSCAN0_base> + $08C8_H$
RS-CAN0	Receive buffer data field 1 register 44	RSCAN0RMDF144	<RSCAN0_base> + $08CC_H$
RS-CAN0	Receive buffer ID register 45	RSCAN0RMID45	<RSCAN0_base> + $08D0_H$
RS-CAN0	Receive buffer pointer register 45	RSCAN0RMPTR45	<RSCAN0_base> + $08D4_H$
RS-CAN0	Receive buffer data field 0 register 45	RSCAN0RMDF045	<RSCAN0_base> + $08D8_H$
RS-CAN0	Receive buffer data field 1 register 45	RSCAN0RMDF145	<RSCAN0_base> + $08DC_H$
RS-CAN0	Receive buffer ID register 46	RSCAN0RMID46	<RSCAN0_base> + $08E0_H$
RS-CAN0	Receive buffer pointer register 46	RSCAN0RMPTR46	<RSCAN0_base> + $08E4_H$
RS-CAN0	Receive buffer data field 0 register 46	RSCAN0RMDF046	<RSCAN0_base> + $08E8_H$
RS-CAN0	Receive buffer data field 1 register 46	RSCAN0RMDF146	<RSCAN0_base> + $08EC_H$
RS-CAN0	Receive buffer ID register 47	RSCAN0RMID47	<RSCAN0_base> + $08F0_H$
RS-CAN0	Receive buffer pointer register 47	RSCAN0RMPTR47	<RSCAN0_base> + $08F4_H$
RS-CAN0	Receive buffer data field 0 register 47	RSCAN0RMDF047	<RSCAN0_base> + $08F8_H$
RS-CAN0	Receive buffer data field 1 register 47	RSCAN0RMDF147	<RSCAN0_base> + $08FC_H$
RS-CAN0	Receive buffer ID register 48	RSCAN0RMID48	<RSCAN0_base> + 0900_H
RS-CAN0	Receive buffer pointer register 48	RSCAN0RMPTR48	<RSCAN0_base> + 0904_H
RS-CAN0	Receive buffer data field 0 register 48	RSCAN0RMDF048	<RSCAN0_base> + 0908_H
RS-CAN0	Receive buffer data field 1 register 48	RSCAN0RMDF148	<RSCAN0_base> + $090C_H$
RS-CAN0	Receive buffer ID register 49	RSCAN0RMID49	<RSCAN0_base> + 0910_H
RS-CAN0	Receive buffer pointer register 49	RSCAN0RMPTR49	<RSCAN0_base> + 0914_H
RS-CAN0	Receive buffer data field 0 register 49	RSCAN0RMDF049	<RSCAN0_base> + 0918_H
RS-CAN0	Receive buffer data field 1 register 49	RSCAN0RMDF149	<RSCAN0_base> + $091C_H$

Table 14.10 Registers (13/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 50	RSCAN0RMID50	<RSCAN0_base> + 0920_H
RS-CAN0	Receive buffer pointer register 50	RSCAN0RMPTR50	<RSCAN0_base> + 0924_H
RS-CAN0	Receive buffer data field 0 register 50	RSCAN0RMDF050	<RSCAN0_base> + 0928_H
RS-CAN0	Receive buffer data field 1 register 50	RSCAN0RMDF150	<RSCAN0_base> + $092C_H$
RS-CAN0	Receive buffer ID register 51	RSCAN0RMID51	<RSCAN0_base> + 0930_H
RS-CAN0	Receive buffer pointer register 51	RSCAN0RMPTR51	<RSCAN0_base> + 0934_H
RS-CAN0	Receive buffer data field 0 register 51	RSCAN0RMDF051	<RSCAN0_base> + 0938_H
RS-CAN0	Receive buffer data field 1 register 51	RSCAN0RMDF151	<RSCAN0_base> + $093C_H$
RS-CAN0	Receive buffer ID register 52	RSCAN0RMID52	<RSCAN0_base> + 0940_H
RS-CAN0	Receive buffer pointer register 52	RSCAN0RMPTR52	<RSCAN0_base> + 0944_H
RS-CAN0	Receive buffer data field 0 register 52	RSCAN0RMDF052	<RSCAN0_base> + 0948_H
RS-CAN0	Receive buffer data field 1 register 52	RSCAN0RMDF152	<RSCAN0_base> + $094C_H$
RS-CAN0	Receive buffer ID register 53	RSCAN0RMID53	<RSCAN0_base> + 0950_H
RS-CAN0	Receive buffer pointer register 53	RSCAN0RMPTR53	<RSCAN0_base> + 0954_H
RS-CAN0	Receive buffer data field 0 register 53	RSCAN0RMDF053	<RSCAN0_base> + 0958_H
RS-CAN0	Receive buffer data field 1 register 53	RSCAN0RMDF153	<RSCAN0_base> + $095C_H$
RS-CAN0	Receive buffer ID register 54	RSCAN0RMID54	<RSCAN0_base> + 0960_H
RS-CAN0	Receive buffer pointer register 54	RSCAN0RMPTR54	<RSCAN0_base> + 0964_H
RS-CAN0	Receive buffer data field 0 register 54	RSCAN0RMDF054	<RSCAN0_base> + 0968_H
RS-CAN0	Receive buffer data field 1 register 54	RSCAN0RMDF154	<RSCAN0_base> + $096C_H$
RS-CAN0	Receive buffer ID register 55	RSCAN0RMID55	<RSCAN0_base> + 0970_H
RS-CAN0	Receive buffer pointer register 55	RSCAN0RMPTR55	<RSCAN0_base> + 0974_H
RS-CAN0	Receive buffer data field 0 register 55	RSCAN0RMDF055	<RSCAN0_base> + 0978_H
RS-CAN0	Receive buffer data field 1 register 55	RSCAN0RMDF155	<RSCAN0_base> + $097C_H$
RS-CAN0	Receive buffer ID register 56	RSCAN0RMID56	<RSCAN0_base> + 0980_H
RS-CAN0	Receive buffer pointer register 56	RSCAN0RMPTR56	<RSCAN0_base> + 0984_H
RS-CAN0	Receive buffer data field 0 register 56	RSCAN0RMDF056	<RSCAN0_base> + 0988_H
RS-CAN0	Receive buffer data field 1 register 56	RSCAN0RMDF156	<RSCAN0_base> + $098C_H$
RS-CAN0	Receive buffer ID register 57	RSCAN0RMID57	<RSCAN0_base> + 0990_H
RS-CAN0	Receive buffer pointer register 57	RSCAN0RMPTR57	<RSCAN0_base> + 0994_H
RS-CAN0	Receive buffer data field 0 register 57	RSCAN0RMDF057	<RSCAN0_base> + 0998_H
RS-CAN0	Receive buffer data field 1 register 57	RSCAN0RMDF157	<RSCAN0_base> + $099C_H$
RS-CAN0	Receive buffer ID register 58	RSCAN0RMID58	<RSCAN0_base> + $09A0_H$
RS-CAN0	Receive buffer pointer register 58	RSCAN0RMPTR58	<RSCAN0_base> + $09A4_H$
RS-CAN0	Receive buffer data field 0 register 58	RSCAN0RMDF058	<RSCAN0_base> + $09A8_H$
RS-CAN0	Receive buffer data field 1 register 58	RSCAN0RMDF158	<RSCAN0_base> + $09AC_H$
RS-CAN0	Receive buffer ID register 59	RSCAN0RMID59	<RSCAN0_base> + $09B0_H$
RS-CAN0	Receive buffer pointer register 59	RSCAN0RMPTR59	<RSCAN0_base> + $09B4_H$
RS-CAN0	Receive buffer data field 0 register 59	RSCAN0RMDF059	<RSCAN0_base> + $09B8_H$
RS-CAN0	Receive buffer data field 1 register 59	RSCAN0RMDF159	<RSCAN0_base> + $09BC_H$
RS-CAN0	Receive buffer ID register 60	RSCAN0RMID60	<RSCAN0_base> + $09C0_H$
RS-CAN0	Receive buffer pointer register 60	RSCAN0RMPTR60	<RSCAN0_base> + $09C4_H$
RS-CAN0	Receive buffer data field 0 register 60	RSCAN0RMDF060	<RSCAN0_base> + $09C8_H$
RS-CAN0	Receive buffer data field 1 register 60	RSCAN0RMDF160	<RSCAN0_base> + $09CC_H$

Table 14.10 Registers (14/23)

Module	Register	Symbol	Address
RS-CAN0	Receive buffer ID register 61	RSCAN0RMID61	<RSCAN0_base> + $09D0_H$
RS-CAN0	Receive buffer pointer register 61	RSCAN0RMPTR61	<RSCAN0_base> + $09D4_H$
RS-CAN0	Receive buffer data field 0 register 61	RSCAN0RMDF061	<RSCAN0_base> + $09D8_H$
RS-CAN0	Receive buffer data field 1 register 61	RSCAN0RMDF161	<RSCAN0_base> + $09DC_H$
RS-CAN0	Receive buffer ID register 62	RSCAN0RMID62	<RSCAN0_base> + $09E0_H$
RS-CAN0	Receive buffer pointer register 62	RSCAN0RMPTR62	<RSCAN0_base> + $09E4_H$
RS-CAN0	Receive buffer data field 0 register 62	RSCAN0RMDF062	<RSCAN0_base> + $09E8_H$
RS-CAN0	Receive buffer data field 1 register 62	RSCAN0RMDF162	<RSCAN0_base> + $09EC_H$
RS-CAN0	Receive buffer ID register 63	RSCAN0RMID63	<RSCAN0_base> + $09F0_H$
RS-CAN0	Receive buffer pointer register 63	RSCAN0RMPTR63	<RSCAN0_base> + $09F4_H$
RS-CAN0	Receive buffer data field 0 register 63	RSCAN0RMDF063	<RSCAN0_base> + $09F8_H$
RS-CAN0	Receive buffer data field 1 register 63	RSCAN0RMDF163	<RSCAN0_base> + $09FC_H$
RS-CAN0	Receive FIFO buffer access ID register 0	RSCAN0RFID0	<RSCAN0_base> + $0E00_H$
RS-CAN0	Receive FIFO buffer access pointer register 0	RSCAN0RFPTR0	<RSCAN0_base> + $0E04_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 0	RSCAN0RFDF00	<RSCAN0_base> + $0E08_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 0	RSCAN0RFDF10	<RSCAN0_base> + $0E0C_H$
RS-CAN0	Receive FIFO buffer access ID register 1	RSCAN0RFID1	<RSCAN0_base> + $0E10_H$
RS-CAN0	Receive FIFO buffer access pointer register 1	RSCAN0RFPTR1	<RSCAN0_base> + $0E14_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 1	RSCAN0RFDF01	<RSCAN0_base> + $0E18_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 1	RSCAN0RFDF11	<RSCAN0_base> + $0E1C_H$
RS-CAN0	Receive FIFO buffer access ID register 2	RSCAN0RFID2	<RSCAN0_base> + $0E20_H$
RS-CAN0	Receive FIFO buffer access pointer register 2	RSCAN0RFPTR2	<RSCAN0_base> + $0E24_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 2	RSCAN0RFDF02	<RSCAN0_base> + $0E28_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 2	RSCAN0RFDF12	<RSCAN0_base> + $0E2C_H$
RS-CAN0	Receive FIFO buffer access ID register 3	RSCAN0RFID3	<RSCAN0_base> + $0E30_H$
RS-CAN0	Receive FIFO buffer access pointer register 3	RSCAN0RFPTR3	<RSCAN0_base> + $0E34_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 3	RSCAN0RFDF03	<RSCAN0_base> + $0E38_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 3	RSCAN0RFDF13	<RSCAN0_base> + $0E3C_H$
RS-CAN0	Receive FIFO buffer access ID register 4	RSCAN0RFID4	<RSCAN0_base> + $0E40_H$
RS-CAN0	Receive FIFO buffer access pointer register 4	RSCAN0RFPTR4	<RSCAN0_base> + $0E44_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 4	RSCAN0RFDF04	<RSCAN0_base> + $0E48_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 4	RSCAN0RFDF14	<RSCAN0_base> + $0E4C_H$
RS-CAN0	Receive FIFO buffer access ID register 5	RSCAN0RFID5	<RSCAN0_base> + $0E50_H$
RS-CAN0	Receive FIFO buffer access pointer register 5	RSCAN0RFPTR5	<RSCAN0_base> + $0E54_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 5	RSCAN0RFDF05	<RSCAN0_base> + $0E58_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 5	RSCAN0RFDF15	<RSCAN0_base> + $0E5C_H$
RS-CAN0	Receive FIFO buffer access ID register 6	RSCAN0RFID6	<RSCAN0_base> + $0E60_H$
RS-CAN0	Receive FIFO buffer access pointer register 6	RSCAN0RFPTR6	<RSCAN0_base> + $0E64_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 6	RSCAN0RFDF06	<RSCAN0_base> + $0E68_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 6	RSCAN0RFDF16	<RSCAN0_base> + $0E6C_H$
RS-CAN0	Receive FIFO buffer access ID register 7	RSCAN0RFID7	<RSCAN0_base> + $0E70_H$
RS-CAN0	Receive FIFO buffer access pointer register 7	RSCAN0RFPTR7	<RSCAN0_base> + $0E74_H$
RS-CAN0	Receive FIFO buffer access data field 0 register 7	RSCAN0RFDF07	<RSCAN0_base> + $0E78_H$
RS-CAN0	Receive FIFO buffer access data field 1 register 7	RSCAN0RFDF17	<RSCAN0_base> + $0E7C_H$

Table 14.10 Registers (15/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit/receive FIFO buffer access ID register 0	RSCAN0CFID0	<RSCAN0_base> + $0E80_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 0	RSCAN0CFPTR0	<RSCAN0_base> + $0E84_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 0	RSCAN0CFDF00	<RSCAN0_base> + $0E88_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 0	RSCAN0CFDF10	<RSCAN0_base> + $0E8C_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 1	RSCAN0CFID1	<RSCAN0_base> + $0E90_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 1	RSCAN0CFPTR1	<RSCAN0_base> + $0E94_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 1	RSCAN0CFDF01	<RSCAN0_base> + $0E98_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 1	RSCAN0CFDF11	<RSCAN0_base> + $0E9C_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 2	RSCAN0CFID2	<RSCAN0_base> + $0EA0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 2	RSCAN0CFPTR2	<RSCAN0_base> + $0EA4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 2	RSCAN0CFDF02	<RSCAN0_base> + $0EA8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 2	RSCAN0CFDF12	<RSCAN0_base> + $0EAC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 3	RSCAN0CFID3	<RSCAN0_base> + $0EB0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 3	RSCAN0CFPTR3	<RSCAN0_base> + $0EB4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 3	RSCAN0CFDF03	<RSCAN0_base> + $0EB8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 3	RSCAN0CFDF13	<RSCAN0_base> + $0EBC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 4	RSCAN0CFID4	<RSCAN0_base> + $0EC0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 4	RSCAN0CFPTR4	<RSCAN0_base> + $0EC4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 4	RSCAN0CFDF04	<RSCAN0_base> + $0EC8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 4	RSCAN0CFDF14	<RSCAN0_base> + $0ECC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 5	RSCAN0CFID5	<RSCAN0_base> + $0ED0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 5	RSCAN0CFPTR5	<RSCAN0_base> + $0ED4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 5	RSCAN0CFDF05	<RSCAN0_base> + $0ED8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 5	RSCAN0CFDF15	<RSCAN0_base> + $0EDC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 6	RSCAN0CFID6	<RSCAN0_base> + $0EE0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 6	RSCAN0CFPTR6	<RSCAN0_base> + $0EE4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 6	RSCAN0CFDF06	<RSCAN0_base> + $0EE8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 6	RSCAN0CFDF16	<RSCAN0_base> + $0EEC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 7	RSCAN0CFID7	<RSCAN0_base> + $0EF0_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 7	RSCAN0CFPTR7	<RSCAN0_base> + $0EF4_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 7	RSCAN0CFDF07	<RSCAN0_base> + $0EF8_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 7	RSCAN0CFDF17	<RSCAN0_base> + $0EFC_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 8	RSCAN0CFID8	<RSCAN0_base> + $0F00_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 8	RSCAN0CFPTR8	<RSCAN0_base> + $0F04_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 8	RSCAN0CFDF08	<RSCAN0_base> + $0F08_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 8	RSCAN0CFDF18	<RSCAN0_base> + $0F0C_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 9	RSCAN0CFID9	<RSCAN0_base> + $0F10_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 9	RSCAN0CFPTR9	<RSCAN0_base> + $0F14_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 9	RSCAN0CFDF09	<RSCAN0_base> + $0F18_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 9	RSCAN0CFDF19	<RSCAN0_base> + $0F1C_H$
RS-CAN0	Transmit/receive FIFO buffer access ID register 10	RSCAN0CFID10	<RSCAN0_base> + $0F20_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 10	RSCAN0CFPTR10	<RSCAN0_base> + $0F24_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 10	RSCAN0CFDF010	<RSCAN0_base> + $0F28_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 10	RSCAN0CFDF110	<RSCAN0_base> + $0F2C_H$

Table 14.10 Registers (16/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit/receive FIFO buffer access ID register 11	RSCAN0CFID11	<RSCAN0_base> + $0F30_H$
RS-CAN0	Transmit/receive FIFO buffer access pointer register 11	RSCAN0CFPTR11	<RSCAN0_base> + $0F34_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 0 register 11	RSCAN0CFDF011	<RSCAN0_base> + $0F38_H$
RS-CAN0	Transmit/receive FIFO buffer access data field 1 register 11	RSCAN0CFDF111	<RSCAN0_base> + $0F3C_H$
RS-CAN0	Transmit buffer ID register 0	RSCAN0TMID0	<RSCAN0_base> + 1000_H
RS-CAN0	Transmit buffer pointer register 0	RSCAN0TMPTR0	<RSCAN0_base> + 1004_H
RS-CAN0	Transmit buffer data field 0 register 0	RSCAN0TMDF00	<RSCAN0_base> + 1008_H
RS-CAN0	Transmit buffer data field 1 register 0	RSCAN0TMDF10	<RSCAN0_base> + $100C_H$
RS-CAN0	Transmit buffer ID register 1	RSCAN0TMID1	<RSCAN0_base> + 1010_H
RS-CAN0	Transmit buffer pointer register 1	RSCAN0TMPTR1	<RSCAN0_base> + 1014_H
RS-CAN0	Transmit buffer data field 0 register 1	RSCAN0TMDF01	<RSCAN0_base> + 1018_H
RS-CAN0	Transmit buffer data field 1 register 1	RSCAN0TMDF11	<RSCAN0_base> + $101C_H$
RS-CAN0	Transmit buffer ID register 2	RSCAN0TMID2	<RSCAN0_base> + 1020_H
RS-CAN0	Transmit buffer pointer register 2	RSCAN0TMPTR2	<RSCAN0_base> + 1024_H
RS-CAN0	Transmit buffer data field 0 register 2	RSCAN0TMDF02	<RSCAN0_base> + 1028_H
RS-CAN0	Transmit buffer data field 1 register 2	RSCAN0TMDF12	<RSCAN0_base> + $102C_H$
RS-CAN0	Transmit buffer ID register 3	RSCAN0TMID3	<RSCAN0_base> + 1030_H
RS-CAN0	Transmit buffer pointer register 3	RSCAN0TMPTR3	<RSCAN0_base> + 1034_H
RS-CAN0	Transmit buffer data field 0 register 3	RSCAN0TMDF03	<RSCAN0_base> + 1038_H
RS-CAN0	Transmit buffer data field 1 register 3	RSCAN0TMDF13	<RSCAN0_base> + $103C_H$
RS-CAN0	Transmit buffer ID register 4	RSCAN0TMID4	<RSCAN0_base> + 1040_H
RS-CAN0	Transmit buffer pointer register 4	RSCAN0TMPTR4	<RSCAN0_base> + 1044_H
RS-CAN0	Transmit buffer data field 0 register 4	RSCAN0TMDF04	<RSCAN0_base> + 1048_H
RS-CAN0	Transmit buffer data field 1 register 4	RSCAN0TMDF14	<RSCAN0_base> + $104C_H$
RS-CAN0	Transmit buffer ID register 5	RSCAN0TMID5	<RSCAN0_base> + 1050_H
RS-CAN0	Transmit buffer pointer register 5	RSCAN0TMPTR5	<RSCAN0_base> + 1054_H
RS-CAN0	Transmit buffer data field 0 register 5	RSCAN0TMDF05	<RSCAN0_base> + 1058_H
RS-CAN0	Transmit buffer data field 1 register 5	RSCAN0TMDF15	<RSCAN0_base> + $105C_H$
RS-CAN0	Transmit buffer ID register 6	RSCAN0TMID6	<RSCAN0_base> + 1060_H
RS-CAN0	Transmit buffer pointer register 6	RSCAN0TMPTR6	<RSCAN0_base> + 1064_H
RS-CAN0	Transmit buffer data field 0 register 6	RSCAN0TMDF06	<RSCAN0_base> + 1068_H
RS-CAN0	Transmit buffer data field 1 register 6	RSCAN0TMDF16	<RSCAN0_base> + $106C_H$
RS-CAN0	Transmit buffer ID register 7	RSCAN0TMID7	<RSCAN0_base> + 1070_H
RS-CAN0	Transmit buffer pointer register 7	RSCAN0TMPTR7	<RSCAN0_base> + 1074_H
RS-CAN0	Transmit buffer data field 0 register 7	RSCAN0TMDF07	<RSCAN0_base> + 1078_H
RS-CAN0	Transmit buffer data field 1 register 7	RSCAN0TMDF17	<RSCAN0_base> + $107C_H$
RS-CAN0	Transmit buffer ID register 8	RSCAN0TMID8	<RSCAN0_base> + 1080_H
RS-CAN0	Transmit buffer pointer register 8	RSCAN0TMPTR8	<RSCAN0_base> + 1084_H
RS-CAN0	Transmit buffer data field 0 register 8	RSCAN0TMDF08	<RSCAN0_base> + 1088_H
RS-CAN0	Transmit buffer data field 1 register 8	RSCAN0TMDF18	<RSCAN0_base> + $108C_H$
RS-CAN0	Transmit buffer ID register 9	RSCAN0TMID9	<RSCAN0_base> + 1090_H
RS-CAN0	Transmit buffer pointer register 9	RSCAN0TMPTR9	<RSCAN0_base> + 1094_H
RS-CAN0	Transmit buffer data field 0 register 9	RSCAN0TMDF09	<RSCAN0_base> + 1098_H
RS-CAN0	Transmit buffer data field 1 register 9	RSCAN0TMDF19	<RSCAN0_base> + $109C_H$

Table 14.10 Registers (17/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer ID register 10	RSCAN0TMID10	<RSCAN0_base> + $10A0_H$
RS-CAN0	Transmit buffer pointer register 10	RSCAN0TMPTR10	<RSCAN0_base> + $10A4_H$
RS-CAN0	Transmit buffer data field 0 register 10	RSCAN0TMDF010	<RSCAN0_base> + $10A8_H$
RS-CAN0	Transmit buffer data field 1 register 10	RSCAN0TMDF110	<RSCAN0_base> + $10AC_H$
RS-CAN0	Transmit buffer ID register 11	RSCAN0TMID11	<RSCAN0_base> + $10B0_H$
RS-CAN0	Transmit buffer pointer register 11	RSCAN0TMPTR11	<RSCAN0_base> + $10B4_H$
RS-CAN0	Transmit buffer data field 0 register 11	RSCAN0TMDF011	<RSCAN0_base> + $10B8_H$
RS-CAN0	Transmit buffer data field 1 register 11	RSCAN0TMDF111	<RSCAN0_base> + $10BC_H$
RS-CAN0	Transmit buffer ID register 12	RSCAN0TMID12	<RSCAN0_base> + $10C0_H$
RS-CAN0	Transmit buffer pointer register 12	RSCAN0TMPTR12	<RSCAN0_base> + $10C4_H$
RS-CAN0	Transmit buffer data field 0 register 12	RSCAN0TMDF012	<RSCAN0_base> + $10C8_H$
RS-CAN0	Transmit buffer data field 1 register 12	RSCAN0TMDF112	<RSCAN0_base> + $10CC_H$
RS-CAN0	Transmit buffer ID register 13	RSCAN0TMID13	<RSCAN0_base> + $10D0_H$
RS-CAN0	Transmit buffer pointer register 13	RSCAN0TMPTR13	<RSCAN0_base> + $10D4_H$
RS-CAN0	Transmit buffer data field 0 register 13	RSCAN0TMDF013	<RSCAN0_base> + $10D8_H$
RS-CAN0	Transmit buffer data field 1 register 13	RSCAN0TMDF113	<RSCAN0_base> + $10DC_H$
RS-CAN0	Transmit buffer ID register 14	RSCAN0TMID14	<RSCAN0_base> + $10E0_H$
RS-CAN0	Transmit buffer pointer register 14	RSCAN0TMPTR14	<RSCAN0_base> + $10E4_H$
RS-CAN0	Transmit buffer data field 0 register 14	RSCAN0TMDF014	<RSCAN0_base> + $10E8_H$
RS-CAN0	Transmit buffer data field 1 register 14	RSCAN0TMDF114	<RSCAN0_base> + $10EC_H$
RS-CAN0	Transmit buffer ID register 15	RSCAN0TMID15	<RSCAN0_base> + $10F0_H$
RS-CAN0	Transmit buffer pointer register 15	RSCAN0TMPTR15	<RSCAN0_base> + $10F4_H$
RS-CAN0	Transmit buffer data field 0 register 15	RSCAN0TMDF015	<RSCAN0_base> + $10F8_H$
RS-CAN0	Transmit buffer data field 1 register 15	RSCAN0TMDF115	<RSCAN0_base> + $10FC_H$
RS-CAN0	Transmit buffer ID register 16	RSCAN0TMID16	<RSCAN0_base> + 1100_H
RS-CAN0	Transmit buffer pointer register 16	RSCAN0TMPTR16	<RSCAN0_base> + 1104_H
RS-CAN0	Transmit buffer data field 0 register 16	RSCAN0TMDF016	<RSCAN0_base> + 1108_H
RS-CAN0	Transmit buffer data field 1 register 16	RSCAN0TMDF116	<RSCAN0_base> + $110C_H$
RS-CAN0	Transmit buffer ID register 17	RSCAN0TMID17	<RSCAN0_base> + 1110_H
RS-CAN0	Transmit buffer pointer register 17	RSCAN0TMPTR17	<RSCAN0_base> + 1114_H
RS-CAN0	Transmit buffer data field 0 register 17	RSCAN0TMDF017	<RSCAN0_base> + 1118_H
RS-CAN0	Transmit buffer data field 1 register 17	RSCAN0TMDF117	<RSCAN0_base> + $111C_H$
RS-CAN0	Transmit buffer ID register 18	RSCAN0TMID18	<RSCAN0_base> + 1120_H
RS-CAN0	Transmit buffer pointer register 18	RSCAN0TMPTR18	<RSCAN0_base> + 1124_H
RS-CAN0	Transmit buffer data field 0 register 18	RSCAN0TMDF018	<RSCAN0_base> + 1128_H
RS-CAN0	Transmit buffer data field 1 register 18	RSCAN0TMDF118	<RSCAN0_base> + $112C_H$
RS-CAN0	Transmit buffer ID register 19	RSCAN0TMID19	<RSCAN0_base> + 1130_H
RS-CAN0	Transmit buffer pointer register 19	RSCAN0TMPTR19	<RSCAN0_base> + 1134_H
RS-CAN0	Transmit buffer data field 0 register 19	RSCAN0TMDF019	<RSCAN0_base> + 1138_H
RS-CAN0	Transmit buffer data field 1 register 19	RSCAN0TMDF119	<RSCAN0_base> + $113C_H$
RS-CAN0	Transmit buffer ID register 20	RSCAN0TMID20	<RSCAN0_base> + 1140_H
RS-CAN0	Transmit buffer pointer register 20	RSCAN0TMPTR20	<RSCAN0_base> + 1144_H
RS-CAN0	Transmit buffer data field 0 register 20	RSCAN0TMDF020	<RSCAN0_base> + 1148_H
RS-CAN0	Transmit buffer data field 1 register 20	RSCAN0TMDF120	<RSCAN0_base> + $114C_H$

Table 14.10 Registers (18/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer ID register 21	RSCAN0TMID21	<RSCAN0_base> + 1150_H
RS-CAN0	Transmit buffer pointer register 21	RSCAN0TMPTR21	<RSCAN0_base> + 1154_H
RS-CAN0	Transmit buffer data field 0 register 21	RSCAN0TMDF021	<RSCAN0_base> + 1158_H
RS-CAN0	Transmit buffer data field 1 register 21	RSCAN0TMDF121	<RSCAN0_base> + $115C_H$
RS-CAN0	Transmit buffer ID register 22	RSCAN0TMID22	<RSCAN0_base> + 1160_H
RS-CAN0	Transmit buffer pointer register 22	RSCAN0TMPTR22	<RSCAN0_base> + 1164_H
RS-CAN0	Transmit buffer data field 0 register 22	RSCAN0TMDF022	<RSCAN0_base> + 1168_H
RS-CAN0	Transmit buffer data field 1 register 22	RSCAN0TMDF122	<RSCAN0_base> + $116C_H$
RS-CAN0	Transmit buffer ID register 23	RSCAN0TMID23	<RSCAN0_base> + 1170_H
RS-CAN0	Transmit buffer pointer register 23	RSCAN0TMPTR23	<RSCAN0_base> + 1174_H
RS-CAN0	Transmit buffer data field 0 register 23	RSCAN0TMDF023	<RSCAN0_base> + 1178_H
RS-CAN0	Transmit buffer data field 1 register 23	RSCAN0TMDF123	<RSCAN0_base> + $117C_H$
RS-CAN0	Transmit buffer ID register 24	RSCAN0TMID24	<RSCAN0_base> + 1180_H
RS-CAN0	Transmit buffer pointer register 24	RSCAN0TMPTR24	<RSCAN0_base> + 1184_H
RS-CAN0	Transmit buffer data field 0 register 24	RSCAN0TMDF024	<RSCAN0_base> + 1188_H
RS-CAN0	Transmit buffer data field 1 register 24	RSCAN0TMDF124	<RSCAN0_base> + $118C_H$
RS-CAN0	Transmit buffer ID register 25	RSCAN0TMID25	<RSCAN0_base> + 1190_H
RS-CAN0	Transmit buffer pointer register 25	RSCAN0TMPTR25	<RSCAN0_base> + 1194_H
RS-CAN0	Transmit buffer data field 0 register 25	RSCAN0TMDF025	<RSCAN0_base> + 1198_H
RS-CAN0	Transmit buffer data field 1 register 25	RSCAN0TMDF125	<RSCAN0_base> + $119C_H$
RS-CAN0	Transmit buffer ID register 26	RSCAN0TMID26	<RSCAN0_base> + $11A0_H$
RS-CAN0	Transmit buffer pointer register 26	RSCAN0TMPTR26	<RSCAN0_base> + $11A4_H$
RS-CAN0	Transmit buffer data field 0 register 26	RSCAN0TMDF026	<RSCAN0_base> + $11A8_H$
RS-CAN0	Transmit buffer data field 1 register 26	RSCAN0TMDF126	<RSCAN0_base> + $11AC_H$
RS-CAN0	Transmit buffer ID register 27	RSCAN0TMID27	<RSCAN0_base> + $11B0_H$
RS-CAN0	Transmit buffer pointer register 27	RSCAN0TMPTR27	<RSCAN0_base> + $11B4_H$
RS-CAN0	Transmit buffer data field 0 register 27	RSCAN0TMDF027	<RSCAN0_base> + $11B8_H$
RS-CAN0	Transmit buffer data field 1 register 27	RSCAN0TMDF127	<RSCAN0_base> + $11BC_H$
RS-CAN0	Transmit buffer ID register 28	RSCAN0TMID28	<RSCAN0_base> + $11C0_H$
RS-CAN0	Transmit buffer pointer register 28	RSCAN0TMPTR28	<RSCAN0_base> + $11C4_H$
RS-CAN0	Transmit buffer data field 0 register 28	RSCAN0TMDF028	<RSCAN0_base> + $11C8_H$
RS-CAN0	Transmit buffer data field 1 register 28	RSCAN0TMDF128	<RSCAN0_base> + $11CC_H$
RS-CAN0	Transmit buffer ID register 29	RSCAN0TMID29	<RSCAN0_base> + $11D0_H$
RS-CAN0	Transmit buffer pointer register 29	RSCAN0TMPTR29	<RSCAN0_base> + $11D4_H$
RS-CAN0	Transmit buffer data field 0 register 29	RSCAN0TMDF029	<RSCAN0_base> + $11D8_H$
RS-CAN0	Transmit buffer data field 1 register 29	RSCAN0TMDF129	<RSCAN0_base> + $11DC_H$
RS-CAN0	Transmit buffer ID register 30	RSCAN0TMID30	<RSCAN0_base> + $11E0_H$
RS-CAN0	Transmit buffer pointer register 30	RSCAN0TMPTR30	<RSCAN0_base> + $11E4_H$
RS-CAN0	Transmit buffer data field 0 register 30	RSCAN0TMDF030	<RSCAN0_base> + $11E8_H$
RS-CAN0	Transmit buffer data field 1 register 30	RSCAN0TMDF130	<RSCAN0_base> + $11EC_H$
RS-CAN0	Transmit buffer ID register 31	RSCAN0TMID31	<RSCAN0_base> + $11F0_H$
RS-CAN0	Transmit buffer pointer register 31	RSCAN0TMPTR31	<RSCAN0_base> + $11F4_H$
RS-CAN0	Transmit buffer data field 0 register 31	RSCAN0TMDF031	<RSCAN0_base> + $11F8_H$
RS-CAN0	Transmit buffer data field 1 register 31	RSCAN0TMDF131	<RSCAN0_base> + $11FC_H$

Table 14.10 Registers (19/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer ID register 32	RSCAN0TMID32	<RSCAN0_base> + 1200_H
RS-CAN0	Transmit buffer pointer register 32	RSCAN0TMPTR32	<RSCAN0_base> + 1204_H
RS-CAN0	Transmit buffer data field 0 register 32	RSCAN0TMDF032	<RSCAN0_base> + 1208_H
RS-CAN0	Transmit buffer data field 1 register 32	RSCAN0TMDF132	<RSCAN0_base> + $120C_H$
RS-CAN0	Transmit buffer ID register 33	RSCAN0TMID33	<RSCAN0_base> + 1210_H
RS-CAN0	Transmit buffer pointer register 33	RSCAN0TMPTR33	<RSCAN0_base> + 1214_H
RS-CAN0	Transmit buffer data field 0 register 33	RSCAN0TMDF033	<RSCAN0_base> + 1218_H
RS-CAN0	Transmit buffer data field 1 register 33	RSCAN0TMDF133	<RSCAN0_base> + $121C_H$
RS-CAN0	Transmit buffer ID register 34	RSCAN0TMID34	<RSCAN0_base> + 1220_H
RS-CAN0	Transmit buffer pointer register 34	RSCAN0TMPTR34	<RSCAN0_base> + 1224_H
RS-CAN0	Transmit buffer data field 0 register 34	RSCAN0TMDF034	<RSCAN0_base> + 1228_H
RS-CAN0	Transmit buffer data field 1 register 34	RSCAN0TMDF134	<RSCAN0_base> + $122C_H$
RS-CAN0	Transmit buffer ID register 35	RSCAN0TMID35	<RSCAN0_base> + 1230_H
RS-CAN0	Transmit buffer pointer register 35	RSCAN0TMPTR35	<RSCAN0_base> + 1234_H
RS-CAN0	Transmit buffer data field 0 register 35	RSCAN0TMDF035	<RSCAN0_base> + 1238_H
RS-CAN0	Transmit buffer data field 1 register 35	RSCAN0TMDF135	<RSCAN0_base> + $123C_H$
RS-CAN0	Transmit buffer ID register 36	RSCAN0TMID36	<RSCAN0_base> + 1240_H
RS-CAN0	Transmit buffer pointer register 36	RSCAN0TMPTR36	<RSCAN0_base> + 1244_H
RS-CAN0	Transmit buffer data field 0 register 36	RSCAN0TMDF036	<RSCAN0_base> + 1248_H
RS-CAN0	Transmit buffer data field 1 register 36	RSCAN0TMDF136	<RSCAN0_base> + $124C_H$
RS-CAN0	Transmit buffer ID register 37	RSCAN0TMID37	<RSCAN0_base> + 1250_H
RS-CAN0	Transmit buffer pointer register 37	RSCAN0TMPTR37	<RSCAN0_base> + 1254_H
RS-CAN0	Transmit buffer data field 0 register 37	RSCAN0TMDF037	<RSCAN0_base> + 1258_H
RS-CAN0	Transmit buffer data field 1 register 37	RSCAN0TMDF137	<RSCAN0_base> + $125C_H$
RS-CAN0	Transmit buffer ID register 38	RSCAN0TMID38	<RSCAN0_base> + 1260_H
RS-CAN0	Transmit buffer pointer register 38	RSCAN0TMPTR38	<RSCAN0_base> + 1264_H
RS-CAN0	Transmit buffer data field 0 register 38	RSCAN0TMDF038	<RSCAN0_base> + 1268_H
RS-CAN0	Transmit buffer data field 1 register 38	RSCAN0TMDF138	<RSCAN0_base> + $126C_H$
RS-CAN0	Transmit buffer ID register 39	RSCAN0TMID39	<RSCAN0_base> + 1270_H
RS-CAN0	Transmit buffer pointer register 39	RSCAN0TMPTR39	<RSCAN0_base> + 1274_H
RS-CAN0	Transmit buffer data field 0 register 39	RSCAN0TMDF039	<RSCAN0_base> + 1278_H
RS-CAN0	Transmit buffer data field 1 register 39	RSCAN0TMDF139	<RSCAN0_base> + $127C_H$
RS-CAN0	Transmit buffer ID register 40	RSCAN0TMID40	<RSCAN0_base> + 1280_H
RS-CAN0	Transmit buffer pointer register 40	RSCAN0TMPTR40	<RSCAN0_base> + 1284_H
RS-CAN0	Transmit buffer data field 0 register 40	RSCAN0TMDF040	<RSCAN0_base> + 1288_H
RS-CAN0	Transmit buffer data field 1 register 40	RSCAN0TMDF140	<RSCAN0_base> + $128C_H$
RS-CAN0	Transmit buffer ID register 41	RSCAN0TMID41	<RSCAN0_base> + 1290_H
RS-CAN0	Transmit buffer pointer register 41	RSCAN0TMPTR41	<RSCAN0_base> + 1294_H
RS-CAN0	Transmit buffer data field 0 register 41	RSCAN0TMDF041	<RSCAN0_base> + 1298_H
RS-CAN0	Transmit buffer data field 1 register 41	RSCAN0TMDF141	<RSCAN0_base> + $129C_H$
RS-CAN0	Transmit buffer ID register 42	RSCAN0TMID42	<RSCAN0_base> + $12A0_H$
RS-CAN0	Transmit buffer pointer register 42	RSCAN0TMPTR42	<RSCAN0_base> + $12A4_H$
RS-CAN0	Transmit buffer data field 0 register 42	RSCAN0TMDF042	<RSCAN0_base> + $12A8_H$
RS-CAN0	Transmit buffer data field 1 register 42	RSCAN0TMDF142	<RSCAN0_base> + $12AC_H$

Table 14.10 Registers (20/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer ID register 43	RSCAN0TMID43	<RSCAN0_base> + $12B0_H$
RS-CAN0	Transmit buffer pointer register 43	RSCAN0TMPTR43	<RSCAN0_base> + $12B4_H$
RS-CAN0	Transmit buffer data field 0 register 43	RSCAN0TMDF043	<RSCAN0_base> + $12B8_H$
RS-CAN0	Transmit buffer data field 1 register 43	RSCAN0TMDF143	<RSCAN0_base> + $12BC_H$
RS-CAN0	Transmit buffer ID register 44	RSCAN0TMID44	<RSCAN0_base> + $12C0_H$
RS-CAN0	Transmit buffer pointer register 44	RSCAN0TMPTR44	<RSCAN0_base> + $12C4_H$
RS-CAN0	Transmit buffer data field 0 register 44	RSCAN0TMDF044	<RSCAN0_base> + $12C8_H$
RS-CAN0	Transmit buffer data field 1 register 44	RSCAN0TMDF144	<RSCAN0_base> + $12CC_H$
RS-CAN0	Transmit buffer ID register 45	RSCAN0TMID45	<RSCAN0_base> + $12D0_H$
RS-CAN0	Transmit buffer pointer register 45	RSCAN0TMPTR45	<RSCAN0_base> + $12D4_H$
RS-CAN0	Transmit buffer data field 0 register 45	RSCAN0TMDF045	<RSCAN0_base> + $12D8_H$
RS-CAN0	Transmit buffer data field 1 register 45	RSCAN0TMDF145	<RSCAN0_base> + $12DC_H$
RS-CAN0	Transmit buffer ID register 46	RSCAN0TMID46	<RSCAN0_base> + $12E0_H$
RS-CAN0	Transmit buffer pointer register 46	RSCAN0TMPTR46	<RSCAN0_base> + $12E4_H$
RS-CAN0	Transmit buffer data field 0 register 46	RSCAN0TMDF046	<RSCAN0_base> + $12E8_H$
RS-CAN0	Transmit buffer data field 1 register 46	RSCAN0TMDF146	<RSCAN0_base> + $12EC_H$
RS-CAN0	Transmit buffer ID register 47	RSCAN0TMID47	<RSCAN0_base> + $12F0_H$
RS-CAN0	Transmit buffer pointer register 47	RSCAN0TMPTR47	<RSCAN0_base> + $12F4_H$
RS-CAN0	Transmit buffer data field 0 register 47	RSCAN0TMDF047	<RSCAN0_base> + $12F8_H$
RS-CAN0	Transmit buffer data field 1 register 47	RSCAN0TMDF147	<RSCAN0_base> + $12FC_H$
RS-CAN0	Transmit buffer ID register 48	RSCAN0TMID48	<RSCAN0_base> + 1300_H
RS-CAN0	Transmit buffer pointer register 48	RSCAN0TMPTR48	<RSCAN0_base> + 1304_H
RS-CAN0	Transmit buffer data field 0 register 48	RSCAN0TMDF048	<RSCAN0_base> + 1308_H
RS-CAN0	Transmit buffer data field 1 register 48	RSCAN0TMDF148	<RSCAN0_base> + $130C_H$
RS-CAN0	Transmit buffer ID register 49	RSCAN0TMID49	<RSCAN0_base> + 1310_H
RS-CAN0	Transmit buffer pointer register 49	RSCAN0TMPTR49	<RSCAN0_base> + 1314_H
RS-CAN0	Transmit buffer data field 0 register 49	RSCAN0TMDF049	<RSCAN0_base> + 1318_H
RS-CAN0	Transmit buffer data field 1 register 49	RSCAN0TMDF149	<RSCAN0_base> + $131C_H$
RS-CAN0	Transmit buffer ID register 50	RSCAN0TMID50	<RSCAN0_base> + 1320_H
RS-CAN0	Transmit buffer pointer register 50	RSCAN0TMPTR50	<RSCAN0_base> + 1324_H
RS-CAN0	Transmit buffer data field 0 register 50	RSCAN0TMDF050	<RSCAN0_base> + 1328_H
RS-CAN0	Transmit buffer data field 1 register 50	RSCAN0TMDF150	<RSCAN0_base> + $132C_H$
RS-CAN0	Transmit buffer ID register 51	RSCAN0TMID51	<RSCAN0_base> + 1330_H
RS-CAN0	Transmit buffer pointer register 51	RSCAN0TMPTR51	<RSCAN0_base> + 1334_H
RS-CAN0	Transmit buffer data field 0 register 51	RSCAN0TMDF051	<RSCAN0_base> + 1338_H
RS-CAN0	Transmit buffer data field 1 register 51	RSCAN0TMDF151	<RSCAN0_base> + $133C_H$
RS-CAN0	Transmit buffer ID register 52	RSCAN0TMID52	<RSCAN0_base> + 1340_H
RS-CAN0	Transmit buffer pointer register 52	RSCAN0TMPTR52	<RSCAN0_base> + 1344_H
RS-CAN0	Transmit buffer data field 0 register 52	RSCAN0TMDF052	<RSCAN0_base> + 1348_H
RS-CAN0	Transmit buffer data field 1 register 52	RSCAN0TMDF152	<RSCAN0_base> + $134C_H$
RS-CAN0	Transmit buffer ID register 53	RSCAN0TMID53	<RSCAN0_base> + 1350_H
RS-CAN0	Transmit buffer pointer register 53	RSCAN0TMPTR53	<RSCAN0_base> + 1354_H
RS-CAN0	Transmit buffer data field 0 register 53	RSCAN0TMDF053	<RSCAN0_base> + 1358_H
RS-CAN0	Transmit buffer data field 1 register 53	RSCAN0TMDF153	<RSCAN0_base> + $135C_H$

Table 14.10 Registers (21/23)

Module	Register	Symbol	Address
RS-CAN0	Transmit buffer ID register 54	RSCAN0TMID54	<RSCAN0_base> + 1360_H
RS-CAN0	Transmit buffer pointer register 54	RSCAN0TMPTR54	<RSCAN0_base> + 1364_H
RS-CAN0	Transmit buffer data field 0 register 54	RSCAN0TMDF054	<RSCAN0_base> + 1368_H
RS-CAN0	Transmit buffer data field 1 register 54	RSCAN0TMDF154	<RSCAN0_base> + $136C_H$
RS-CAN0	Transmit buffer ID register 55	RSCAN0TMID55	<RSCAN0_base> + 1370_H
RS-CAN0	Transmit buffer pointer register 55	RSCAN0TMPTR55	<RSCAN0_base> + 1374_H
RS-CAN0	Transmit buffer data field 0 register 55	RSCAN0TMDF055	<RSCAN0_base> + 1378_H
RS-CAN0	Transmit buffer data field 1 register 55	RSCAN0TMDF155	<RSCAN0_base> + $137C_H$
RS-CAN0	Transmit buffer ID register 56	RSCAN0TMID56	<RSCAN0_base> + 1380_H
RS-CAN0	Transmit buffer pointer register 56	RSCAN0TMPTR56	<RSCAN0_base> + 1384_H
RS-CAN0	Transmit buffer data field 0 register 56	RSCAN0TMDF056	<RSCAN0_base> + 1388_H
RS-CAN0	Transmit buffer data field 1 register 56	RSCAN0TMDF156	<RSCAN0_base> + $138C_H$
RS-CAN0	Transmit buffer ID register 57	RSCAN0TMID57	<RSCAN0_base> + 1390_H
RS-CAN0	Transmit buffer pointer register 57	RSCAN0TMPTR57	<RSCAN0_base> + 1394_H
RS-CAN0	Transmit buffer data field 0 register 57	RSCAN0TMDF057	<RSCAN0_base> + 1398_H
RS-CAN0	Transmit buffer data field 1 register 57	RSCAN0TMDF157	<RSCAN0_base> + $139C_H$
RS-CAN0	Transmit buffer ID register 58	RSCAN0TMID58	<RSCAN0_base> + $13A0_H$
RS-CAN0	Transmit buffer pointer register 58	RSCAN0TMPTR58	<RSCAN0_base> + $13A4_H$
RS-CAN0	Transmit buffer data field 0 register 58	RSCAN0TMDF058	<RSCAN0_base> + $13A8_H$
RS-CAN0	Transmit buffer data field 1 register 58	RSCAN0TMDF158	<RSCAN0_base> + $13AC_H$
RS-CAN0	Transmit buffer ID register 59	RSCAN0TMID59	<RSCAN0_base> + $13B0_H$
RS-CAN0	Transmit buffer pointer register 59	RSCAN0TMPTR59	<RSCAN0_base> + $13B4_H$
RS-CAN0	Transmit buffer data field 0 register 59	RSCAN0TMDF059	<RSCAN0_base> + $13B8_H$
RS-CAN0	Transmit buffer data field 1 register 59	RSCAN0TMDF159	<RSCAN0_base> + $13BC_H$
RS-CAN0	Transmit buffer ID register 60	RSCAN0TMID60	<RSCAN0_base> + $13C0_H$
RS-CAN0	Transmit buffer pointer register 60	RSCAN0TMPTR60	<RSCAN0_base> + $13C4_H$
RS-CAN0	Transmit buffer data field 0 register 60	RSCAN0TMDF060	<RSCAN0_base> + $13C8_H$
RS-CAN0	Transmit buffer data field 1 register 60	RSCAN0TMDF160	<RSCAN0_base> + $13CC_H$
RS-CAN0	Transmit buffer ID register 61	RSCAN0TMID61	<RSCAN0_base> + $13D0_H$
RS-CAN0	Transmit buffer pointer register 61	RSCAN0TMPTR61	<RSCAN0_base> + $13D4_H$
RS-CAN0	Transmit buffer data field 0 register 61	RSCAN0TMDF061	<RSCAN0_base> + $13D8_H$
RS-CAN0	Transmit buffer data field 1 register 1	RSCAN0TMDF161	<RSCAN0_base> + $13DC_H$
RS-CAN0	Transmit buffer ID register 62	RSCAN0TMID62	<RSCAN0_base> + $13E0_H$
RS-CAN0	Transmit buffer pointer register 62	RSCAN0TMPTR62	<RSCAN0_base> + $13E4_H$
RS-CAN0	Transmit buffer data field 0 register 62	RSCAN0TMDF062	<RSCAN0_base> + $13E8_H$
RS-CAN0	Transmit buffer data field 1 register 62	RSCAN0TMDF162	<RSCAN0_base> + $13EC_H$
RS-CAN0	Transmit buffer ID register 63	RSCAN0TMID63	<RSCAN0_base> + $13F0_H$
RS-CAN0	Transmit buffer pointer register 63	RSCAN0TMPTR63	<RSCAN0_base> + $13F4_H$
RS-CAN0	Transmit buffer data field 0 register 63	RSCAN0TMDF063	<RSCAN0_base> + $13F8_H$
RS-CAN0	Transmit buffer data field 1 register 63	RSCAN0TMDF163	<RSCAN0_base> + $13FC_H$
RS-CAN0	Transmit history access register 0	RSCAN0THLACC0	<RSCAN0_base> + 1800_H
RS-CAN0	Transmit history access register 1	RSCAN0THLACC1	<RSCAN0_base> + 1804_H
RS-CAN0	Transmit history access register 2	RSCAN0THLACC2	<RSCAN0_base> + 1808_H
RS-CAN0	Transmit history access register 3	RSCAN0THLACC3	<RSCAN0_base> + $180C_H$

Table 14.10 Registers (22/23)

Module	Register	Symbol	Address
RS-CAN0	RAM test page access register 0	RSCAN0RPGACC0	<RSCAN0_base> + 1900_H
RS-CAN0	RAM test page access register 1	RSCAN0RPGACC1	<RSCAN0_base> + 1904_H
RS-CAN0	RAM test page access register 2	RSCAN0RPGACC2	<RSCAN0_base> + 1908_H
RS-CAN0	RAM test page access register 3	RSCAN0RPGACC3	<RSCAN0_base> + $190C_H$
RS-CAN0	RAM test page access register 4	RSCAN0RPGACC4	<RSCAN0_base> + 1910_H
RS-CAN0	RAM test page access register 5	RSCAN0RPGACC5	<RSCAN0_base> + 1914_H
RS-CAN0	RAM test page access register 6	RSCAN0RPGACC6	<RSCAN0_base> + 1918_H
RS-CAN0	RAM test page access register 7	RSCAN0RPGACC7	<RSCAN0_base> + $191C_H$
RS-CAN0	RAM test page access register 8	RSCAN0RPGACC8	<RSCAN0_base> + 1920_H
RS-CAN0	RAM test page access register 9	RSCAN0RPGACC9	<RSCAN0_base> + 1924_H
RS-CAN0	RAM test page access register 10	RSCAN0RPGACC10	<RSCAN0_base> + 1928_H
RS-CAN0	RAM test page access register 11	RSCAN0RPGACC11	<RSCAN0_base> + $192C_H$
RS-CAN0	RAM test page access register 12	RSCAN0RPGACC12	<RSCAN0_base> + 1930_H
RS-CAN0	RAM test page access register 13	RSCAN0RPGACC13	<RSCAN0_base> + 1934_H
RS-CAN0	RAM test page access register 14	RSCAN0RPGACC14	<RSCAN0_base> + 1938_H
RS-CAN0	RAM test page access register 15	RSCAN0RPGACC15	<RSCAN0_base> + $193C_H$
RS-CAN0	RAM test page access register 16	RSCAN0RPGACC16	<RSCAN0_base> + 1940_H
RS-CAN0	RAM test page access register 17	RSCAN0RPGACC17	<RSCAN0_base> + 1944_H
RS-CAN0	RAM test page access register 18	RSCAN0RPGACC18	<RSCAN0_base> + 1948_H
RS-CAN0	RAM test page access register 19	RSCAN0RPGACC19	<RSCAN0_base> + $194C_H$
RS-CAN0	RAM test page access register 20	RSCAN0RPGACC20	<RSCAN0_base> + 1950_H
RS-CAN0	RAM test page access register 21	RSCAN0RPGACC21	<RSCAN0_base> + 1954_H
RS-CAN0	RAM test page access register 22	RSCAN0RPGACC22	<RSCAN0_base> + 1958_H
RS-CAN0	RAM test page access register 23	RSCAN0RPGACC23	<RSCAN0_base> + $195C_H$
RS-CAN0	RAM test page access register 24	RSCAN0RPGACC24	<RSCAN0_base> + 1960_H
RS-CAN0	RAM test page access register 25	RSCAN0RPGACC25	<RSCAN0_base> + 1964_H
RS-CAN0	RAM test page access register 26	RSCAN0RPGACC26	<RSCAN0_base> + 1968_H
RS-CAN0	RAM test page access register 27	RSCAN0RPGACC27	<RSCAN0_base> + $196C_H$
RS-CAN0	RAM test page access register 28	RSCAN0RPGACC28	<RSCAN0_base> + 1970_H
RS-CAN0	RAM test page access register 29	RSCAN0RPGACC29	<RSCAN0_base> + 1974_H
RS-CAN0	RAM test page access register 30	RSCAN0RPGACC30	<RSCAN0_base> + 1978_H
RS-CAN0	RAM test page access register 31	RSCAN0RPGACC31	<RSCAN0_base> + $197C_H$
RS-CAN0	RAM test page access register 32	RSCAN0RPGACC32	<RSCAN0_base> + 1980_H
RS-CAN0	RAM test page access register 33	RSCAN0RPGACC33	<RSCAN0_base> + 1984_H
RS-CAN0	RAM test page access register 34	RSCAN0RPGACC34	<RSCAN0_base> + 1988_H
RS-CAN0	RAM test page access register 35	RSCAN0RPGACC35	<RSCAN0_base> + $198C_H$
RS-CAN0	RAM test page access register 36	RSCAN0RPGACC36	<RSCAN0_base> + 1990_H
RS-CAN0	RAM test page access register 37	RSCAN0RPGACC37	<RSCAN0_base> + 1994_H
RS-CAN0	RAM test page access register 38	RSCAN0RPGACC38	<RSCAN0_base> + 1998_H
RS-CAN0	RAM test page access register 39	RSCAN0RPGACC39	<RSCAN0_base> + $199C_H$
RS-CAN0	RAM test page access register 40	RSCAN0RPGACC40	<RSCAN0_base> + $19A0_H$
RS-CAN0	RAM test page access register 41	RSCAN0RPGACC41	<RSCAN0_base> + $19A4_H$
RS-CAN0	RAM test page access register 42	RSCAN0RPGACC42	<RSCAN0_base> + $19A8_H$
RS-CAN0	RAM test page access register 43	RSCAN0RPGACC43	<RSCAN0_base> + $19AC_H$

Table 14.10 Registers (23/23)

Module	Register	Symbol	Address
RS-CAN0	RAM test page access register 44	RSCAN0RPGACC44	<RSCAN0_base> + $19B0_H$
RS-CAN0	RAM test page access register 45	RSCAN0RPGACC45	<RSCAN0_base> + $19B4_H$
RS-CAN0	RAM test page access register 46	RSCAN0RPGACC46	<RSCAN0_base> + $19B8_H$
RS-CAN0	RAM test page access register 47	RSCAN0RPGACC47	<RSCAN0_base> + $19BC_H$
RS-CAN0	RAM test page access register 48	RSCAN0RPGACC48	<RSCAN0_base> + $19C0_H$
RS-CAN0	RAM test page access register 49	RSCAN0RPGACC49	<RSCAN0_base> + $19C4_H$
RS-CAN0	RAM test page access register 50	RSCAN0RPGACC50	<RSCAN0_base> + $19C8_H$
RS-CAN0	RAM test page access register 51	RSCAN0RPGACC51	<RSCAN0_base> + $19CC_H$
RS-CAN0	RAM test page access register 52	RSCAN0RPGACC52	<RSCAN0_base> + $19D0_H$
RS-CAN0	RAM test page access register 53	RSCAN0RPGACC53	<RSCAN0_base> + $19D4_H$
RS-CAN0	RAM test page access register 54	RSCAN0RPGACC54	<RSCAN0_base> + $19D8_H$
RS-CAN0	RAM test page access register 55	RSCAN0RPGACC55	<RSCAN0_base> + $19DC_H$
RS-CAN0	RAM test page access register 56	RSCAN0RPGACC56	<RSCAN0_base> + $19E0_H$
RS-CAN0	RAM test page access register 57	RSCAN0RPGACC57	<RSCAN0_base> + $19E4_H$
RS-CAN0	RAM test page access register 58	RSCAN0RPGACC58	<RSCAN0_base> + $19E8_H$
RS-CAN0	RAM test page access register 59	RSCAN0RPGACC59	<RSCAN0_base> + $19EC_H$
RS-CAN0	RAM test page access register 60	RSCAN0RPGACC60	<RSCAN0_base> + $19F0_H$
RS-CAN0	RAM test page access register 61	RSCAN0RPGACC61	<RSCAN0_base> + $19F4_H$
RS-CAN0	RAM test page access register 62	RSCAN0RPGACC62	<RSCAN0_base> + $19F8_H$
RS-CAN0	RAM test page access register 63	RSCAN0RPGACC63	<RSCAN0_base> + $19FC_H$

Table 14.11 Transmit Buffer p Allocated to Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 14.12 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 14.13 Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 14.14 Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000_B	Setting prohibited
0001_B	Setting prohibited
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

14.3.2 RSCAN0CmCFG — Channel Configuration Register (m = 0 to 3)

Access: RSCAN0CmCFG can be read/written in 32-bit units.
RSCAN0CmCFGL and RSCAN0CmCFGH can be read/written in 16-bit units.
RSCAN0CmCFGLL, RSCAN0CmCFGLH, RSCAN0CmCFGHL, and RSCAN0CmCFGHH can be read/written in 8-bit units.

Address: RSCAN0CmCFG: <RSCAN0_base> + 0000_H + (10_H × m)
RSCAN0CmCFGL: <RSCAN0_base> + 0000_H + (10_H × m),
RSCAN0CmCFGH: <RSCAN0_base> + 0002_H + (10_H × m)
RSCAN0CmCFGLL: <RSCAN0_base> + 0000_H + (10_H × m),
RSCAN0CmCFGLH: <RSCAN0_base> + 0001_H + (10_H × m),
RSCAN0CmCFGHL: <RSCAN0_base> + 0002_H + (10_H × m),
RSCAN0CmCFGHH: <RSCAN0_base> + 0003_H + (10_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW[1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.15 RSCAN0CmCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	SJW[1:0]	Resynchronization Jump Width Control 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	—	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	TSEG2[2:0]	Time Segment 2 Control 0 0 0: Setting prohibited*1 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Table 14.15 RSCAN0CmCFG Register Contents (2/2)

Bit Position	Bit Name	Function
19 to 16	TSEG1[3:0]	Time Segment 1 Control 0 0 0 0: Setting prohibited*1 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 10	—	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	BRP[9:0]	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.

Note 1. Since the value after a reset is “setting prohibited”, be sure to set an appropriate value.

Modify the RSCAN0CmCFG register in channel reset mode or channel halt mode. Set this register before requesting a transition to channel communication mode or channel wait mode. For a description of the bit timing parameters and settings, see **Section 14.5.1, Initial Settings**.

SJW[1:0] Bits

These bits are used to specify a Tq value for the resynchronization jump width. Allowed values are 1 Tq to 4 Tq, inclusive.
Set a value less than or equal to the value of the TSEG2 bits.

TSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase buffer segment 2 (PHASE_SEG2). Allowed values are 2 Tq to 8 Tq, inclusive.
Set a value smaller than the value of the TSEG1 bits.

TSEG1[3:0] Bits

These bits are used to specify a Tq value for the total length of the propagation time segment (PROP_SEG) and phase buffer segment 1 (PHASE_SEG1).
Allowed values are 4 Tq to 16 Tq, inclusive.

BRP[9:0] Bits

The CANmTq (fCANTQn) clock is calculated by the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0]) + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).

14.3.3 RSCAN0CmCTR — Channel Control Register (m = 0 to 3)

Access: RSCAN0CmCTR can be read/written in 32-bit units.
RSCAN0CmCTRL and RSCAN0CmCTRH can be read/written in 16-bit units.
RSCAN0CmCTRLL, RSCAN0CmCTRLH, RSCAN0CmCTRHL, and RSCAN0CmCTRHH can be read/written in 8-bit units.

Address: RSCAN0CmCTR: <RSCAN0_base> + 0004_H + (10_H × m)
RSCAN0CmCTRL: <RSCAN0_base> + 0004_H + (10_H × m),
RSCAN0CmCTRH: <RSCAN0_base> + 0006_H + (10_H × m)
RSCAN0CmCTRLL: <RSCAN0_base> + 0004_H + (10_H × m),
RSCAN0CmCTRLH: <RSCAN0_base> + 0005_H + (10_H × m),
RSCAN0CmCTRHL: <RSCAN0_base> + 0006_H + (10_H × m),
RSCAN0CmCTRHH: <RSCAN0_base> + 0007_H + (10_H × m)

Value after reset: $0000\ 0005_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	—	—	—	TAIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.16 RSCAN0CmCTR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. The write value should always be 0.
26, 25	CTMS[1:0]	Communication Test Mode Select 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error event after bits 14 to 8 in RSCAN0CmERFL are all cleared. 1: Error flags for all error events are displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (during bus-off recovery period) by program request
20 to 17	—	Reserved These bits are always read as 0. The write value should always be 0.
16	TAIE	Transmission Abort Interrupt Enable 0: Transmission abort interrupt is disabled. 1: Transmission abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.

Table 14.16 RSCAN0CmCTR Register Contents (2/2)

Bit Position	Bit Name	Function
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control the display mode of bits 14 to 8 in the RSCAN0CmERFL register.
When this bit is clear to 0, only the flags of the first error event are set to 1. If two or more errors occur in the first error event, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.

Modify this bit only in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select the bus off recovery mode of the RS-CAN module.

When the BOM[1:0] bits are set to 00_B, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CAN module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10_B (channel halt mode) before recessive bits are detected 128 times, the RS-CAN module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 01_B, the CHMDC[1:0] bits in the RSCAN0CmCTR register (m = 0 to 3) are set to 10_B and the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated and the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are cleared to 00_H.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 10_B, the CHMDC[1:0] bits are set to 10_B and the RS-CAN module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H.

When the BOM[1:0] bits are set to 11_B and the CHMDC[1:0] bits are set to 10_B while the RS-CAN module is in the bus off state, the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CAN module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10_B, a bus off recovery interrupt request is generated.

If the CPU requests a transition to channel reset mode at the same time as the RS-CAN module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01_B or at bus off end when the BOM[1:0] bits are 10_B), the CPU request takes precedence. Modify the BOM bits only in channel reset mode.

TAIE Bit

When transmission abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

ALIE Bit

When the ALF flag in the RSCAN0CmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BLIE Bit

When the BLF flag in the RSCAN0CmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCAN0CmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BORIE Bit

When the BORF flag in the RSCAN0CmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCAN0CmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EPIE Bit

When the EPF flag in the RSCAN0CmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

EWIE Bit

When the EWF flag in the RSCAN0CmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

BEIE Bit

When the BEF flag in the RSCAN0CmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

RTBO Bit

Setting this bit to 1 in the bus off state forcibly return the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register to 00_H and also clears the BOSTS flag in the RSCAN0CmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from then bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCAN0CmCTR register are 00_B (ISO11898-1 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel into channel stop mode.

Clearing this bit to 0 makes the channel exit channel stop mode.

Modify this bit from 0 to 1 only in channel reset mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 14.4.2.2, Channel Modes**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11_B. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically becomes 10_B.

14.3.4 RSCAN0CmSTS — Channel Status Register (m = 0 to 3)

Access: RSCAN0CmSTS is read-only in 32-bit units.
RSCAN0CmSTSL and RSCAN0CmSTSH are read-only in 16-bit units.
RSCAN0CmSTSLL, RSCAN0CmSTSLH, RSCAN0CmSTSHL, and RSCAN0CmSTSHH are read-only in 8-bit units.

Address: RSCAN0CmSTS: <RSCAN0_base> + 0008_H + (10_H × m)
RSCAN0CmSTSL: <RSCAN0_base> + 0008_H + (10_H × m),
RSCAN0CmSTSH: <RSCAN0_base> + $000A_H$ + (10_H × m)
RSCAN0CmSTSLL: <RSCAN0_base> + 0008_H + (10_H × m),
RSCAN0CmSTSLH: <RSCAN0_base> + 0009_H + (10_H × m),
RSCAN0CmSTSHL: <RSCAN0_base> + $000A_H$ + (10_H × m),
RSCAN0CmSTSHH: <RSCAN0_base> + $000B_H$ + (10_H × m)

Value after reset: $0000\ 0005_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COM STS	REC STS	TRM STS	BOSTS	EPSTS	CSLP STS	CHLT STS	CRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.17 RSCAN0CmSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read. Counter Value: 00_H to FF_H
23 to 16	REC[7:0]	The receive error counter (REC) can be read. Counter Value: 00_H to FF_H
15 to 8	—	Reserved These bits are always read as 0.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

COMSTS Flag

This bit indicates that communication is ready.

This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the bus off state (TEC[7:0] > 255) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the RS-CAN module has entered the error passive state ((128 ≤ TEC[7:0] ≤ 255) or (128 ≤ REC[7:0] ≤ 255)), It is cleared to 0 when the RS-CAN module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

14.3.5 RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 3)

Access: RSCAN0CmERFL can be read/written in 32-bit units.
RSCAN0CmERFLL and RSCAN0CmERFLH can be read/written in 16-bit units.
RSCAN0CmERFLLL, RSCAN0CmERFLLH, RSCAN0CmERFLHL, and RSCAN0CmERFLHH can be read/written in 8-bit units.

Address: RSCAN0CmERFL: <RSCAN0_base> + $000C_H$ + (10_H × m)
RSCAN0CmERFLL: <RSCAN0_base> + $000C_H$ + (10_H × m),
RSCAN0CmERFLH: <RSCAN0_base> + $000E_H$ + (10_H × m)
RSCAN0CmERFLLL: <RSCAN0_base> + $000C_H$ + (10_H × m),
RSCAN0CmERFLLH: <RSCAN0_base> + $000D_H$ + (10_H × m),
RSCAN0CmERFLHL: <RSCAN0_base> + $000E_H$ + (10_H × m),
RSCAN0CmERFLHH: <RSCAN0_base> + $000F_H$ + (10_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.18 RSCAN0CmERFL Register Contents (1/2)

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	CRCREG[14:0]	CRC Calculation Data A CRC value calculated based on the transmit message or receive message is indicated.
15	—	Reserved This bit is always read as 0. The write value should always be 0.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.

Table 14.18 RSCAN0CmERFL Register Contents (2/2)

Bit Position	Bit Name	Function
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	Bus Lock Flag 0: No Bus lock is detected. 1: Bus lock is detected
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No bus error is detected. 1: Bus Error is detected.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these error occurs at the same time that the program writes 0 to the flag, the flag is set to 1. The channel reset mode transition clears all of these flags to 0. If the ERRD bit in the RSCAN0CmCTR register is set to 0 (i.e., only the flags from the first error event are displayed) and an error related to bits 14 to 8 of RSCAN0CmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the time the error occurred.

CRCREG[14:0] Bits

When the CTME bit in the RSCAN0CmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration-lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, the detection processing restarts either when.

- a recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- the CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01_B (channel reset mode).
- The RTBO bit in the RSCAN0CmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCAN0CmCTR register are set to 01_B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 10_B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11_B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCAN0CmCTR register (m = 0 to 3) set to 01_B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).

This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCAN0CmERFL register is set to 1.

NOTE

To clear any of the flags of this register, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

14.3.6 RSCAN0GCFG — Global Configuration Register

Access: RSCAN0GCFG can be read/written in 32-bit units.
RSCAN0GCFGL and RSCAN0GCFGH can be read/written in 16-bit units.
RSCAN0GCFGLL, RSCAN0GCFGLH, RSCAN0GCFGHL, and RSCAN0GCFGHH can be read/written in 8-bit units.

Address: RSCAN0GCFG: <RSCAN0_base> + 0084_H
RSCAN0GCFGL: <RSCAN0_base> + 0084_H, RSCAN0GCFGH: <RSCAN0_base> + 0086_H
RSCAN0GCFGLL: <RSCAN0_base> + 0084_H, RSCAN0GCFGLH: <RSCAN0_base> + 0085_H,
RSCAN0GCFGHL: <RSCAN0_base> + 0086_H, RSCAN0GCFGHH: <RSCAN0_base> + 0087_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	—	DCS	MME	DRE	DCE	TPRI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 14.19 RSCAN0GCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to q, the pclk is divided by q. Setting 0000_H is prohibited.*3
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select 0 0 0: Channel 0 bit time clock 0 0 1: Channel 1 bit time clock 0 1 0: Channel 2 bit time clock 0 1 1: Channel 3 bit time clock 1 0 0: Setting prohibited 1 0 1: Setting prohibited 1 1 0: Setting prohibited 1 1 1: Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Bit time clock
11 to 8	TSP[3:0]	Timestamp Clock Source Division 0 0 0 0: Not divided 0 0 0 1: Divided by 2 0 0 1 0: Divided by 4 0 0 1 1: Divided by 8 0 1 0 0: Divided by 16 0 1 0 1: Divided by 32 0 1 1 0: Divided by 64 0 1 1 1: Divided by 128 1 0 0 0: Divided by 256 1 0 0 1: Divided by 512 1 0 1 0: Divided by 1024 1 0 1 1: Divided by 2048 1 1 0 0: Divided by 4096 1 1 0 1: Divided by 8192 1 1 1 0: Divided by 16384 1 1 1 1: Divided by 32768
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.

Table 14.19 RSCAN0GCFG Register Contents (2/2)

Bit Position	Bit Name	Function
4	DCS	CAN Clock Source Select*2 0: clkc 1: clk_xincan
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying pclk/2 as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.
Note 2. The frequency of the CAN clock must be no greater than pclk/2.
Note 3. Since the value after a reset is “setting prohibited”, be sure to set an appropriate value.

Modify the RSCAN0GCFG register only in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. Refer to **(1), Interval Transmission Function**

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

TSSS Bit

This bit is used to select a clock source of the timestamp counter.

TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (f_{CAN}).

When this bit is set to 1, clk_xincan is used as the clock source of the CAN clock (f_{CAN}).

The frequency of the CAN clock (f_{CAN}) must be no greater than pclk/2.

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00_H is stored in each data byte beyond the first n bytes, where n is the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

DCE Bit

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCAN0GAFLP0j register to 0000_B before clearing the DCE bit in the RSCAN0GCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those with pending messages has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

14.3.7 RSCAN0GCTR — Global Control Register

Access: RSCAN0GCTR can be read/written in 32-bit units.
RSCAN0GCTRL and RSCAN0GCTRH can be read/written in 16-bit units.
RSCAN0GCTRLL, RSCAN0GCTRLH, RSCAN0GCTRHL, and RSCAN0GCTRHH can be read/written in 8-bit units.

Address: RSCAN0GCTR: <RSCAN0_base> + 0088_H
RSCAN0GCTRL: <RSCAN0_base> + 0088_H, RSCAN0GCTRH: <RSCAN0_base> + $008A_H$
RSCAN0GCTRLL: <RSCAN0_base> + 0088_H, RSCAN0GCTRLH: <RSCAN0_base> + 0089_H,
RSCAN0GCTRHL: <RSCAN0_base> + $008A_H$, RSCAN0GCTRHH: <RSCAN0_base> + $008B_H$

Value after reset: $0000\ 0005_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.20 RSCAN0GCTR Register Contents

Bit Position	Bit Name	Function
31 to 17	—	Reserved These bits are always read as 0. The write value should always be 0.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	—	Reserved These bits are always read as 0. The write value should always be 0.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCAN0GTSC register is cleared to 0000_H.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

GSLPR Bit

When the RCAN module is in global reset mode, setting this bit to 1 places the RCAN module into global stop mode.

Clearing this bit to 0 makes the RCAN module leave from global stop mode.

Modify this bit only in global reset mode.

GMDC[1:0] Bits

These bits are used to select the mode of entire RS-CAN module (global operating mode, global reset mode, or global test mode). For details, see **Section 14.4.2.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CAN module into global stop mode.

14.3.8 RSCAN0GSTS — Global Status Register

Access: RSCAN0GSTS is read-only in 32-bit units.
RSCAN0GSTSL and RSCAN0GSTSH are read-only in 16-bit units.
RSCAN0GSTSLL, RSCAN0GSTSLH, RSCAN0GSTSHL, and RSCAN0GSTSHH are read-only in 8-bit units.

Address: RSCAN0GSTS: <RSCAN0_base> + $008C_H$
RSCAN0GSTSL: <RSCAN0_base> + $008C_H$, RSCAN0GSTSH: <RSCAN0_base> + $008E_H$
RSCAN0GSTSLL: <RSCAN0_base> + $008C_H$, RSCAN0GSTSLH: <RSCAN0_base> + $008D_H$,
RSCAN0GSTSHL: <RSCAN0_base> + $008E_H$, RSCAN0GSTSHH: <RSCAN0_base> + $008F_H$

Value after reset: $0000\ 000D_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAM INIT	GSLP STS	GHLT STS	GRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.21 RSCAN0GSTS Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

14.3.9 RSCAN0GERFL — Global Error Flag Register

Access: RSCAN0GERFL can be read/written in 32-bit units.
RSCAN0GERFLL and RSCAN0GERFLH can be read/written in 16-bit units.
RSCAN0GERFLLL, RSCAN0GERFLLH, RSCAN0GERFLHL, and RSCAN0GERFLHH can be read/written in 8-bit units.

Address: RSCAN0GERFL: <RSCAN0_base> + 0090_H
RSCAN0GERFLL: <RSCAN0_base> + 0090_H, RSCAN0GERFLH: <RSCAN0_base> + 0092_H
RSCAN0GERFLLL: <RSCAN0_base> + 0090_H, RSCAN0GERFLLH: <RSCAN0_base> + 0091_H,
RSCAN0GERFLHL: <RSCAN0_base> + 0092_H, RSCAN0GERFLHH: <RSCAN0_base> + 0093_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.22 RSCAN0GERFL Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved The read value is undefined. The write value should always be 0.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCAN0GERFL register are cleared to 0 in global reset mode.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCAN0THLSTSm register (m = 0 to 3) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCAN0RFSTSm register (m = 0 to 7) or the CFMLT flags in the RSCAN0CFSTSk register (k = 0 to 11) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

14.3.10 RSCAN0GTINTSTS0 — Global TX Interrupt Status Register 0

Access: RSCAN0GTINTSTS0 is read-only in 32-bit units.
RSCAN0GTINTSTS0L and RSCAN0GTINTSTS0H are read-only in 16-bit units.
RSCAN0GTINTSTS0LL, RSCAN0GTINTSTS0LH, RSCAN0GTINTSTS0HL, and RSCAN0GTINTSTS0HH are read-only in 8-bit units.

Address: RSCAN0GTINTSTS0: <RSCAN0_base> + 0460_H
RSCAN0GTINTSTS0L: <RSCAN0_base> + 0460_H, RSCAN0GTINTSTS0H: <RSCAN0_base> + 0462_H
RSCAN0GTINTSTS0LL: <RSCAN0_base> + 0460_H, RSCAN0GTINTSTS0LH: <RSCAN0_base> + 0461_H,
RSCAN0GTINTSTS0HL: <RSCAN0_base> + 0462_H, RSCAN0GTINTSTS0HH: <RSCAN0_base> + 0463_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	—	—	—	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in global reset or channel reset mode.

Table 14.23 RSCAN0GTINTSTS0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. The write value should always be 0.
28	THIF3	Channel 3 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
27	CFTIF3	Channel 3 Transmit/Receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
26	TQIF3	Channel 3 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
25	TAIF3	Channel 3 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
24	TSIF3	Channel 3 Transmit Buffer Transmission Complete Interrupt Status Flag 0: Transmit buffer transmission complete interrupt is not requested. 1: Transmit buffer transmission complete interrupt is requested.
23 to 21	—	Reserved These bits are always read as 0. The write value should always be 0.
20	THIF2	Channel 2 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
19	CFTIF2	Channel 2 Transmit/Receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
18	TQIF2	Channel 2 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
17	TAIF2	Channel 2 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.

Table 14.23 RSCAN0GTINTSTS0 Register Contents (2/2)

Bit Position	Bit Name	Function
16	TSIF2	Channel 2 Transmit Buffer Interrupt Status Flag 0: Transmit buffer interrupt is not requested. 1: Transmit buffer interrupt is requested.
15 to 13	—	Reserved These bits are always read as 0. The write value should always be 0.
12	THIF1	Channel 1 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 Transmit/Receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
10	TQIF1	Channel 1 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer transmission abort interrupt is not requested. 1: Transmit buffer transmission abort interrupt is requested.
8	TSIF1	Channel 1 Transmit Buffer Interrupt Status Flag 0: Transmit buffer interrupt is not requested. 1: Transmit buffer interrupt is requested.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	THIF0	Channel 0 Transmit History Interrupt Status Flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 Transmit/Receive FIFO Transmit Interrupt Status Flag 0: Transmit/receive FIFO transmit interrupt is not requested. 1: Transmit/receive FIFO transmit interrupt is requested.
2	TQIF0	Channel 0 Transmit Queue Interrupt Status Flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 Transmit Buffer Abort Interrupt Status Flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF0	Channel 0 Transmit Buffer Transmission Complete Interrupt Status Flag 0: Transmit buffer transmission complete interrupt is not requested. 1: Transmit buffer transmission complete interrupt is requested.

TSIFn Bits

The TSIFn bit is set to 1 when the TMIE bit in the RSCAN0TMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 10_B (transmission completed without abort request) or 11_B (transmission completed with abort request).

When the TMTRF[1:0] flags are cleared to 00_B under the condition that the TSIFn bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIE bit to 0 also clears this flag to 0.

TAIFn Bits

The TAIFn bit is set to 1 when the TAIE bit in the RSCAN0CmCTR register is 1 (transmission abort interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 01_B (transmission abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00_B after the transmission abort is completed.

TQIFn Bits

When the TXQIE bit in the RSCAN0TXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCAN0TXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFn bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCAN0TXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

CFTIFn Bits

When the CFTXIE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO transmission interrupt enabled) and the CFTXIF bit in the RSCAN0CFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFn bit is set to 1.

When all the CFTXIF bits in the RSCAN0CFSTSk register are cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

THIFn Bits

When the THLIE bit in the RSCAN0THLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCAN0THLSTSm register is set to 1 (transmit history interrupt request), the THIFn bit is set to 1.

When the THLIF bit in the RSCAN0THLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

14.3.11 RSCAN0GTSC — Global Timestamp Counter Register

Access: RSCAN0GTSC is read-only in 32-bit units.
RSCAN0GTSCL and RSCAN0GTSCH are read-only in 16-bit units.
RSCAN0GTSCLL, RSCAN0GTSCLH, RSCAN0GTSCHL, and RSCAN0GTSCHH are read-only in 8-bit units.

Address: RSCAN0GTSC: <RSCAN0_base> + 0094_H
RSCAN0GTSCL: <RSCAN0_base> + 0094_H, RSCAN0GTSCH: <RSCAN0_base> + 0096_H
RSCAN0GTSCLL: <RSCAN0_base> + 0094_H, RSCAN0GTSCLH: <RSCAN0_base> + 0095_H,
RSCAN0GTSCHL: <RSCAN0_base> + 0096_H, RSCAN0GTSCHH: <RSCAN0_base> + 0097_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.24 RSCAN0GTSC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter starts and stops counting differently, depending on the count source.

- When the TSSS bit in the RSCAN0GCFG register is 0 (pclk):
  The timestamp counter starts counting when the RCAN module has transitioned to global operating mode.
  This counter stops counting when the RCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.
  This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

14.3.12 RSCAN0GAFLECTR — Receive Rule Entry Control Register

Access: RSCAN0GAFLECTR can be read/written in 32-bit units.
RSCAN0GAFLECTRL and RSCAN0GAFLECTRH can be read/written in 16-bit units.
RSCAN0GAFLECTRLL, RSCAN0GAFLECTRLH, RSCAN0GAFLECTRHL, and RSCAN0GAFLECTRHH can be read/written in 8-bit units.

Address: RSCAN0GAFLECTR: <RSCAN0_base> + 0098_H
RSCAN0GAFLECTRL: <RSCAN0_base> + 0098_H, RSCAN0GAFLECTRH: <RSCAN0_base> + $009A_H$
RSCAN0GAFLECTRLL: <RSCAN0_base> + 0098_H, RSCAN0GAFLECTRLH: <RSCAN0_base> + 0099_H,
RSCAN0GAFLECTRHL: <RSCAN0_base> + $009A_H$, RSCAN0GAFLECTRHH: <RSCAN0_base> + $009B_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFL DAE	—	—	—	AFLPN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 14.25 RSCAN0GAFLECTR Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved These bits are always read as 0. The write value should always be 0.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 15 (01111_B).

AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.

Set these bits to a value within the range of 00000_B to 01111_B.

14.3.13 RSCAN0GAFLCFG0 — Receive Rule Configuration Register 0

Access: RSCAN0GAFLCFG0 can be read/written in 32-bit units.
RSCAN0GAFLCFG0L and RSCAN0GAFLCFG0H can be read/written in 16-bit units.
RSCAN0GAFLCFG0LL, RSCAN0GAFLCFG0LH, RSCAN0GAFLCFG0HL, and RSCAN0GAFLCFG0HH can be read/written in 8-bit units.

Address: RSCAN0GAFLCFG0: <RSCAN0_base> + $009C_H$
RSCAN0GAFLCFG0L: <RSCAN0_base> + $009C_H$, RSCAN0GAFLCFG0H: <RSCAN0_base> + $009E_H$
RSCAN0GAFLCFG0LL: <RSCAN0_base> + $009C_H$, RSCAN0GAFLCFG0LH: <RSCAN0_base> + $009D_H$,
RSCAN0GAFLCFG0HL: <RSCAN0_base> + $009E_H$, RSCAN0GAFLCFG0HH: <RSCAN0_base> + $009F_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNC2[7:0]								RNC3[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.26 RSCAN0GAFLCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 8	RNC2[7:0]	Number of Rules for Channel 2 Set the number of receive rules exclusively used for channel 2.
7 to 0	RNC3[7:0]	Number of Rules for Channel 3 Set the number of receive rules exclusively used for channel 3.

Modify the RSCAN0GAFLCFG0 register only in global reset mode.

Up to 64 × (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC2[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 2 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

RNC3[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 3 receive rule table.

Set these bits to a value within the range of 00_H to 80_H.

14.3.14 RSCAN0GAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access: RSCAN0GAFLIDj can be read/written in 32-bit units.
RSCAN0GAFLIDjL and RSCAN0GAFLIDjH can be read/written in 16-bit units.
RSCAN0GAFLIDjLL, RSCAN0GAFLIDjLH, RSCAN0GAFLIDjHL, and RSCAN0GAFLIDjHH can be read/written in 8-bit units.

Address: RSCAN0GAFLIDj: <RSCAN0_base> + 0500_H + (10_H × j)
RSCAN0GAFLIDjL: <RSCAN0_base> + 0500_H + (10_H × j),
RSCAN0GAFLIDjH: <RSCAN0_base> + 0502_H + (10_H × j)
RSCAN0GAFLIDjLL: <RSCAN0_base> + 0500_H + (10_H × j),
RSCAN0GAFLIDjLH: <RSCAN0_base> + 0501_H + (10_H × j),
RSCAN0GAFLIDjHL: <RSCAN0_base> + 0502_H + (10_H × j),
RSCAN0GAFLIDjHH: <RSCAN0_base> + 0503_H + (10_H × j)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFL IDE	GAFL RTR	GAFL LB	GAFLID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.27 RSCAN0GAFLIDj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
28 to 0	GAFLID[28:0]	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCAN0GAFLIDj register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

14.3.15 RSCAN0GAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access: RSCAN0GAFLMj can be read/written in 32-bit units.
RSCAN0GAFLMjL and RSCAN0GAFLMjH can be read/written in 16-bit units.
RSCAN0GAFLMjLL, RSCAN0GAFLMjLH, RSCAN0GAFLMjHL, and RSCAN0GAFLMjHH can be read/written in 8-bit units.

Address: RSCAN0GAFLMj: <RSCAN0_base> + 0504_H + (10_H × j)
RSCAN0GAFLMjL: <RSCAN0_base> + 0504_H + (10_H × j),
RSCAN0GAFLMjH: <RSCAN0_base> + 0506_H + (10_H × j)
RSCAN0GAFLMjLL: <RSCAN0_base> + 0504_H + (10_H × j),
RSCAN0GAFLMjLH: <RSCAN0_base> + 0505_H + (10_H × j),
RSCAN0GAFLMjHL: <RSCAN0_base> + 0506_H + (10_H × j),
RSCAN0GAFLMjHH: <RSCAN0_base> + 0507_H + (10_H × j)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFL IDEM	GAF LRTRM	—	GAFLIDM[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.28 RSCAN0GAFLMj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	—	Reserved This bit is always read as 0. The write value should always be 0.
28 to 0	GAFLIDM[28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCAN0GAFLMj register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCAN0GAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

14.3.16 RSCAN0GAFLP0j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access: RSCAN0GAFLP0j can be read/written in 32-bit units.
RSCAN0GAFLP0jL and RSCAN0GAFLP0jH can be read/written in 16-bit units.
RSCAN0GAFLP0jLL, RSCAN0GAFLP0jLH, RSCAN0GAFLP0jHL, and RSCAN0GAFLP0jHH can be read/written in 8-bit units.

Address: RSCAN0GAFLP0j: <RSCAN0_base> + 0508_H + (10_H × j)
RSCAN0GAFLP0jL: <RSCAN0_base> + 0508_H + (10_H × j),
RSCAN0GAFLP0jH: <RSCAN0_base> + $050A_H$ + (10_H × j)
RSCAN0GAFLP0jLL: <RSCAN0_base> + 0508_H + (10_H × j),
RSCAN0GAFLP0jLH: <RSCAN0_base> + 0509_H + (10_H × j),
RSCAN0GAFLP0jHL: <RSCAN0_base> + $050A_H$ + (10_H × j),
RSCAN0GAFLP0jHH: <RSCAN0_base> + $050B_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFL RVM	GAFLRMDP[6:0]							—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 14.29 RSCAN0GAFLP0j Register Contents

Bit Position	Bit Name	Function
31 to 28	GAFLDLC[3:0]	Receive Rule DLC\ 0 0 0 0: DLC check is disabled. 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	GAFLPTR[11:0]	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP[6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Modify the RSCAN0GAFLP0j register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCAN0RMNB register.

14.3.17 RSCAN0GAFLP1j — Receive Rule Pointer 1 Register (j = 0 to 15)

Access: RSCAN0GAFLP1j can be read/written in 32-bit units.
RSCAN0GAFLP1jL and RSCAN0GAFLP1jH can be read/written in 16-bit units.
RSCAN0GAFLP1jLL, RSCAN0GAFLP1jLH, RSCAN0GAFLP1jHL, and RSCAN0GAFLP1jHH can be read/written in 8-bit units.

Address: RSCAN0GAFLP1j: <RSCAN0_base> + $050C_H$ + (10_H × j)
RSCAN0GAFLP1jL: <RSCAN0_base> + $050C_H$ + (10_H × j),
RSCAN0GAFLP1jH: <RSCAN0_base> + $050E_H$ + (10_H × j)
RSCAN0GAFLP1jLL: <RSCAN0_base> + $050C_H$ + (10_H × j),
RSCAN0GAFLP1jLH: <RSCAN0_base> + $050D_H$ + (10_H × j),
RSCAN0GAFLP1jHL: <RSCAN0_base> + $050E_H$ + (10_H × j),
RSCAN0GAFLP1jHH: <RSCAN0_base> + $050F_H$ + (10_H × j)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.30 RSCAN0GAFLP1j Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0. The write value should always be 0.
19 to 0	GAFLFDP [19:0]	FIFO Buffer z Select (z = 0 to 19) z = 0 to 7 0: Receive FIFO buffer z is not selected. 1: Receiver FIFO buffer z is selected. z = 8 to 19 0: Transmit/receive FIFO buffer z-8 is not selected. 1: Transmit/receive FIFO buffer z-8 is selected.

Modify the RSCAN0GAFLP1j register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

GAFLFDP [19:0] Bits

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCAN0GAFLP0j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected.

Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCAN0CFCCk register are set to 00_B (receive mode) or 10_B (gateway mode) are selectable.

14.3.18 RSCAN0RMNB — Receive Buffer Number Register

Access: RSCAN0RMNB can be read/written in 32-bit units.
RSCAN0RMNBL and RSCAN0RMNBH can be read/written in 16-bit units.
RSCAN0RMNBLL, RSCAN0RMNBLH, RSCAN0RMNBHL, and RSCAN0RMNBHH can be read/written in 8-bit units.

Address: RSCAN0RMNB: <RSCAN0_base> + $00A4_H$
RSCAN0RMNBL: <RSCAN0_base> + $00A4_H$, RSCAN0RMNBH: <RSCAN0_base> + $00A6_H$
RSCAN0RMNBLL: <RSCAN0_base> + $00A4_H$, RSCAN0RMNBLH: <RSCAN0_base> + $00A5_H$,
RSCAN0RMNBHL: <RSCAN0_base> + $00A6_H$, RSCAN0RMNBHH: <RSCAN0_base> + $00A7_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.31 RSCAN0RMNB Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 64.

Modify the RSCAN0RMNB register only in global reset mode.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CAN module. The maximum value is 16 × (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.

14.3.19 RSCAN0RMNDy — Receive Buffer New Data Register y (y = 0, 1)

Access: RSCAN0RMNDy can be read/written in 32-bit units.
RSCAN0RMNDyL and RSCAN0RMNDmH can be read/written in 16-bit units.
RSCAN0RMNDyLL, RSCAN0RMNDyLH, RSCAN0RMNDyHL, and RSCAN0RMNDyHH can be read/written in 8-bit units.

Address: RSCAN0RMNDy: <RSCAN0_base> + $00A8_H$ + (04_H × m)
RSCAN0RMNDyL: <RSCAN0_base> + $00A8_H$ + (04_H × m),
RSCAN0RMNDyH: <RSCAN0_base> + $00AA_H$ + (04_H × m)
RSCAN0RMNDyLL: <RSCAN0_base> + $00A8_H$ + (04_H × m),
RSCAN0RMNDyLH: <RSCAN0_base> + $00A9_H$ + (04_H × m),
RSCAN0RMNDyHL: <RSCAN0_base> + $00AA_H$ + (04_H × m),
RSCAN0RMNDyHH: <RSCAN0_base> + $00AB_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.32 RSCAN0RMNDy Register Contents

Bit Position	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Reception Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Reception Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCAN0RMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 63)

Each RMNS flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear any of these flags, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message.

These flags are cleared to 0 in global reset mode.

14.3.20 RSCAN0RMIDq — Receive Buffer ID Register (q = 0 to 63)

Access: RSCAN0RMIDq is read-only in 32-bit units.
RSCAN0RMIDqL and RSCAN0RMIDqH are read-only in 16-bit units.
RSCAN0RMIDqLL, RSCAN0RMIDqLH, RSCAN0RMIDqHL, and RSCAN0RMIDqHH are read-only in 8-bit units.

Address: RSCAN0RMIDq: <RSCAN0_base> + 0600_H + (10_H × q)
RSCAN0RMIDqL: <RSCAN0_base> + 0600_H + (10_H × q),
RSCAN0RMIDqH: <RSCAN0_base> + 0602_H + (10_H × q)
RSCAN0RMIDqLL: <RSCAN0_base> + 0600_H + (10_H × q),
RSCAN0RMIDqLH: <RSCAN0_base> + 0601_H + (10_H × q),
RSCAN0RMIDqHL: <RSCAN0_base> + 0602_H + (10_H × q),
RSCAN0RMIDqHH: <RSCAN0_base> + 0603_H + (10_H × q)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 14.33 RSCAN0RMIDq Register Contents

Bit Position	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR 0: Data frame 1: Remote frame
29	—	Reserved This bit is always read as 0.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits indicate the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

RMID[28:0] Bits

These bits indicate the ID of the message stored in the receive buffer.

14.3.21 RSCAN0RMPTRq — Receive Buffer Pointer Register (q = 0 to 63)

Access: RSCAN0RMPTRq is read-only in 32-bit units.
RSCAN0RMPTRqL and RSCAN0RMPTRqH are read-only in 16-bit units.
RSCAN0RMPTRqLL, RSCAN0RMPTRqLH, RSCAN0RMPTRqHL, and RSCAN0RMPTRqHH are read-only in 8-bit units.

Address: RSCAN0RMPTRq: <RSCAN0_base> + 0604_H + (10_H × q)
RSCAN0RMPTRqL: <RSCAN0_base> + 0604_H + (10_H × q),
RSCAN0RMPTRqH: <RSCAN0_base> + 0606_H + (10_H × q)
RSCAN0RMPTRqLL: <RSCAN0_base> + 0604_H + (10_H × q),
RSCAN0RMPTRqLH: <RSCAN0_base> + 0605_H + (10_H × q),
RSCAN0RMPTRqHL: <RSCAN0_base> + 0606_H + (10_H × q),
RSCAN0RMPTRqHH: <RSCAN0_base> + 0607_H + (10_H × q)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 14.34 RSCAN0RMPTRq Register Contents

Bit Position	Bit Name	Function
31 to 28	RMDLC[3:0]	Receive Buffer DLC Data 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR[11:0]	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

14.3.22 RSCAN0RMDF0q — Receive Buffer Data Field 0 Register (q = 0 to 63)

Access: RSCAN0RMDF0q is read-only in 32-bit units.
RSCAN0RMDF0qL and RSCAN0RMDF0qH are read-only in 16-bit units.
RSCAN0RMDF0qLL, RSCAN0RMDF0qLH, RSCAN0RMDF0qHL, and RSCAN0RMDF0qHH are read-only in 8-bit units.

Address: RSCAN0RMDF0q: <RSCAN0_base> + 0608_H + (10_H × q)
RSCAN0RMDF0qL: <RSCAN0_base> + 0608_H + (10_H × q),
RSCAN0RMDF0qH: <RSCAN0_base> + $060A_H$ + (10_H × q)
RSCAN0RMDF0qLL: <RSCAN0_base> + 0608_H + (10_H × q),
RSCAN0RMDF0qLH: <RSCAN0_base> + 0609_H + (10_H × q),
RSCAN0RMDF0qHL: <RSCAN0_base> + $060A_H$ + (10_H × q),
RSCAN0RMDF0qHH: <RSCAN0_base> + $060B_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.35 RSCAN0RMDF0q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB3[7:0]	Receive Buffer Data Byte 3
23 to 16	RMDB2[7:0]	Receive Buffer Data Byte 2
15 to 8	RMDB1[7:0]	Receive Buffer Data Byte 1
7 to 0	RMDB0[7:0]	Receive Buffer Data Byte 0 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.23 RSCAN0RMDF1q — Receive Buffer Data Field 1 Register (q = 0 to 63)

Access: RSCAN0RMDF1q is read-only in 32-bit units.
RSCAN0RMDF1qL and RSCAN0RMDF1qH are read-only in 16-bit units.
RSCAN0RMDF1qLL, RSCAN0RMDF1qLH, RSCAN0RMDF1qHL, and RSCAN0RMDF1qHH are read-only in 8-bit units.

Address: RSCAN0RMDF1q: <RSCAN0_base> + $060C_H$ + (10_H × q)
RSCAN0RMDF1qL: <RSCAN0_base> + $060C_H$ + (10_H × q),
RSCAN0RMDF1qH: <RSCAN0_base> + $060E_H$ + (10_H × q)
RSCAN0RMDF1qLL: <RSCAN0_base> + $060C_H$ + (10_H × q),
RSCAN0RMDF1qLH: <RSCAN0_base> + $060D_H$ + (10_H × q),
RSCAN0RMDF1qHL: <RSCAN0_base> + $060E_H$ + (10_H × q),
RSCAN0RMDF1qHH: <RSCAN0_base> + $060F_H$ + (10_H × q)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.36 RSCAN0RMDF1q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB7[7:0]	Receive Buffer Data Byte 7
23 to 16	RMDB6[7:0]	Receive Buffer Data Byte 6
15 to 8	RMDB5[7:0]	Receive Buffer Data Byte 5
7 to 0	RMDB4[7:0]	Receive Buffer Data Byte 4 Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.24 RSCAN0RFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

Access: RSCAN0RFCCx can be read/written in 32-bit units.
RSCAN0RFCCxL and RSCAN0RFCCxH can be read/written in 16-bit units.
RSCAN0RFCCxLL, RSCAN0RFCCxLH, RSCAN0RFCCxHL, and RSCAN0RFCCxHH can be read/written in 8-bit units.

Address: RSCAN0RFCCx: <RSCAN0_base> + $00B8_H$ + (04_H × x)
RSCAN0RFCCxL: <RSCAN0_base> + $00B8_H$ + (04_H × x),
RSCAN0RFCCxH: <RSCAN0_base> + $00BA_H$ + (04_H × x)
RSCAN0RFCCxLL: <RSCAN0_base> + $00B8_H$ + (04_H × x),
RSCAN0RFCCxLH: <RSCAN0_base> + $00B9_H$ + (04_H × x),
RSCAN0RFCCxHL: <RSCAN0_base> + $00BA_H$ + (04_H × x),
RSCAN0RFCCxHH: <RSCAN0_base> + $00BB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 14.37 RSCAN0RFCCx Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should always be 0.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	—	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.

Table 14.37 RSCAN0RFCCx Register Contents (2/2)

Bit Position	Bit Name	Function
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to select the timing for generating a receive FIFO interrupt request when the RFIM bit is cleared to 0. An interrupt request is generated when the number of stored messages reaches the specified ratio (in fraction) of the storable messages set with the RFDC[2:0] bits.

When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits only in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000_B, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCAN0RFSTSx register to 1 (the receive FIFO buffer contains no unread message (buffer empty)). Modify this bit in global operating mode or global test mode.

14.3.25 RSCAN0RFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access: RSCAN0RFSTSx can be read/written in 32-bit units.
RSCAN0RFSTSxL and RSCAN0RFSTSxH can be read/written in 16-bit units.
RSCAN0RFSTSxLL, RSCAN0RFSTSxLH, RSCAN0RFSTSxHL, and RSCAN0RFSTSxHH can be read/written in 8-bit units.

Address: RSCAN0RFSTSx: <RSCAN0_base> + $00D8_H$ + (04_H × x)
RSCAN0RFSTSxL: <RSCAN0_base> + $00D8_H$ + (04_H × x),
RSCAN0RFSTSxH: <RSCAN0_base> + $00DA_H$ + (04_H × x)
RSCAN0RFSTSxLL: <RSCAN0_base> + $00D8_H$ + (04_H × x),
RSCAN0RFSTSxLH: <RSCAN0_base> + $00D9_H$ + (04_H × x),
RSCAN0RFSTSxHL: <RSCAN0_base> + $00DA_H$ + (04_H × x),
RSCAN0RFSTSxHH: <RSCAN0_base> + $00DB_H$ + (04_H × x)

Value after reset: $0000\ 0001_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.38 RSCAN0RFSTSx Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flags

These flags indicate the number of unread messages in the receive FIFO buffer. These flags become 00_H when the RFE bit in the RSCAN0RFCCx register is set to 0.

RFIF Flag

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCAN0RFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

RFMLT Flag

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCAN0RFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCAN0RFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCAN0RFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

14.3.26 RSCAN0RFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: RSCAN0RFPCTRx can be read/written in 32-bit units.
RSCAN0RFPCTRxL and RSCAN0RFPCTRxH can be read/written in 16-bit units.
RSCAN0RFPCTRxLL, RSCAN0RFPCTRxLH, RSCAN0RFPCTRxHL, and RSCAN0RFPCTRxHH can be read/written in 8-bit units.

Address: RSCAN0RFPCTRx: <RSCAN0_base> + $00F8_H$ + (04_H × x)
RSCAN0RFPCTRxL: <RSCAN0_base> + $00F8_H$ + (04_H × x),
RSCAN0RFPCTRxH: <RSCAN0_base> + $00FA_H$ + (04_H × x)
RSCAN0RFPCTRxLL: <RSCAN0_base> + $00F8_H$ + (04_H × x),
RSCAN0RFPCTRxLH: <RSCAN0_base> + $00F9_H$ + (04_H × x),
RSCAN0RFPCTRxHL: <RSCAN0_base> + $00FA_H$ + (04_H × x),
RSCAN0RFPCTRxHH: <RSCAN0_base> + $00FB_H$ + (04_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.39 RSCAN0RFPCTRx Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved The write value should always be 0.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer.

RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF_H, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCAN0RFSTSx register is decremented. Read the RSCAN0RFID, RSCAN0RFPTR, RSCAN0RFDF0, and RSCAN0RFDF1 registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits.

Write FF_H to these bits when the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCAN0RFSTSx register is 0 (the receive FIFO buffer contains unread messages).

14.3.27 RSCAN0RFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: RSCAN0RFIDx is read-only in 32-bit units.
RSCAN0RFIDxL and RSCAN0RFIDxH are read-only in 16-bit units.
RSCAN0RFIDxLL, RSCAN0RFIDxLH, RSCAN0RFIDxHL, and RSCAN0RFIDxHH are read-only in 8-bit units.

Address: RSCAN0RFIDx: <RSCAN0_base> + $0E00_H$ + (10_H × x)
RSCAN0RFIDxL: <RSCAN0_base> + $0E00_H$ + (10_H × x),
RSCAN0RFIDxH: <RSCAN0_base> + $0E02_H$ + (10_H × x)
RSCAN0RFIDxLL: <RSCAN0_base> + $0E00_H$ + (10_H × x),
RSCAN0RFIDxLH: <RSCAN0_base> + $0E01_H$ + (10_H × x),
RSCAN0RFIDxHL: <RSCAN0_base> + $0E02_H$ + (10_H × x),
RSCAN0RFIDxHH: <RSCAN0_base> + $0E03_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 14.40 RSCAN0RFIDx Register Contents

Bit Position	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	—	Reserved This bit is always read as 0.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

14.3.28 RSCAN0RFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: RSCAN0RFPTRx is read-only in 32-bit units.
RSCAN0RFPTRxL and RSCAN0RFPTRxH are read-only in 16-bit units.
RSCAN0RFPTRxLL, RSCAN0RFPTRxLH, RSCAN0RFPTRxHL, and RSCAN0RFPTRxHH are read-only in 8-bit units.

Address: RSCAN0RFPTRx: <RSCAN0_base> + $0E04_H$ + (10_H × x)
RSCAN0RFPTRxL: <RSCAN0_base> + $0E04_H$ + (10_H × x),
RSCAN0RFPTRxH: <RSCAN0_base> + $0E06_H$ + (10_H × x)
RSCAN0RFPTRxLL: <RSCAN0_base> + $0E04_H$ + (10_H × x),
RSCAN0RFPTRxLH: <RSCAN0_base> + $0E05_H$ + (10_H × x),
RSCAN0RFPTRxHL: <RSCAN0_base> + $0E06_H$ + (10_H × x),
RSCAN0RFPTRxHH: <RSCAN0_base> + $0E07_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 14.41 RSCAN0RFPTRx Register Contents

Bit Position	Bit Name	Function
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

14.3.29 RSCAN0RFDF0x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

Access: RSCAN0RFDF0x is read-only in 32-bit units.
RSCAN0RFDF0xL and RSCAN0RFDF0xH are read-only in 16-bit units.
RSCAN0RFDF0xLL, RSCAN0RFDF0xLH, RSCAN0RFDF0xHL, and RSCAN0RFDF0xHH are read-only in 8-bit units.

Address: RSCAN0RFDF0x: <RSCAN0_base> + $0E08_H$ + (10_H × x)
RSCAN0RFDF0xL: <RSCAN0_base> + $0E08_H$ + (10_H × x),
RSCAN0RFDF0xH: <RSCAN0_base> + $0E0A_H$ + (10_H × x)
RSCAN0RFDF0xLL: <RSCAN0_base> + $0E08_H$ + (10_H × x),
RSCAN0RFDF0xLH: <RSCAN0_base> + $0E09_H$ + (10_H × x),
RSCAN0RFDF0xHL: <RSCAN0_base> + $0E0A_H$ + (10_H × x),
RSCAN0RFDF0xHH: <RSCAN0_base> + $0E0B_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.42 RSCAN0RFDF0x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB3[7:0]	Receive FIFO Buffer Data Byte 3
23 to 16	RFDB2[7:0]	Receive FIFO Buffer Data Byte 2
15 to 8	RFDB1[7:0]	Receive FIFO Buffer Data Byte 1
7 to 0	RFDB0[7:0]	Receive FIFO Buffer Data Byte 0 Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.30 RSCAN0RFDF1x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

Access: RSCAN0RFDF1x is read-only in 32-bit units.
RSCAN0RFDF1xL and RSCAN0RFDF1xH are read-only in 16-bit units.
RSCAN0RFDF1xLL, RSCAN0RFDF1xLH, RSCAN0RFDF1xHL, and RSCAN0RFDF1xHH are read-only in 8-bit units.

Address: RSCAN0RFDF1x: <RSCAN0_base> + $0E0C_H$ + (10_H × x)
RSCAN0RFDF1xL: <RSCAN0_base> + $0E0C_H$ + (10_H × x),
RSCAN0RFDF1xH: <RSCAN0_base> + $0E0E_H$ + (10_H × x)
RSCAN0RFDF1xLL: <RSCAN0_base> + $0E0C_H$ + (10_H × x),
RSCAN0RFDF1xLH: <RSCAN0_base> + $0E0D_H$ + (10_H × x),
RSCAN0RFDF1xHL: <RSCAN0_base> + $0E0E_H$ + (10_H × x),
RSCAN0RFDF1xHH: <RSCAN0_base> + $0E0F_H$ + (10_H × x)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.43 RSCAN0RFDF1x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB7[7:0]	Receive FIFO Buffer Data Byte 7 Receive FIFO Buffer Data Byte 6 Receive FIFO Buffer Data Byte 5 Receive FIFO Buffer Data Byte 4 Data for a message stored in the receive FIFO buffer can be read.
23 to 16	RFDB6[7:0]	
15 to 8	RFDB5[7:0]	
7 to 0	RFDB4[7:0]	

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

14.3.31 RSCAN0CFCCk — Transmit/Receive FIFO Buffer Configuration and Control Register k (k = 0 to 11)

Access: RSCAN0CFCCk can be read/written in 32-bit units.
RSCAN0CFCCkL and RSCAN0CFCCkH can be read/written in 16-bit units.
RSCAN0CFCCkLL, RSCAN0CFCCkLH, RSCAN0CFCCkHL, and RSCAN0CFCCkHH can be read/written in 8-bit units.

Address: RSCAN0CFCCk: <RSCAN0_base> + 0118_H + (04_H × k)
RSCAN0CFCCkL: <RSCAN0_base> + 0118_H + (04_H × k),
RSCAN0CFCCkH: <RSCAN0_base> + $011A_H$ + (04_H × k)
RSCAN0CFCCkLL: <RSCAN0_base> + 0118_H + (04_H × k),
RSCAN0CFCCkLH: <RSCAN0_base> + 0119_H + (04_H × k),
RSCAN0CFCCkHL: <RSCAN0_base> + $011A_H$ + (04_H × k),
RSCAN0CFCCkHH: <RSCAN0_base> + $011B_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	—	—	—	—	CFTXIE	CFRXIE	CFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 14.44 RSCAN0CFCCk Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk/2 by (ITRCP [15:0] bits × 1) 1: Clock dividing pclk/2 by (ITRCP [15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.
17, 16	CFM[1:0]	Transmit/Receive FIFO Mode Select 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.

Table 14.44 RSCAN0CFCCk Register Contents (2/2)

Bit Position	Bit Name	Function
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: • Receive mode/gateway mode When the number of received messages has met the condition set by the CFIGCV[2:0] bits, a FIFO receive interrupt request is generated. • Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated. 1: • Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received. • Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.
11	—	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number n of FIFO buffer k is calculated as n = k/3 (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be ((16 × m) + CFTML[3:0]).

See **Table 14.11** and **Table 14.12** for the relationship between transmit/receive FIFO buffer k and transmit buffer p.

Setting the CFDC[2:0] bits to 001_B or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFITR Bit

This bit is valid when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the ITRCP[15:0] bits in the RSCAN0GCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register × 10).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITR bit.

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITSS bit.

CFM[1:0] Bits

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

CFIGCV[2:0] Bits

When the CFIM bit is cleared to 0 and the CFM[1:0] bits are set to 00_B (receive mode) or 10_B (gateway mode), these bits are used to select the timing for generating a transmit/receive FIFO receive interrupt request.

An interrupt request is generated when the number of stored messages reaches the specified ratio (in fraction) of the storable messages set with the CFDC[2:0] bits.

When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.

Modify these bits only in global reset mode.

CFIM Bit

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

CFDC[2:0] Bits

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000_B, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCAN0CFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFRXIF flag in the RSCAN0CFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

14.3.32 RSCAN0CFSTSk — Transmit/Receive FIFO Buffer Status Register (k = 0 to 11)

Access: RSCAN0CFSTSk can be read/written in 32-bit units.
RSCAN0CFSTSkL and RSCAN0CFSTSkH can be read/written in 16-bit units.
RSCAN0CFSTSkLL, RSCAN0CFSTSkLH, RSCAN0CFSTSkHL, and RSCAN0CFSTSkHH can be read/written in 8-bit units.

Address: RSCAN0CFSTSk: <RSCAN0_base> + 0178_H + (04_H × k)
RSCAN0CFSTSkL: <RSCAN0_base> + 0178_H + (04_H × k),
RSCAN0CFSTSkH: <RSCAN0_base> + $017A_H$ + (04_H × k)
RSCAN0CFSTSkLL: <RSCAN0_base> + 0178_H + (04_H × k),
RSCAN0CFSTSkLH: <RSCAN0_base> + 0179_H + (04_H × k),
RSCAN0CFSTSkHL: <RSCAN0_base> + $017A_H$ + (04_H × k),
RSCAN0CFSTSkHH: <RSCAN0_base> + $017B_H$ + (04_H × k)

Value after reset: $0000\ 0001_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.45 RSCAN0CFSTSk Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCAN0CFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: In global reset mode
- When CFM[1:0] value is 01_B or 10_B: In channel reset mode

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B, and the factor selected by the CFIM bit in the RSCAN0CFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B or 10_B, and the factor selected by the CFIM bit in the RSCAN0CFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Clear this flag to 0 in global operating mode or global test mode.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCAN0CFCCk register.

The CFFLL flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCAN0CFCCk register is 0 (no transmit/receive FIFO buffer is used): When not in the transmission abort
- When the CFM[1:0] bits are set to 00_B: In global reset mode
- When the CFM[1:0] bits are set to 01_B or 10_B: In channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01_B or 10_B: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01_B: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmission abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00_B: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01_B or 10_B: A value of FF_H has been written to the RSCAN0CFPCTRk register after data was written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers.

14.3.33 RSCAN0CFPCTRk — Transmit/Receive FIFO Buffer Pointer Control Register (k = 0 to 11)

Access: RSCAN0CFPCTRk can be read/written in 32-bit units.
RSCAN0CFPCTRkL and RSCAN0CFPCTRkH can be read/written in 16-bit units.
RSCAN0CFPCTRkLL, RSCAN0CFPCTRkLH, RSCAN0CFPCTRkHL, and RSCAN0CFPCTRkHH can be read/written in 8-bit units.

Address: RSCAN0CFPCTRk: <RSCAN0_base> + $01D8_H$ + (04_H × k)
RSCAN0CFPCTRkL: <RSCAN0_base> + $01D8_H$ + (04_H × k),
RSCAN0CFPCTRkH: <RSCAN0_base> + $01DA_H$ + (04_H × k)
RSCAN0CFPCTRkLL: <RSCAN0_base> + $01D8_H$ + (04_H × k),
RSCAN0CFPCTRkLH: <RSCAN0_base> + $01D9_H$ + (04_H × k),
RSCAN0CFPCTRkHL: <RSCAN0_base> + $01DA_H$ + (04_H × k),
RSCAN0CFPCTRkHH: <RSCAN0_base> + $01DB_H$ + (04_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.46 RSCAN0CFPCTRk Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved The write value should always be 0.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FF_H to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FF_H to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCAN0CFCCk register is 00_B):
  Writing FF_H to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCAN0CFSTSk register is decremented. Read the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers to read messages from the transmit/receive FIFO buffer, and then write FF_H to the CFPC[7:0] bits.
  Write FF_H to these bits when the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCAN0CFSTSk register is cleared to 0 (the transmit/receive FIFO buffer contains messages).

- Transmit mode (CFM[1:0] value in the RSCAN0CFCCk register is 01_B):
  Writing FF_H to the CFPC[7:0] bits stores the data written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers before writing FF_H to the CFPC[7:0] bits.
  Write FF_H to these bits when the CFE bit in the RSCAN0CFCCk register is set to 1 and the CFFLL flag in the RSCAN0CFSTSk register is cleared to 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCAN0CFCCk register is 10_B):
  Setting prohibited

14.3.34 RSCAN0CFIDk — Transmit/Receive FIFO Buffer Access ID Register (k = 0 to 11)

Access: RSCAN0CFIDk can be read/written in 32-bit units.
RSCAN0CFIDkL and RSCAN0CFIDkH can be read/written in 16-bit units.
RSCAN0CFIDkLL, RSCAN0CFIDkLH, RSCAN0CFIDkHL, and RSCAN0CFIDkHH can be read/written in 8-bit units.

Address: RSCAN0CFIDk: <RSCAN0_base> + $0E80_H$ + (10_H × k)
RSCAN0CFIDkL: <RSCAN0_base> + $0E80_H$ + (10_H × k),
RSCAN0CFIDkH: <RSCAN0_base> + $0E82_H$ + (10_H × k)
RSCAN0CFIDkLL: <RSCAN0_base> + $0E80_H$ + (10_H × k),
RSCAN0CFIDkLH: <RSCAN0_base> + $0E81_H$ + (10_H × k),
RSCAN0CFIDkHL: <RSCAN0_base> + $0E82_H$ + (10_H × k),
RSCAN0CFIDkHH: <RSCAN0_base> + $0E83_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.47 RSCAN0CFIDk Register Contents

Bit Position	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits b10 to b0 and write 0 to bits b28 to b11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits b10 to b0. Bits b28 to b11 are read as 0. **NOTE** To clear the CFTXIF, CFRXIF, and CFMLT flags to 0, the program must write 0 to these flags. Use a store instruction to write 0 to these flags and write 1 to the other flags.

This register is writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This

RSCAN0CFIDk register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is valid when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.

When the CFM[1:0] value is 01_B, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

14.3.35 RSCAN0CFPTRk — Transmit/Receive FIFO Buffer Access Pointer Register (k = 0 to 11)

Access: RSCAN0CFPTRk can be read/written in 32-bit units.
RSCAN0CFPTRkL and RSCAN0CFPTRkH can be read/written in 16-bit units.
RSCAN0CFPTRkLL, RSCAN0CFPTRkLH, RSCAN0CFPTRkHL, and RSCAN0CFPTRkHH can be read/written in 8-bit units.

Address: RSCAN0CFPTRk: <RSCAN0_base> + $0E84_H$ + (10_H × k)
RSCAN0CFPTRkL: <RSCAN0_base> + $0E84_H$ + (10_H × k),
RSCAN0CFPTRkH: <RSCAN0_base> + $0E86_H$ + (10_H × k)
RSCAN0CFPTRkLL: <RSCAN0_base> + $0E84_H$ + (10_H × k),
RSCAN0CFPTRkLH: <RSCAN0_base> + $0E85_H$ + (10_H × k),
RSCAN0CFPTRkHL: <RSCAN0_base> + $0E86_H$ + (10_H × k),
RSCAN0CFPTRkHH: <RSCAN0_base> + $0E87_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.48 RSCAN0CFPTRk Register Contents

Bit Position	Bit Name	Function
31 to 28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, this bit is used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 9 bytes or more, the actual transmit data defaults to 8 bytes.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.

These bits are valid when the CFM[1:0] value is 00_B.

14.3.36 RSCAN0CFDF0k — Transmit/Receive FIFO Buffer Access Data Field 0 Register (k = 0 to 11)

Access: RSCAN0CFDF0k can be read/written in 32-bit units.
RSCAN0CFDF0kL and RSCAN0CFDF0kH can be read/written in 16-bit units.
RSCAN0CFDF0kLL, RSCAN0CFDF0kLH, RSCAN0CFDF0kHL, and RSCAN0CFDF0kHH can be read/written in 8-bit units.

Address: RSCAN0CFDF0k: <RSCAN0_base> + $0E88_H$ + (10_H × k)
RSCAN0CFDF0kL: <RSCAN0_base> + $0E88_H$ + (10_H × k),
RSCAN0CFDF0kH: <RSCAN0_base> + $0E8A_H$ + (10_H × k)
RSCAN0CFDF0kLL: <RSCAN0_base> + $0E88_H$ + (10_H × k),
RSCAN0CFDF0kLH: <RSCAN0_base> + $0E89_H$ + (10_H × k),
RSCAN0CFDF0kHL: <RSCAN0_base> + $0E8A_H$ + (10_H × k),
RSCAN0CFDF0kHH: <RSCAN0_base> + $0E8B_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.49 RSCAN0CFDF0k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB3[7:0]	Transmit/Receive FIFO Buffer Data Byte 3 Transmit/Receive FIFO Buffer Data Byte 2 Transmit/Receive FIFO Buffer Data Byte 1 Transmit/Receive FIFO Buffer Data Byte 0 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.
23 to 16	CFDB2[7:0]	
15 to 8	CFDB1[7:0]	
7 to 0	CFDB0[7:0]	

This register is writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H. This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

14.3.37 RSCAN0CFDF1k — Transmit/Receive FIFO Buffer Access Data Field 1 Register (k = 0 to 11)

Access: RSCAN0CFDF1k can be read/written in 32-bit units.
RSCAN0CFDF1kL and RSCAN0CFDF1kH can be read/written in 16-bit units.
RSCAN0CFDF1kLL, RSCAN0CFDF1kLH, RSCAN0CFDF1kHL, and RSCAN0CFDF1kHH can be read/written in 8-bit units.

Address: RSCAN0CFDF1k: <RSCAN0_base> + $0E8C_H$ + (10_H × k)
RSCAN0CFDF1kL: <RSCAN0_base> + $0E8C_H$ + (10_H × k),
RSCAN0CFDF1kH: <RSCAN0_base> + $0E8E_H$ + (10_H × k)
RSCAN0CFDF1kLL: <RSCAN0_base> + $0E8C_H$ + (10_H × k),
RSCAN0CFDF1kLH: <RSCAN0_base> + $0E8D_H$ + (10_H × k),
RSCAN0CFDF1kHL: <RSCAN0_base> + $0E8E_H$ + (10_H × k),
RSCAN0CFDF1kHH: <RSCAN0_base> + $0E8F_H$ + (10_H × k)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.50 RSCAN0CFDF1k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB7[7:0]	Transmit/Receive FIFO Buffer Data Byte 7 Transmit/Receive FIFO Buffer Data Byte 6 Transmit/Receive FIFO Buffer Data Byte 5 Transmit/Receive FIFO Buffer Data Byte 4 • When CFM[1:0] value is 01_B (transmit mode): Set the transmit/receive FIFO buffer data. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.
23 to 16	CFDB6[7:0]	
15 to 8	CFDB5[7:0]	
7 to 0	CFDB4[7:0]	

This register is writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H. This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

14.3.38 RSCAN0FESTS — FIFO Empty Status Register

Access: RSCAN0FESTS is read-only in 32-bit units.
RSCAN0FESTSL and RSCAN0FESTSH are read-only in 16-bit units.
RSCAN0FESTSLL, RSCAN0FESTSLH, RSCAN0FESTSHL, and RSCAN0FESTSHH are read-only in 8-bit units.

Address: RSCAN0FESTS: <RSCAN0_base> + 0238_H
RSCAN0FESTSL: <RSCAN0_base> + 0238_H, RSCAN0FESTSH: <RSCAN0_base> + $023A_H$
RSCAN0FESTSLL: <RSCAN0_base> + 0238_H, RSCAN0FESTSLH: <RSCAN0_base> + 0239_H,
RSCAN0FESTSHL: <RSCAN0_base> + $023A_H$, RSCAN0FESTSHH: <RSCAN0_base> + $023B_H$

Value after reset: $007F\ FFFF_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 EMP	CF10 EMP	CF9 EMP	CF8 EMP
Value after reset	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 EMP	CF6 EMP	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.51 RSCAN0FESTS Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0.
22 to 20	—	Reserved These bits are always read as 1.
19	CF11EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 11)
18	CF10EMP	
17	CF9EMP	
16	CF8EMP	
15	CF7EMP	
14	CF6EMP	
13	CF5EMP	
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message (buffer empty). (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCAN0FESTS register is set to $007F\ FFFF_H$ in global reset mode.

CFkEMP Flag (k = 0 to 11)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCAN0RFSTS register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

14.3.39 RSCAN0FFSTS — FIFO Full Status Register

Access: RSCAN0FFSTS is read-only in 32-bit units.
RSCAN0FFSTSL and RSCAN0FFSTSH are read-only in 16-bit units.
RSCAN0FFSTSLL, RSCAN0FFSTSLH, RSCAN0FFSTSHL, and RSCAN0FFSTSHH are read-only in 8-bit units.

Address: RSCAN0FFSTS: <RSCAN0_base> + $023C_H$
RSCAN0FFSTSL: <RSCAN0_base> + $023C_H$, RSCAN0FFSTSH: <RSCAN0_base> + $023E_H$
RSCAN0FFSTSLL: <RSCAN0_base> + $023C_H$, RSCAN0FFSTSLH: <RSCAN0_base> + $023D_H$,
RSCAN0FFSTSHL: <RSCAN0_base> + $023E_H$, RSCAN0FFSTSHH: <RSCAN0_base> + $023F_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11FLL	CF10FLL	CF9FLL	CF8FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7FLL	CF6FLL	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.52 RSCAN0FFSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0.
19	CF11FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full. (k = 0 to 11)
18	CF10FLL	
17	CF9FLL	
16	CF8FLL	
15	CF7FLL	
14	CF6FLL	
13	CF5FLL	
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer m is not full. 1: Receive FIFO buffer m is full. (m = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCAN0FFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 11)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer is full). When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCAN0RFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

14.3.40 RSCAN0FMSTS — FIFO Message Lost Status Register

Access: RSCAN0FMSTS is read-only in 32-bit units.
RSCAN0FMSTSL and RSCAN0FMSTSH are read-only in 16-bit units.
RSCAN0FMSTSLL, RSCAN0FMSTSLH, RSCAN0FMSTSHL, and RSCAN0FMSTSHH are read-only in 8-bit units.

Address: RSCAN0FMSTS: <RSCAN0_base> + 0240_H
RSCAN0FMSTSL: <RSCAN0_base> + 0240_H, RSCAN0FMSTSH: <RSCAN0_base> + 0242_H
RSCAN0FMSTSLL: <RSCAN0_base> + 0240_H, RSCAN0FMSTSLH: <RSCAN0_base> + 0241_H,
RSCAN0FMSTSHL: <RSCAN0_base> + 0242_H, RSCAN0FMSTSHH: <RSCAN0_base> + 0243_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 MLT	CF10 MLT	CF9 MLT	CF8 MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 MLT	CF6 MLT	CF5 MLT	CF4 MLT	CF3 MLT	CF2 MLT	CF1 MLT	CF0 MLT	RF7 MLT	RF6 MLT	RF5 MLT	RF4 MLT	RF3 MLT	RF2 MLT	RF1 MLT	RF0 MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.53 RSCAN0FMSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0.
19	CF11MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 11)
18	CF10MLT	
17	CF9MLT	
16	CF8MLT	
15	CF7MLT	
14	CF6MLT	
13	CF5MLT	
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCAN0FMSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkMLT Flag (k = 0 to 11)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

14.3.41 RSCAN0RFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access: RSCAN0RFISTS is read-only in 32-bit units.
RSCAN0RFISTSL and RSCAN0RFISTSH are read-only in 16-bit units.
RSCAN0RFISTSLL, RSCAN0RFISTSLH, RSCAN0RFISTSHL, and RSCAN0RFISTSHH are read-only in 8-bit units.

Address: RSCAN0RFISTS: <RSCAN0_base> + 0244_H
RSCAN0RFISTSL: <RSCAN0_base> + 0244_H, RSCAN0RFISTSH: <RSCAN0_base> + 0246_H
RSCAN0RFISTSLL: <RSCAN0_base> + 0244_H, RSCAN0RFISTSLH: <RSCAN0_base> + 0245_H,
RSCAN0RFISTSHL: <RSCAN0_base> + 0246_H, RSCAN0RFISTSHH: <RSCAN0_base> + 0247_H

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.54 RSCAN0RFISTS Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCAN0RFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

14.3.42 RSCAN0CFRISTS — Transmit/Receive FIFO Buffer Reception Interrupt Flag Status Register

Access: RSCAN0CFRISTS is read-only in 32-bit units.
RSCAN0CFRISTSL and RSCAN0CFRISTSH are read-only in 16-bit units.
RSCAN0CFRISTSLL, RSCAN0CFRISTSLH, RSCAN0CFRISTSHL, and RSCAN0CFRISTSHH are read-only in 8-bit units.

Address: RSCAN0CFRISTS: <RSCAN0_base> + 0248_H
RSCAN0CFRISTSL: <RSCAN0_base> + 0248_H, RSCAN0CFRISTSH: <RSCAN0_base> + $024A_H$
RSCAN0CFRISTSLL: <RSCAN0_base> + 0248_H, RSCAN0CFRISTSLH: <RSCAN0_base> + 0249_H,
RSCAN0CFRISTSHL: <RSCAN0_base> + $024A_H$, RSCAN0CFRISTSHH: <RSCAN0_base> + $024B_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 RXIF	CF10 RXIF	CF9 RXIF	CF8 RXIF	CF7 RXIF	CF6 RXIF	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.55 RSCAN0CFRISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0.
11	CF11RXIF	Transmit/Receive FIFO Buffer Reception Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k receive interrupt request is present. 1: A transmit/receive FIFO buffer k receive interrupt request is present. (k = 0 to 11)
10	CF10RXIF	
9	CF9RXIF	
8	CF8RXIF	
7	CF7RXIF	
6	CF6RXIF	
5	CF5RXIF	
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCAN0CFRISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkRXIF Flag (k = 0 to 11)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

14.3.43 RSCAN0CFTISTS — Transmit/Receive FIFO Buffer Transmission Interrupt Flag Status Register

Access: RSCAN0CFTISTS is read-only in 32-bit units.
RSCAN0CFTISTSL and RSCAN0CFTISTSH are read-only in 16-bit units.
RSCAN0CFTISTSLL, RSCAN0CFTISTSLH, RSCAN0CFTISTSHL, and RSCAN0CFTISTSHH are read-only in 8-bit units.

Address: RSCAN0CFTISTS: <RSCAN0_base> + $024C_H$
RSCAN0CFTISTSL: <RSCAN0_base> + $024C_H$, RSCAN0CFTISTSH: <RSCAN0_base> + $024E_H$
RSCAN0CFTISTSLL: <RSCAN0_base> + $024C_H$, RSCAN0CFTISTSLH: <RSCAN0_base> + $024D_H$,
RSCAN0CFTISTSHL: <RSCAN0_base> + $024E_H$, RSCAN0CFTISTSHH: <RSCAN0_base> + $024F_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 TXIF	CF10 TXIF	CF9 TXIF	CF8 TXIF	CF7 TXIF	CF6 TXIF	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CF1 TXIF	CF0 TXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.56 RSCAN0CFTISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0.
11	CF11TXIF	Transmit/Receive FIFO Buffer Transmission Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmit interrupt request is present. 1: A transmit/receive FIFO buffer k transmit interrupt request is present. (k = 0 to 11)
10	CF10TXIF	
9	CF9TXIF	
8	CF8TXIF	
7	CF7TXIF	
6	CF6TXIF	
5	CF5TXIF	
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCAN0CFTISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkTXIF Flag (k = 0 to 11)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

14.3.44 RSCAN0TMCp — Transmit Buffer Control Register (p = 0 to 63)

Access: RSCAN0TMCp can be read/written in 8-bit units.

Address: RSCAN0TMCp: <RSCAN0_base> + 0250_H + (01_H × p)

Value after reset: 00_H This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

Table 14.57 RSCAN0TMCp Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmission Abort Request 0: Transmission abort is not requested. 1: Transmission abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCAN0TMCp register meets any of the following conditions, set it to 00_H.

- The RSCAN0TMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCAN0CFCCk register (p = m × 16 + the value of CFTML[3:0] bits).
- The RSCAN0TMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCAN0TXQCCm (m = 0 to 3) register (p = (m × 16 + 15) to (m × 16 + 15 – the value of TXQDC[3:0] bits)).

Bits in the RSCAN0TMCp register are all cleared to 0 in channel reset mode. Modify the RSCAN0TMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCAN0TMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

TMTAR Bit

Setting this bit to 1 generates a transmission abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

When the TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmission abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmission abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCAN0TMSTSp register is 00_B.

14.3.45 RSCAN0TMSTSp — Transmit Buffer Status Register (p = 0 to 63)

Access: RSCAN0TMSTSp can be read/written in 8-bit units.

Address: RSCAN0TMSTSp: <RSCAN0_base> + $02D0_H$ + (01_H × p)

Value after reset: 00_H This register is initialized by any reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Table 14.58 RSCAN0TMSTSp Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	TMTARM	Transmit Buffer Transmission Abort Request Status Flag 0: No transmission abort request is present. 1: A transmission abort request is present.
3	TMTRM	Transmit Buffer Transmission Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmission Result Status Flag 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmission abort has been completed. 1 0: Transmission has been completed (without transmission abort request). 1 1: Transmission has been completed (with transmission abort request).
0	TMTSTS	Transmit Buffer Transmission Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCAN0TMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCAN0TMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCAN0TMCp register is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCAN0TMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCAN0TMCp register is set to 0

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00_B: Transmission is in progress or no transmit request is present.

01_B: Transmission from the transmit buffer was aborted.

10_B: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 0 (transmission abort is not requested).

11_B: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 1 (transmission abort is requested).

Write 00_B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00_B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

14.3.46 RSCAN0TMTRSTSy — Transmit Buffer Transmission Request Status Register y (y = 0, 1)

Access: RSCAN0TMTRSTSy is read-only in 32-bit units.
RSCAN0TMTRSTSyL and RSCAN0TMTRSTSyH are read-only in 16-bit units.
RSCAN0TMTRSTSyLL, RSCAN0TMTRSTSyLH, RSCAN0TMTRSTSyHL, and RSCAN0TMTRSTSyHH are read-only in 8-bit units.

Address: RSCAN0TMTRSTSy: <RSCAN0_base> + 0350_H + (04_H × y)
RSCAN0TMTRSTSyL: <RSCAN0_base> + 0350_H + (04_H × y),
RSCAN0TMTRSTSyH: <RSCAN0_base> + 0352_H + (04_H × y)
RSCAN0TMTRSTSyLL: <RSCAN0_base> + 0350_H + (04_H × y),
RSCAN0TMTRSTSyLH: <RSCAN0_base> + 0351_H + (04_H × y),
RSCAN0TMTRSTSyHL: <RSCAN0_base> + 0352_H + (04_H × y),
RSCAN0TMTRSTSyHH: <RSCAN0_base> + 0253_H + (04_H × y)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.59 RSCAN0TMTRSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmission Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmission is requested. 1: Transmission is requested.
15 to 0	TMTRSTSp	Transmit Buffer Transmission Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTR bit in the RSCAN0TMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 14.60 shows the bit assignment.

Table 14.60 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.47 RSCAN0TMTARSTSy — Transmit Buffer Transmission Abort Request Status Register y (y = 0, 1)

Access: RSCAN0TMTARSTSy is read-only in 32-bit units.
RSCAN0TMTARSTSyL and RSCAN0TMTARSTSyH are read-only in 16-bit units.
RSCAN0TMTARSTSyLL, RSCAN0TMTARSTSyLH, RSCAN0TMTARSTSyHL, and RSCAN0TMTARSTSyHH are read-only in 8-bit units.

Address: RSCAN0TMTARSTSy: <RSCAN0_base> + 0360_H + (04_H × y)
RSCAN0TMTARSTSyL: <RSCAN0_base> + 0360_H + (04_H × y),
RSCAN0TMTARSTSyH: <RSCAN0_base> + 0362_H + (04_H × y)
RSCAN0TMTARSTSyLL: <RSCAN0_base> + 0360_H + (04_H × y),
RSCAN0TMTARSTSyLH: <RSCAN0_base> + 0361_H + (04_H × y),
RSCAN0TMTARSTSyHL: <RSCAN0_base> + 0362_H + (04_H × y),
RSCAN0TMTARSTSyHH: <RSCAN0_base> + 0263_H + (04_H × y)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.61 RSCAN0TMTARSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmission Abort Request Status Flag p (p = m × 32 + 31 to m × 32 + 16) 0: No transmission abort is requested. 1: Transmission abort is requested.
15 to 0	TMTARSTSp	Transmit Buffer Transmission Abort Request Status Flag p (p = m × 32 + 15 to m × 32 + 0) 0: No transmission abort request is present. 1: A transmission abort request is present.

TMTARSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTAR bit in the RSCAN0TMCp register.

When the TMTAR bit is set to 1 (transmission abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmission abort is not requested) or in channel reset mode.

Table 14.62 shows the bit assignment.

Table 14.62 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.48 RSCAN0TMTCSTSy — Transmit Buffer Transmission Complete Status Register y (y = 0, 1)

Access: RSCAN0TMTCSTSy is read-only in 32-bit units.
RSCAN0TMTCSTSyL and RSCAN0TMTCSTSyH are read-only in 16-bit units.
RSCAN0TMTCSTSyLL, RSCAN0TMTCSTSyLH, RSCAN0TMTCSTSyHL, and RSCAN0TMTCSTSyHH are read-only in 8-bit units.

Address: RSCAN0TMTCSTSy: <RSCAN0_base> + 0370_H + (04_H × y)
RSCAN0TMTCSTSyL: <RSCAN0_base> + 0370_H + (04_H × y),
RSCAN0TMTCSTSyH: <RSCAN0_base> + 0372_H + (04_H × y)
RSCAN0TMTCSTSyLL: <RSCAN0_base> + 0370_H + (04_H × y),
RSCAN0TMTCSTSyLH: <RSCAN0_base> + 0371_H + (04_H × y),
RSCAN0TMTCSTSyHL: <RSCAN0_base> + 0372_H + (04_H × y),
RSCAN0TMTCSTSyHH: <RSCAN0_base> + 0373_H + (04_H × y)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.63 RSCAN0TMTCSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmission Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not completed 1: Transmission is completed
15 to 0	TMTCSTSp	Transmit Buffer Transmission Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 10_B (transmission has been completed (without transmission abort request)) or 11_B (transmission has been completed (with transmission abort request)), the corresponding TMTCSTSp flag is set to 1.

A TMTCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.64 shows the bit assignment.

Table 14.64 TMTCSTS Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.49 RSCAN0TMTASTSy — Transmit Buffer Transmission Abort Status Register y (y = 0, 1)

Access: RSCAN0TMTASTSy is read-only in 32-bit units.
RSCAN0TMTASTSyL and RSCAN0TMTASTSyH are read-only in 16-bit units.
RSCAN0TMTASTSyLL, RSCAN0TMTASTSyLH, RSCAN0TMTASTSyHL, and RSCAN0TMTASTSyHH are read-only in 8-bit units.

Address: RSCAN0TMTASTSy: <RSCAN0_base> + 0380_H + (04_H × y)
RSCAN0TMTASTSyL: <RSCAN0_base> + 0380_H + (04_H × y),
RSCAN0TMTASTSyH: <RSCAN0_base> + 0382_H + (04_H × y)
RSCAN0TMTASTSyLL: <RSCAN0_base> + 0380_H + (04_H × y),
RSCAN0TMTASTSyLH: <RSCAN0_base> + 0381_H + (04_H × y),
RSCAN0TMTASTSyHL: <RSCAN0_base> + 0382_H + (04_H × y),
RSCAN0TMTASTSyHH: <RSCAN0_base> + 0383_H + (04_H × y)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31 to y × 32 × 16 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.65 RSCAN0TMTASTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmission Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	Transmit Buffer Transmission Abort Status Flag p(p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01_B (transmission abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 14.66 shows the bit assignment.

Table 14.66 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.50 RSCAN0TMIECy — Transmit Buffer Interrupt Enable Configuration Register y (y = 0, 1)

Access: RSCAN0TMIECy can be read/written in 32-bit units.
RSCAN0TMIECyL and RSCAN0TMIECyH can be read/written in 16-bit units.
RSCAN0TMIECyLL, RSCAN0TMIECyLH, RSCAN0TMIECyHL, and RSCAN0TMIECyHH can be read/written in 8-bit units.

Address: RSCAN0TMIECy: <RSCAN0_base> + 0390_H + (04_H × y)
RSCAN0TMIECyL: <RSCAN0_base> + 0390_H + (04_H × y),
RSCAN0TMIECyH: <RSCAN0_base> + 0392_H + (04_H × y)
RSCAN0TMIECyLL: <RSCAN0_base> + 0390_H + (04_H × y),
RSCAN0TMIECyLH: <RSCAN0_base> + 0391_H + (04_H × y),
RSCAN0TMIECyHL: <RSCAN0_base> + 0392_H + (04_H × y),
RSCAN0TMIECyHH: <RSCAN0_base> + 0393_H + (04_H × y)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 16 to y × 32 + 31 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 0 to y × 32 + 15 (y = 0, 1))															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.67 RSCAN0TMIECy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable Bit p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	Transmit Buffer Interrupt Enable Bit p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 63)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCAN0TMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 14.68 shows the bit assignment.

Table 14.68 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

14.3.51 RSCAN0TMIDp — Transmit Buffer ID Register (p = 0 to 63)

Access: RSCAN0TMIDp can be read/written in 32-bit units.
RSCAN0TMIDpL and RSCAN0TMIDpH can be read/written in 16-bit units.
RSCAN0TMIDpLL, RSCAN0TMIDpLH, RSCAN0TMIDpHL, and RSCAN0TMIDpHH can be read/written in 8-bit units.

Address: RSCAN0TMIDp: <RSCAN0_base> + 1000_H + (10_H × p)
RSCAN0TMIDpL: <RSCAN0_base> + 1000_H + (10_H × p),
RSCAN0TMIDpH: <RSCAN0_base> + 1002_H + (10_H × p)
RSCAN0TMIDpLL: <RSCAN0_base> + 1000_H + (10_H × p),
RSCAN0TMIDpLH: <RSCAN0_base> + 1001_H + (10_H × p),
RSCAN0TMIDpHL: <RSCAN0_base> + 1002_H + (10_H × p),
RSCAN0TMIDpHH: <RSCAN0_base> + 1003_H + (10_H × p)

Value after reset: 0000 0000_H This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.69 RSCAN0TMIDp Register Contents

Bit Position	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

THLEN Bit

When this bit is set to 1, the transmit buffer label information and the number and type of the transmit source buffer are stored in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

14.3.52 RSCAN0TMPTRp — Transmit Buffer Pointer Register (p = 0 to 63)

Access: RSCAN0TMPTRp can be read/written in 32-bit units.
RSCAN0TMPTRpL and RSCAN0TMPTRpH can be read/written in 16-bit units.
RSCAN0TMPTRpLL, RSCAN0TMPTRpLH, RSCAN0TMPTRpHL, and RSCAN0TMPTRpHH can be read/written in 8-bit units.

Address: RSCAN0TMPTRp: <RSCAN0_base> + 1004_H + (10_H × p)
RSCAN0TMPTRpL: <RSCAN0_base> + 1004_H + (10_H × p),
RSCAN0TMPTRpH: <RSCAN0_base> + 1006_H + (10_H × p)
RSCAN0TMPTRpLL: <RSCAN0_base> + 1004_H + (10_H × p),
RSCAN0TMPTRpLH: <RSCAN0_base> + 1005_H + (10_H × p),
RSCAN0TMPTRpHL: <RSCAN0_base> + 1006_H + (10_H × p),
RSCAN0TMPTRpHH: <RSCAN0_base> + 1007_H + (10_H × p)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 14.70 RSCAN0TMPTRp Register Contents

Bit Position	Bit Name	Function
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 24	—	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCAN0TMIDp register is set to 0 (data frame). If the data length is set to 9 bytes or more, the transmit data is 8 bytes long.

When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

14.3.53 RSCAN0TMDF0p — Transmit Buffer Data Field 0 Register (p = 0 to 63)

Access: RSCAN0TMDF0p can be read/written in 32-bit units.
RSCAN0TMDF0pL and RSCAN0TMDF0pH can be read/written in 16-bit units.
RSCAN0TMDF0pLL, RSCAN0TMDF0pLH, RSCAN0TMDF0pHL, and RSCAN0TMDF0pHH can be read/written in 8-bit units.

Address: RSCAN0TMDF0p: <RSCAN0_base> + 1008_H + (10_H × p)
RSCAN0TMDF0pL: <RSCAN0_base> + 1008_H + (10_H × p),
RSCAN0TMDF0pH: <RSCAN0_base> + $100A_H$ + (10_H × p)
RSCAN0TMDF0pLL: <RSCAN0_base> + 1008_H + (10_H × p),
RSCAN0TMDF0pLH: <RSCAN0_base> + 1009_H + (10_H × p),
RSCAN0TMDF0pHL: <RSCAN0_base> + $100A_H$ + (10_H × p),
RSCAN0TMDF0pHH: <RSCAN0_base> + $100B_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.71 RSCAN0TMDF0p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB3[7:0]	Transmit Buffer Data Byte 3 Transmit Buffer Data Byte 2 Transmit Buffer Data Byte 1 Transmit Buffer Data Byte 0 Set the transmit buffer data.
23 to 16	TMDB2[7:0]	
15 to 8	TMDB1[7:0]	
7 to 0	TMDB0[7:0]	

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

14.3.54 RSCAN0TMDF1p — Transmit Buffer Data Field 1 Register (p = 0 to 63)

Access: RSCAN0TMDF1p can be read/written in 32-bit units.
RSCAN0TMDF1pL and RSCAN0TMDF1pH can be read/written in 16-bit units.
RSCAN0TMDF1pLL, RSCAN0TMDF1pLH, RSCAN0TMDF1pHL, and RSCAN0TMDF1pHH can be read/written in 8-bit units.

Address: RSCAN0TMDF1p: <RSCAN0_base> + $100C_H$ + (10_H × p)
RSCAN0TMDF1pL: <RSCAN0_base> + $100C_H$ + (10_H × p),
RSCAN0TMDF1pH: <RSCAN0_base> + $100E_H$ + (10_H × p)
RSCAN0TMDF1pLL: <RSCAN0_base> + $100C_H$ + (10_H × p),
RSCAN0TMDF1pLH: <RSCAN0_base> + $100D_H$ + (10_H × p),
RSCAN0TMDF1pHL: <RSCAN0_base> + $100E_H$ + (10_H × p),
RSCAN0TMDF1pHH: <RSCAN0_base> + $100F_H$ + (10_H × p)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.72 RSCAN0TMDF1p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB7[7:0]	Transmit Buffer Data Byte 7 Transmit Buffer Data Byte 6 Transmit Buffer Data Byte 5 Transmit Buffer Data Byte 4 Set the transmit buffer data.
23 to 16	TMDB6[7:0]	
15 to 8	TMDB5[7:0]	
7 to 0	TMDB4[7:0]	

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

14.3.55 RSCAN0TXQCCm — Transmit Queue Configuration and Control Register (m = 0 to 3)

Access: RSCAN0TXQCCm can be read/written in 32-bit units.
RSCAN0TXQCCmL and RSCAN0TXQCCmH can be read/written in 16-bit units.
RSCAN0TXQCCmLL, RSCAN0TXQCCmLH, RSCAN0TXQCCmHL, and RSCAN0TXQCCmHH can be read/written in 8-bit units.

Address: RSCAN0TXQCCm: <RSCAN0_base> + $03A0_H$ + (4_H × m)
RSCAN0TXQCCmL: <RSCAN0_base> + $03A0_H$ + (4_H × m),
RSCAN0TXQCCmH: <RSCAN0_base> + $03A2_H$ + (4_H × m)
RSCAN0TXQCCmLL: <RSCAN0_base> + $03A0_H$ + (4_H × m),
RSCAN0TXQCCmLH: <RSCAN0_base> + $03A1_H$ + (4_H × m),
RSCAN0TXQCCmHL: <RSCAN0_base> + $03A2_H$ + (4_H × m),
RSCAN0TXQCCmHH: <RSCAN0_base> + $03A3_H$ + (4_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.73 RSCAN0TXQCCm Register Contents

Bit Position	Bit Name	Function
31 to 14	—	Reserved These bits are always read as 0. The write value should always be 0.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the buffer becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to y (y = 2 to 15) makes the (y + 1)-buffer transmission queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 before modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from (m × 16 + 15) to (m × 16 + 0). For examples of how buffer allocation is done, see **Figure 14.9**. Modify these bits only in channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to 0010_B or more.

14.3.56 RSCAN0TXQSTSm — Transmit Queue Status Register (m = 0 to 3)

Access: RSCAN0TXQSTSm can be read/written in 32-bit units.
RSCAN0TXQSTSmL and RSCAN0TXQSTSmH can be read/written in 16-bit units.
RSCAN0TXQSTSmLL, RSCAN0TXQSTSmLH, RSCAN0TXQSTSmHL, and RSCAN0TXQSTSmHH can be read/written in 8-bit units.

Address: RSCAN0TXQSTSm: <RSCAN0_base> + $03C0_H$ + (04_H × m)
RSCAN0TXQSTSmL: <RSCAN0_base> + $03C0_H$ + (04_H × m),
RSCAN0TXQSTSmH: <RSCAN0_base> + $03C2_H$ + (04_H × m)
RSCAN0TXQSTSmLL: <RSCAN0_base> + $03C0_H$ + (04_H × m),
RSCAN0TXQSTSmLH: <RSCAN0_base> + $03C1_H$ + (04_H × m),
RSCAN0TXQSTSmHL: <RSCAN0_base> + $03C2_H$ + (04_H × m),
RSCAN0TXQSTSmHH: <RSCAN0_base> + $03C3_H$ + (04_H × m)

Value after reset: $0000\ 0001_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQ FLL	TXQ EMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.74 RSCAN0TXQSTSm Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCAN0TXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCAN0TXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCAN0TXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is pending in the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

14.3.57 RSCAN0TXQPCTRm — Transmit Queue Pointer Control Register (m = 0 to 3)

Access: RSCAN0TXQPCTRm can be read/written in 32-bit units.
RSCAN0TXQPCTRmL and RSCAN0TXQPCTRmH can be read/written in 16-bit units.
RSCAN0TXQPCTRmLL, RSCAN0TXQPCTRmLH, RSCAN0TXQPCTRmHL, and RSCAN0TXQPCTRmHH can be read/written in 8-bit units.

Address: RSCAN0TXQPCTRm: <RSCAN0_base> + $03E0_H$ + (04_H × m)
RSCAN0TXQPCTRmL: <RSCAN0_base> + $03E0_H$ + (04_H × m),
RSCAN0TXQPCTRmH: <RSCAN0_base> + $03E2_H$ + (04_H × m)
RSCAN0TXQPCTRmLL: <RSCAN0_base> + $03E0_H$ + (04_H × m),
RSCAN0TXQPCTRmLH: <RSCAN0_base> + $03E1_H$ + (04_H × m),
RSCAN0TXQPCTRmHL: <RSCAN0_base> + $03E2_H$ + (04_H × m),
RSCAN0TXQPCTRmHH: <RSCAN0_base> + $03E3_H$ + (04_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.75 RSCAN0TXQPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved The write value should always be 0.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FF_H to these bits moves the write pointer of the transmit queue to the next queue buffer.

TXQPC[7:0] Bits

Writing FF_H to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCAN0TMIDp, RSCAN0TMPTRp, RSCAN0TMDF0p, and RSCAN0TMDF1p registers (p = 15, 31, 47, and 63) before writing FF_H to the TXQPC[7:0] bits.

Write FF_H only when the TXQE bit in the RSCAN0TXQCCm register set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCAN0TXQSTSm register set to 0 (not full).

14.3.58 RSCAN0THLCCm — Transmit History Configuration and Control Register (m = 0 to 3)

Access: RSCAN0THLCCm can be read/written in 32-bit units.
RSCAN0THLCCmL and RSCAN0THLCCmH can be read/written in 16-bit units.
RSCAN0THLCCmLL, RSCAN0THLCCmLH, RSCAN0THLCCmHL, and RSCAN0THLCCmHH can be read/written in 8-bit units.

Address: RSCAN0THLCCm: <RSCAN0_base> + 0400_H + (04_H × m)
RSCAN0THLCCmL: <RSCAN0_base> + 0400_H + (04_H × m),
RSCAN0THLCCmH: <RSCAN0_base> + 0402_H + (04_H × m)
RSCAN0THLCCmLL: <RSCAN0_base> + 0400_H + (04_H × m),
RSCAN0THLCCmLH: <RSCAN0_base> + 0401_H + (04_H × m),
RSCAN0THLCCmHL: <RSCAN0_base> + 0402_H + (04_H × m),
RSCAN0THLCCmHH: <RSCAN0_base> + 0403_H + (04_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 14.76 RSCAN0THLCCm Register Contents

Bit Position	Bit Name	Function
31 to 11	—	Reserved These bits are always read as 0. The write value should always be 0.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only while the THLE bit is set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.

14.3.59 RSCAN0THLSTSm — Transmit History Status Register (m = 0 to 3)

Access: RSCAN0THLSTSm can be read/written in 32-bit units.
RSCAN0THLSTSmL and RSCAN0THLSTSmH can be read/written in 16-bit units.
RSCAN0THLSTSmLL, RSCAN0THLSTSmLH, RSCAN0THLSTSmHL, and RSCAN0THLSTSmHH can be read/written in 8-bit units.

Address: RSCAN0THLSTSm: <RSCAN0_base> + 0420_H + (04_H × m)
RSCAN0THLSTSmL: <RSCAN0_base> + 0420_H + (04_H × m),
RSCAN0THLSTSmH: <RSCAN0_base> + 0422_H + (04_H × m)
RSCAN0THLSTSmLL: <RSCAN0_base> + 0420_H + (04_H × m),
RSCAN0THLSTSmLH: <RSCAN0_base> + 0421_H + (04_H × m),
RSCAN0THLSTSmHL: <RSCAN0_base> + 0422_H + (04_H × m),
RSCAN0THLSTSmHH: <RSCAN0_base> + 0423_H + (04_H × m)

Value after reset: $0000\ 0001_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THL EMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

Table 14.77 RSCAN0THLSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	THLIF	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCAN0THLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

THLELT Flag

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

To clear the flag, the program must write 0 to the corresponding bit to be cleared and 1 to all other bits, using the Store instruction.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear the THLIF and THLELT flags to 0, the program must write 0 to these flags. Use a store instruction to write 0 to these flags and write 1 to the other flags.

14.3.60 RSCAN0THLACCm — Transmit History Access Register (m = 0 to 3)

Access: RSCAN0THLACCm is read-only in 32-bit units.
RSCAN0THLACCmL and RSCAN0THLACCmH are read-only in 16-bit units.
RSCAN0THLACCmLL, RSCAN0THLACCmLH, RSCAN0THLACCmHL, and RSCAN0THLACCmHH are read-only in 8-bit units.

Address: RSCAN0THLACCm: <RSCAN0_base> + 1800_H + (04_H × m)
RSCAN0THLACCmL: <RSCAN0_base> + 1800_H + (04_H × m),
RSCAN0THLACCmH: <RSCAN0_base> + 1802_H + (04_H × m)
RSCAN0THLACCmLL: <RSCAN0_base> + 1800_H + (04_H × m),
RSCAN0THLACCmLH: <RSCAN0_base> + 1801_H + (04_H × m),
RSCAN0THLACCmHL: <RSCAN0_base> + 1802_H + (04_H × m),
RSCAN0THLACCmHH: <RSCAN0_base> + 1803_H + (04_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 14.78 RSCAN0THLACCm Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	—	Reserved This bit is always read as 0.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the transmit history data stored in the transmit history buffer.

14.3.61 RSCAN0THLPCTRm — Transmit History Pointer Control Register (m = 0 to 3)

Access: RSCAN0THLPCTRm can be read/written in 32-bit units.
RSCAN0THLPCTRmL and RSCAN0THLPCTRmH can be read/written in 16-bit units.
RSCAN0THLPCTRmLL, RSCAN0THLPCTRmLH, RSCAN0THLPCTRmHL, and RSCAN0THLPCTRmHH can be read/written in 8-bit units.

Address: RSCAN0THLPCTRm: <RSCAN0_base> + 0440_H + (04_H × m)
RSCAN0THLPCTRmL: <RSCAN0_base> + 0440_H + (04_H × m),
RSCAN0THLPCTRmH: <RSCAN0_base> + 0442_H + (04_H × m)
RSCAN0THLPCTRmLL: <RSCAN0_base> + 0440_H + (04_H × m),
RSCAN0THLPCTRmLH: <RSCAN0_base> + 0441_H + (04_H × m),
RSCAN0THLPCTRmHL: <RSCAN0_base> + 0442_H + (04_H × m),
RSCAN0THLPCTRmHH: <RSCAN0_base> + 0443_H + (04_H × m)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.79 RSCAN0THLPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved The write value should always be 0.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FF_H to these bits moves the read pointer to the next unread data in the transmit history buffer. Set Value: FF_H

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF_H, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCAN0THLSTSm register is decremented. Write FF_H to the THLPC[7:0] bits after reading from the RSCAN0THLACCm register.

Write FF_H to the THLPC[7:0] bits only when the THLE bit in the RSCAN0THLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCAN0THLSTSm register is 0.

14.3.62 RSCAN0GTSTCFG — Global Test Configuration Register

Access: RSCAN0GTSTCFG can be read/written in 32-bit units.
RSCAN0GTSTCFGL and RSCAN0GTSTCFGH can be read/written in 16-bit units.
RSCAN0GTSTCFGLL, RSCAN0GTSTCFGLH, RSCAN0GTSTCFGHL, and RSCAN0GTSTCFGHH can be read/written in 8-bit units.

Address: RSCAN0GTSTCFG: <RSCAN0_base> + 0468_H
RSCAN0GTSTCFGL: <RSCAN0_base> + 0468_H, RSCAN0GTSTCFGH: <RSCAN0_base> + $046A_H$
RSCAN0GTSTCFGLL: <RSCAN0_base> + 0468_H, RSCAN0GTSTCFGLH: <RSCAN0_base> + 0469_H,
RSCAN0GTSTCFGHL: <RSCAN0_base> + $046A_H$, RSCAN0GTSTCFGHH: <RSCAN0_base> + $046B_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 14.80 RSCAN0GTSTCFG Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0. The write value should always be 0.
22 to 16	RTMPS[6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 47 ($2F_H$).
15 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	C3ICBCE	CAN3 Inter-channel Communication Test Enable 0: CAN3 inter-channel communication test is disabled. 1: CAN3 inter-channel communication test is enabled.
2	C2ICBCE	CAN2 Inter-channel Communication Test Enable 0: CAN2 inter-channel communication test is disabled 1: CAN2 inter-channel communication test is enabled.
1	C1ICBCE	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCAN0GTSTCFG register only in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value in the range of 00_H to $2F_H$, inclusive.

C3ICBCE Bit

Setting this bit to 1 enables the channel 3 inter-channel communication test.

C2ICBCE Bit

Setting this bit to 1 enables the channel 2 inter-channel communication test.

C1ICBCE Bit

Setting this bit to 1 enables the channel 1 inter-channel communication test.

C0ICBCE Bit

Setting this bit to 1 enables the channel 0 inter-channel communication test.

14.3.63 RSCAN0GTSTCTR — Global Test Control Register

Access: RSCAN0GTSTCTR can be read/written in 32-bit units.
RSCAN0GTSTCTRL and RSCAN0GTSTCTRH can be read/written in 16-bit units.
RSCAN0GTSTCTRLL, RSCAN0GTSTCTRLH, RSCAN0GTSTCTRHL, and RSCAN0GTSTCTRHH can be read/written in 8-bit units.

Address: RSCAN0GTSTCTR: <RSCAN0_base> + $046C_H$
RSCAN0GTSTCTRL: <RSCAN0_base> + $046C_H$, RSCAN0GTSTCTRH: <RSCAN0_base> + $046E_H$
RSCAN0GTSTCTRLL: <RSCAN0_base> + $046C_H$, RSCAN0GTSTCTRLH: <RSCAN0_base> + $046D_H$,
RSCAN0GTSTCTRHL: <RSCAN0_base> + $046E_H$, RSCAN0GTSTCTRHH: <RSCAN0_base> + $046F_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.81 RSCAN0GTSTCTR Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	RTME	RAM Test Enable 0: RAM test is stopped. 1: RAM test is started.
1	—	Reserved This bit is always read as 0. The write value should always be 0.
0	ICBCTME	Inter-Channel Communication Test Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

RTME Bit

Setting this bit to 1 enables the RAM test. Modify this bit only in global test mode.

1. Set the GMDC[1:0] bits in the RSCAN0GCTR register to 10_B (global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 to 3) in the RSCAN0GTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.

14.3.64 RSCAN0GLOCKK — Global Lock Key Register

Access: RSCAN0GLOCKK can be written-only in 32-bit units.
RSCAN0GLOCKKL and RSCAN0GLOCKKH can be written-only in 16-bit units.

Address: RSCAN0GLOCKK: <RSCAN0_base> + $047C_H$
RSCAN0GLOCKKL: <RSCAN0_base> + $047C_H$, RSCAN0GLOCKKH: <RSCAN0_base> + $047E_H$

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CAN module is in global test mode.

Table 14.82 RSCAN0GLOCKK Register Contents

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, an undefined value is returned. The write value should be 0.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCAN0GLOCKK register releases protection of special test bits and is write-only.

For the protection release data, refer to **Section 14.5.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCAN0GTSTCTR register.

After the protection has been released, writing to the I/O register area (<RCAN0_base> + 0000_H to <RCAN0_base> + $04FF_H$) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/wring to other areas does not enable the protection.

14.3.65 RSCAN0RPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access: RSCAN0RPGACCr can be read/written in 32-bit units.
RSCAN0RPGACCrL and RSCAN0RPGACCrH can be read/written in 16-bit units.
RSCAN0RPGACCrLL, RSCAN0RPGACCrLH, RSCAN0RPGACCrHL, and RSCAN0RPGACCrHH can be read/written in 8-bit units.

Address: RSCAN0RPGACCr: <RSCAN0_base> + 1900_H + (04_H × r)
RSCAN0RPGACCrL: <RSCAN0_base> + 1900_H + (04_H × r),
RSCAN0RPGACCrH: <RSCAN0_base> + 1902_H + (04_H × r)
RSCAN0RPGACCrLL: <RSCAN0_base> + 1900_H + (04_H × r),
RSCAN0RPGACCrLH: <RSCAN0_base> + 1901_H + (04_H × r),
RSCAN0RPGACCrHL: <RSCAN0_base> + 1902_H + (04_H × r),
RSCAN0RPGACCrHH: <RSCAN0_base> + 1903_H + (04_H × r)

Value after reset: $0000\ 0000_H$ This register is initialized by any reset.

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

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

Table 14.83 RSCAN0RPGACCr Register Contents

Bit Position	Bit Name	Function
31 to 0	RDTA [31:0]	RAM Data Test Access Data can be read and written in RSCAN RAM.

Modify the RSCAN0RPGACCr register in global test mode with the RTME bit in the RSCAN0GTSTCTR register set to 1 (RAM test is enabled).

The RSCAN0RPGACCr register is readable and writable when the RTME bit is set to 1.

14.4 Function

14.4.1 Interrupt Sources

The RS-CAN module has 14 interrupts that are grouped into global interrupts and channel interrupts.

- Global interrupts (2 sources):
  Receive FIFO interrupt
  Global error interrupt
- Channel interrupts (3 sources/channel):
  CANm transmit interrupt (m = 0 to 3)
  - CANm transmission complete interrupt
  - CANm transmission abort interrupt
  - CANm transmit/receive FIFO transmission complete interrupt (in transmit mode, gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue Interrupt

  CANm transmit/receive FIFO reception complete interrupt (in transmit mode, gateway mode)
  CANm error interrupt

NOTE

CANm wakeup interrupt is generated using the external interrupt request.

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CAN module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.

Table 14.84 lists the CAN interrupt sources. **Figure 14.2** shows the CAN global interrupt block diagram. **Figure 14.3** shows the CAN channel interrupt block diagram.

Table 14.84 List of CAN Interrupt Sources

	Interrupt Source	Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit
Global interrupts	Receive FIFO 0	RFIF in the RSCAN0RFSTS0 register	RFIE in the RSCAN0RFCC0 register
	Receive FIFO 1	RFIF in the RSCAN0RFSTS1 register	RFIE in the RSCAN0RFCC1 register
	Receive FIFO 2	RFIF in the RSCAN0RFSTS2 register	RFIE in the RSCAN0RFCC2 register
	Receive FIFO 3	RFIF in the RSCAN0RFSTS3 register	RFIE in the RSCAN0RFCC3 register
	Receive FIFO 4	RFIF in the RSCAN0RFSTS4 register	RFIE in the RSCAN0RFCC4 register
	Receive FIFO 5	RFIF in the RSCAN0RFSTS5 register	RFIE in the RSCAN0RFCC5 register
	Receive FIFO 6	RFIF in the RSCAN0RFSTS6 register	RFIE in the RSCAN0RFCC6 register
	Receive FIFO 7	RFIF in the RSCAN0RFSTS7 register	RFIE in the RSCAN0RFCC7 register
	Global error	DEF in the RSCAN0GERFL register MES in the RSCAN0GERFL register THLES in the RSCAN0GERFL register	DEIE in the RSCAN0GCTR register MEIE in the RSCAN0GCTR register THLEIE in the RSCAN0GCTR register
Channel interrupts (m = 0 to 3)	CANm transmission complete	TMTRF[1:0] in the RSCAN0TMSTSp register	TMIE in the RSCAN0TMIECn register
	CANm transmission abort	TMTRF[1:0] in the RSCAN0TMSTSp register	TAIE in the RSCAN0CmCTR register
	CANm transmit/receive FIFO transmission complete	CFTXIF in the RSCAN0CFSTSk register	CFTXIE in the RSCAN0CFCCk register
	CANm transmit/receive FIFO reception complete	CFRXIF in the RSCAN0CFSTSk register	CFRXIE in the RSCAN0CFCCk register
	CANm transmit queue	TXQIF in the RSCAN0TXQSTSm register	TXQIE in the RSCAN0TXQCCm register
	CANm transmit history	THLIF in the RSCAN0THLSTSm register	THLIE in the RSCAN0THLCCm register
	CANm error	• BEF in the RSCAN0CmERFL register • ALF in the RSCAN0CmERFL register • BLF in the RSCAN0CmERFL register • OVLF in the RSCAN0CmERFL register • BORF in the RSCAN0CmERFL register • BOEF in the RSCAN0CmERFL register • EPF in the RSCAN0CmERFL register • EWF in the RSCAN0CmERFL register	• BEIE in the RSCAN0CmCTR register • ALIE in the RSCAN0CmCTR register • BLIE in the RSCAN0CmCTR register • OLIE in the RSCAN0CmCTR register • BORIE in the RSCAN0CmCTR register • BOEIE in the RSCAN0CmCTR register • EPIE in the RSCAN0CmCTR register • EWIE in the RSCAN0CmCTR register

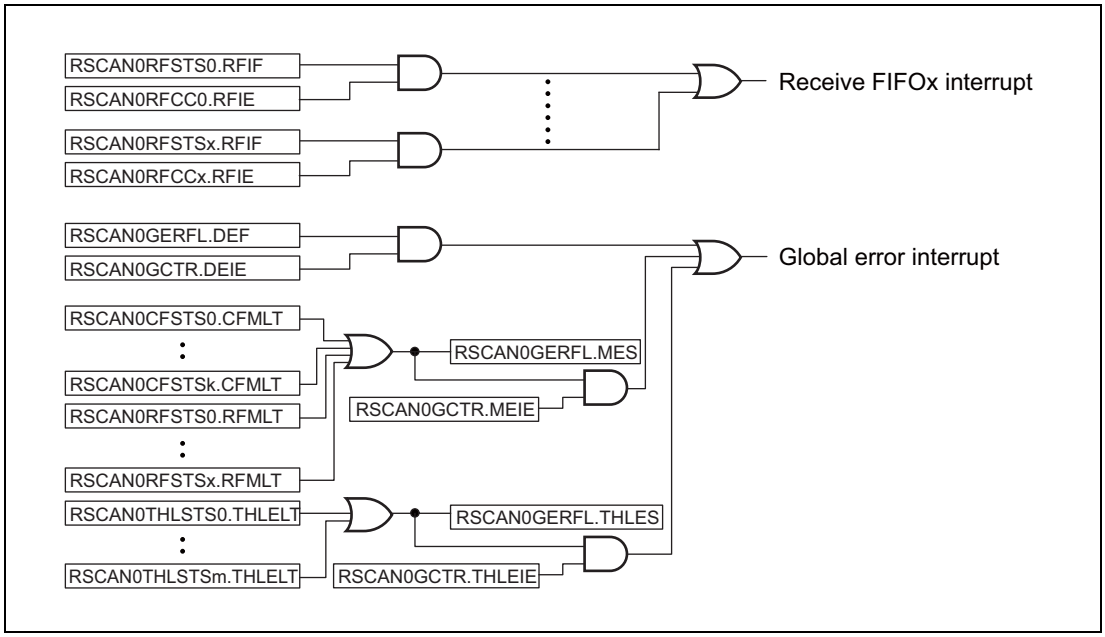


Figure 14.2 CAN Global Interrupt Block Diagram

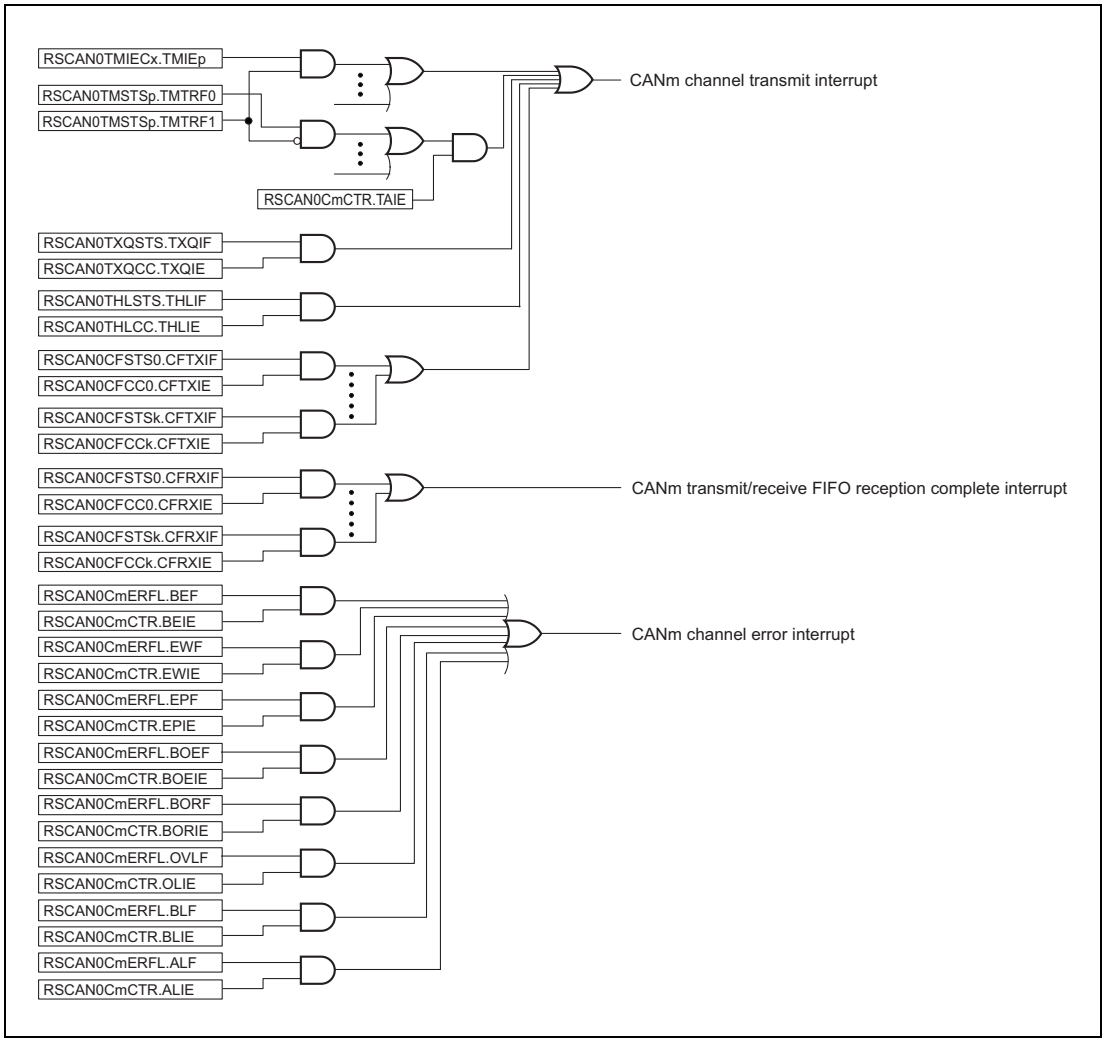


Figure 14.3 CAN Channel Interrupt Block Diagram

14.4.2 CAN Modes

The RS-CAN module has four global modes to control the entire RS-CAN module status and four channel modes to control individual channel status. Details of global modes are described in **Section 14.4.2.1, Global Modes**, and details of channel modes are described in **Section 14.4.2.2, Channel Modes**.

- Global stop mode: Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode: Performs initial settings for the entire module.
- Global test mode: Performs test settings and performs the RAM test.
- Global operating mode: Makes the entire module operable.
- Channel stop mode: Stops the channel clock.
- Channel reset mode: Performs initial settings for the channels.
- Channel halt mode: Stops CAN communication and allows channel testing.
- Channel communication mode: Performs CAN communication.

14.4.2.1 Global Modes

Figure 14.4 shows the transitions of global modes.

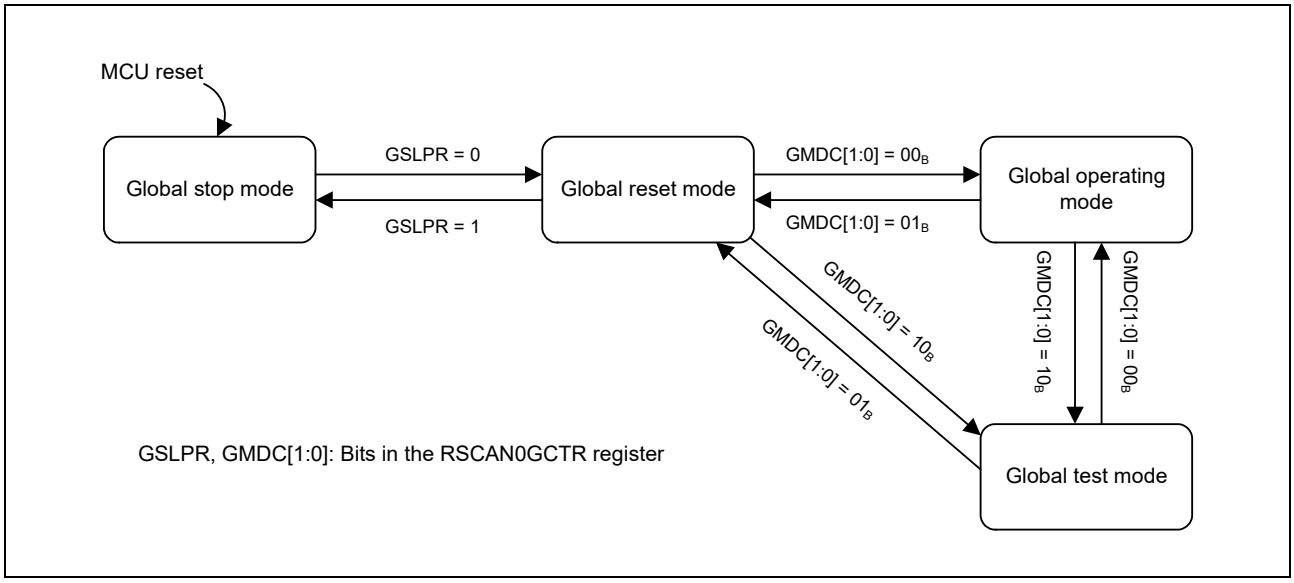


Figure 14.4 Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. **Table 14.85** shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

Table 14.85 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

Channel Mode before Setting	Channel Mode after Setting			
	GMDC[1:0] = 00_B GSLPR = 0 (Global Operation)	GMDC[1:0] = 10_B GSLPR = 0 (Global Test)	GMDC[1:0] = 01_B GSLPR = 0 (Global Reset)	GMDC[1:0] = 01_B GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

Note: GMDC[1:0] and GSLPR: Bits in the RSCAN0GCTR register

Table 14.86 shows the global mode transition time.

Table 14.86 Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three pclk cycles
Global reset	Global stop	Three pclk cycles
Global reset	Global test	Ten pclk cycles
Global reset	Global operating	Ten pclk cycles
Global test	Global reset	Three pclk cycles
Global test	Global operating	Three pclk cycles
Global operating	Global reset	Three pclk cycles
Global operating	Global test	Two CAN frames*1

Note 1. CAN frame time of the lowest communication speed of the channels in use

(1) Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.

After the MCU is reset, the CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCAN0GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCAN0CmCTR register to 1 (channel stop mode). After that, if all channels are forced to transition to channel stop mode, the CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

(2) Global Reset Mode

In global reset mode, RS-CAN module settings are performed. When the RS-CAN module transitions to global reset mode, some registers are initialized. **Table 14.89** and **Table 14.90** list the registers to be initialized.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 01_B sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR registers (m = 0 to 3) to 01_B (channel reset mode). If all channels are forced to transition to channel reset mode, the CAN module transitions to global reset mode. Channels that are

already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01_B).

(3) Global Test Mode

In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 10_B sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR register to 10_B (channel halt mode). If all channels are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

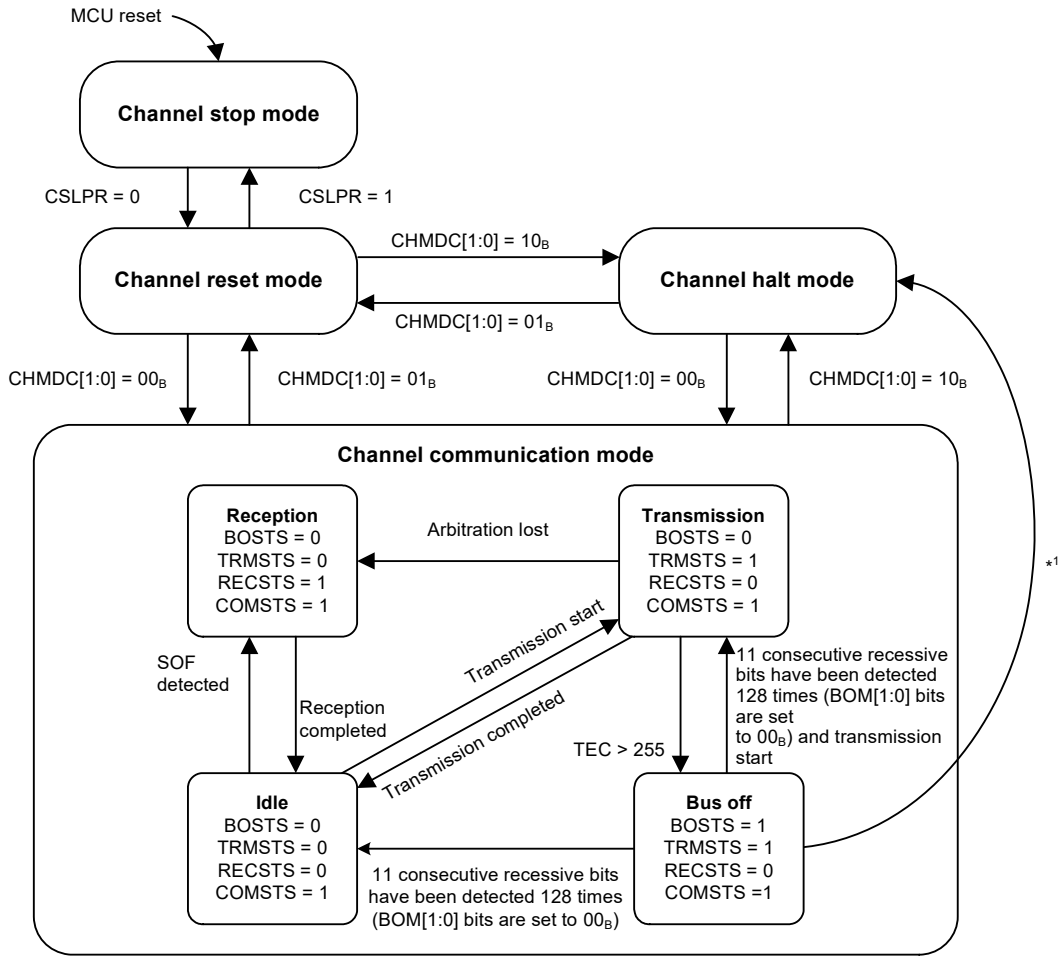
(4) Global Operating Mode

The RS-CAN module operates in global operating mode.

When the GMDC[1:0] bits in the RSCAN0GCTR register are set to 00_B, the RS-CAN module transitions to global operating mode.

14.4.2.2 Channel Modes

Figure 14.5 shows a channel mode state transition chart. **Table 14.87** shows the channel mode transition time.



CHMDC[1:0], CSLPR, BOM[1:0]: Bits in the RSCAN0CmCTR register (m = 0 to 3)
BOSTS, TRMSTS, RECSTS, COMSTS: Bits in the RSCAN0CmSTS register

Note 1. Timing of transition from bus off state to channel halt mode
- When BOM[1:0] = 01_B: Transition to channel halt mode when TEC exceeds 255
- When BOM[1:0] = 10_B: Transition to channel halt mode when 11 consecutive recessive bits have been detected 128 times
- When BOM[1:0] = 11_B: Transition to channel halt mode when the CHMDC[1:0] bits are set to 10_B

Figure 14.5 Transitions of Global Modes

Table 14.87 Channel Mode Transition Time (m = 0 to 3)

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel reset	Channel halt	Three CANm bit times
Channel reset	Channel communication	Two CANm bit times
Channel halt	Channel reset	Three pclk cycles
Channel halt	Channel communication	Three CANm bit times
Channel communication	Channel reset	Three pclk cycles
Channel communication	Channel halt	Two CANm frames

(1) Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. Channel-related registers can be read, but writing data to them is prohibited (except for the CSLPR bit). Register values are retained.

Each channel enters channel stop mode after the MCU is reset. Channels also transition to channel stop mode when the CSLPR bit in the RSCAN0CmCTR register (m = 0 to 3) is set to 1 (channel stop mode) in channel reset mode. The CSLPR bit should not be modified in channel communication mode and channel halt mode.

(2) Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. **Table 14.89** lists the registers to be initialized.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01_B (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. **Table 14.88** shows the operation when the CHMDC[1:0] bits are set to 01_B (channel reset mode) during CAN communication.

(3) Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

Table 14.88 shows operation when the CHMDC[1:0] bits are set to 10_B (channel halt mode) during CAN communication.

Table 14.88 Operation a Channel Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01_B)	Transitions to channel reset mode before reception is completed.*1	Transitions to channel reset mode before transmission is completed.*1	Transitions to channel reset mode before bus off recovery.
Channel halt*3 (CHMDC[1:0] = 10_B)	Transitions to channel halt mode after reception is completed.*2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00_B] Transitions to channel halt mode (CHMDC[1:0] = 10_B) only after bus off recovery. [When BOM[1:0] = 01_B] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10_B] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11_B] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10_B before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10_B and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01_B.

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCAN0CmERFL register that becomes 1 when dominant lock is detected.

Note 3. When the transition from channel reset mode to channel wait mode is to be made, set the RSCAN0CmCFG register in channel reset mode and then shift to channel wait mode.

(4) Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle : Neither reception nor transmission is in progress.
- Reception : Receiving a message sent from another node.
- Transmission : Transmitting a message.
- Bus off : Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 00_B, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCAN0CmSTS register (m = 0 to 3) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

(5) Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.

The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCAN0CmCTR register.

- When BOM[1:0] = 00_B:
  Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are initialized to 00_H and the BORF flag in the RSCAN0CmERFL register is set to 1 (bus off recovery is detected). When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 10_B (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01_B:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10_B and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H. The BORF flag is not set to 1 and no bus off recovery interrupt request is generated.
- When BOM[1:0] = 10_B:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10_B. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H and the BORF flag is set to 1.
- When BOM[1:0] = 11_B:
  When the CHMDC[1:0] bits are set to 10_B in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H but the BORF flag is not set to 1.
  However, the BORF flag becomes 1 if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10_B.

If the RS-CAN module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01_B or 10_B is made only when the CHMDC[1:0] bits are 00_B (channel communication mode). Furthermore, setting the RTBO bit in the RSCAN0CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to 00_H. Write 1 to the RTBO bit only when the BOM[1:0] value is 00_B.

Table 14.89 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit / Flag
RSCAN0CmCTR register	CTMS[1:0], CTME, CHMDC[1:0]
RSCAN0CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, REC[7:0], TEC[7:0]
RSCAN0CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCAN0CFTISTS register	CFnTXIF
RSCAN0TMCp register	TMOM, TMTAR, TMTR
RSCAN0TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCAN0TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTCSTSy register	TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTASTSy register	TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TXQCCm register	TXQE
RSCAN0TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCAN0THLCCm register	THLE
RSCAN0THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCAN0GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m = 0 to 3)

Table 14.90 Registers Initialized Only in Global Reset Mode

Register	Bit / Flag
RSCAN0GSTS register	GHLTSTS
RSCAN0GERFL register	THLES, MES, DEF
RSCAN0GTSC register	TS[15:0]
RSCAN0RMNDy register	RMNSq
RSCAN0RFCCm register	RFE
RSCAN0RFSTSm register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCAN0FESTS register	CFnEMP, RFnEMP
RSCAN0FFSTS register	CFnFLL, RFnFLL
RSCAN0FMSTS register	CFnMLT, RFnMLT
RSCAN0RFISTS register	RFnIF
RSCAN0CFRISTS register	CFnRXIF
RSCAN0GTSTCFG register	RTMPS[6:0], C0ICBCE, C1ICBCE, C2ICBCE, C3ICBCE
RSCAN0GTSTCTR register	RTME, ICBCTME

14.4.3 Reception Function

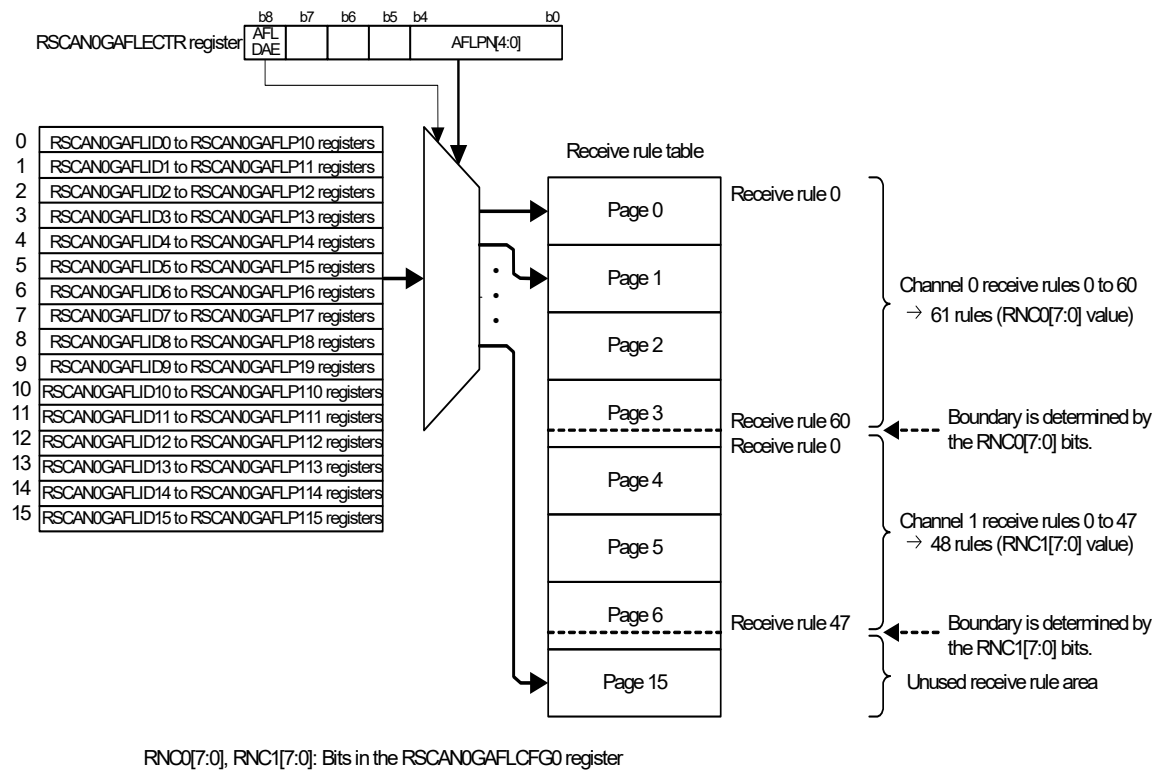
There are two reception types.

- Reception by receive buffers:
  Zero to 64 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
  Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

14.4.3.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 256 receive rules can be registered in this module that has four channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. **Figure 14.6** illustrates how receive rules are registered.



Note: Receive rules for each channel are set in contiguous blocks.
It is not possible to configure channel 1 rules in the block reserved for channel 0 rules.

Figure 14.6 Entry of Receive Rules (for Setting Channel 0 and 1)

Each receive rule consists of 16 bytes in the RSCAN0GAFLIDj, RSCAN0GAFLMj, RSCAN0GAFLP0j, and RSCAN0GAFLP1j registers (j = 0 to 15). The RSCAN0GAFLIDj register (j = 0 to 15) is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCAN0GAFLMj register is used to set mask, the RSCAN0GAFLP0j register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCAN0GAFLP1j register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

(1) Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCAN0GAFLMj register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

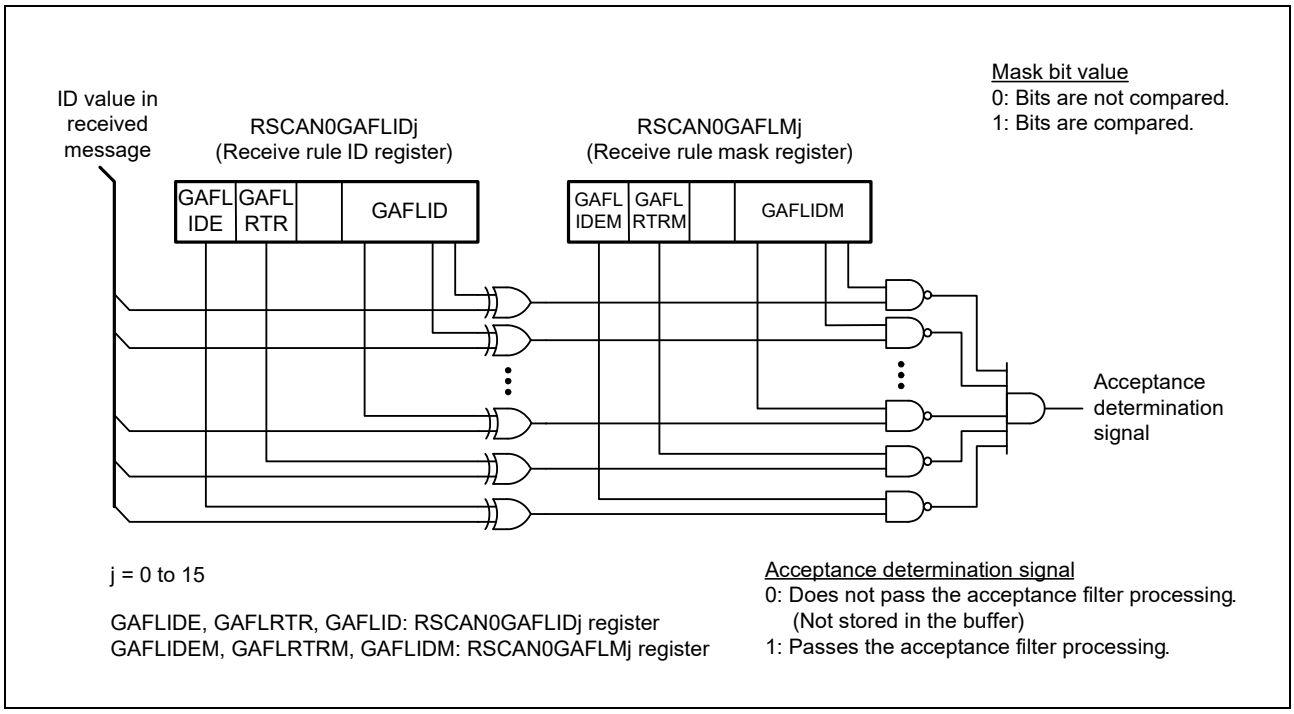


Figure 14.7 Acceptance Filter Function

(2) DLC Filter Processing

When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of 00_H is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCAN0GERFL register is set to 1 (a DLC error is present).

(3) Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCAN0GAFLP0j register (j = 0 to 15) and by the RSCAN0GAFLP1j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

(4) Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCAN0GAFLP0j register.

(5) Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCAN0GCFG register to 1 (mirror function is enabled).

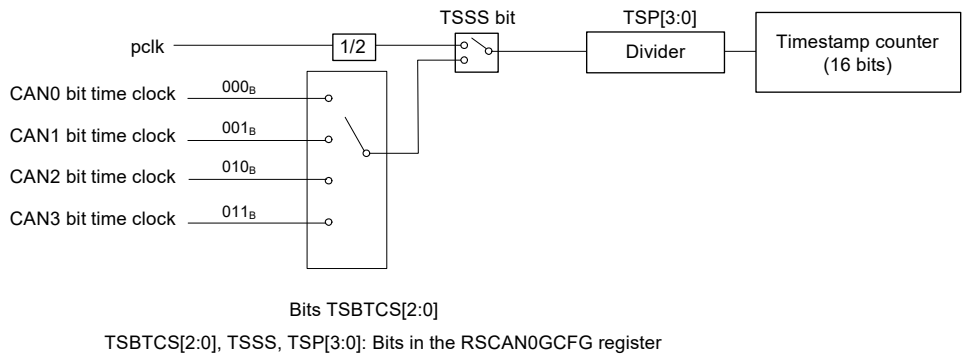
When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCAN0GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

(6) Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message receive time. The timestamp counter value is fetched at the start-of-frame (SOF) timing of a message and is then stored in a receive buffer or a FIFO buffer together with the message ID and data. Either pclk/2 or the CANm bit time clock (m = 0 to 3) may be selected as a timestamp counter clock source using the TSBTCS[2:0] and TSSS bits in the RSCAN0GCFG register. The timestamp counter count source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCAN0GCFG register.

When the CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to 0000_H by setting the TSRST bit in the RSCAN0GCTR register to 1.



Note: When specifying pclk as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.

Figure 14.8 Timestamp Function Block Diagram

14.4.4 Transmission Functions

There are three types of transmission.

- Transmission using transmit buffers:
  Each channel has 16 buffers.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:
  Up to 16 transmit buffers per channel can be allocated to the transmit queues. Transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 14.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

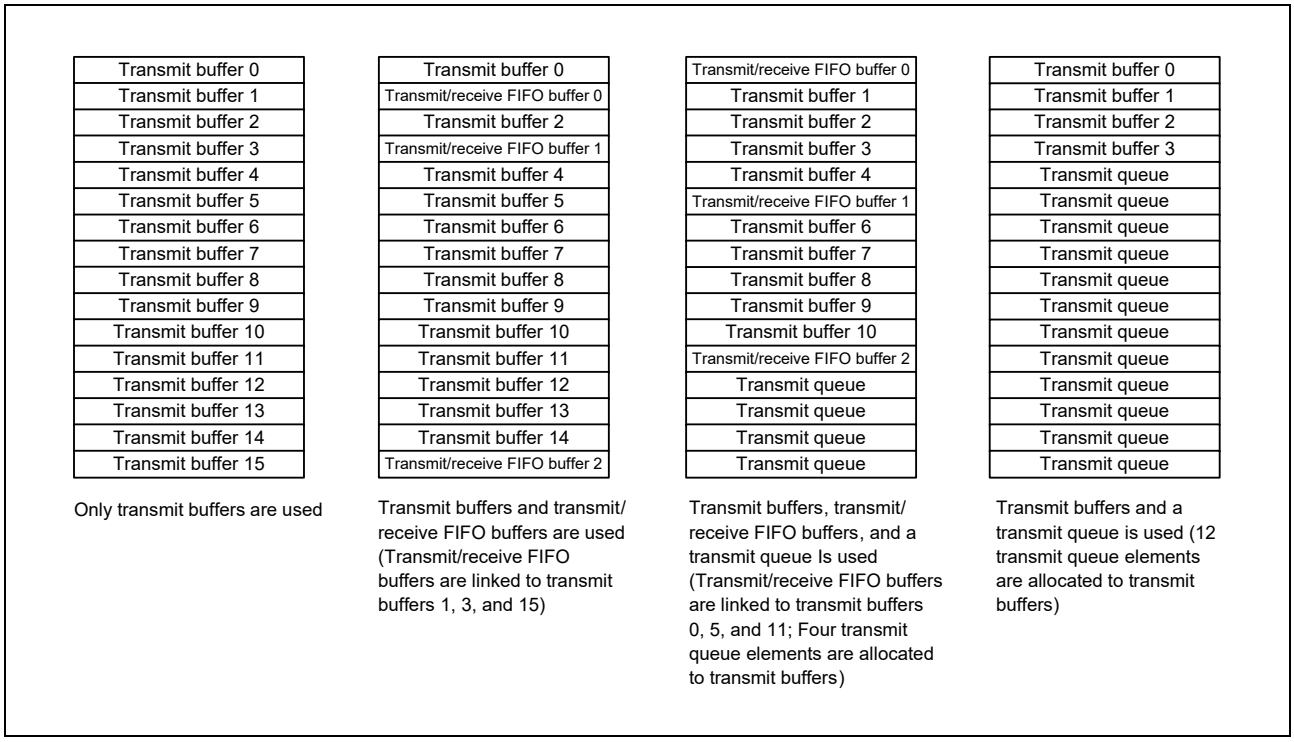


Figure 14.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

14.4.4.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCAN0GCFG register.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence.

When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.

When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit.

14.4.4.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCAN0TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register (p = 0 to 63). When transmission completes successfully, the TMTRF[1:0] flag is set to 10_B (transmission has been completed (without transmission abort request)) or 11_B (transmission has been completed (with transmission abort request)).

(1) Transmission Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCAN0TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCAN0TMCp register is set to 1 (transmission abort is requested), the transmit request is canceled. When transmission abort is completed, the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01_B (transmission abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

(2) One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCAN0TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.

The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10_B or 11_B. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01_B (transmission abort has been completed).

14.4.4.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCAN0CFCCk register (k = 0 to 11). Messages are transmitted sequentially on a first-in, first-out basis.

Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCAN0CFCCk register. When the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

(1) Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCAN0CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCAN0CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00_H.

Select an interval timer count source using the CFITR and CFITSS bits in the RSCAN0CFCCk register. When the CFITR and CFITSS bits are set to 00_B, the count source is obtained by dividing pclk/2 clock by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10_B, the count source is obtained by dividing pclk/2 by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register × 10). When the CFITR and CFITSS bits are set to $x1_B$, the CANm bit time clock is used as a count source.

The interval time is calculated by the following equations where m is the value set to ITRCP[15:0] and n is the set CFITT[7:0] value.

- When CFITR and CFITSS = 00_B (fPBA is the frequency of pclk):

$$\frac{1}{fPBA} \times 2 \times m \times n$$

- When CFITR and CFITSS = 10_B:

$$\frac{1}{fPBA} \times 2 \times m \times 10 \times n$$

- When CFITR and CFITSS = $x1_B$ (fCANBIT is the frequency of CANm bit time clock):

$$\frac{1}{fCANBIT} \times n$$

Figure 14.10 shows the interval timer block diagram.

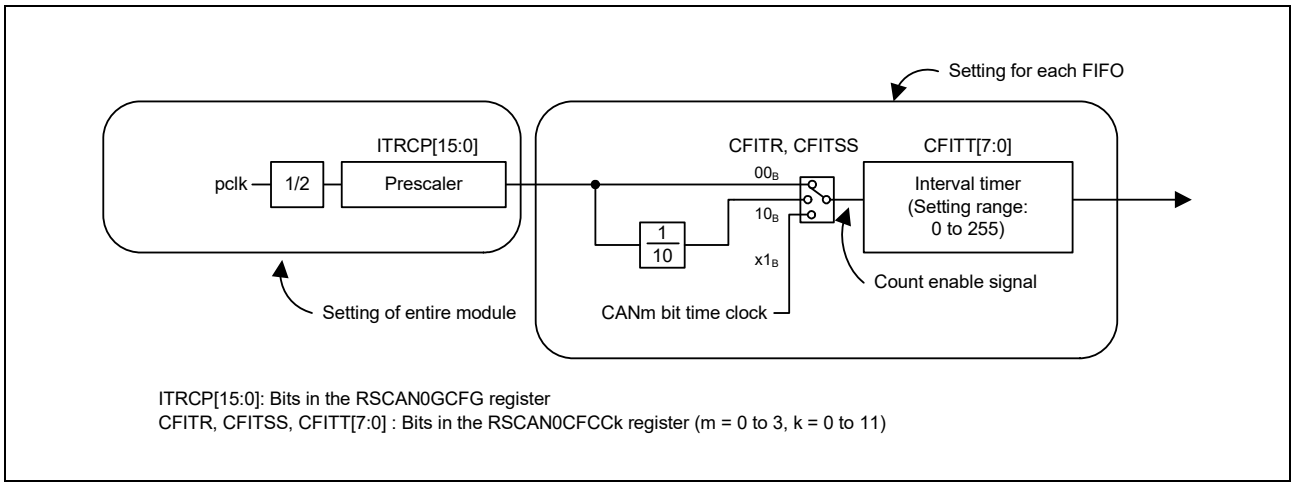


Figure 14.10 Interval Timer Block Diagram

Figure 14.11 shows the interval timer timing diagram.

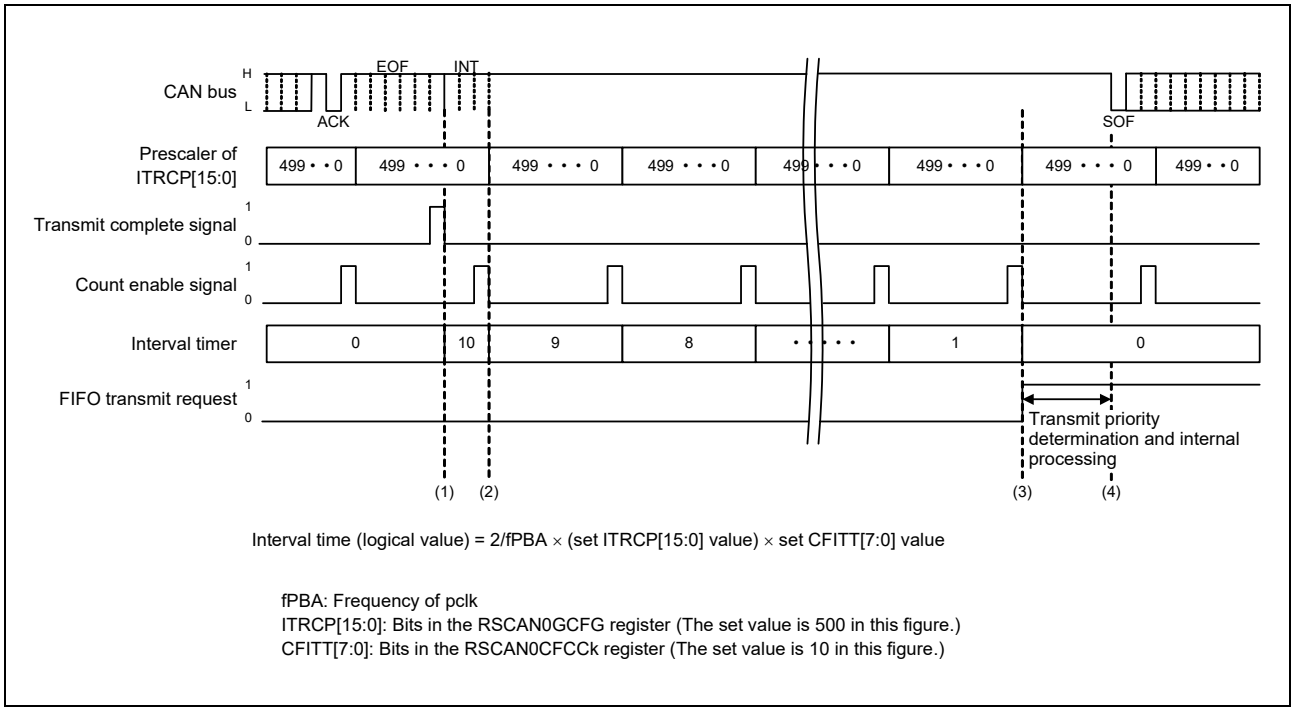


Figure 14.11 Interval Timer Timing Chart

(1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.

(2) The interval timer is decremented by the next count enable signal.

(3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.

(4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 504 cycles of the pclk may be generated.

14.4.4.4 Transmission Using Transmit Queues

Three to sixteen buffers are allocated to a transmit queue for each channel, and transmit buffer ((16 × m) + 15) is used as an access window of a corresponding channel.

All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue.

Setting the TXQE bit in the RSCAN0TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCAN0TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

14.4.4.5 Transmit History Function

Information about transmitted messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

A message transmit source buffer type can be selected by the THLDTE bit in the RSCAN0THLCCm register. The THLEN bit in the RSCAN0CFIDk register (k = 0 to 11) determines whether transmit history data is stored for each message. If enabled, the following information about a transmitted message will be stored in the transmit history buffer after transmission completes successfully.

Storage of the transmission history data after the successful completion of transmission may take up to 144 cycles of pclk.

- Buffer type
  001_B: Transmit buffer
  010_B: Transmit/receive FIFO buffer
  100_B: Transmit queue
- Buffer number
  Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer.
  This number depends on buffer types. See **Table 14.91**.
- Label data
  Label information of the transmit message

Table 14.91 Transmit History Data Buffer Numbers

Buffer type / Buffer No.	001_B	010_B	100_B
0000_B	Transmit buffer 16 × m + 0	Buffer numbers of the transmit buffer linked to the transmit/ receive FIFO buffer by the CFTML[3:0] bits in the RSCAN0CFCCk register (k = 0 to 11)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001_B	Transmit buffer 16 × m + 1		
0010_B	Transmit buffer 16 × m + 2		
0011_B	Transmit buffer 16 × m + 3		
0100_B	Transmit buffer 16 × m + 4		
0101_B	Transmit buffer 16 × m + 5		
0110_B	Transmit buffer 16 × m + 6		
0111_B	Transmit buffer 16 × m + 7		
1000_B	Transmit buffer 16 × m + 8		
1001_B	Transmit buffer 16 × m + 9		
1010_B	Transmit buffer 16 × m + 10		
1011_B	Transmit buffer 16 × m + 11		
1100_B	Transmit buffer 16 × m + 12		
1101_B	Transmit buffer 16 × m + 13		
1110_B	Transmit buffer 16 × m + 14		
1111_B	Transmit buffer 16 × m + 15		

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.

Transmit history data can be read from the RSCAN0THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

14.4.5 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.

When a transmit/receive FIFO buffer of a channel being used for transmission for which the CFM[1:0] bits in the RSCAN0CFCCk register are set to 10_B (gateway mode) is selected by the RSCAN0GAFLP1j register, messages that passed through the filter processing of the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCAN0CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the oldest message in it is not being transmitted and will not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

14.4.6 Test Function

The test function is classified into communication tests and global tests.

Communication tests: Performed for each channel.

- Standard test mode
- Listen-only mode
- Self-test mode 0 (external loopback mode)
- Self-test mode 1 (internal loopback mode)

Global tests: Performed for the entire module

- RAM test (read/write test)
- Inter-channel communication test

14.4.6.1 Standard Test Mode

Standard test mode allows CRC test.

14.4.6.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.

Listen-only mode is available for detecting the communication speed.

Do not make a transmit request from any buffer or queue in listen-only mode.

Figure 14.12 shows the connection when listen-only mode is selected.

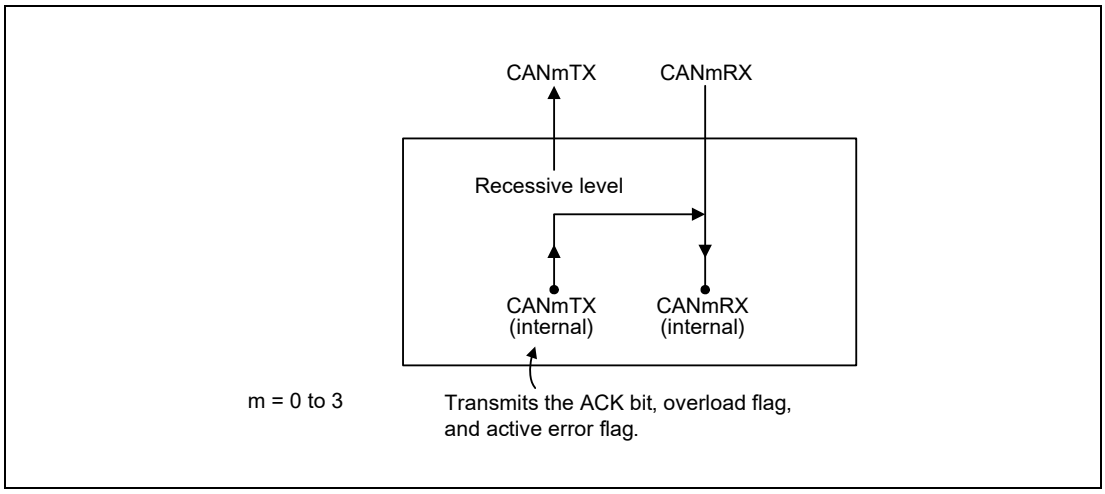


Figure 14.12 Connection when Listen-Only Mode is Selected

14.4.6.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCAN0GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).

If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

(1) Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

Figure 14.13 shows the connection when self-test mode 0 is selected.

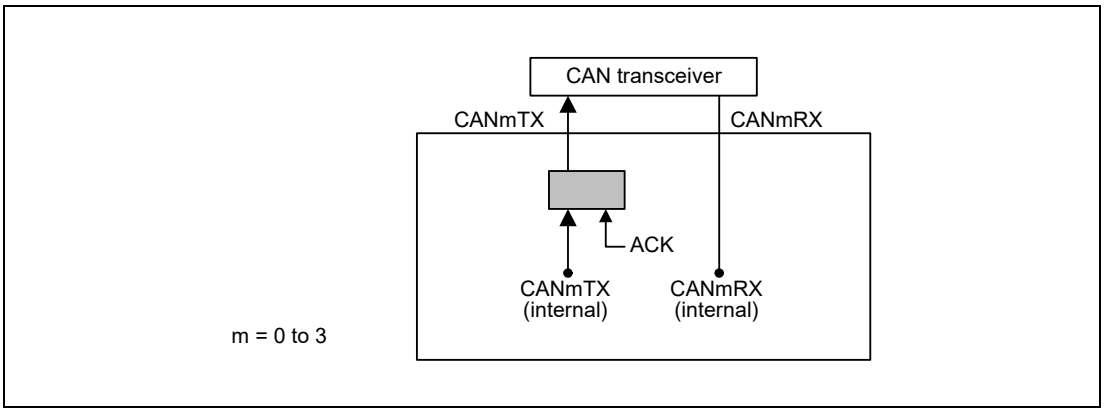


Figure 14.13 Connection when Self-Test Mode 0 is Selected

(2) Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal CANmTX pin (m = 0 to 3) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.

Figure 14.14 shows the connection when self-test mode 1 is selected.

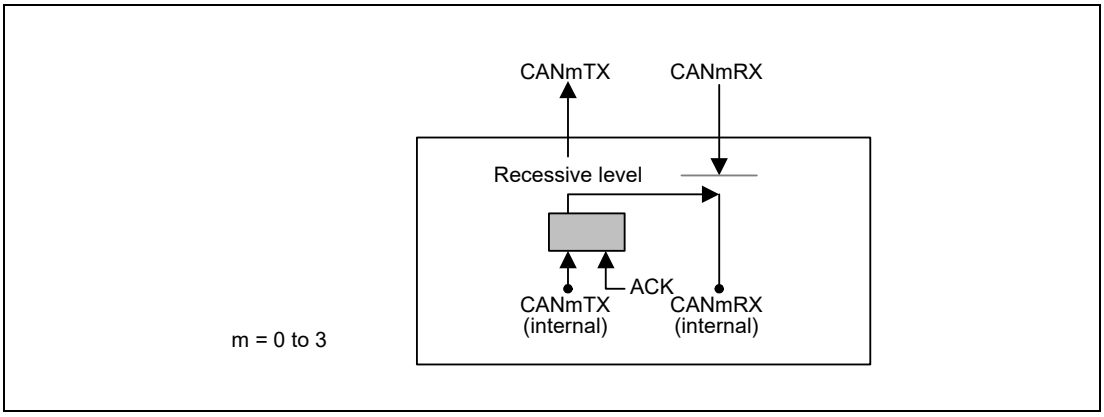


Figure 14.14 Connection when Self-Test Mode 1 is Selected

14.4.6.4 RAM Test

The RAM test function allows accesses to all CAN RAM addresses.

When the RAM test function is used, the RAM is divided into pages of 256 bytes each. RAM test page is set by the RTMPS[6:0] bits in the RSCAN0GTSTCFG register. Data in the set page can be read from and written to the RSCAN0RPGACCr register (r = 0 to 63). The available total RAM size is 12160 bytes ($2F80_H$).

14.4.6.5 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/ reception settings for each channel.

Figure 14.15 shows the connection for inter-channel communication test.

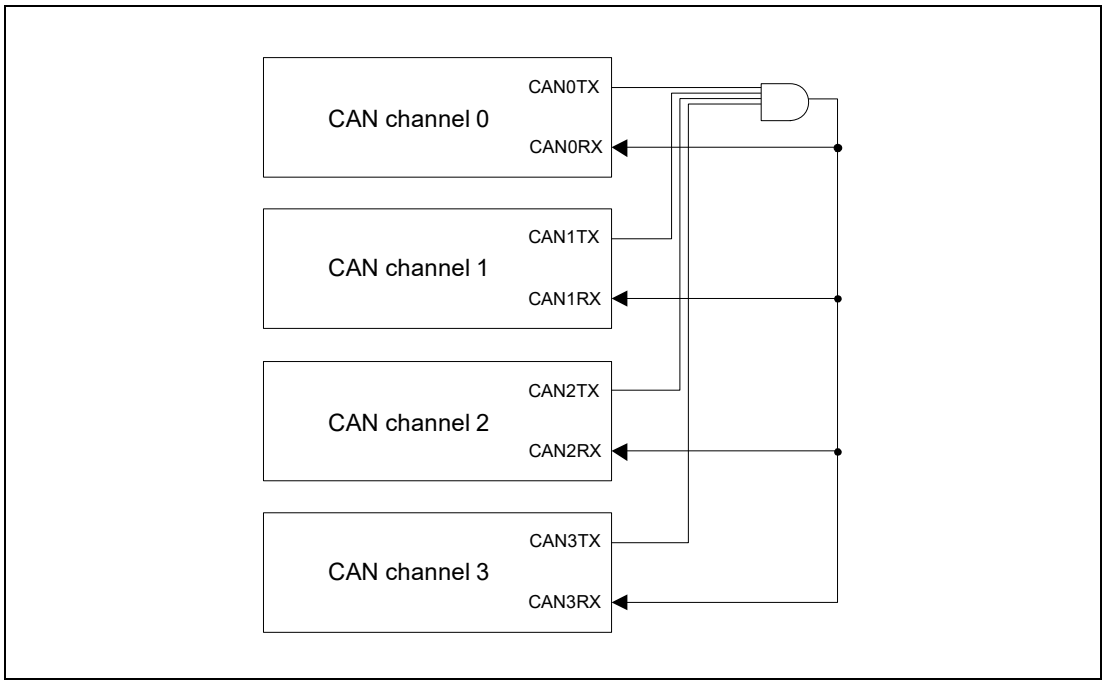
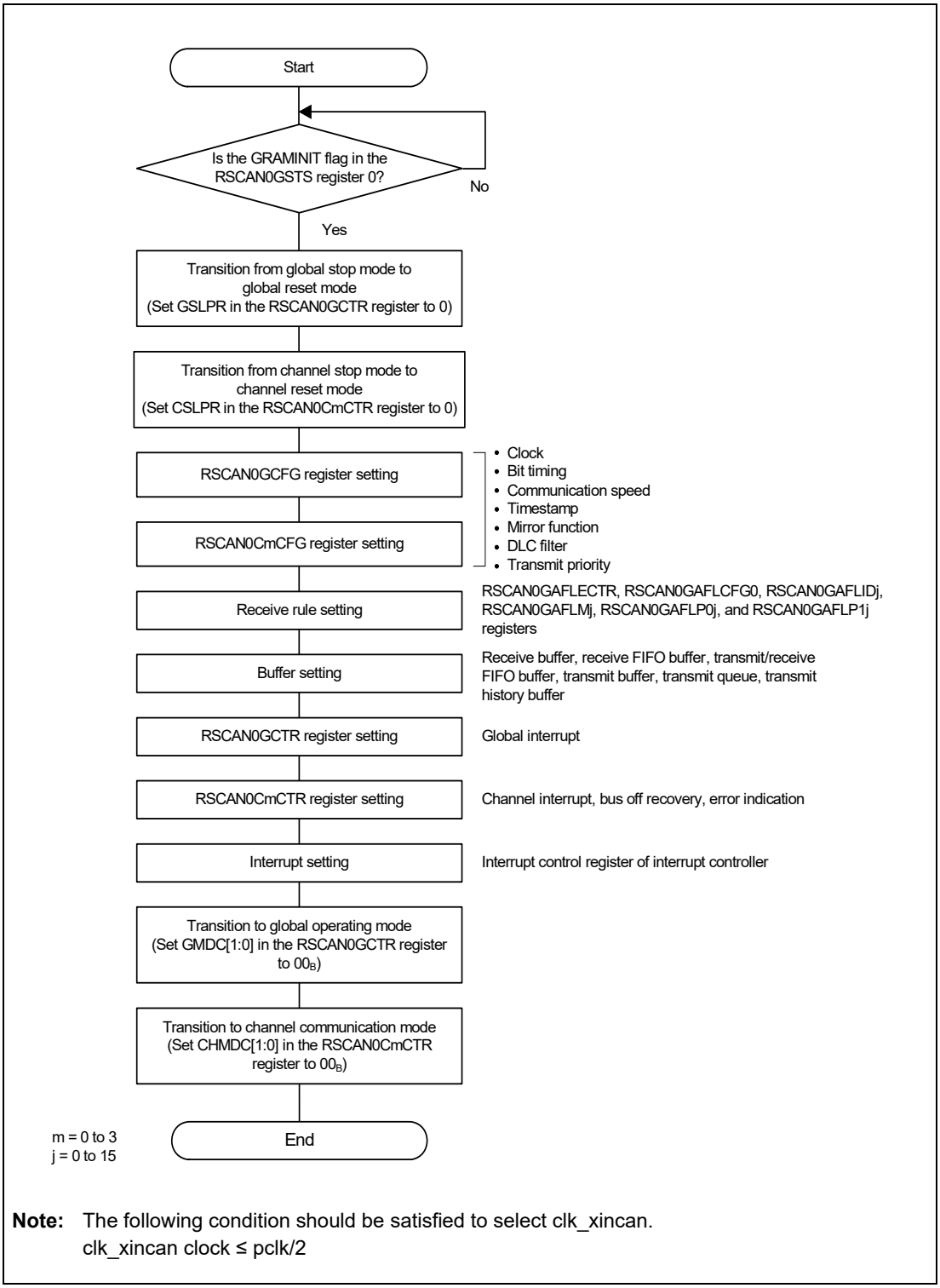


Figure 14.15 Connection for Inter-Channel Communication Test

14.5 RS-CAN Setting Procedure

14.5.1 Initial Settings

The RS-CAN module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 6082 cycles of the pclk. The GRAMINIT flag in the RSCAN0GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. **Figure 14.16** shows the CAN setting procedure after the MCU is reset.



Note: The following condition should be satisfied to select clk_xincan.
clk_xincan clock ≤ pclk/2

Figure 14.16 CAN Setting Procedure after the MCU is Reset

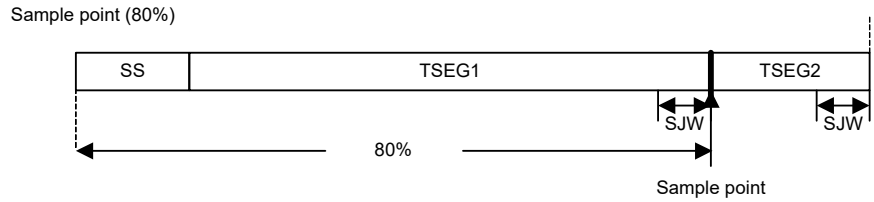
14.5.1.1 Clock Setting

Set the CAN clock (fCAN) as a clock source of the RS-CAN module. Select the clk_xincan or clkc using the DCS bit in the RSCAN0GCFG register.

14.5.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2, of which two segments TSEG1 and TSEG2 can be set by the RSCAN0CmCFG register (m = 0 to 3) for each channel. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (referred to as Tq hereinafter). 1 Tq is equal to one CANmTq clock cycle. The CANmTq clock is obtained by selecting the clock source with the DCS bit in the RSCAN0GCFG register and selecting the clock division ratio with the BRP[9:0] bits in the RSCAN0CmCFG register.

Figure 14.17 shows the bit timing chart. **Table 14.92** shows an example of bit timing setting.



SS = 1 Tq fixed
Set TSEG1 to a range of 4 Tq to 16 Tq
Set TSEG2 to a range of 2 Tq to 8 Tq
Set SJW to a range of 1 Tq to 4 Tq
Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
TSEG1 > TSEG2 ≥ SJW

- SS (synchronization segment):
  The SS is a segment that performs synchronization by monitoring the edge from recessive to dominant bits in the Interframe Space.
  Interframe Space consists of Intermission, Suspend Transmission, and Bus Idle. All nodes can start transmission during Bus Idle.
- TSEG1 (Time segment 1):
  TSEG1 is a segment that absorbs physical delay on the CAN network. The physical delay on the CAN network is twice the total of the delay on the CAN bus, the delay in the input comparator, and the delay in the output driver.
- TSEG2 (Time segment 2):
  TSEG2 is a segment that compensates phase error due to an error in frequency.
- SJW (Resynchronization jump width)
  The SJW is a length to extend or reduce the time segment to compensate for an error in phase due to phase error.

Figure 14.17 Bit Timing Chart

Table 14.92 Example of Bit Timing Setting

1 Bit	Set Value (Tq)				Sample Point (%) Note: See Figure 14.17.
	SS	TSEG1	TSEG2	SJW	
20 Tq	1	12	7	1	65.00
	1	13	6	1	70.00

14.5.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value (BRP[9:0] bits in the RSCAN0CmCFG register), and Tq count per bit time.

Figure 14.18 shows the CAN clock control block diagram, and **Table 14.93** shows an example of the communication speed setting.

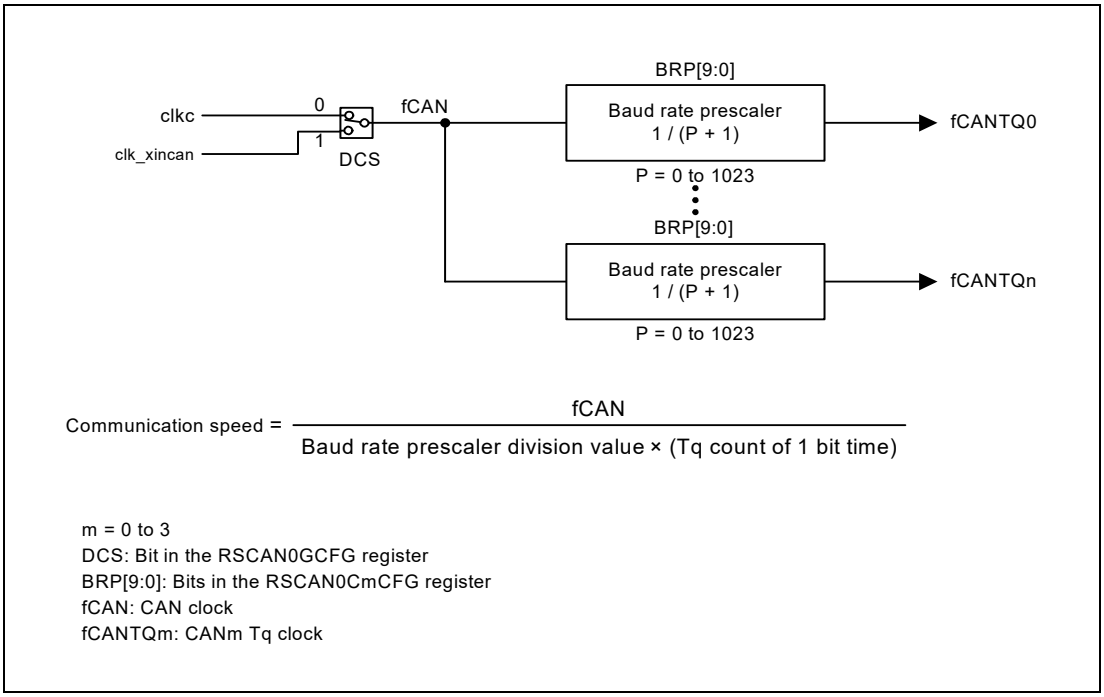


Figure 14.18 CAN Clock Control Block Diagram

Table 14.93 Example of Communication Speed Setting

fCAN / Communication speed	40 MHz
1 Mbps	8Tq (5) 20Tq (2)
500 Kbps	8Tq (10) 20Tq (4)
250 Kbps	8Tq (20) 20Tq (8)

Note: Values in ( ) are baud rate prescaler division values.

14.5.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.

Up to 16 receive rules can be registered per page. Specify pages 0 to 15 by the AFLPN[4:0] bits in the RSCAN0GAFLECTR register.

Set receive rule table write enable/disable using the AFLDAE bit.

Figure 14.19 shows the receive rule setting procedure.

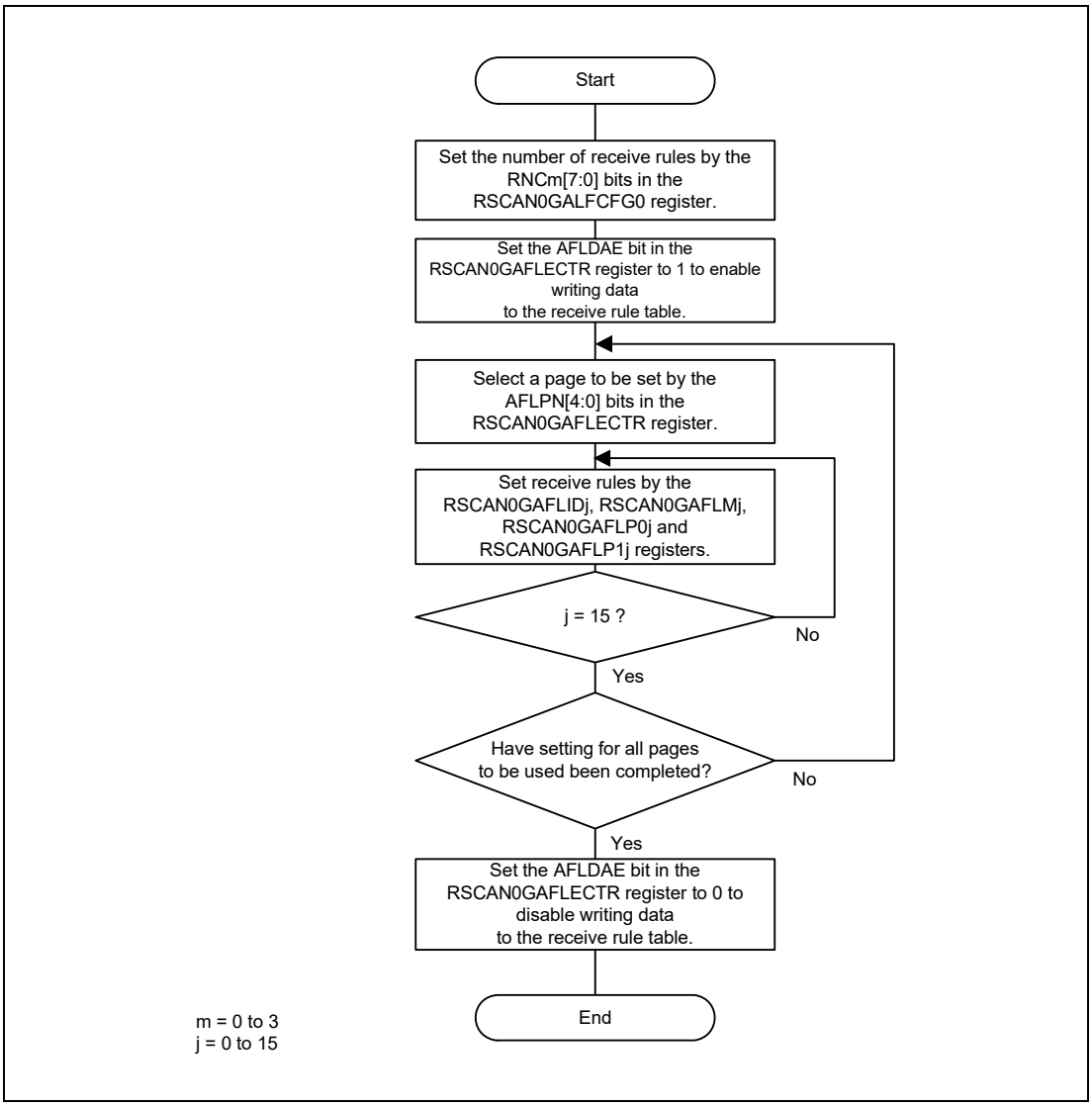
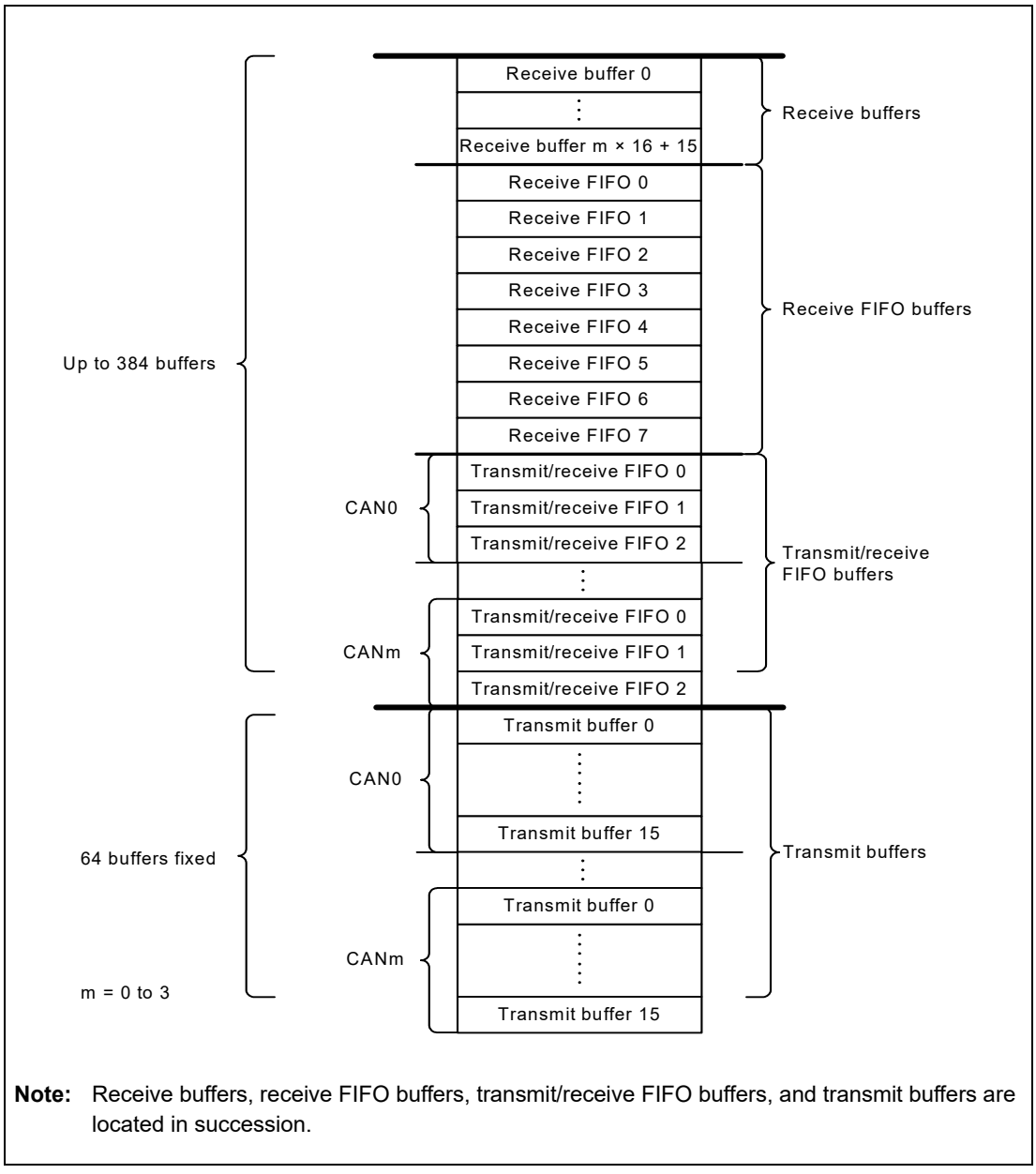


Figure 14.19 Receive Rule Setting Procedure

14.5.1.5 Buffer Setting

Set sizes and interrupt sources of buffers. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

Figure 14.20 shows the buffer configuration. **Figure 14.21** shows the buffer setting procedure.



Note: Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.

Figure 14.20 Buffer Configuration

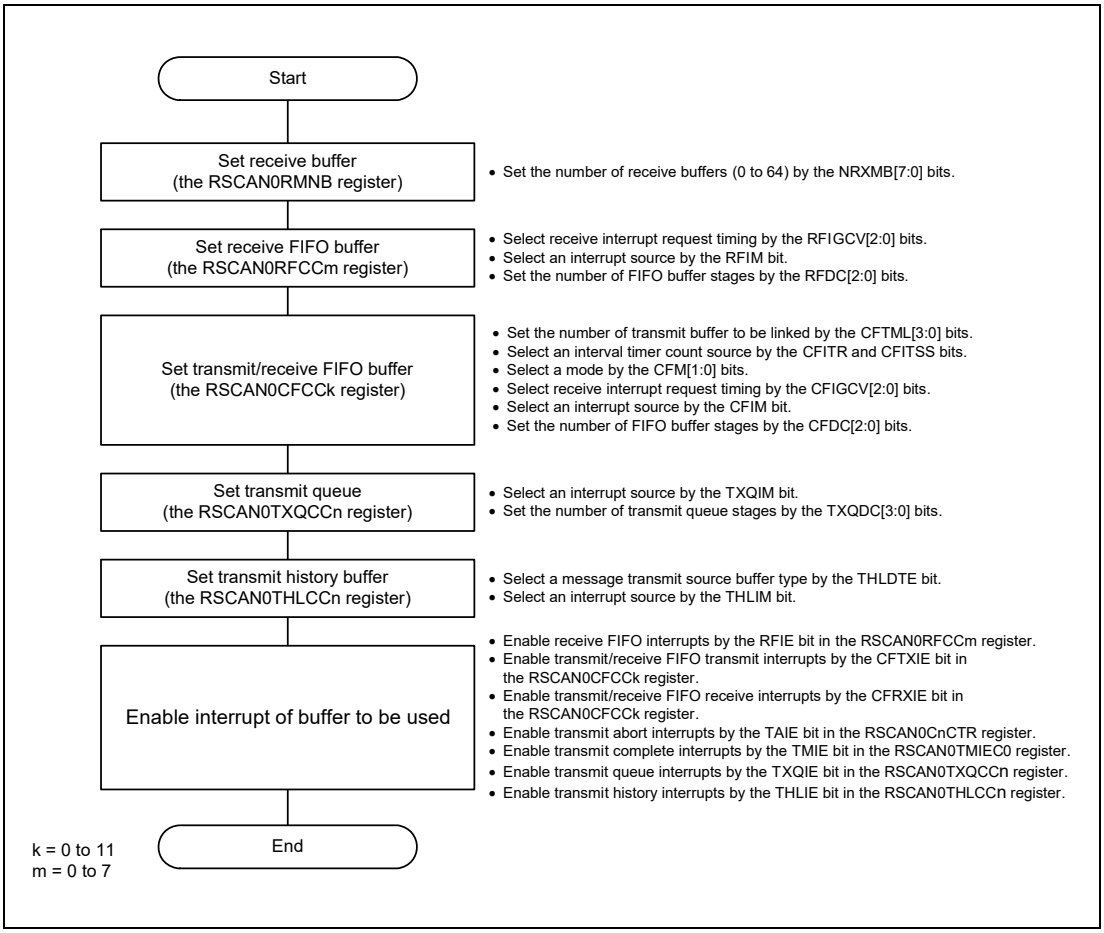


Figure 14.21 Buffer Setting Procedure

14.5.2 Reception Procedure

14.5.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCAN0RMNDy register (y = 0, 1, q = 0 to 63) is set to 1 (receive buffer q contains a new message). Messages can be read from the RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers. If the next message has been received before the current message is read from the receive buffer, the message is overwritten. **Figure 14.22** shows the receive buffer reading procedure.

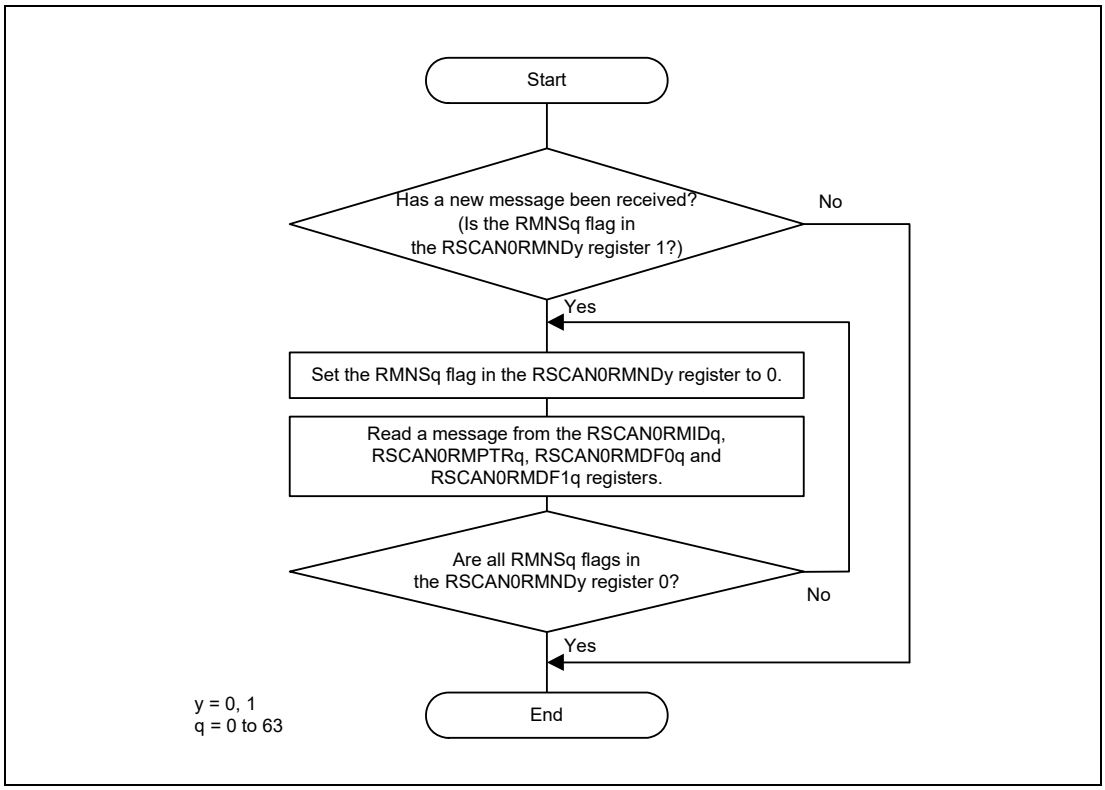


Figure 14.22 Receive Buffer Reading Procedure

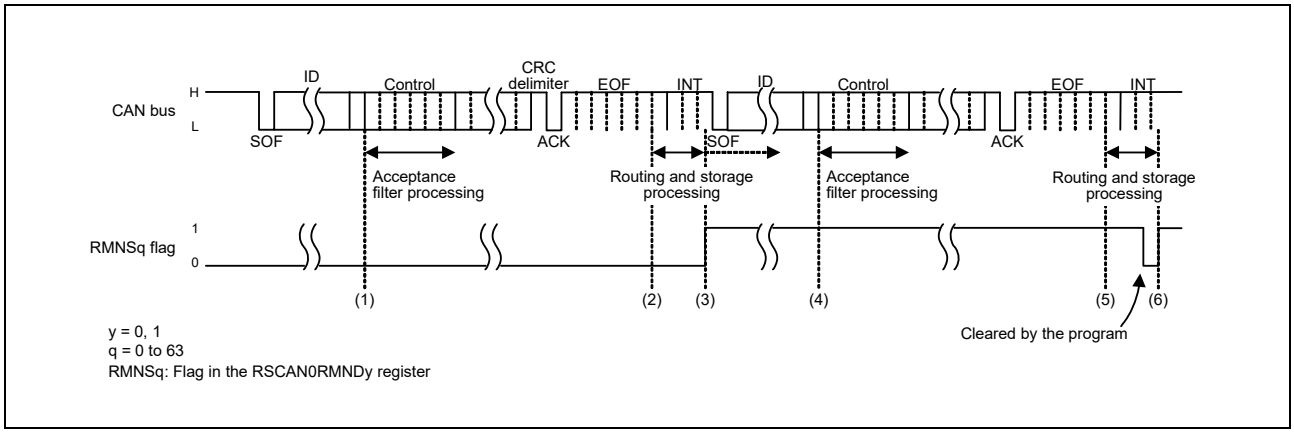


Figure 14.23 Receive Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.
When the message storage processing starts, the RMNSq flag in the corresponding RSCAN0RMNDy register is set to 1 (the receive buffer contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(6) When the corresponding RMNSq flag is cleared to 0 (receive buffer contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

14.5.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCAN0RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCAN0CFSTSk register (k = 0 to 11)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCAN0RFCCm register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCAN0CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCAN0RFIDm, RSCAN0RFPTRm, RSCAN0RFDF0m, and RSCAN0RFDF1m registers for receive FIFO buffers, or from the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCAN0RFCCm register or the CFDC[2:0] bits in the RSCAN0CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCAN0RFSTSm register or the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCAN0RFSTSm register or CFRXIF flag in the RSCAN0CFSTSk register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.

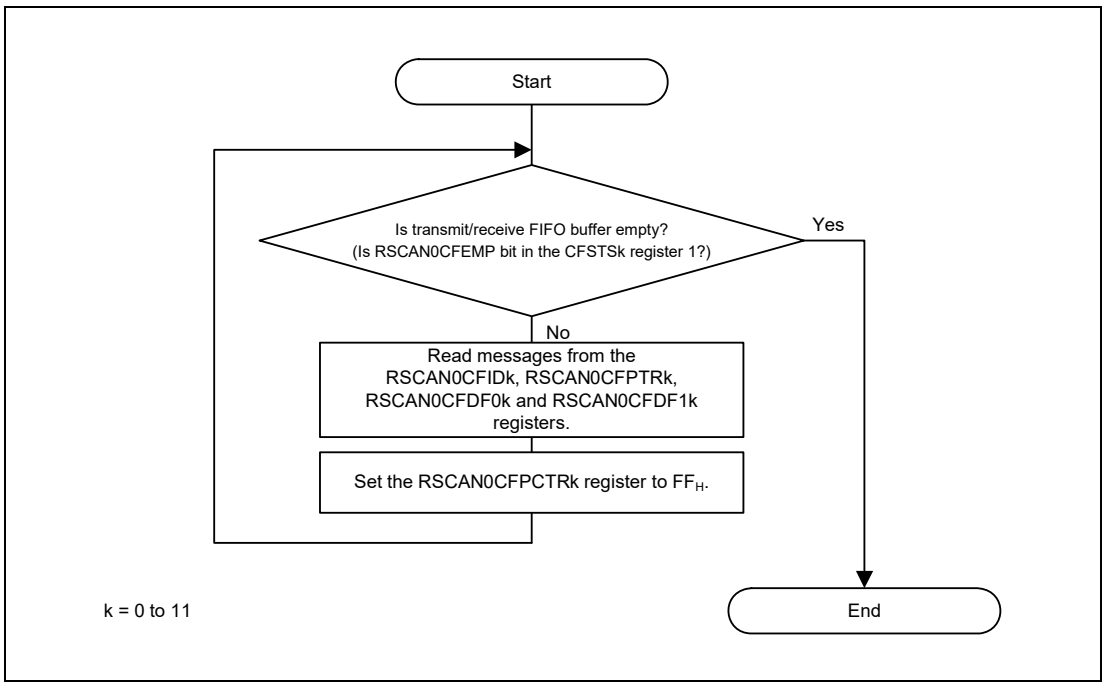


Figure 14.24 Transmit/Receive FIFO Buffer Reading Procedure

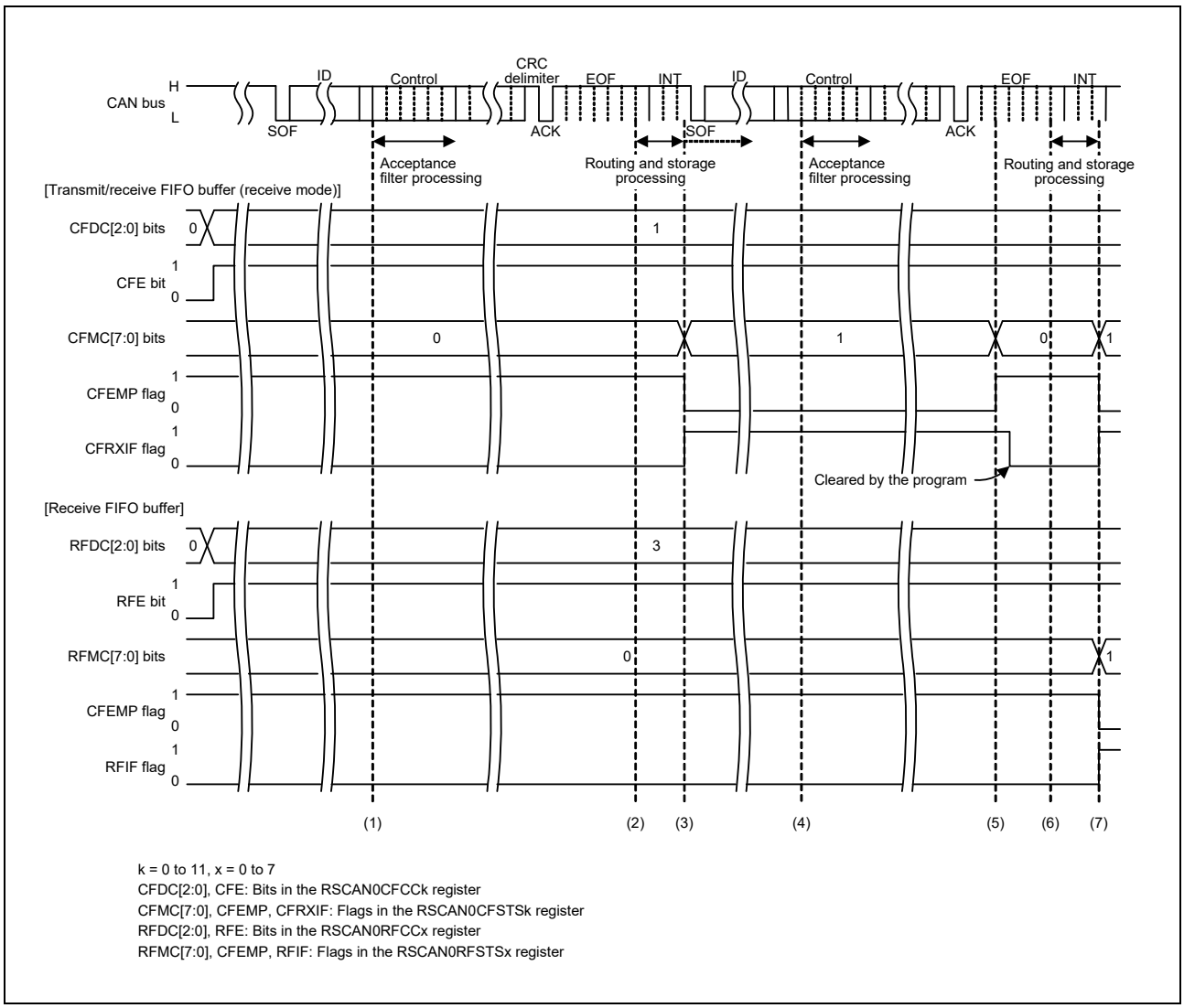


Figure 14.25 FIFO Buffer Reception Timing Chart

(1) When the ID field in a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing and the CFE value in the RSCAN0CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCk register is 001_B or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCAN0CFSTSk register is incremented and becomes 01_H. When the CFIM bit in the RSCAN0CFCCk register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) Read received messages from the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers and write FF_H to the RSCAN0CFPCTRk register. This causes the

CFMC[7:0] bits in the RSCAN0CFSTSk register to be decremented. When CFMC[7:0] becomes 00_H, the CFEMP flag in the RSCAN0CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), the RFE bit in the RSCAN0RFCCx register is set to 1, and the CFDC[2:0] bits are set to 001_B or more. The CFMC[7:0] bit value is incremented by 1 to be 01_H. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present).
The message is stored in the receive FIFO buffer if the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCAN0RFCCx register are set to 001_B or more. The RFMC[7:0] bits in the RSCAN0RFSTSx register are set to 01_H by being incremented by 1. When the RFIM bit in the RSCAN0RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

14.5.3 Transmission Procedure

14.5.3.1 Procedure for Transmission from Transmit Buffers

Figure 14.26 shows the procedure for transmission from transmit buffers.

Figure 14.27 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. **Figure 14.28** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission abort has been completed.

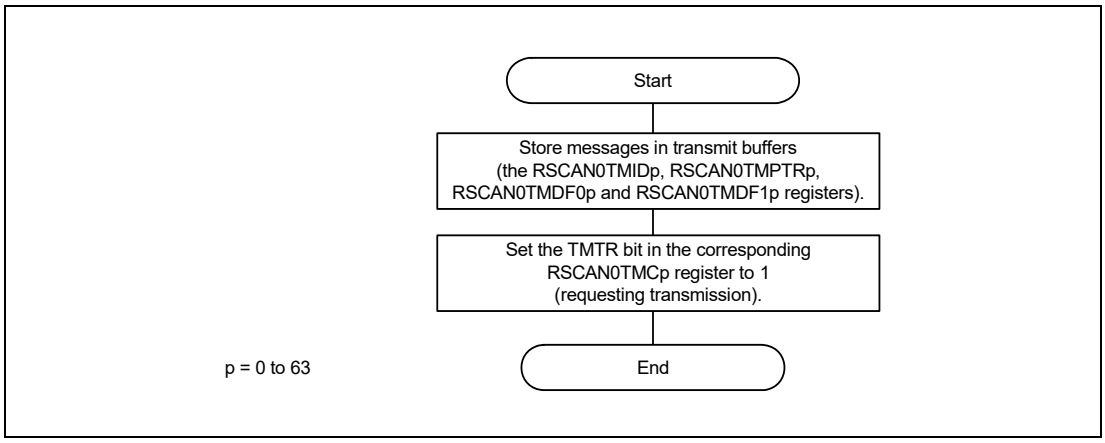


Figure 14.26 Procedure for Transmission from Transmit Buffers

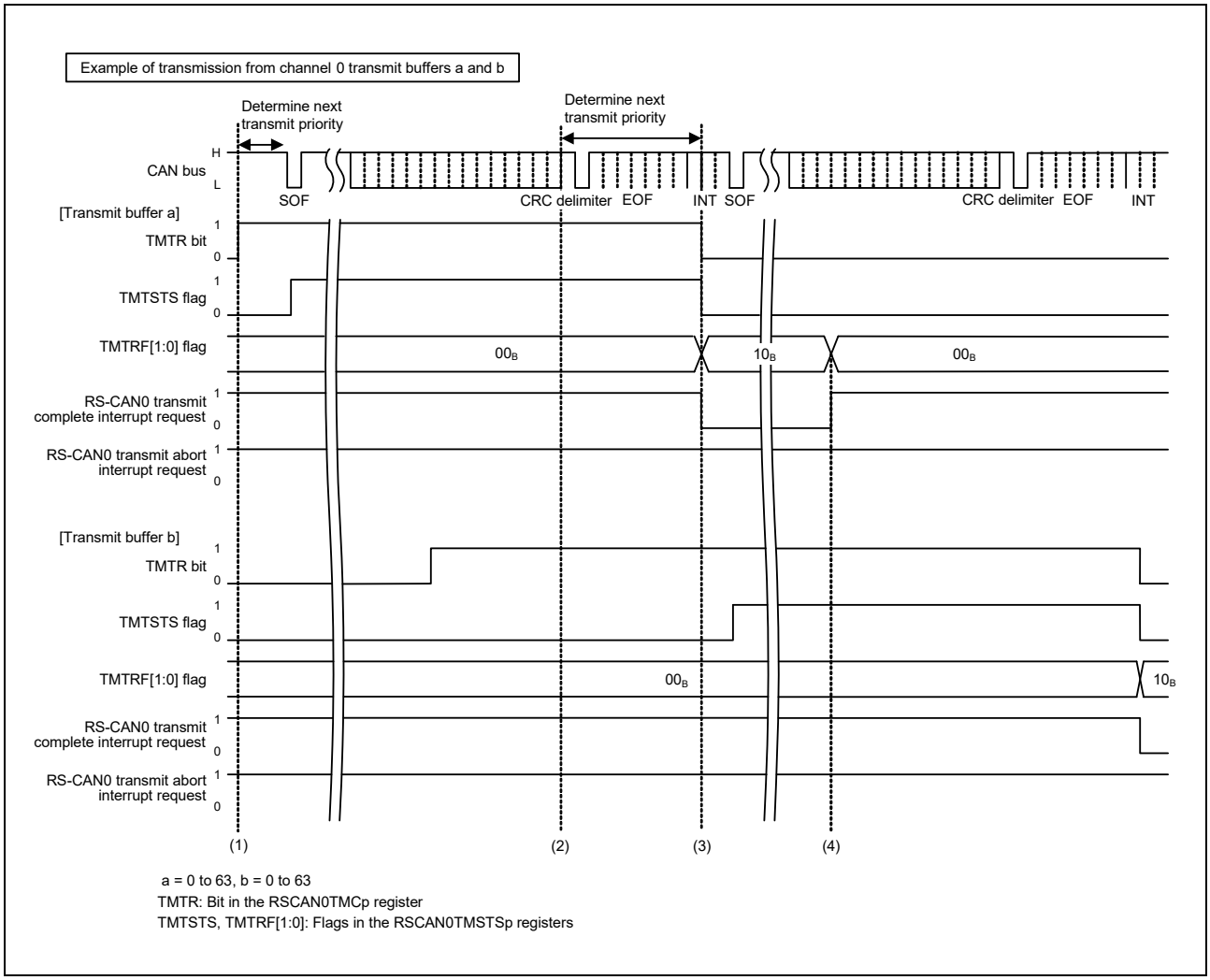


Figure 14.27 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding the RSCAN0TMCa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission.

(3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMCa register is set to 10_B (transmission has been completed (without transmission abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa value in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmission complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00_B. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00_B.

If an arbitration-lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

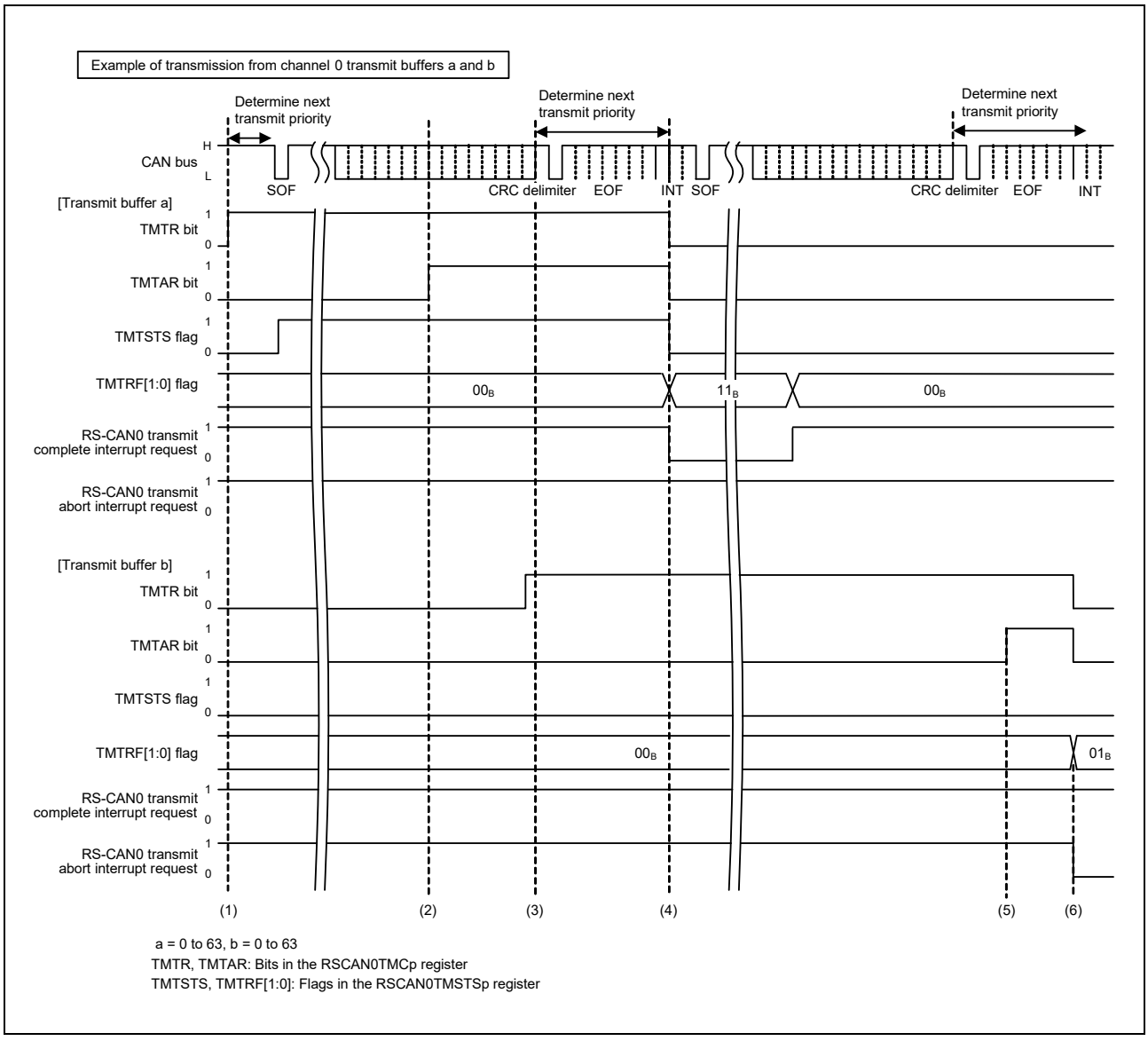


Figure 14.28 Transmit Buffer Transmission Timing Chart (Transmission Abort Completed)

(1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCAN0TMCa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmission abort is requested).

(3) The priority determination starts with the CRC delimiter for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. Some delay may occur in the

determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMCa register is set to 11_B (transmission has been completed (with transmission abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa value in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmission complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.

(6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01_B. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01_B. At this time, the TMTR and TMTAR bits are cleared to 0. When transmission abort is completed with the TAIE bit in the RSCAN0CmCTR register set to 1 (transmission abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B.

If an arbitration loss has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

14.5.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 14.29 shows the procedure for transmission from transmit/receive FIFO buffers.

Figure 14.30 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. **Figure 14.31** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission abort has been completed.

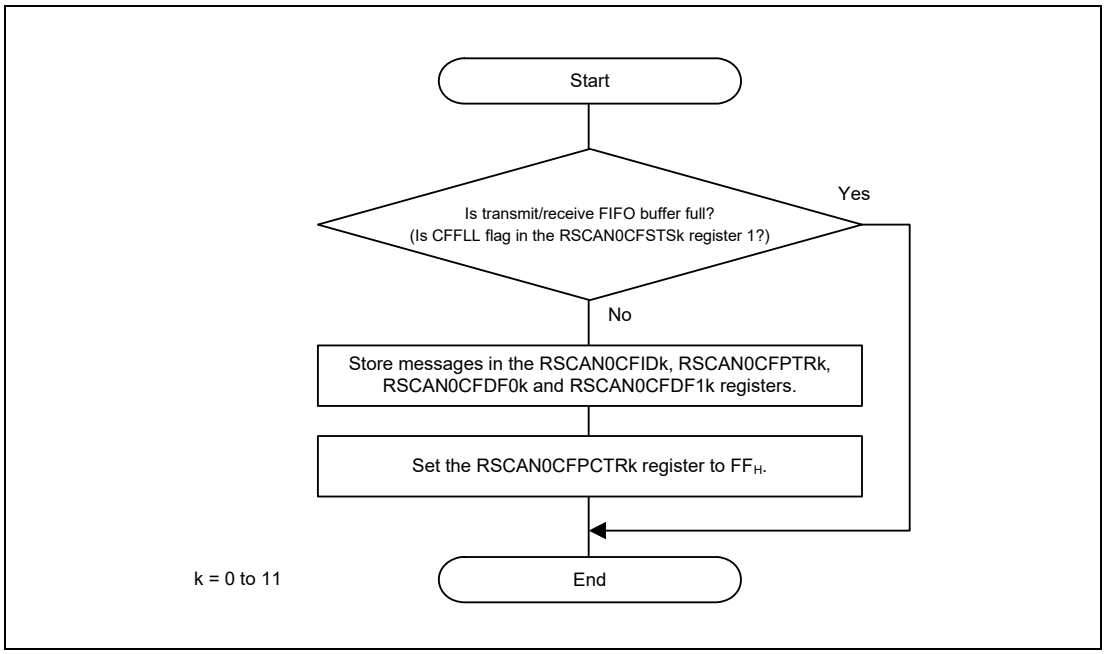


Figure 14.29 Procedure for Transmission from Transmit/Receive FIFO Buffers

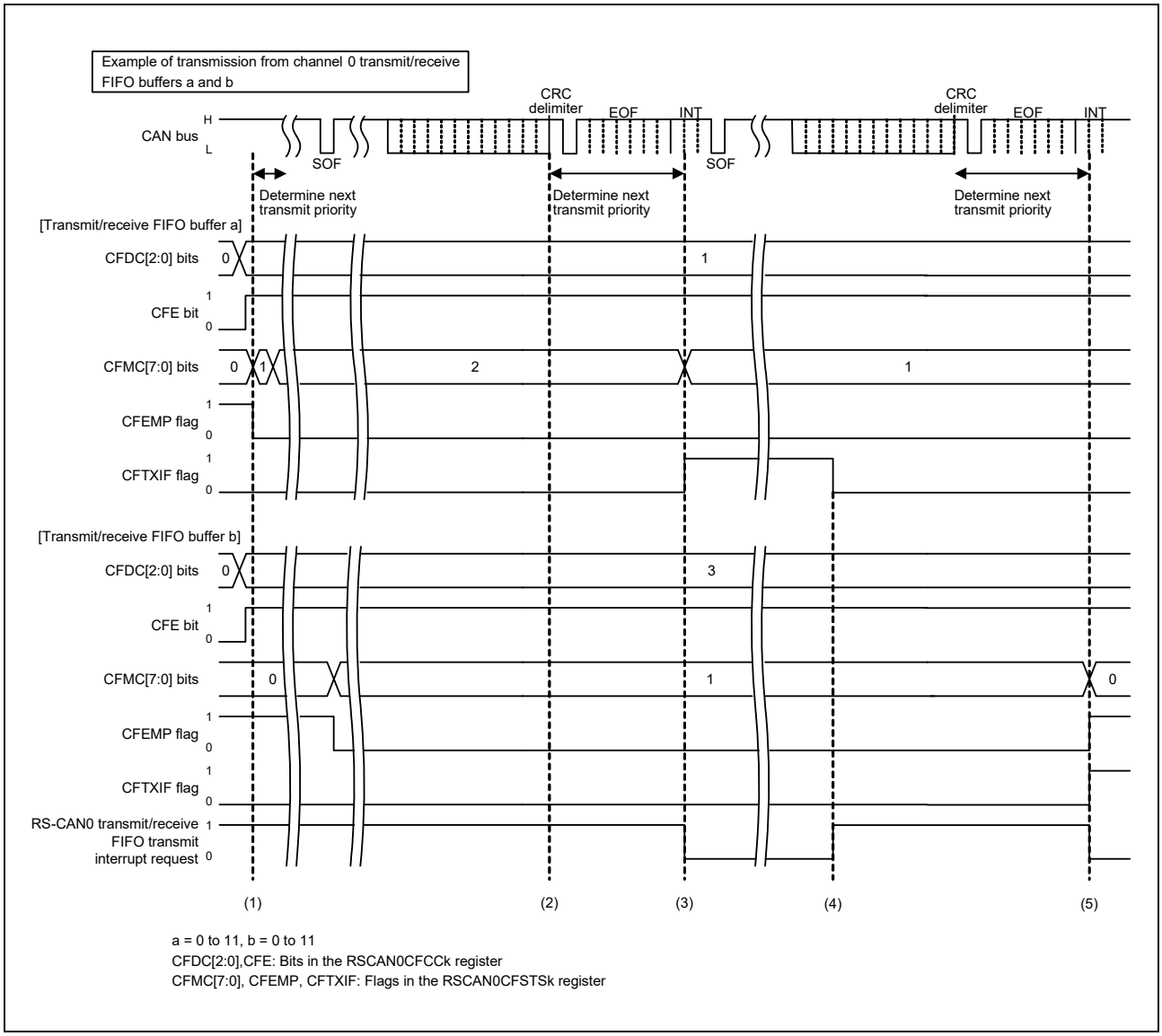


Figure 14.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, when the CFE value in the RSCA0CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(3) When transmission completes successfully, the CFMC[7:0] value in the RSCAN0CFSTSa register is decremented. Setting the CFIM bit in the RSCAN0CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSk register to 1 (a transmit/receive FIFO transmit interrupt request is present).

(4) The program can clear the CFTXIF flag.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCAN0CFSTSb register is decremented. The CFMC[7:0] bits are cleared to 00_H and therefore the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCAN0CFSTSa and RSCAN0CFSTSb register is set to 1 (the transmit/receive FIFO buffer is full).

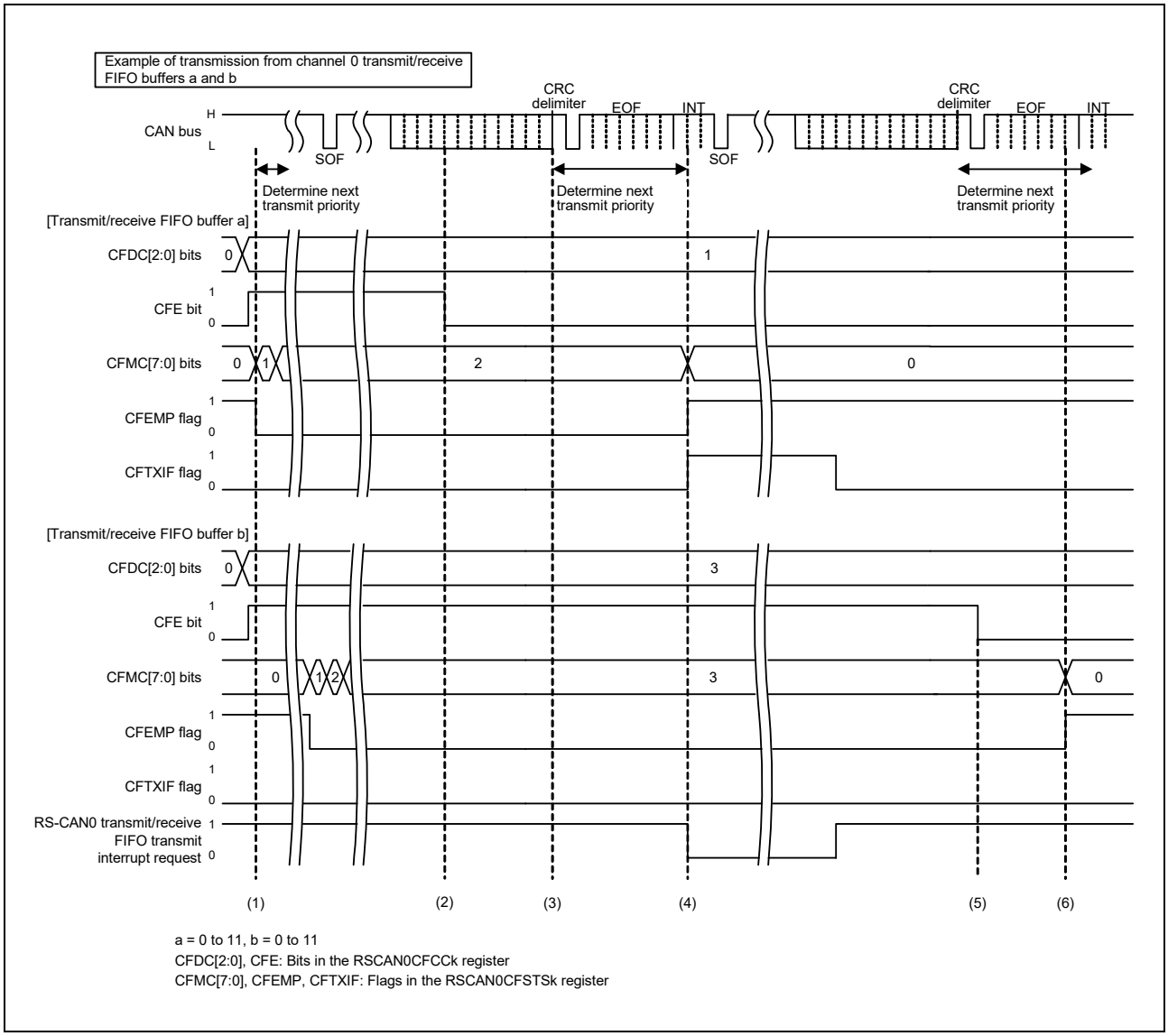


Figure 14.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Abort Completed)

(1) While the CAN bus is idle, when the CFE value in the RSCAN0CFCCa register (a = 0 to 11) is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCa register is 001_B (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01_H or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the

message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

(3) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(4) When transmission completes successfully, the CFMC[7:0] value is cleared to 00_H. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.

(5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCAN0CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCAN0CFSTSb register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCAN0CFSTSb register are cleared to 00_H and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to 00_H and the CFEMP flag is set to 1.)

14.5.3.3 Procedure for Transmission from the Transmit Queue

Figure 14.32 shows the procedure for transmission from the transmit queue.

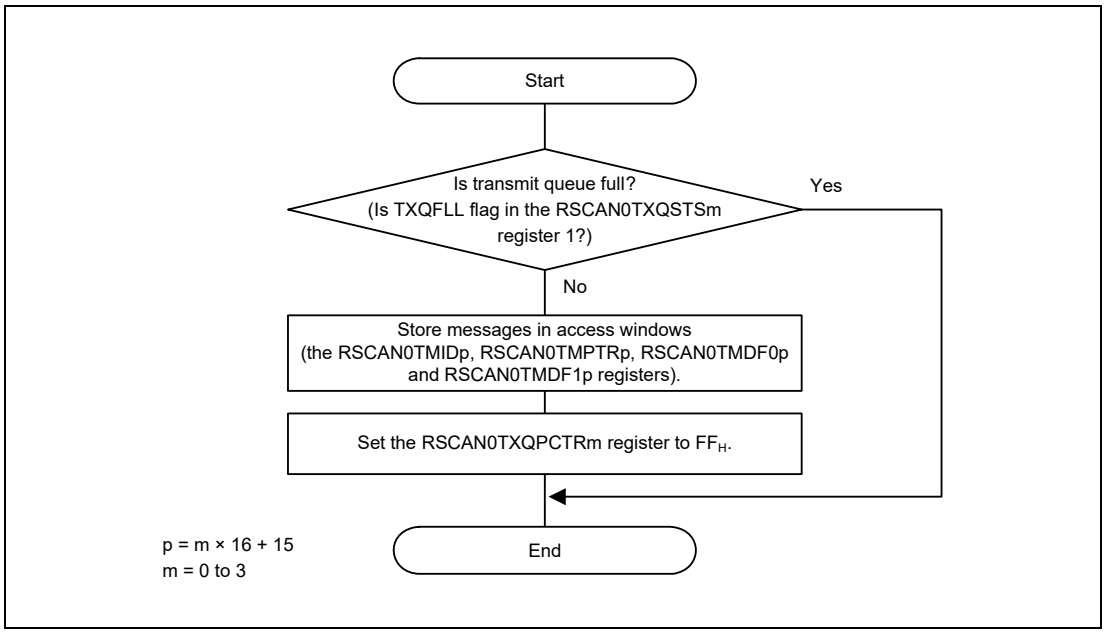


Figure 14.32 Procedure for Transmission from the Transmit Queue

14.5.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCAN0THLACCm register. The next data can be accessed by writing FF_H to the corresponding RSCAN0THLPCTRm register (m = 0 to 3) after reading a set of data. **Figure 14.33** shows the transmit history buffer reading procedure.

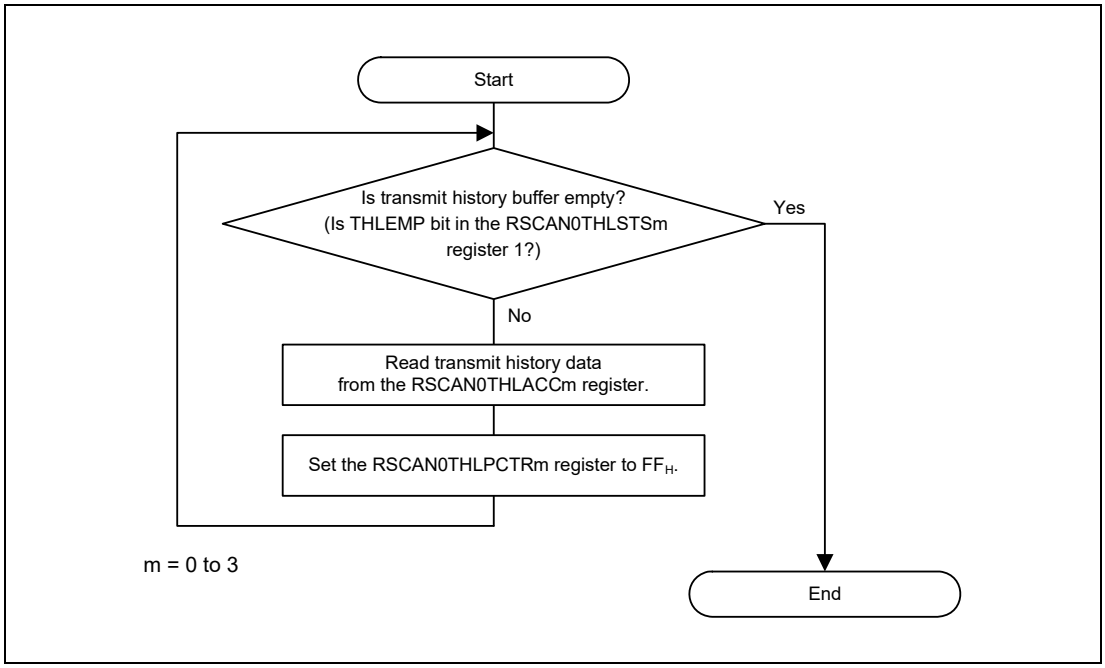


Figure 14.33 Transmit History Buffer Reading Procedure

14.5.4 Test Settings

14.5.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 14.34 shows the self-test mode setting procedure.

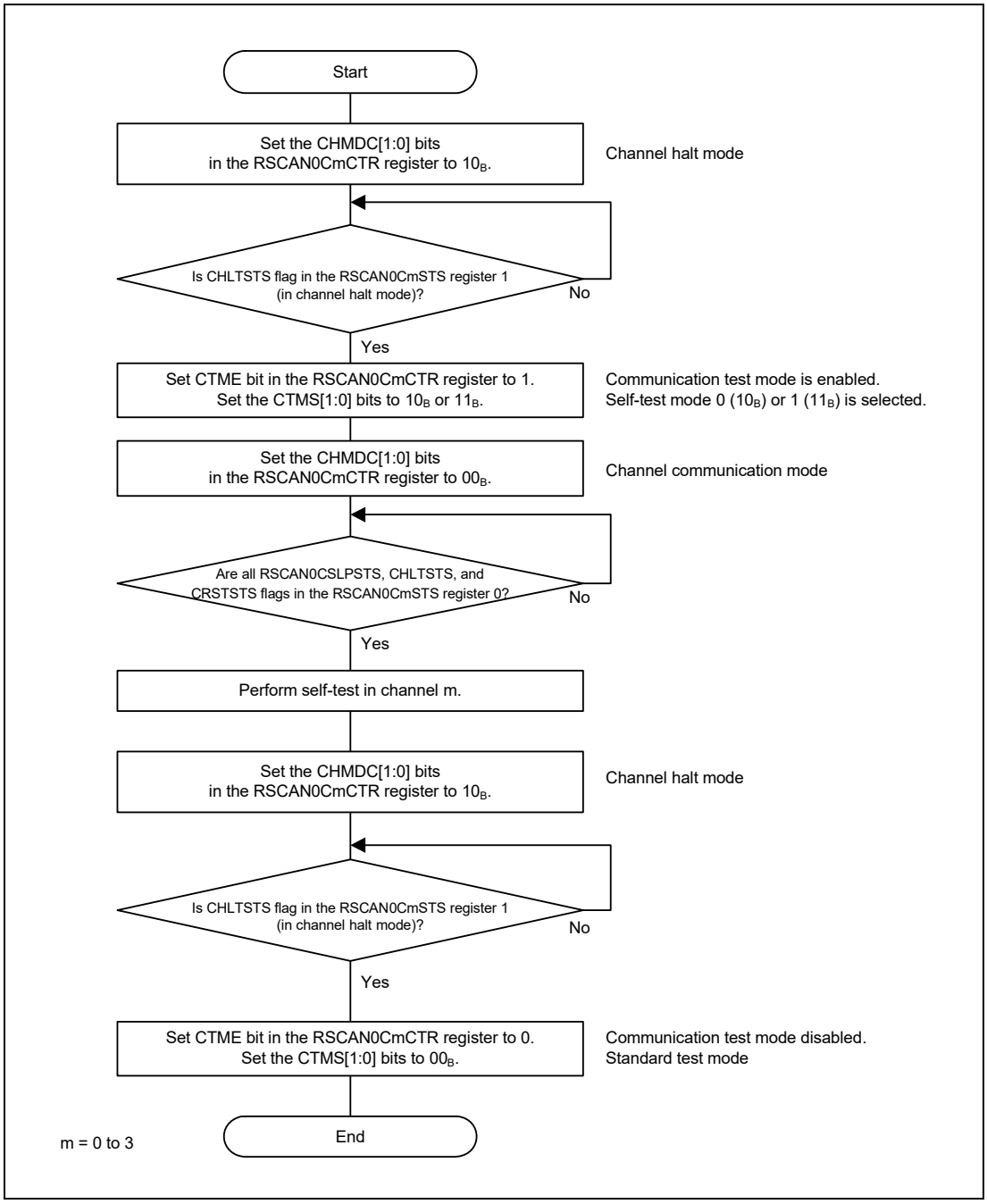


Figure 14.34 Self-Test Mode Setting Procedure

14.5.4.2 Procedure for Releasing the Protection

Since the global test function in **Table 14.94** is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCAN0GLOCKK register, then set the target test bit to 1.

Table 14.94 Protection Release Data for Test Function

Test Function	Protection Release Data 1	Protection Release Data 2	Target Bit
RAM test	7575_H	$8A8A_H$	RTME bit in the RSCAN0GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. **Figure 14.35** shows the procedure for releasing the protection.

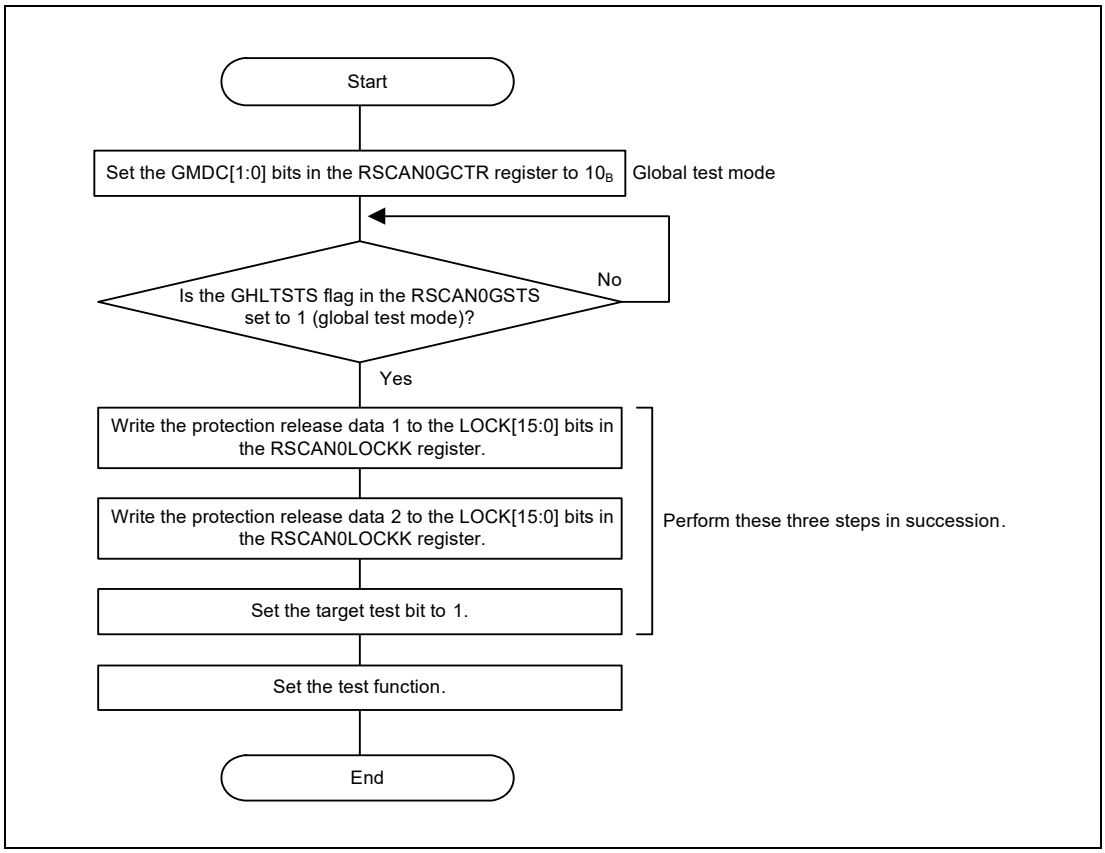


Figure 14.35 Protection Release Procedure

14.5.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write 0000 0000_H to all pages of the CAN RAM.

Figure 14.36 shows the RAM test setting procedure.

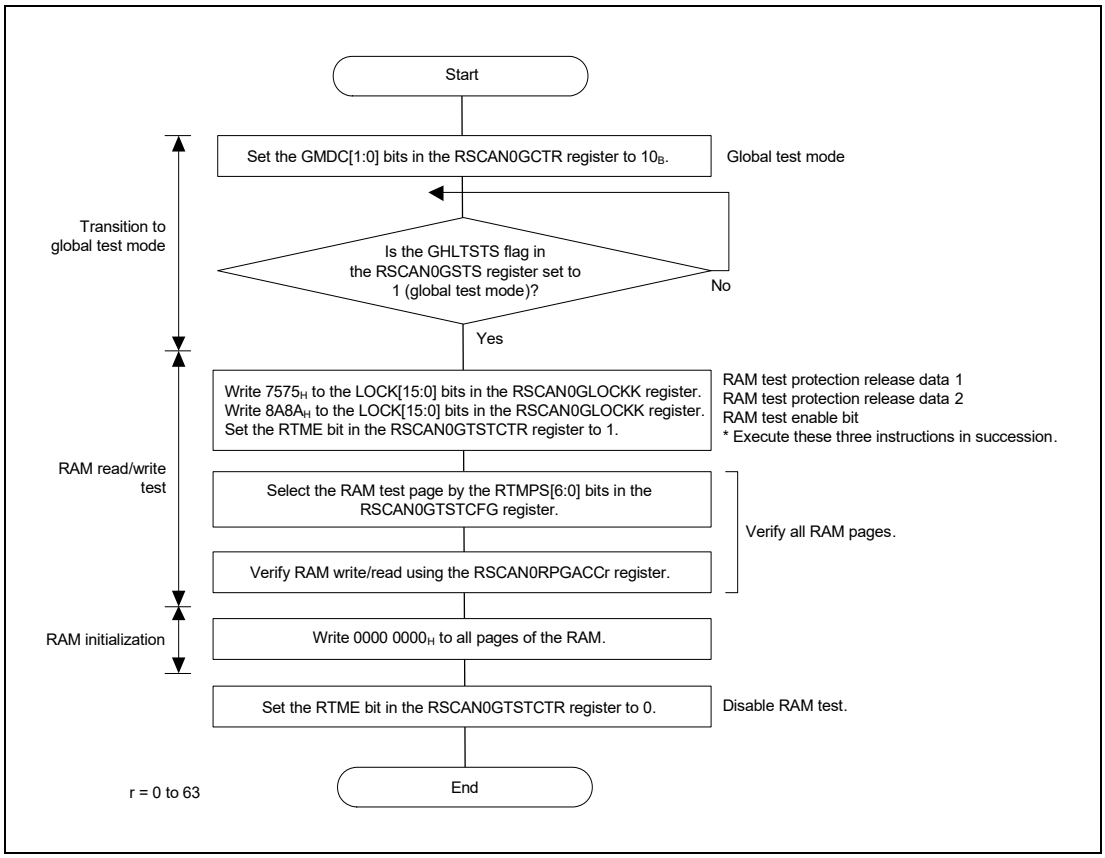


Figure 14.36 RAM Test Setting Procedure

14.5.4.4 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.

Figure 14.37 shows the inter-channel communication test setting procedure.

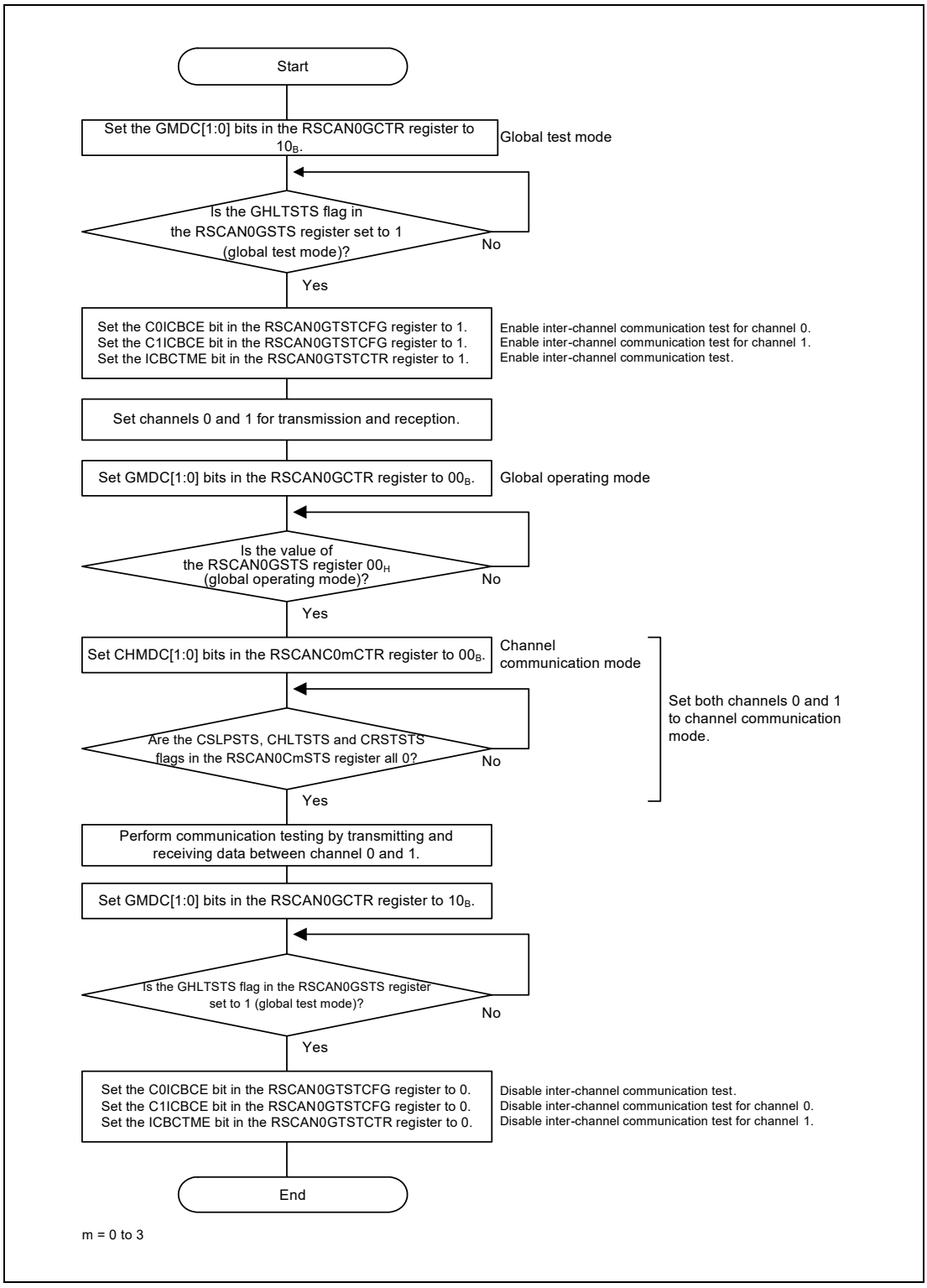


Figure 14.37 Inter-Channel Communication Test Setting Procedure (Example of Communication Test between Channel 0 and Channel 1)

14.6 Notes

- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCAN0GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCAN0CmSTS register (m = 0 to 3) for transitions.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCAN0TMCp) of the corresponding transmit buffer to 00_H. The status register (RSCAN0TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCAN0TMTRSTS0 to RSCAN0TMTRSTS2, RSCAN0TMTARSTS0 to RSCAN0TMTARSTS2, RSCAN0TMTCSTS0 to RSCAN0TMTCSTS2, and RSCAN0TMTASTS0 to RSCAN0TMTASTS2), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (the RSCAN0TMIEC0, RSCAN0TMIEC1 and RSCAN0TMIEC2) to 0 (transmit buffer interrupt is disabled).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not bet allocated to transmit queues.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.

Section 15 Window Watchdog Timer A (WDTA)

This section contains a generic description of the window watchdog timer (WDTA).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of WDTA.

15.1 Features of RH850/C1x WDTA

15.1.1 Number of Units

This microcontroller has the following number of WDTA units.

Table 15.1 Number of Units

Product	RH850/C1H	RH850/C1M
Channels	2	1
Name	WDTAn (n = 0 to 1)	WDTA0

Table 15.2 Index

Index	Meaning
n	Throughout this section, the individual WDTA units are identified by the index “n” (n = 0, 1) (n = 0,1 for C1H, and n = 0 for C1M); for example, WDTAnWDTE is the WDTAn enable register.

15.1.2 Register Base Address

WDTA base addresses are listed in the following table.

WDTA register addresses are given as offsets from the base addresses in general.

Table 15.3 Register Base Address

Base Address Name	Base Address
<WDTA0_base>	FFED 0000_H
<WDTA1_base>	FFED 1000_H

15.1.3 Clock Supply

The WDTA clock supply is shown in the following table.

Table 15.4 WDTAn Clock Supply

WDTAn	Clock Name for the Unit	Supply Clock Name
WDTA0	WDTATCKI	WDTCLKI
	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)
WDTA1	WDTATCKI	WDTCLKI
	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)

15.1.4 Interrupts

WDTA can generate the following interrupt requests.

Table 15.5 Interrupt Requests

WDTAn Signals	Function	Interrupt Number
WDTA0		
WDTA0TIT	75% interrupt	54
WDTA1		
WDTA1TIT	75% interrupt	55*1

Note 1. This is not supported by RH850/C1M.

15.1.5 Reset Sources

WDTA is initialized by reset sources listed in the following table.

Table 15.6 Reset Sources

Unit Name	Reset Source
WDTA0	All reset sources
WDTA1	All reset sources

15.1.6 WDTA Start-Up Options

The start-up options determine the start-up configuration of WDTA0 after reset release. The start-up options are described in **Table 15.7, WDTA0 Start-Up Options**. They are not supported in WDTA1.

Table 15.7 WDTA0 Start-Up Options

Start-Up Option	Function	Flash Option Assignment
OPWDOVF[2:0]	Setting for overflow interval time	OPBT0[27:25]
OPWDWS[1:0]	Setting for window open period	OPBT0[29:28]
OPWDINT	Setting for 75% interrupt request	OPBT0[30]
OPWDRUN	Setting for start mode	OPBT0[31]

15.2 Overview

15.2.1 Functional Overview

The WDTA has the following functions:

- Selection of the operating mode after release from reset using the option byte
  Starting/stopping of the counter after the WDTA is reset, enabling or disabling of 75% interrupt requests, the window-open period, and overflow time can be selected.
- WDTA trigger function
  Writing an activation code to the WDTA trigger register starts WDTA and restarts the counter.
- 75% interrupt request signals
  An interrupt request signal can be generated when the WDTA counter reaches 75% of the overflow interval time (this function can be enabled or disabled by the setting of WDTAnMD.WDTAnWIE).
- Window function
  The period during which writing to the WDTA trigger register is valid (window-open period) can be set. Writing to the WDTA trigger register at a time outside the window-open period causes an error.
- WDTA error detection
  When an error is detected, the WDTAnTRES signal indicates the error to the ECM. For details about the error sources, refer to **Section 15.5.3, Error Detection.**

15.2.2 Block Diagram

Figure 15.1 shows the main components of the WDTA.

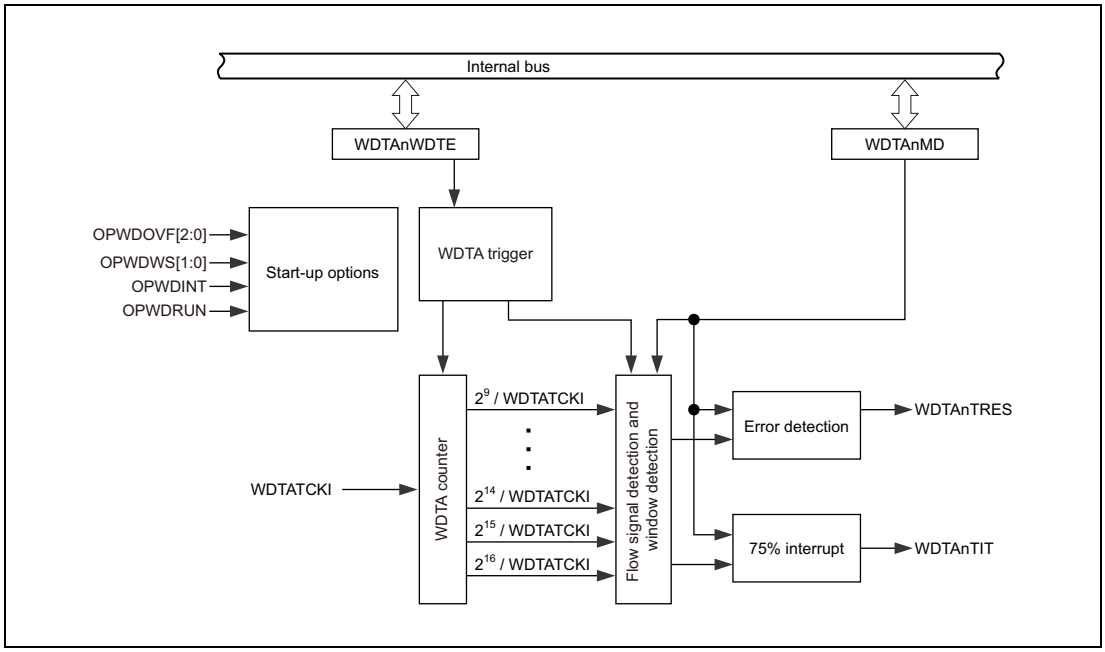


Figure 15.1 Block Diagram of the WDTA

15.3 Register

15.3.1 List of Registers

WDTA registers are listed in the following table.

For details about <WDTAn_base>, see **Section 15.1.2, Register Base Address**.

Table 15.8 WDTA Registers Overview

Module Name	Register Name	Symbol	Address
WDTAn	WDTAn enable register	WDTAnWDTE	<WDTAn_base> + 0_H
WDTAn	WDTAn mode register	WDTAnMD	<WDTAn_base> + C_H

15.3.2 WDTAnWDTE — WDTA Enable Register

This register is the WDTA start control and trigger register.

Writing AC_H to this register generates a WDTA trigger and starts or restarts the WDTA counter. Refer to **Section 15.5.2, WDTA Trigger**, for details.

The writable value of this register is only AC_H.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + 0000_H

Initial Value: (1) WDTA0
The value after reset of WDTA0 depends on the settings of OPWDRUN.
The initial state of OPWDRUN in the product as shipped is OPWDRUN = 0_B.
Accordingly, the initial register value for WDTA0 at the time of shipment is $2C_H$.
(2) WDTA1 *1
The initial register value for WDTA1 is $2C_H$.

For details on the OPWDRUN settings, see **Section 15.5, Function**.
Any reset source triggers initialization.

Note 1. The RH850/C1M does not have WDTA1.

Bit	7	6	5	4	3	2	1	0
	WDTAnRUN[7:0]							
Value after reset	0	0	1	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.9 WDTAnWDTE Register Contents

Bit Position	Bit Name	Function
7 to 0	WDTAnRUN [7:0]	Writing a fixed activation code (AC_H) generates a WDTA trigger to control WDTAn count start and restart. Writing a value other than AC_H generates an error. After WDTAn starts, it cannot be stopped.

The value of bit WDTAnRUN[7] after reset depends on other start-up options as listed below.

Table 15.10 WDTAnRUN[7] Values after Reset

Start-Up Options	
OPWDRUN	**WDTAnRUN[7] Value after Reset**
1	1
0	0

15.3.3 WDTAnMD — WDTA Mode Register

This register specifies the overflow interval time, the 75% interrupt enable/disable, and the window-open period.

It can be updated only once after reset release and before the first trigger. The updated value will be effective after the next WDTA trigger.

Updating this register after the first WDTA trigger has been generated leads to error detection, but an error does not occur if the same value has been written to it.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + $000C_H$

Initial Value: (1) WDTA0
The value after reset of WDTA0 depends on the settings of OPWDOVF2 to OPWDOVF0, OPWDINT, and OPWDWS1 to OPWDWS0. Since the settings of the option bytes are for the initial state of WDTA0 in the product as shipped to be the same as that of WDTA1, OPWDOVF2 to OPWDOVF0, OPWDINT, and OPWDWS1 and OPWDWS0 are all set to 1_B. Accordingly, the initial register value for WDTA0 at the time of shipment is $7F_H$.
(2) WDTA1 *1
The initial register value for WDTA1 is $7F_H$.

This register is initialized by a reset of any type.

Note 1. The RH850/C1M does not have WDTA1.

Bit	7	6	5	4	3	2	1	0
	—	WDTAnOVF[2:0]			WDTAnWIE	—	WDTAnWS[1:0]	
Value after reset	0	1	1	1	1	1	1	1
R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Table 15.11 WDTAnMD Register Contents (1/2)

Bit Position	Bit Name	Function
7	Reserved	When written, write the value after reset.
6 to 4	WDTAnOVF[2:0]	Select the overflow interval time:
3	WDTAnWIE	Enables/disables the 75% interrupt request WDTAnTIT. 0: WDTAnTIT disabled 1: WDTAnTIT enabled
2	Reserved	When written, write the value after reset.

WDTAn OVF2	WDTAn OVF1	WDTAn OVF0	Overflow Interval Time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

Table 15.11 WDTAnMD Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	WDTAnWS[1:0]	Select the window-open period:

WDTAnWS1	WDTAnWS0	Window-Open Period
0	0	25%
0	1	50%
1	0	75%
1	1	100%

15.4 Interrupt Sources

The WDTA detects the state of the WDTA counter value or illegal accesses to the WDTA-related registers, and generates an interrupt request. A WDTA interrupt request is described below.

(1) WDTAnTIT (WDTA timer count 75% interrupt request)
An interrupt request signal is generated at 75% of the counter overflow time of the WDTA timer. An interrupt request signal can be enabled or disabled by using the WDTA mode register (WDTAnMD).

15.5 Function

15.5.1 WDTA after Reset Release

15.5.1.1 Start modes

The WDTA provides two modes for the counter start after release from the reset state:

- Software trigger start mode (for WDTA0 and WDTA1)
  The counter value remains 0000_H after reset release.
  The counter is started with the first WDTA trigger.
- Default start mode (only for WDTA0)
  The counter starts automatically after release from the reset state. However, default start mode is disabled in serial programming mode even if OPWDRUN is set to 1_B for the star-up options.

WDTA1 is fixed to software trigger start mode, and cannot be set by the option byte.

15.5.1.2 Start mode selection (only for WDTA0)

The start mode can be selected by the start-up options.

The start mode selection is listed in **Table 15.12**.

Table 15.12 Start Mode Selection

Start-Up Options		
OPWDRUN	Reset Type	Start Mode
0	Ignored	Software trigger
1		Default

15.5.1.3 WDTA settings after reset release

The WDTA settings are as follows between reset release and the first trigger:

Function	Setting after WDTA0 is Reset	Setting after WDTA1 is Reset
Start mode	Specified by start-up options	Software trigger mode
Overflow interval time	Specified by start-up options	2^{16}/WDTATCKI
75% interrupt mode	Specified by start-up options	75% interrupt enabled
Window-open period	Specified by start-up options	100%

Change WDTA settings

The setting of the WDTA mode register (WDTAnMD) is effective when the first WDTA trigger is generated (writing an activation code to WDTAnWDTE). Change the setting of the WDTAnMD register before a WDTA trigger is generated.

Setting of the WDTA by using WDTAnMD is possible only once. If the value set for WDTAnMD is changed after a WDTA trigger is generated, an error occurs. However, an error does not occur if the same value has been set.

15.5.1.4 Default start mode timing (only for WDTA0)

The default start mode timing and the changes to the WDTA settings are illustrated in **Figure 15.2**.

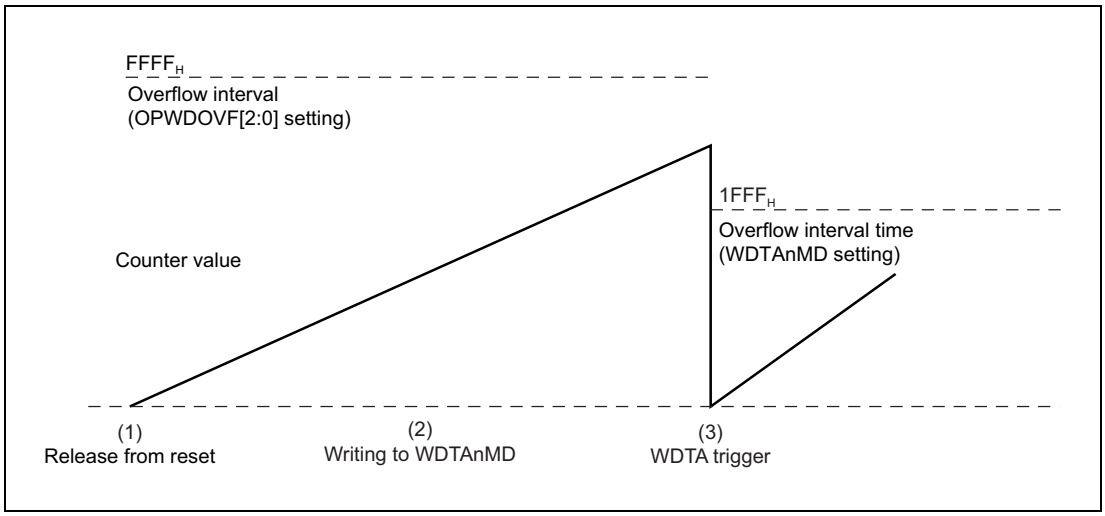


Figure 15.2 Timing Diagram of WDTA Start in Default Start Mode

The timing diagram in **Figure 15.2** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by start-up options.

Example: Overflow interval time after release from the reset state
$= 2^{16}$/WDTATCKI (OPWDOVF[2:0] = 111_B)

(2) WDTAnMD is set before a WDTA trigger is generated. Note, however, that the setting is not applied immediately.

(3) Write to the WDTA trigger register before the WDTA counter overflows.
The WDTAnMD setting is applied due to the WDTA trigger.

Example: Overflow interval time after a WDTA trigger is generated
$= 2^{13}$/WDTATCKI

15.5.1.5 Software trigger start mode timing (common to WDTA0 and WDTA1)

The software trigger start mode timing and the changes to the WDTA settings are illustrated in **Figure 15.3**.

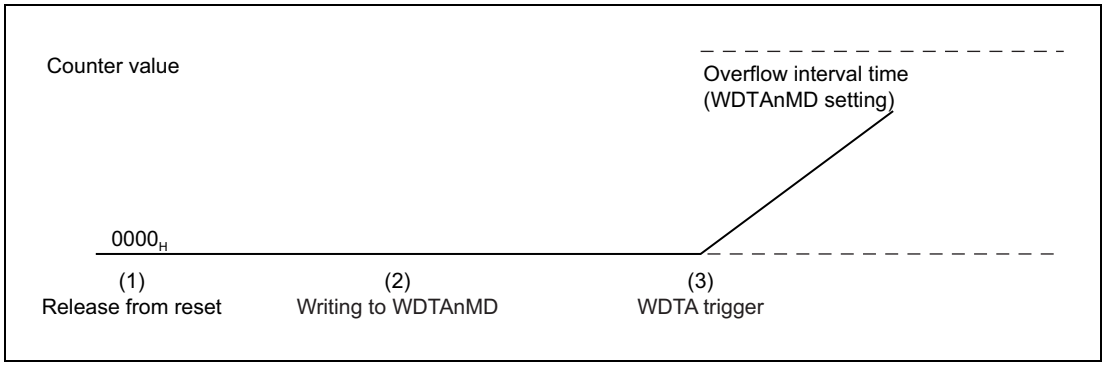


Figure 15.3 Timing Diagram of WDTA Start in Software Trigger Start Mode

The timing diagram in **Figure 15.3** shows the following behaviors:

(1) After release from the reset state, the counter remains 0000_H until the first WDTA trigger.
The overflow interval time is set by using the start-up options, but it does not have any effect.

(2) WDTAnMD is set before the first WDTA trigger. However, the settings are not applied immediately.

(3) The WDTA counter starts at the first WDTA trigger.
The overflow interval time and other settings specified in WDTAnMD are applied.

15.5.2 WDTA Trigger

Writing a special value called an activation code to the WDTA enable register (WDTAnWDTE) leads to generation of a WDTA trigger.

The WDTA trigger has the following functions:

- Starting the WDTA counter in software trigger start mode
- Restarting the WDTA counter
- WDTA mode setting by the WDTAnMD register (only for the first WDTA trigger after release from the reset state)

The WDTA can be triggered by writing a fixed activation code to the trigger register.

Table 15.13 Trigger Register and Activation Code

Type of Activation Code	Trigger Register	Activation Code
Fixed	WDTAnWDTE	AC_H

15.5.3 Error Detection

The WDTA detects an overflow of the WDTA counter and illegal operations as an error.

The conditions for error detection are:

- WDTA counter overflow
- Wrong activation code is written to the WDTA trigger register.
- Writing to the trigger register at the time outside the window-open period
- When an attempt is made to change the setting value of the WDTA mode register WDTAnMD after the first WDTA trigger has been generated
- When updating the setting value of the WDTA mode register WDTAnMD twice before the first WDTA trigger is generated

15.5.4 WDTA Error Mode

When an error is detected, the WDTAnTRES signal indicates the error to the ECM.

Figure 15.4 shows generation of a reset when the counter overflows and default start mode is selected.

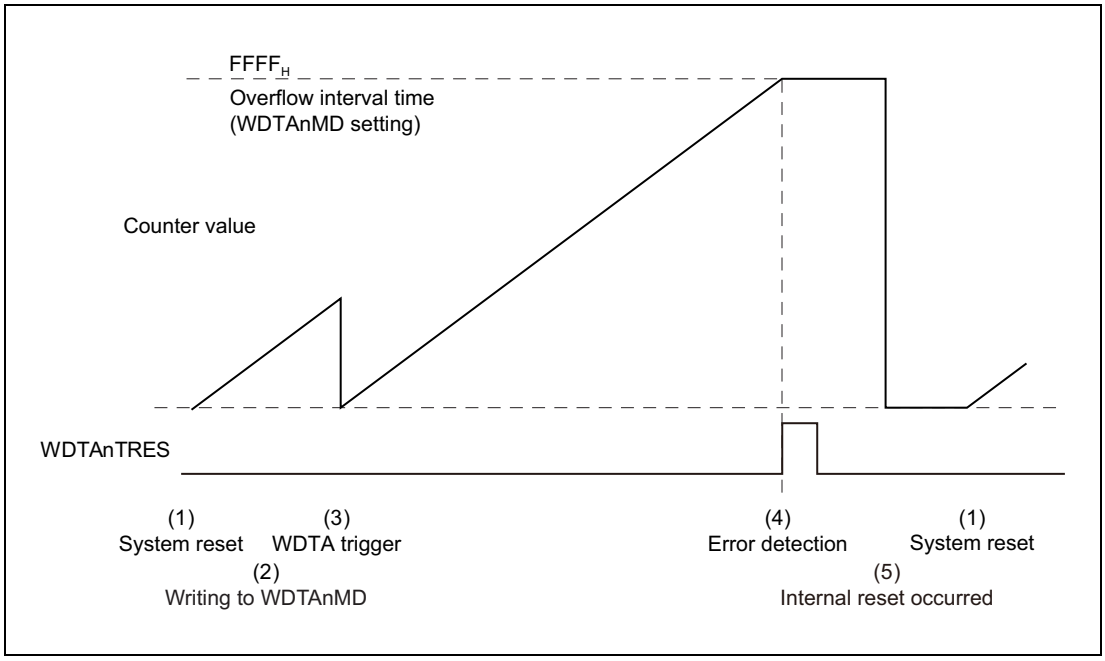


Figure 15.4 Timing Diagram of WDTA Internal Reset Generation

The timing diagram in **Figure 15.4** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by using start-up options.

(2) WDTAnMD is set before a WDTA trigger is generated.
In this case, 2^{16}/WDTATCKI is set for the overflow interval time.

(3) The WDTAnMD setting is applied due to the WDTA trigger.

(4) If the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(5) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

15.5.5 75% Interrupt Request Signals

When the WDTA counter reaches 75% of the time set for the overflow interval, an interrupt request WDTAnTIT is generated.

By use of WDTAnMD.WDTAnWIE this function can be enabled or disabled afterwards.

Figure 15.5 shows the 75% interrupt request generation under following conditions:

- Default start mode is selected.
- 75% interrupt request is enabled after the first WDTA trigger is generated.
- WDTA overflow interval time: 2^{16}/WDTATCKI.

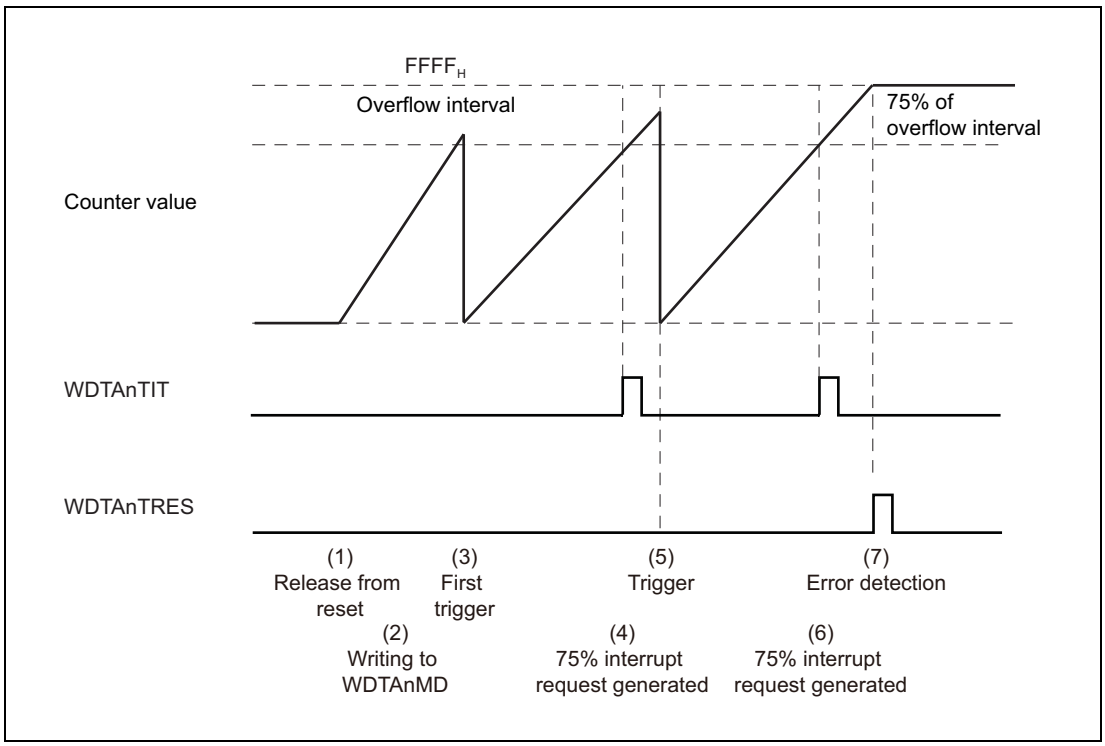


Figure 15.5 Timing Diagram of WDTA 75% Interrupt Request Signals

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) When the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(5) The counter restarts at the WDTA trigger.

(6) When the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(7) When the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

15.5.6 Window Function

The period when a WDTA trigger is valid (window-open period) can be set.
If the window-open period is set to the value less than 100%, an error occurs by the WDTA trigger generated at the time outside the window-open period. The window-open period after release from the reset state is 100%. The period is set to the value set in the WDTAnMD.WDTAnWS[1:0] after the first WDTA trigger is generated.

Figure 15.6 shows window function operations under the following conditions.

- Default start mode is selected.
- The 25% window open period is valid (WDTAnWS[1:0] = 00_B) after the first WDTA trigger.
- WDTA overflow interval time period: 2^{16}/WDTATCKI

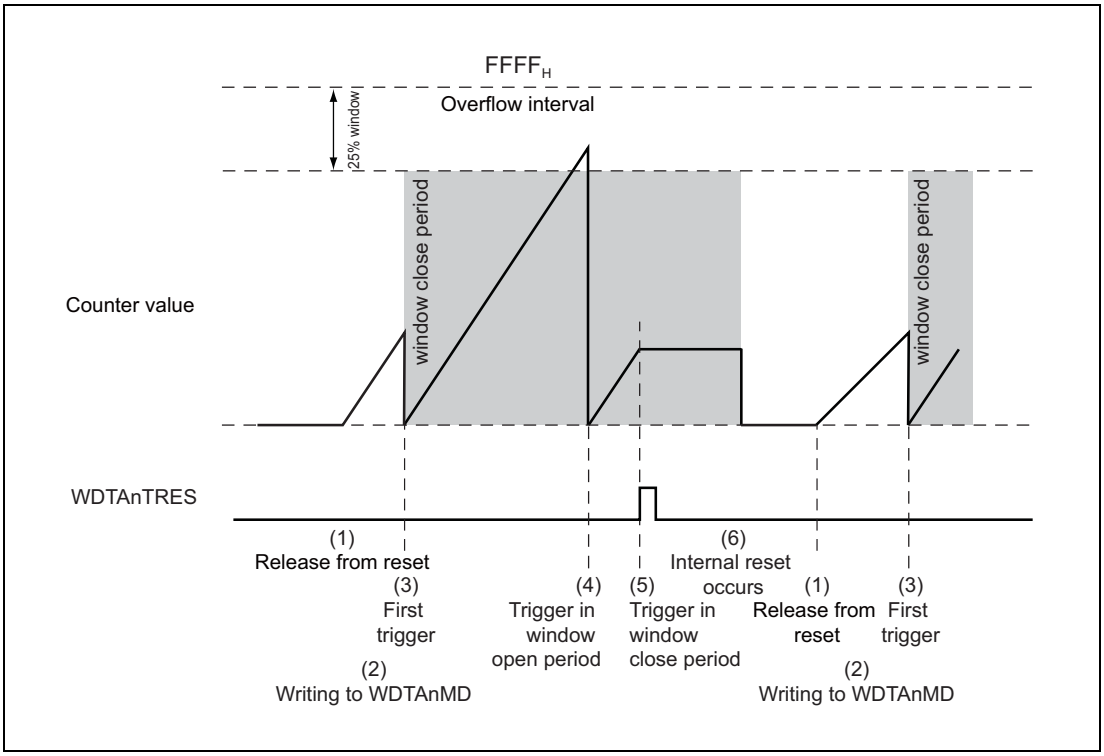


Figure 15.6 Timing Diagram of WDTA Window Function

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) The WDTA counter restarts at the WDTA trigger during the window open period.

(5) An error is detected at the WDTA trigger during the window close period, and then the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(6) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

Section 16 OS Timer (OSTM)

This section contains a generic description of the OS timer (OSTM).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of the OSTM.

16.1 Features of RH850/C1x OSTM

16.1.1 Number of Units

This microcontroller has the following number of OSTM units.

Table 16.1 Number of Units

Product	C1H	C1M
Number of Units	3	2
Name	OSTMn (n = 0 to 2)	OSTMn (n = 0 to 1)

Table 16.2 Index

Index	Meaning
n	Throughout this section, the individual OSTM units are identified by the index “n” (n = 0 to 2 for C1H; n = 0, 1 for C1M); for example, OSTMnCNT is the OSTM counter register.

16.1.2 Register Base Address

OSTM base addresses are listed in the following table.

OSTM register addresses are given as offsets from the base addresses in general.

Table 16.3 Register Base Address

Base Address Name	Base Address
<OSTM0_base>	FFEC 0000_H
<OSTM1_base>	FFEC 1000_H
<OSTM2_base>	FFEC 2000_H

Note: OSTM2 is not supported by RH850/C1M.

16.1.3 Clock Supply

The OSTM clock supply is shown in the following table.

Table 16.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
OSTMn	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)

Note: OSTM2 is not supported by RH850/C1M.

16.1.4 Interrupts

OSTM interrupt requests are listed in the following table.

Table 16.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number
OSTM0TINT	OSTM0 interrupt	25
OSTM1TINT	OSTM1 interrupt	26
OSTM2TINT	OSTM2 interrupt	27

Note: OSTM2 is not supported by RH850/C1M.

16.1.5 Reset Sources

OSTM reset sources are listed in the following table. The OSTM is initialized by these reset sources.

Table 16.6 Reset Source

Unit Name	Reset Source
OSTMn	All reset sources

16.2 Overview

The OSTM is a 32-bit timer/counter.

It can be used in interval timer mode or in free-run compare mode. The settings for operating mode specify the direction of counting (up or down) to control the generation of interrupt requests.

A counter-start signal (OSTMnTSST) is input to the OSTM, allowing synchronization of these timers with other peripheral functions.

16.2.1 Functional Overview

The OSTM has the following features.

- Two operating modes
  - Interval timer mode
  - Free-run compare mode
- Simultaneous start trigger function between the units
- OSTMnTINT interrupt

16.2.2 Block Diagram

The following block diagram shows the main components of the OSTM.

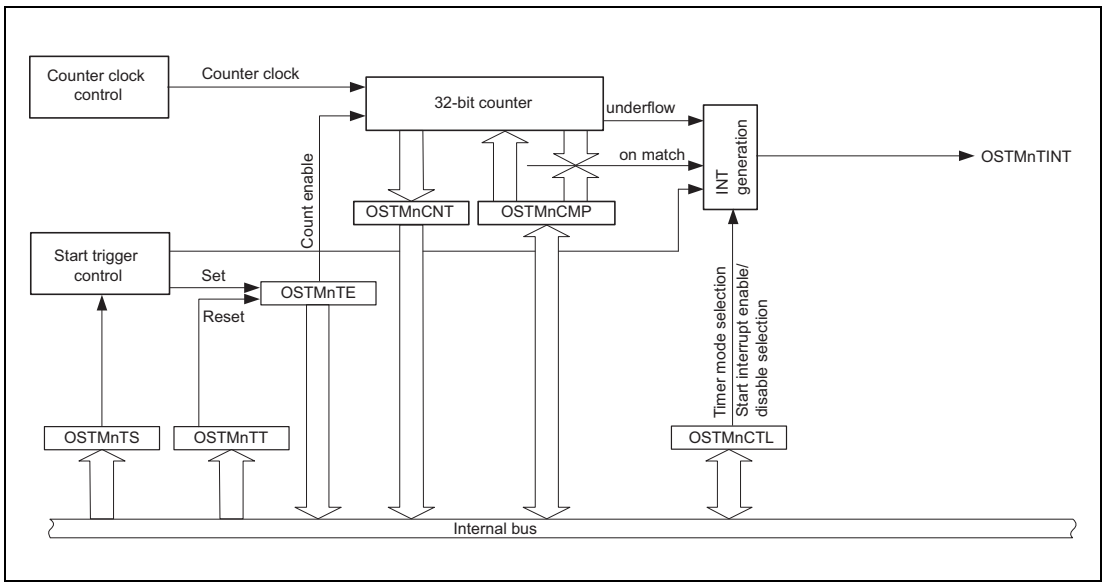


Figure 16.1 Block Diagram of the OSTM

16.2.3 Counter Clock

The OSTM uses PCLK as the counter clock.

16.2.4 Interrupt Request (OSTMnTINT)

By default, an OSTMnTINT interrupt request is generated on counter underflow (interval timer mode) or when the counter matches the compare value (free-run compare mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

Since OSTMnTINT triggers toggling of the OSTMnTTOUT output in timer-output toggling mode (OSTMnTOE.OSTMnTOE is 1), the setting of the OSTMnCTL.OSTMnMD0 bit also affects the output (OSTMnTTOUT).

This is illustrated in the following figure.

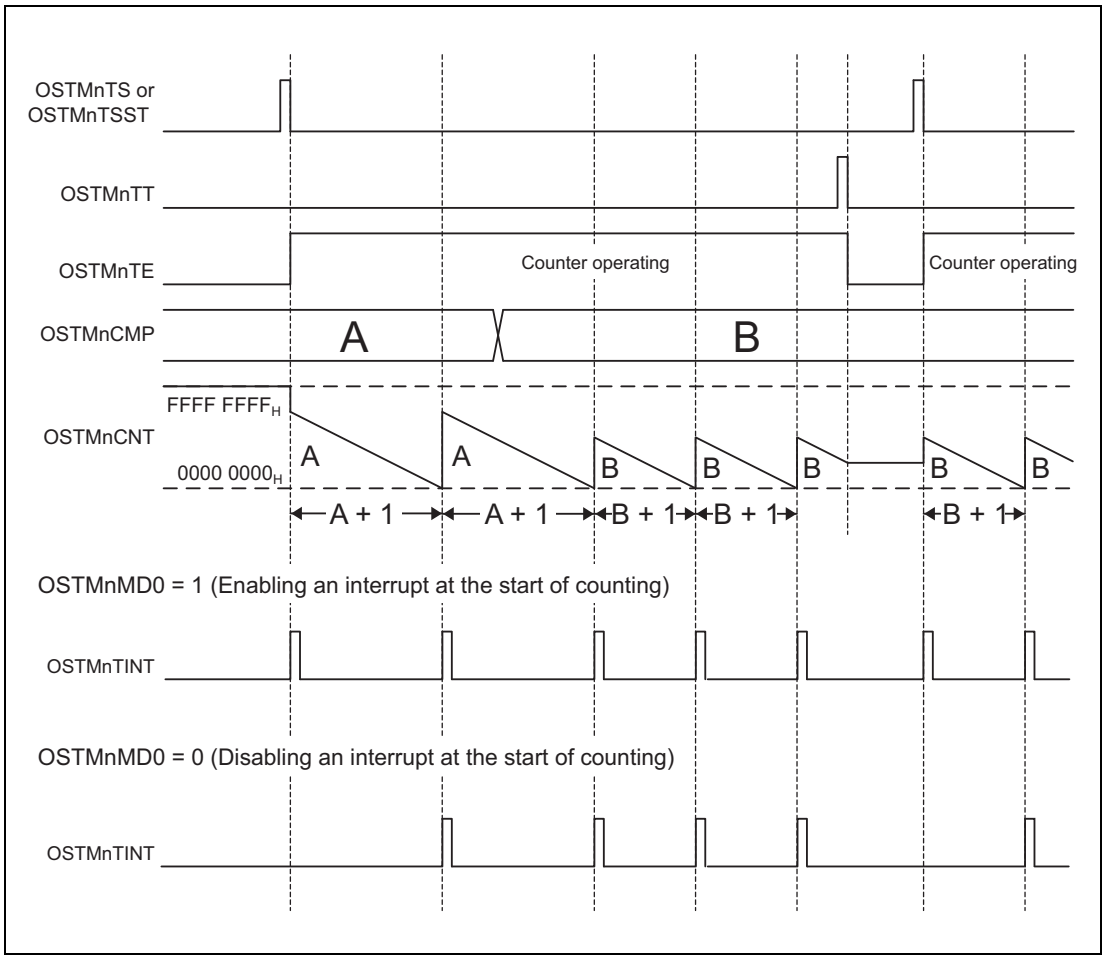


Figure 16.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode)

16.3 Registers

16.3.1 List of Registers

OSTM registers are listed in the following table.

For details about <OSTMn_base>, see **Section 16.1.2, Register Base Address**.

Table 16.7 Registers

Module	Register	Symbol	Address
OSTMn	OSTMn compare register	OSTMnCMP	<OSTMn_base> + 00_H
OSTMn	OSTMn counter register	OSTMnCNT	<OSTMn_base> + 04_H
OSTMn	OSTMn output register	OSTMnTO	<OSTMn_base> + 08_H
OSTMn	OSTMn output enable register	OSTMnTOE	<OSTMn_base> + $0C_H$
OSTMn	OSTMn count enable status register	OSTMnTE	<OSTMn_base> + 10_H
OSTMn	OSTMn count start trigger register	OSTMnTS	<OSTMn_base> + 14_H
OSTMn	OSTMn count stop trigger register	OSTMnTT	<OSTMn_base> + 18_H
OSTMn	OSTMn control register	OSTMnCTL	<OSTMn_base> + 20_H

16.3.2 OSTMnCMP — OSTMn Compare Register

This register stores the start value of the counter or the value with which the counter is compared, depending on the operation mode.

Access: This register can be read/written in 32-bit units.

Address: <OSTMn_base> + 00_H

Value after reset: $0000\ 0000_H$

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

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

Table 16.8 OSTMnCMP Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCMP [31:0]	• In interval timer mode: start value of the counter • In free-run compare mode: compare value

16.3.3 OSTMnCNT — OSTMn Counter Register

This register indicates the counter value of the timer.

Access: This register can be read in 32-bit units.

Address: <OSTMn_base> + 04_H

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 16.9 OSTMnCNT Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCNT [31:0]	Timer counter value

Table 16.10 shows the correspondence between OSTMn's operating mode, counting direction, and initial value. The initial value is the value read from the counter after a change to the operating mode.

Table 16.10 Correspondence between Operating Mode, Counting Direction and Initial Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial Value
Interval timer mode	0*1	Down	FFFF $FFFF_H$
Free-run compare mode	1	Up	0000 0000_H

Note 1. Value after reset.

16.3.4 OSTMnTO — OSTMn Output Register

This register is used to specify and read the level of an OSTMnTTOUT output signal. The setting of this register is only valid in OSTMn (n = 0).

Access: This register can be read/written in 8-bit units. Writing can only proceed when the software control mode is selected (OSTMnTOE.OSTMnTOE = 0).

Address: <OSTMn_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTO
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 16.11 OSTMnTO Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTO	This bit is used to specify and read the level of the OSTMnTTOUT output signal. 0: Low level 1: High level

16.3.5 OSTMnTOE — OSTMn Output Enable Register

This register specifies OSTMnTTOUT output mode. The setting of this register is only valid in OSTMn (n = 0).

Access: This register can be read/written in 8-bit units.

Address: <OSTMn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTOE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 16.12 OSTMnTOE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTOE	This bit specifies the OSTMnTTOUT output mode. 0: Software control mode: The level corresponding to the setting of OSTMnTO.OSTMnTO is output to OSTMnTTOUT. 1: Timer-output toggling mode: OSTMnTTOUT output is toggled whenever an OSTMnTINT interrupt request is generated.

16.3.6 OSTMnTE — OSTMn Count Enable Status Register

This register indicates whether the counter is enabled or disabled.

Access: This register can be read in 8-bit units.

Address: <OSTMn_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 16.13 OSTMnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	OSTMnTE	Indicates whether the counter is enabled or disabled: 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1, or to OSTMnTSST being 1. Setting OSTMnTT.OSTMnTT to 1 resets this bit to 0.

NOTE

If the counter is disabled, the counter value OSTMnCNT retains its value.

If the counter is restarted, it

- restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or
- restarts counting up from the counter value 0000 0000_H if it is in free-run compare mode.

16.3.7 OSTMnTS — OSTMn Count Start Trigger Register

This register starts the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 16.14 OSTMnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTS	Starts the counter: 0: This setting disables the counter. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. • In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. • In free-run compare mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

16.3.8 OSTMnTT — OSTMn Count Stop Trigger Register

This register stops the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 16.15 OSTMnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTT	Stops the counter: 0: This setting disables the counter. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

16.3.9 OSTMnCTL — OSTMn Control Register

This register specifies the operating mode for the counter and controls the generation of OSTMnTINT interrupt requests when counting starts.

Although this register is readable and writable, writing to it is only possible when OSTMnTE.OSTMnTE = 0; that is, the register becomes read-only when OSTMnTE.OSTMnTE = 1.

Access: This register can be read/written in 8-bit units.

Address: <OSTMn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OSTMnMD1	OSTMnMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 16.16 OSTMnCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When writing, write the value after reset.
1	OSTMnMD1	Specifies the operating mode for the counter: 0: Interval timer mode 1: Free-run compare mode
0	OSTMnMD0	Controls the generation of OSTMnTINT interrupt requests at the start of counting: 0: Interrupts when counting starts are disabled. 1: Interrupts when counting starts are enabled.

16.4 Functions

16.4.1 Starting and Stopping the Timer

The OSTM is started and stopped as follows:

Starting the timer

The timer is started in either of the following ways:

- Setting the OSTMnTS.OSTMnTS bit to 1 or
- Setting the OSTMnTSST signal to the high level (when the Simultaneous start trigger function is in use)

The OSTMnTE.OSTMnTE status bit is set to 1.

The counter starts to count up or down in accord with the settings for operating mode. For details, refer to **Section 16.4.2, Interval Timer Mode** and **Section 16.4.3, Free-Run Compare Mode**.

If the OSTMnTS.OSTMnTS bit is to be used to start the timer, the OSTMnTSST input must correspond to logical zero.

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

When the counter is stopped, values in the OSTMnTO and OSTMnCNT registers and the level of the OSTMnTTOUT output are retained until further counting operations start.

Simultaneous start trigger function

The OSTMnTSST signal output from the PIC1A module can be used to start multiple timers at the same time. Refer to **Section 23, Peripheral Interconnection (PIC)**.

16.4.2 Interval Timer Mode

In interval timer mode, the OSTM can be used as a reference timer generating interrupt requests at fixed intervals.

16.4.2.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter underflows (reaches 0000 0000_H).

To select interval timer mode, set OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000 0000_H is reached. Then the counter continues with the new value.

OSTMnTINT period

The periods of OSTMnTINT is:

- OSTMnTINT generation period = counter clock period × (OSTMnCMP + 1)

The following figure shows the basic operation of the OSTM in interval timer mode when interrupts at the start of counting are enabled.

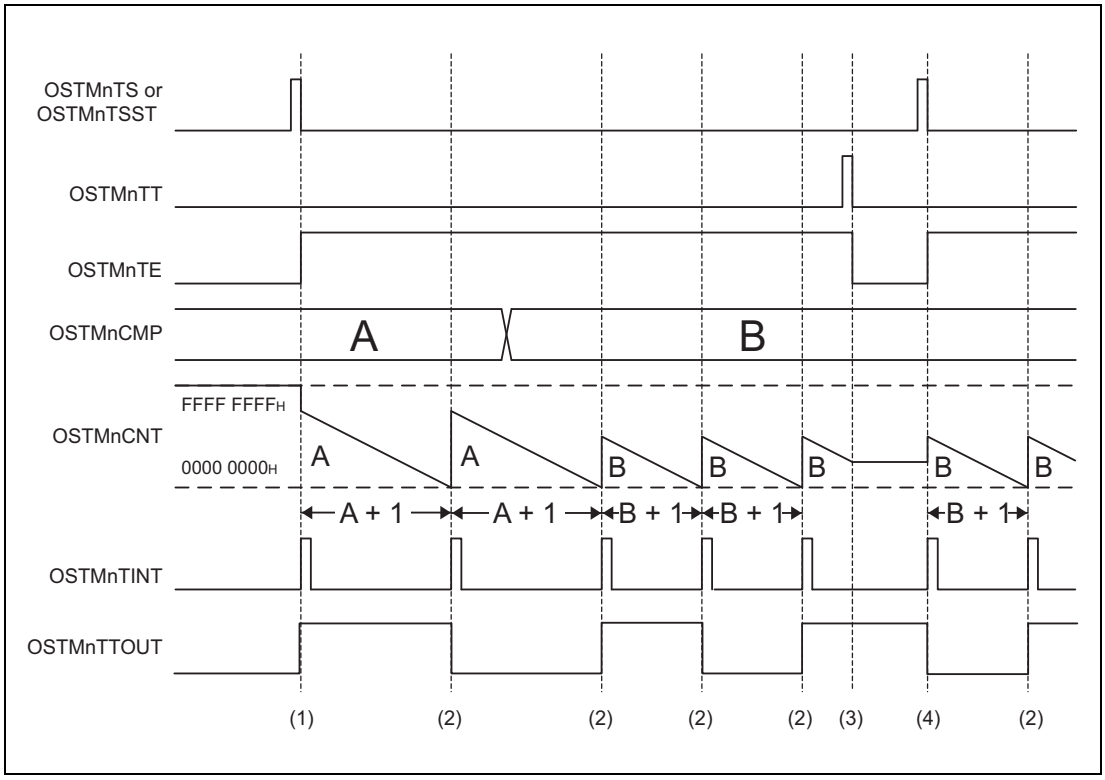


Figure 16.3 Timing Diagram of OSTM in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting-down from the value of OSTMnCMP.
If OSTMnCTL.OSTMnMD0 is 1, OSTMnTINT interrupt requests are generated at the start of counting. The OSTMnCNT register indicates the current value of the counter.

(2) When the counter reaches $0000\ 0000_H$, an OSTMnTINT interrupt request is generated. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1, or OSTMnTSST = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 or by a transition of the OSTMnTSST signal from the high level to the low level during counting.

The counter loads the start value from the OSTMnCMP register and continues to counting down.

The following figure shows the forced restart of the OSTM in interval timer mode, with interrupts at the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

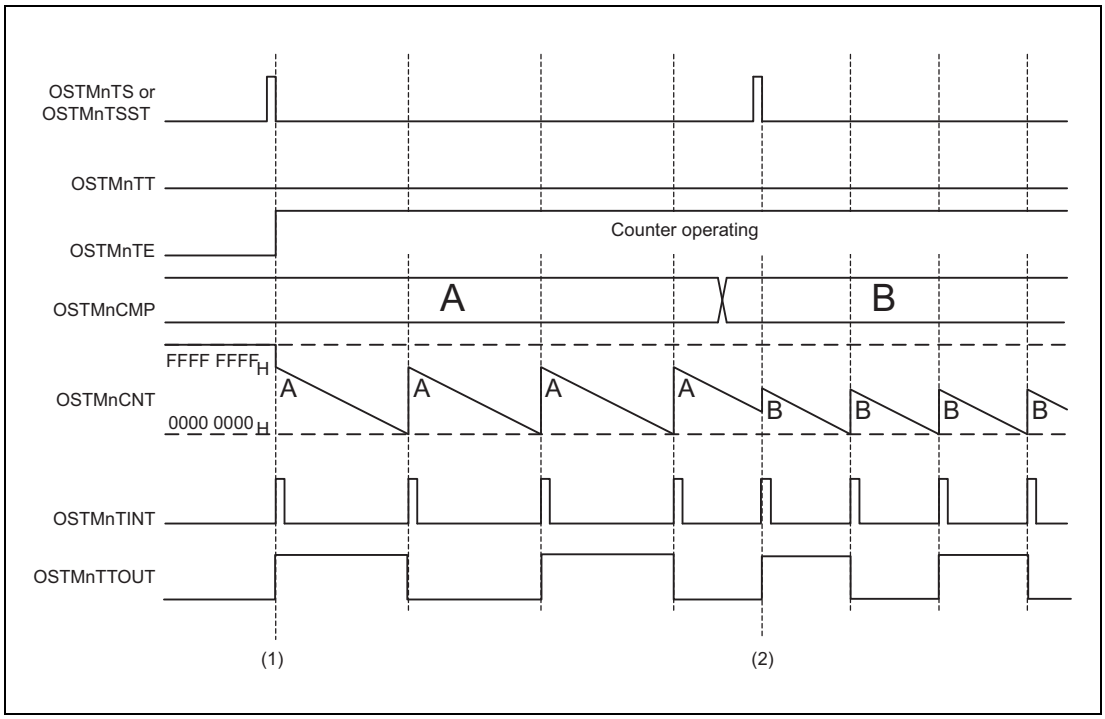


Figure 16.4 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 16.3, Timing Diagram of OSTM in Interval Timer Mode**.

(2) Setting OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
The counter immediately restarts counting down, starting with the current value of OSTMnCMP.
When OSTMnCTL.OSTMnMD0 = 1, an OSTMnTINT interrupt request is generated when counting starts.

16.4.2.2 Operation when OSTMnCMP = 0000 0000$_H$

When OSTMnCMP = 0000 0000$_H$ the OSTM behaves as follows.

- When the counter is enabled, the OSTMnTINT interrupt request is always set to 1.
- When the OSTMnTTOUT signal is in timer-output toggling mode, the output is toggled on every cycle of PCLK.

The following figure shows operations of the OSTM when OSTMnCMP = 0000 0000$_H$, and interrupts at the start of counting are enabled.

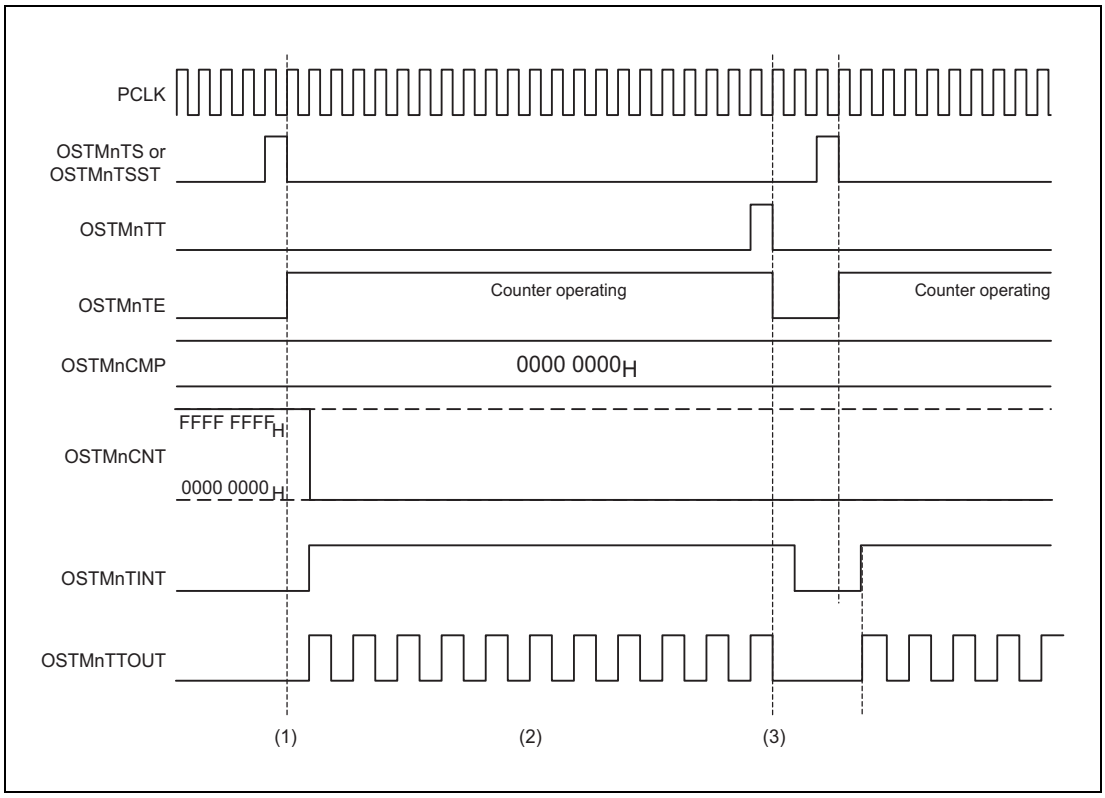


Figure 16.5 Timing Diagram when OSTMnCMP = 0000 0000$_H$ in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP, so the value 0000 0000$_H$ is retained in OSTMnCMP.

(2) The OSTMnTINT interrupt request is continuously asserted.

(3) After the counter stops, the OSTMnTINT interrupt request signal is deasserted.

When interrupts at the start of counting are disabled, interrupts are not generated at the start timing of the counter.

16.4.2.3 Setting Procedure for Interval Timer Mode

The setting procedure in interval timer mode after reset release is described below:

Setting procedure

(1) Set the start value of the counter in the OSTMnCMP register.

(2) Select interval timer mode by clearing the OSTMnCTL.OSTMnMD1 bit.

(3) Select enabling or disabling of interrupts when counting starts (OSTMnCTL.OSTMnMD0).

16.4.3 Free-Run Compare Mode

16.4.3.1 Basic Operation in Free-Run Compare Mode

In free-run compare mode, the counter counts up from 0000 0000_H to FFFF $FFFF_H$. When the value of the OSTMnCMP register matches the current counter value, an OSTMnTINT interrupt request is output.

When free-run compare mode is used, set OSTMnCTL.OSTMnMD1 to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of the OSTM in free-run compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1).

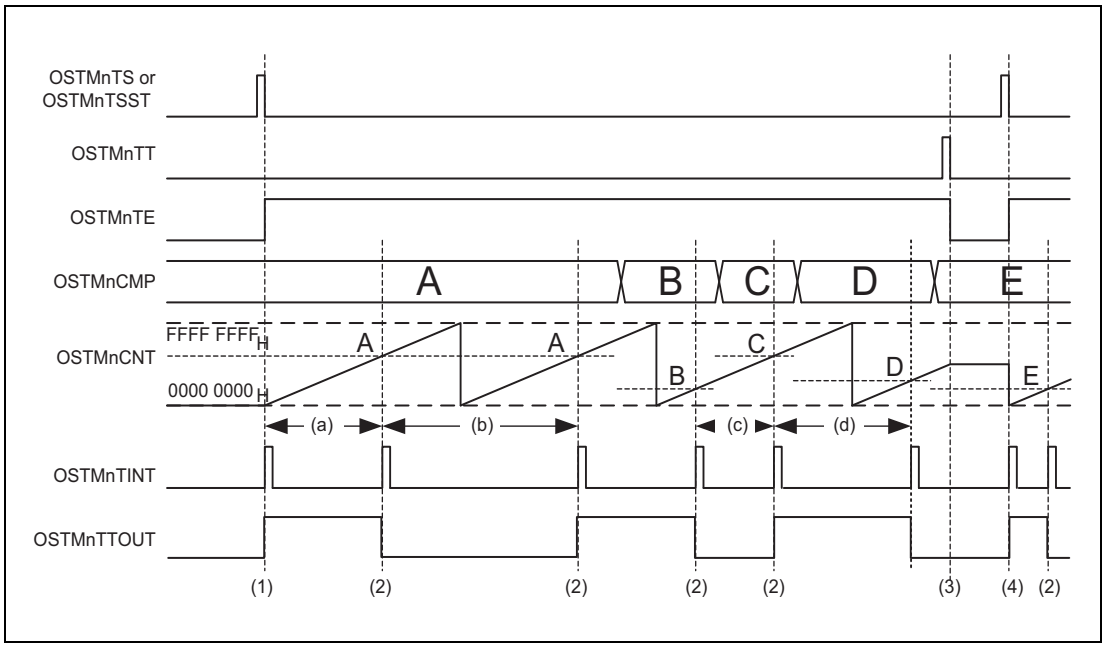


Figure 16.6 Timing Diagram of OSTM in Free-Run Compare Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from 0000 0000_H to FFFF $FFFF_H$. The OSTMnCNT register indicates the current value of the counter. When OSTMnCTL.OSTMnMD0 = 1, an interrupt request OSTMnTINT is generated when the counter starts.

(2) When the current counter value matches the value in the OSTMnCMP register, an OSTMnTINT interrupt request is generated.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from 0000 0000_H when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1.

OSTMnTINT period

The OSTMnTINT generation period is different at the start of counting and depends on the old and new compare value if OSTMnCMP is rewritten during operation.

Table 16.17 OSTMnTINT Generation Timing

Old value for comparison	New value for comparison	Counter value at time of rewriting	Period of OSTMnTINT Generation	Label in timing diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(FFFF $FFFF_H$ + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C - B) × counter clock period	(c)
C	D < C	Counter value > D,C	(FFFF $FFFF_H$ - C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set, or if OSTMnTSST = 1. The counter ignores the attempted setting and continues counting.

16.4.3.2 Operation when OSTMnCMP = 0000 0000_H

The following figure shows the operation of the OSTM when OSTMnCMP = 0000 0000_H, interrupts at the start of counting are enabled (OSTMnCTL.OSTMnMD0 = 1).

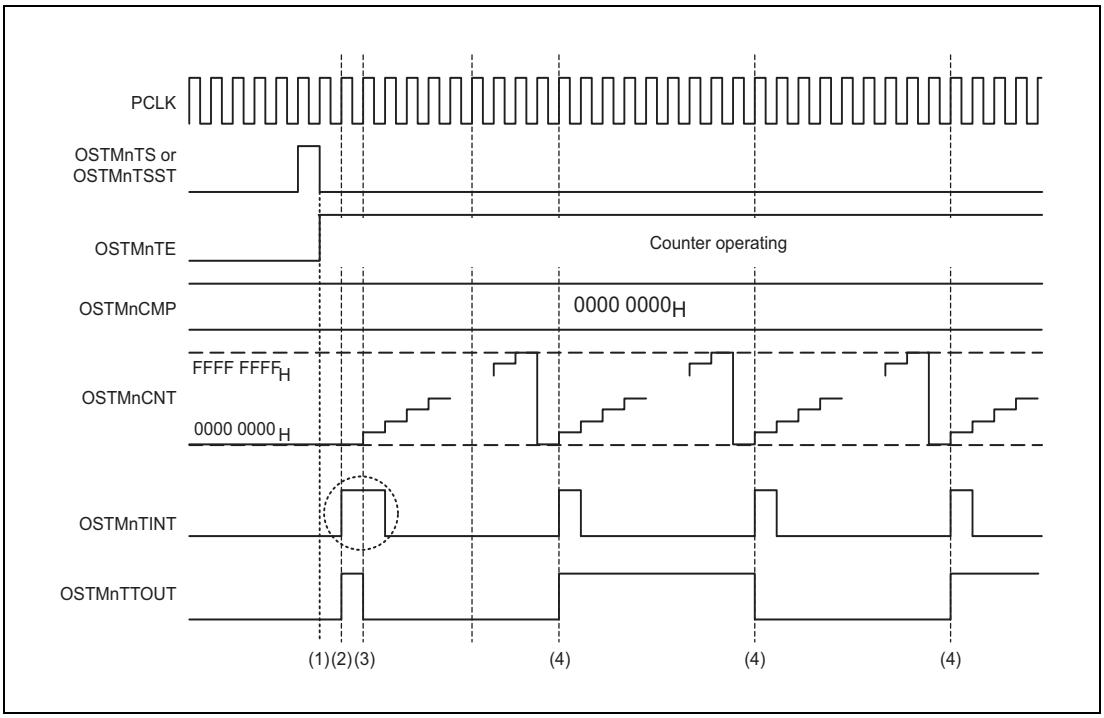


Figure 16.7 Timing Diagram when OSTMnCMP = 0000 0000_H in Free-Run Compare Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from 0000 0000_H to FFFF $FFFF_H$.

(2) An OSTMnTINT interrupt request is generated when counting starts.

(3) If the current counter value matches OSTMnCMP, an interrupt request OSTMnTINT is generated.

If OSTMnCMP = 0000 0000_H in the above case, OSTMnTINT is generated over two clock cycles.

(4) Every (FFFF $FFFF_H$ + 1) clock cycle the OSTMnTINT interrupt request is asserted.

When interrupts at the start of counting are disabled, interrupts are not generated at the start timing of the counter.

16.4.3.3 Setting Procedure for Free-Run Compare Mode

The setting procedure in free-run compare mode after reset release is described below:

Setting Procedure

(1) Set the compare value in the OSTMnCMP register.

(2) Select free-run compare mode by setting the OSTMnCTL.OSTMnMD1 bit.

(3) Enable or disable interrupts when counting is started by the OSTMnCTL.OSTMnMD0 bit.

Section 17 Timer Array Unit D (TAUD)

17.1 Overview of RH850/C1x TAUD

17.1.1 Units

This LSI has the following number of TAUD units.

Table 17.1 Unit of TAUD

Product	RH850/C1x
Number of units	2
Name	TAUDn (n = 0 to 1)

Table 17.2 Index

Index	Meaning
n	Throughout this section, the individual TAUD units are identified by the index “n” (n = 0, 1); for example, TAUD0CNT0 is the TAUDn control register 0.
m	The TAUD has up to 16 channels. Throughout this section, the individual channels are identified by the index “m” (m = 0 to 15).

The number of channels included in the individual TAUD units is listed in the following table.

Table 17.3 Channels

Unit Name	Number of Channels
TAUDn	16

17.1.2 Register Base Addresses

TAUD base addresses are listed in the following table.

TAUD register addresses are given as offsets from the base addresses in general.

Table 17.4 Register Base Addresses

Base Address Name	Base Address
<TAUD0_base>	FFE2 0000_H
<TAUD1_base>	FFE2 1000_H

17.1.3 Clock Supply

TAUD clock is listed in following table.

Table 17.5 TAUDn Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TAUDn	PCLK	CLKC_HSB (Unmodulated high-speed peripheral clock)

17.1.4 Interrupts Requests

TAUD interrupt requests are listed in the following table.

Table 17.6 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMAC Trigger Number	DTS Trigger Number
TAUD0				
INTTAUD0I0-15	Channels 0 to 15 interrupt	82 to 97	8 to 23	16 to 31
TAUD1				
INTTAUD1I0-15	Channels 0 to 15 interrupt	98 to 113	24 to 39	32 to 47

17.1.5 Reset Sources

TAUD reset sources are listed in the following table. TAUD is initialized by these reset sources.

Table 17.7 Reset Sources

Unit Name	Reset Source
TAUDn	Reset by any reset source

17.1.6 External Input/Output Signals

External input/output signals of TAUD are listed in the following table.

Table 17.8 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TAUD0		
TAUDTTINm	Channel m input	TAUD0Im
TAUDTTOUTm	Channel m output	TAUD0Om
TAUD1		
TAUDTTINm	Channel m input	TAUD1Im
TAUDTTOUTm	Channel m output	TAUD1Om

CAUTION

When channel input pins are to be used, noise filters must be set for the corresponding port pin functions.

17.2 Overview

17.2.1 Functional Overview

The TAUD has the following functions:

- 16 channels
- 16-bit counter and 16-bit data register per channel
- Independent channel operation
- Synchronous channel operation (master and slave operation)
- Generation of different types of output signal
- Real-time output
- Counter can be triggered by external signal
- Interrupt generation

The Timer Array Unit D is used to perform various count or timer operations and to output a signal which depends on the result of the operation. It contains one prescaler block for count clock generation and 16 channels, each equipped with a 16 bit counter TAUDnCNTm and a 16-bit data register TAUDnCDRm to hold the start or compare value of the counter.

It also contains several control and status registers.

Independent and synchronous operation

Every channel can operate in different operation modes, either independently or in combination with other channels (synchronously), i.e. multiple channels depend on each other with one master and one or more slave channels.

When a channel is operated independently, its operation mode and functions are not affected by those of other channels. When a channel is operated synchronously it is either a master or a slave. A master channel can have multiple slaves, and the state of one channel affects that of the other channels. For example, this means that one channel can control when another starts to count, is reset, etc.

CAUTION

The timing chart described in this section shows an operating timing image. A delay time is added to a timer input. For details, see Section 17.4.8, TAUDTTINm Edge Detection.

17.2.2 Terms

In this chapter, the following terms are used:

Independent / synchronous channel operation

Independent or synchronous channel operation describes the dependency of channels on each other:

- If a channel operates independent of all other channels, this is called independent channel operation.
- If a channel operates depending on other channels, this is called synchronous channel operation.

Channel group

In synchronous channel operation, all channels that depend on each other are referred to as a “channel group”.

A channel group has one master channel and one or more slave channels.

Operation mode

An operation mode can be selected for every channel m. The operation mode defines the basic operation and features of a channel.

In synchronous channel operation, every channel in the channel group can operate in a different operation mode.

Examples are “Capture Mode”, “Event Count Mode”, and “Interval Timer Mode”.

Channel output mode

The channel output mode defines the operation of TAUDTTOUTm

- of a single channel (independent output operation) or
- of all channels in a channel group (synchronous output operation).

Examples are “Independent Channel Output Mode 1” and “Synchronous Channel Output Mode 2 with Dead Time Output”.

Channel operation function

The channel operation function defines the complete function and all features

- of a single channel (independent channel operation) or
- of all channels in a channel group (synchronous channel operation).

Upper/lower channel

Depending on the channel number m, a channel with a smaller number or with a larger number is referred to as “upper” or “lower” channel, respectively.

- Upper channel: Channel with a smaller channel number
- Lower channel: Channel with a larger channel number

Example:

For channel 5, channel 3 is an upper channel and channel 9 is a lower channel.

17.2.3 Functional List of Timer Operations

This timer provides the following functions by operating individual channels independently or a combination of channels.

Table 17.9 Functional List of TAUD Operations

Operation function	Example
Independent Channel Operation Functions	**Section 17.4.9**
Interval Timer Function	**Section 17.4.9.1**
TAUDTTINm Input Interval Timer Function	**Section 17.4.9.2**
Clock Divide Function	**Section 17.4.9.3**
External Event Count Function	**Section 17.4.9.4**
Delay Count Function	**Section 17.4.9.5**
One-Pulse Output Function	**Section 17.4.9.6**
TAUDTTINm Input Pulse Interval Measurement Function	**Section 17.4.9.7**
TAUDTTINm Input Signal Width Measurement Function	**Section 17.4.9.8**
TAUDTTINm Input Position Detection Function	**Section 17.4.9.9**
TAUDTTINm Input Period Count Detection Function	**Section 17.4.9.10**
TAUDTTINm Input Pulse Interval Judgment Function	**Section 17.4.9.11**
TAUDTTINm Input Signal Width Judgment Function	**Section 17.4.9.12**
One-Phase PWM Output Function	**Section 17.4.12.11**
Independent Channel Real-Time Functions	**Section 17.4.10**
Real-Time Output Function Type 1	**Section 17.4.10.1**
Real-Time Output Function Type 2	**Section 17.4.10.2**
Independent Channel Simultaneous Rewrite Functions	**Section 17.4.11**
Simultaneous Rewrite Trigger Generation Function Type 1	**Section 17.4.11.1**
Synchronous Channel Operation Functions	**Section 17.4.12**
PWM Output Function	**Section 17.4.12.1**
One-Shot Pulse Output Function	**Section 17.4.12.2**
Trigger Start PWM Output Function	**Section 17.4.12.3**
Delay Pulse Output Function	**Section 17.4.12.4**
Offset Trigger Output Function	**Section 17.4.12.5**
A/D Conversion Trigger Output Function Type 1	**Section 17.4.12.6**
Triangle PWM Output Function	**Section 17.4.12.7**
Triangle PWM Output Function with Dead Time	**Section 17.4.12.8**
A/D Conversion Trigger Output Function Type 2	**Section 17.4.12.9**
Interrupt Request Signals Culling Function	**Section 17.4.12.10**
Synchronous Non-Complementary and Complementary Modulation Output Functions	**Section 17.4.13**
Non-Complementary Modulation Output Function Type 1	**Section 17.4.13.1**
Non-Complementary Modulation Output Function Type 2	**Section 17.4.13.2**
Complementary Modulation Output Function	**Section 17.4.13.3**

17.2.4 TAUD I/O and Interrupt Request Signals

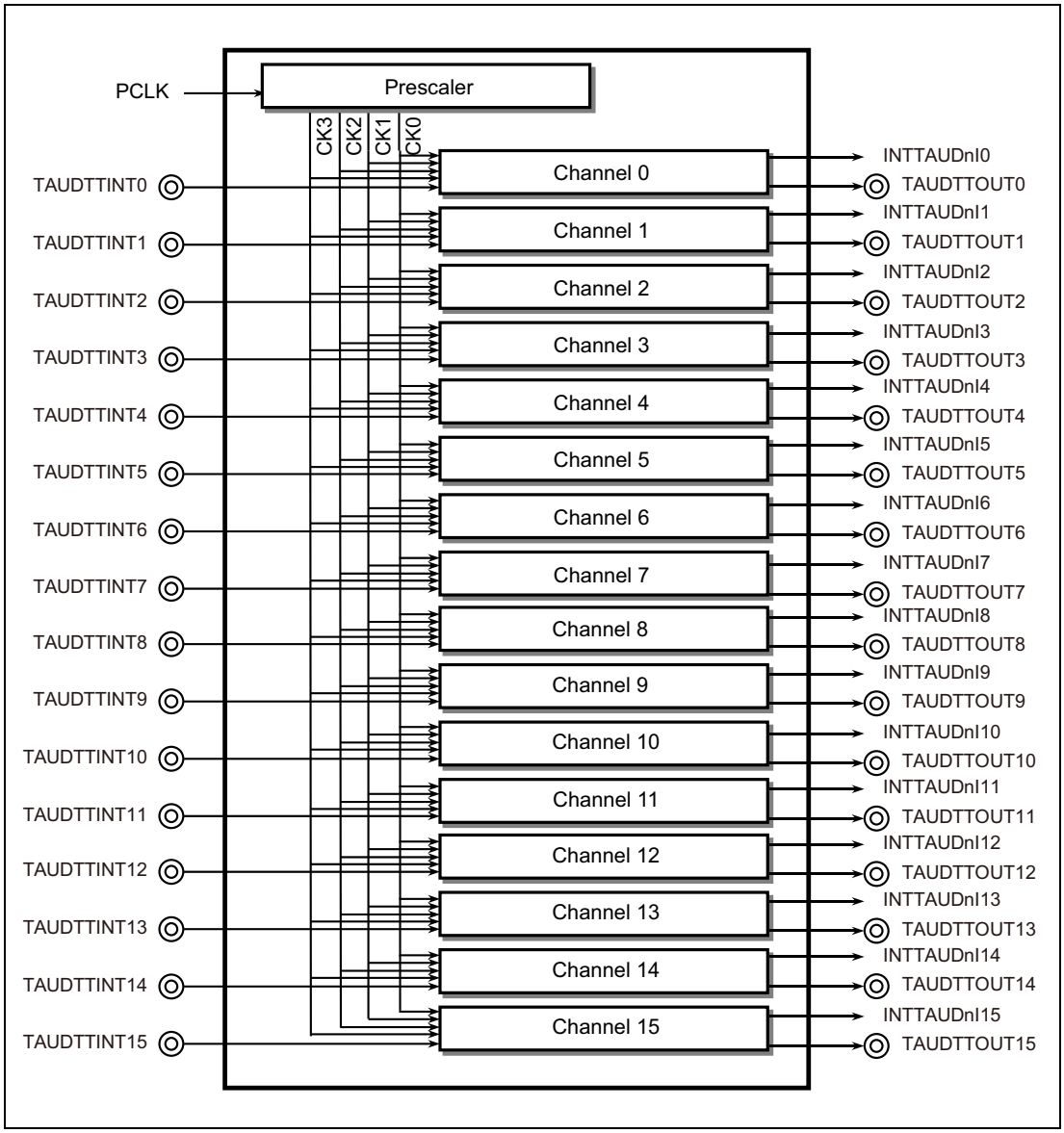


Figure 17.1 TAUD I/O and Interrupt Request Signals

17.2.5 Block Diagram

Figure 17.2 shows the main components of the TAUD:

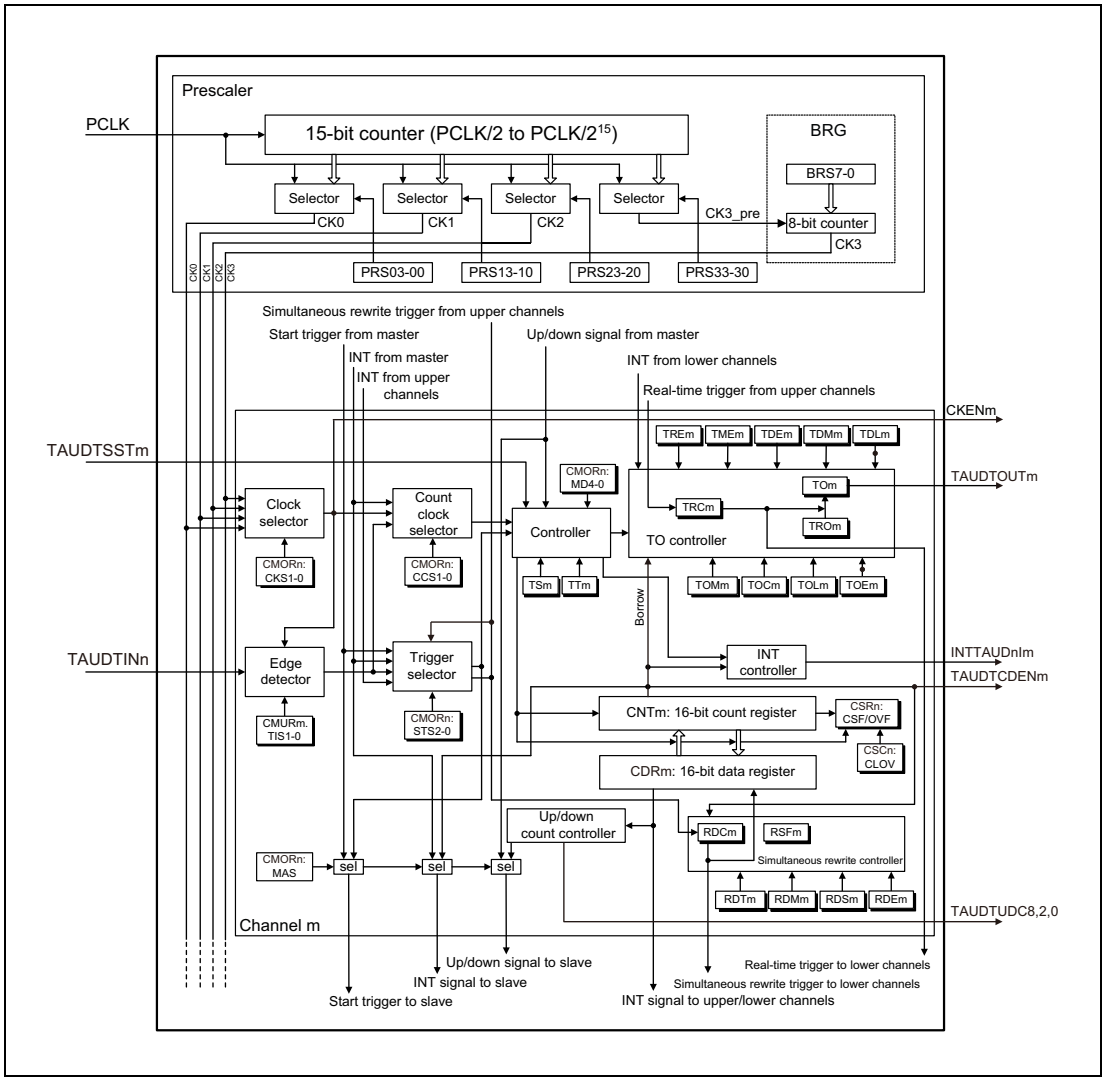


Figure 17.2 Block Diagram of the TAUD

The prefix “TAUDn” has been omitted from the register names for the sake of clarity in the above figure.

- TAUDTSSTm: Simultaneous start trigger (input from PIC1A)

17.2.6 Description of Blocks

The following describes the functional blocks:

Prescaler block

The prescaler block provides up to 4 clock signals (CK0 to CK3) that can be used as count clocks for all channels.

Count clocks CK0 to CK2 are derived from PCLK by a configurable prescaler division factor of 2^0 to 2^{15}. The fourth count clock CK3 can be adjusted more precisely by an additional division factor that is not a power of 2.

Clock and count clock selection

For every channel, the count clock selector selects which of the following is used as the clock source:

- One of the clocks CK0 to CK3 (selected by the clock selector)
- INTTAUDnIm from master channel
- TAUDTTINm input signal effective edge

Controller

The controller controls the main operations of the counter:

- Operation mode (selected by bits TAUDnCMORm.TAUDnMD[4:0])
- Counter start enable (TAUDnTS.TAUDnTSm) and counter stop (TAUDnTT.TAUDnTTm)
  When counter start is enabled, status flag TAUDnTE.TAUDnTEm is set.
- Count direction (up/down) (can be controlled by master channel)

Trigger selector

Depending on the selected operation mode, the counter starts automatically when it is enabled (TAUDnTE.TAUDnTEm = 1), or it waits for an external start trigger signal. Any of the following signals can be used as the start trigger:

- Synchronous channel start trigger input TAUDTSSTm
  For details on how to make a simultaneous start between the units, see **Section 23.2.3.1, Simultaneous Start Trigger Function**.
- TAUDTTINm input effective edge
- INTTAUDnIm from the master or any upper channel
- Up/down output trigger signal of the master channel
- Dead-time output signal of the TAUDTTOUTm generation unit.

Simultaneous rewrite controller

Simultaneous rewrite control is a special function that can be used in synchronous operation modes. The data registers (TAUDnCDRm) of all channels in a channel group can be rewritten at any time. The simultaneous rewrite controller ensures that new data register values of all channels become effective at the same time.

TAUDnTO controller

The output control of every channel enables the generation of various output signal forms such as PWM signals or triangular waves.

17.3 Registers

17.3.1 List of Registers

TAUD registers are listed in the following table.

See **Section 17.1.2** for <TAUDn_base>.

Table 17.10 TAUDn Registers Overview

Module name	Register name	Shortcut	Address
TAUDn prescaler registers			
TAUDn	TAUDn prescaler clock select register	TAUDnTPS	<TAUDn_base> + 240_H
TAUDn	TAUDn prescaler baud rate setting register	TAUDnBRS	<TAUDn_base> + 244_H
TAUDn control registers			
TAUDn	TAUDn channel data register m	TAUDnCDRm	<TAUDn_base> + m × 4_H
TAUDn	TAUDn channel counter register m	TAUDnCNTm	<TAUDn_base> + 80_H + m × 4_H
TAUDn	TAUDn channel mode OS register m	TAUDnCMORm	<TAUDn_base> + 200_H + m × 4_H
TAUDn	TAUDn channel mode user register m	TAUDnCMURm	<TAUDn_base> + $C0_H$ + m × 4_H
TAUDn	TAUDn channel status register m	TAUDnCSRm	<TAUDn_base> + 140_H + m × 4_H
TAUDn	TAUDn channel status clear register m	TAUDnCSCm	<TAUDn_base> + 180_H + m × 4_H
TAUDn	TAUDn channel start trigger register	TAUDnTS	<TAUDn_base> + $1C4_H$
TAUDn	TAUDn channel enable status register	TAUDnTE	<TAUDn_base> + $1C0_H$
TAUDn	TAUDn channel stop trigger register	TAUDnTT	<TAUDn_base> + $1C8_H$
TAUDn output registers			
TAUDn	TAUDn channel output enable register	TAUDnTOE	<TAUDn_base> + $5C_H$
TAUDn	TAUDn channel output register	TAUDnTO	<TAUDn_base> + 58_H
TAUDn	TAUDn channel output mode register	TAUDnTOM	<TAUDn_base> + 248_H
TAUDn	TAUDn channel output configuration register	TAUDnTOC	<TAUDn_base> + $24C_H$
TAUDn	TAUDn channel output active level register	TAUDnTOL	<TAUDn_base> + 40_H
TAUDn	TAUDn channel dead time output enable register	TAUDnTDE	<TAUDn_base> + 250_H
TAUDn	TAUDn channel dead time output mode register	TAUDnTDM	<TAUDn_base> + 254_H
TAUDn	TAUDn channel dead time output level register	TAUDnTDL	<TAUDn_base> + 54_H
TAUDn	TAUDn channel real-time output register	TAUDnTRO	<TAUDn_base> + $4C_H$
TAUDn	TAUDn channel real-time output enable register	TAUDnTRE	<TAUDn_base> + 258_H
TAUDn	TAUDn channel real-time output control register	TAUDnTRC	<TAUDn_base> + $25C_H$
TAUDn	TAUDn channel modulation output enable register	TAUDnTME	<TAUDn_base> + 50_H
TAUDn reload registers			
TAUDn	TAUDn channel reload data enable register	TAUDnRDE	<TAUDn_base> + 260_H
TAUDn	TAUDn channel reload data mode register	TAUDnRDM	<TAUDn_base> + 264_H
TAUDn	TAUDn channel reload data control CH select register	TAUDnRDS	<TAUDn_base> + 268_H
TAUDn	TAUDn channel reload data control register	TAUDnRDC	<TAUDn_base> + $26C_H$
TAUDn	TAUDn channel reload data trigger register	TAUDnRDT	<TAUDn_base> + 44_H
TAUDn	TAUDn channel reload status register	TAUDnRSF	<TAUDn_base> + 48_H

17.3.2 TAUDnTPS — TAUDn Prescaler Clock Select Register

This register specifies clocks CK0, CK1, CK2, and CK3_PRE for all channels of the PCLK prescaler. CK3 is generated by dividing CK3_PRE by the factor specified in TAUDnBRS.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 240_H

Value after reset: $FFFF_H$ This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnPRS3 [3:0]				TAUDnPRS2 [3:0]				TAUDnPRS1 [3:0]				TAUDnPRS0 [3:0]			
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.11 TAUDnTPS Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	TAUDnPRS3 [3:0]	These bits specify CK3_PRE clock. CK3_PRE clock is an input clock to BRG unit which supplies the CK3 operation clock to all channels.

TAUDnPRS3[3:0]	CK3_PRE Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK3 are stopped (TAUDnTE.TAUDnTEm = 0).

Table 17.11 TAUDnTPS Register Contents (2/3)

Bit Position	Bit Name	Function
11 to 8	TAUDnPRS2 [3:0]	These bits specify the CK2 clock.

TAUDnPRS2[3:0]	CK2 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK2 are stopped (TAUDnTE.TAUDnTEm = 0).

7 to 4	TAUDnPRS1 [3:0]	These bits specify the CK1 clock.

TAUDnPRS1[3:0]	CK1 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK1 are stopped (TAUDnTE.TAUDnTEm = 0).

Table 17.11 TAUDnTPS Register Contents (3/3)

Bit Position	Bit Name	Function
3 to 0	TAUDnPRS0 [3:0]	These bits specify the CK0 clock.

TAUDnPRS0[3:0]	CK0 Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK0 are stopped (TAUDnTE.TAUDnTEm = 0).

NOTE

The TAUDn clock input PCLK is specified in the first part of this section, **Section 17.1.3, Clock Supply**.

17.3.3 TAUDnBRS — TAUDn Prescaler Baud Rate Setting Register

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUDnTPS.TAUDnPRS3[3:0].

Access: Readable/writable in 8-bit units.

Address: <TAUDn_base> + 244_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	TAUDnBRS[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.12 TAUDnBRS Register Contents

Bit Position	Bit Name	Function
7 to 0	TAUDnBRS[7:0]	These bits specify a CK3_PRE clock division factor for generating CK3.

TAUDnBRS[7:0]	CK3 Clock
$0000\ 0000_B$	CK3_PRE / 1
$0000\ 0001_B$	CK3_PRE / 2
$0000\ 0010_B$	CK3_PRE / 3
$0000\ 0011_B$	CK3_PRE / 4
...	...
$1111\ 1110_B$	CK3_PRE / 255
$1111\ 1111_B$	CK3_PRE / 256

17.3.4 TAUDnCDRm — TAUDn Channel Data Register

This register functions either as a compare register or as a capture register, depending on the operating mode specified in TAUDnCMORm.TAUDnMD[4:1].

Access: Readable/writable in 16-bit units.
- Readable in capture mode. Any write operation is ignored.
- Readable/writable in compare mode.

Address: <TAUDn_base> + 0_H + m × 4_H

Value after reset: 0000_H This register is initialized by any reset source.

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

Table 17.13 TAUDnCDRm Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnCDR [15:0]	Data register for capture/compare values

17.3.5 TAUDnCNTm — TAUDn Channel Counter Register

This is a channel m counter register.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + 80_H + m × 4_H

Value after reset: $FFFF_H$ An value after reset depends on an operating mode. See **Table 17.15, TAUDnCNTm Read Values after Re-Enabling Counter**. This register is initialized by any reset source.

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

Table 17.14 TAUDnCNTm Register Contents

Bit Position	Bit Name	Function
15 to 0	TAUDnCNT [15:0]	16-bit counter value

A read value depends on a counter value, a changed operating mode, TAUDnTS.TAUDnTSm or TAUDnTT.TAUDnTTm bit value.

The initial read value of the counter depends on an operating mode and how the counter is stopped.

- Stop by a reset
- Stop by a counter stop trigger (TAUDnTT.TAUDnTTm = 1)

Table 17.15 lists the initial counter read values after the counter is stopped (TAUDnTE.TAUDnTEm = 0) and re-enabled (TAUDnTS.TAUDnTSm = 1).

The table also contains the counter read value one count after the counter is enabled (TAUDnTS.TAUDnTSm = 1) with the counter waiting for a start trigger.

Table 17.15 TAUDnCNTm Read Values after Re-Enabling Counter

Mode Name	Count Method (Up/Down)	TAUDnCNTm Value		
		After Reset	After Stop Trigger	After One Count
Interval timer mode	Count down	$FFFF_H$	Stop value	—
Judge mode	Count down	$FFFF_H$	Stop value	—
Capture mode	Count up	0000_H	Stop value	—
Event count mode	Count down	$FFFF_H$	Stop value	—
One-count mode	Count down	$FFFF_H$	Stop value	Stop value
Capture and one-count mode	Count up	0000_H	Stop value	Capture value + 1 (TAUDnCDRm)
Judge and one-count mode	Count down	$FFFF_H$	Stop value	TAUDnCNTm value – 1
Up/down count mode	Count down/up	$FFFF_H$	Stop value	—
Pulse one count mode	Count down	$FFFF_H$	Stop value	0000_H
Count capture mode	Count up	0000_H	Stop value	—
Capture and gate count mode	Count up	0000_H	Stop value	Stop value

NOTE

If the operating mode is changed while the counter is stopped, the initial counter value after a counter restart becomes undefined. The operating mode is changed by the TAUDnCMORm.TAUDnMD[4:1] register.

17.3.6 TAUDnCMORm — TAUDn Channel Mode OS Register

This register controls channel m operation.

Access: Readable/writable in 16-bit units. Writable when the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 200_H + m × 4_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.16 TAUDnCMORm Register Contents (1/3)

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select an operation clock. An operation clock is used for the TAUDTTINm input edge detection circuit. TAUDnCMORm.TAUDnCCS[1:0] bit setting enables use as a counter clock.

TAUDn CKS1	TAUDn CKS0	Selection of Operation Clock
0	0	CK0
0	1	CK1
1	0	CK2
1	1	CK3

Bit Position	Bit Name	Function
13, 12	TAUDnCCS [1:0]	These bits select a count clock for TAUDnCNTm counter.

TAUDn CCS1	TAUDn CCS0	Selection of Count Clock
0	0	Operation clock specified by TAUDnCMORm.TAUDnCKS[1:0]
0	1	Effective edge of TAUDTTINm input signal
1	0	Setting prohibited
1	1	INTTAUDnIm signal of master channel

Bit Position	Bit Name	Function
11	TAUDnMAS	This bit specifies whether the channel is a master channel or slave channel during synchronous channel operation. 0: Slave 1: Master This bit setting is valid only for even-numbered channels (CHm_even). Odd-numbered channels (CHm_odd) are fixed to 0.

Table 17.16 TAUDnCMORm Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>10 to 8</td><td>TAUDnSTS[2:0]</td><td>These bits select an external start trigger.
<table>
<tr><th>TAUDn STS2</th><th>TAUDn STS1</th><th>TAUDn STS0</th><th>Functional description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Software trigger</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Effective edge of TAUDTTINm input signal, which is specified by TAUDnCMURm.TAUDnTIS[1:0].</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Effective edge of TAUDTTINm input signal is used as a start trigger and the opposite edge as a stop trigger.</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Triggers simultaneous rewrite.</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>INTTAUDnIm is the start trigger of master channel</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>INTTAUDnIm of upper channel (m – 1) is the start trigger regardless of master setting</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>Dead time output signal of TAUDTTOUTm generating unit</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>Up/down output trigger signal of master channel</td></tr>
</table></td></tr>
<tr><td>7, 6</td><td>TAUDnCOS [1:0]</td><td>These bits specify the timing for updating capture register TAUDnCDRm and overflow flag TAUDnCSRm.TAUDnOVF of channel m.
These bits are valid only when channel m is in capture mode.
<table>
<tr><th>TAUDn COS1</th><th>TAUDn COS0</th><th>TAUDnCDRm</th><th>TAUDnCSRm.TAUDnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated upon detection of effective edge of TAUDTTINm input.</td><td>Updated (cleared or set) by detecting effective edge of TAUDTTINm input:
• If a counter overflow has occurred since the last detection of effective edge, set TAUDnCSRm.TAUDnOVF.
• If no counter overflow has occurred since the last detection of effective edge, clear TAUDnCSRm.TAUDnOVF.</td></tr>
<tr><td>0</td><td>1</td><td>Set when a counter overflow occurs, and cleared when TAUDnCSCm.TAUDnCLOV is set to 1.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated upon detection of effective edge of TAUDTTINm input and at the occurrence of counter overflow:
• Detection of an effective TAUDTTINm input edge: Counter value is written into TAUDnCDRm.
• Occurrence of overflow: $FFFF_H$ is loaded into TAUDnCDRm. The next detection of an effective TAUDTTINm input edge is ignored.</td><td>Not set</td></tr>
<tr><td>1</td><td>1</td><td>Set when a counter overflow occurs, and cleared when TAUDnCSCm.TAUDnCLOV is set to 1.</td></tr>
</table></td></tr>
<tr><td>5</td><td>Reserved</td><td>When read, the value after reset is read. When writing, write the value after reset.</td></tr>
</table>

Table 17.16 TAUDnCMORm Register Contents (3/3)

Bit Position	Bit Name	Function
4 to 0	TAUDnMD[4:0]	These bits specify an operating mode.

TAUDnMD4	TAUDnMD3	TAUDnMD2	TAUDnMD1	TAUDnMD0	Functional Description
0	0	0	0	1/0	Interval timer mode
0	0	0	1	1/0	Judge mode
0	0	1	0	1/0	Capture mode
0	0	1	1	0	Event count mode
0	1	0	0	1/0	One-count mode
0	1	1	0	0	Capture and one-count mode
0	1	1	1	1/0	Judge and one-count mode
1	0	0	1	0	Up/down count mode
1	0	1	0	1/0	Pulse one-count mode
1	0	1	1	1/0	Count capture mode
1	1	0	1	0	Capture and gate count mode

Settings other than the above are prohibited.

Mode	Role of TAUDnMD0 Bit
Interval timer mode Capture mode Count capture mode	Specifies whether INTTAUDnIm is generated at the beginning of count operation (when a start trigger is entered) or not. 0: INTTAUDnIm is not generated. 1: INTTAUDnIm is generated.
Event count mode Up/down count mode	This bit should be set to 0 (INTTAUDnIm signal is not output at the beginning of count operation).
One-count mode Pulse one-count mode	Enables/disables start trigger detection during counting. 0: Disables detection. 1: Enables detection.
Capture and one-count mode Capture and gate count mode	This bit should be set to 0. **CAUTION** INTTAUDnIm signal is not output at the beginning of count operation. In addition, start trigger detected during counting is disabled.
Judge mode Judge and one-count mode	Specifies INTTAUDnIm output timing. 0: When TAUDnCNTm ≤ TAUDnCDRm 1: When TAUDnCNTm > TAUDnCDRm

17.3.7 TAUDnCMURm — TAUDn Channel Mode User Register

This register specifies a type of effective edge detection used for TAUDTTINm input.

Access: Readable/writable in 8-bit units.

Address: <TAUDn_base> + $C0_H$ + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.17 TAUDnCMURm Register Contents

Bit position	Bit name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	These bits specify an effective edge of TAUDTTINm input signal.

TAUDn TIS1	TAUDn TIS0	Functional Description
0	0	Falling edge
0	1	Rising edge
1	0	Detection of rising and falling edges (selects low width measurement) Start trigger: Falling edge Stop trigger (capture): Rising edge
1	1	Detection of rising and falling edges (selects high width measurement) Start trigger: Rising edge Stop trigger (capture): Falling edge

- Edge detection of TAUDTTINm input signal is based on the operation clock selected by TAUDnCMORm.TAUDnCKS[1:0].

17.3.8 TAUDnCSRm — TAUDn channel status register

This register indicates the count direction and overflow status of channel m counter.

Access: Readable in 8-bit units.

Address: <TAUDn_base> + 140_H + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnCSF	TAUDnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 17.18 TAUDnCSRm Register Contents

Bit position	Bit name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	TAUDnCSF	Indicates a count direction. 0: Count-up 1: Count-down The read value of this bit is valid only in the following mode: • Up/down count mode
0	TAUDnOVF	Indicates counter overflow status. 0: No overflow occurs. 1: Overflow occurs. This bit is used only in the following modes: • Capture mode • Capture and one-count mode The function of this bit depends on the setting of control bit TAUDnCMORm.TAUDnCOS[1:0].

17.3.9 TAUDnCSCm — TAUDn Channel Status Clear Register

This is a trigger register for clearing the overflow flag TAUDnCSRm.TAUDnOVF of channel m.

Access: Writable in 8-bit units. This value is always read as 00_H.

Address: <TAUDn_base> + 180_H + m × 4_H

Value after reset: 00_H This register is initialized by any reset source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUDnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 17.19 TAUDnCSCm Register Contents

Bit position	Bit name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAUDnCLOV	0: No function 1: Clears overflow flag TAUDnCSRm.TAUDnOVF.

17.3.10 TAUDnTS — TAUDn Channel Start Trigger Register

This register enables the counter operation of each channel.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + $1C4_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTS15	TAUDnTS14	TAUDnTS13	TAUDnTS12	TAUDnTS11	TAUDnTS10	TAUDnTS09	TAUDnTS08	TAUDnTS07	TAUDnTS06	TAUDnTS05	TAUDnTS04	TAUDnTS03	TAUDnTS02	TAUDnTS01	TAUDnTS00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 17.20 TAUDnTS Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTSm	These bits enable the counter operation of channel m. 0: No function 1: Enables the counter operation and sets TAUDnTE.TAUDnTEm to 1. The counter operation is only enabled when TAUDnTE.TAUDnTEm is set to 1. Whether counting is started or not depends on a selected operating mode.

17.3.11 TAUDnTE — TAUDn Channel Enable Status Register

This register enables/disables a counter operation.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + $1C0_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTE15	TAUDnTE14	TAUDnTE13	TAUDnTE12	TAUDnTE11	TAUDnTE10	TAUDnTE09	TAUDnTE08	TAUDnTE07	TAUDnTE06	TAUDnTE05	TAUDnTE04	TAUDnTE03	TAUDnTE02	TAUDnTE01	TAUDnTE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.21 TAUDnTE Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTEm	These bits enable or disable the counter operation of channel m. 0: Disables counter operation. 1: Enables counter operation. This bit is set to 1 when trigger input of TAUDTSSTm (synchronous channel start trigger signal) is detected or when TAUDnTS.TAUDnTSm is set to 1. This bit is set to 0 when TAUDnTT.TAUDnTTm is set to 1.

NOTE

For details on how to make a simultaneous start between the units, see **Section 23.2.3.1, Simultaneous Start Trigger Function**.

17.3.12 TAUDnTT — TAUDn Channel Stop Trigger Register

This register stops the counter operation of each channel.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + $1C8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTT15	TAUDnTT14	TAUDnTT13	TAUDnTT12	TAUDnTT11	TAUDnTT10	TAUDnTT09	TAUDnTT08	TAUDnTT07	TAUDnTT06	TAUDnTT05	TAUDnTT04	TAUDnTT03	TAUDnTT02	TAUDnTT01	TAUDnTT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 17.22 TAUDnTT Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTTm	These bits are used to stop the counter operation of channel m. 0: No function 1: Stops the counter operation and resets TAUDnTE.TAUDnTEm. TAUDnCNTm, TAUDnTO.TAUDnTOm, and TAUDTTOUTm retain the values provided before the counter is stopped.

17.3.13 TAUDnRDE — TAUDn Channel Reload Data Enable Register

This register enables/disables simultaneous rewrite of TAUDnCDRm/TAUDnTOLm data register.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 260_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnRDE15	TAUDnRDE14	TAUDnRDE13	TAUDnRDE12	TAUDnRDE11	TAUDnRDE10	TAUDnRDE09	TAUDnRDE08	TAUDnRDE07	TAUDnRDE06	TAUDnRDE05	TAUDnRDE04	TAUDnRDE03	TAUDnRDE02	TAUDnRDE01	TAUDnRDE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.23 TAUDnRDE Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRDEm	These bits enable or disable simultaneous rewrite of the data register of channel m. 0: Disables simultaneous rewrite 1: Enables simultaneous rewrite

17.3.14 TAUDnRDS — TAUDn Channel Reload Data Control Channel Select Register

This register selects a channel that controls simultaneous rewrite.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 268_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnRDS15	TAUDnRDS14	TAUDnRDS13	TAUDnRDS12	TAUDnRDS11	TAUDnRDS10	TAUDnRDS09	TAUDnRDS08	TAUDnRDS07	TAUDnRDS06	TAUDnRDS05	TAUDnRDS04	TAUDnRDS03	TAUDnRDS02	TAUDnRDS01	TAUDnRDS00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.24 TAUDnRDS Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRDSm	These bits select a channel that controls a simultaneous rewrite trigger. 0: Master channel 1: Another upper channel

17.3.15 TAUDnRDM — TAUDn Channel Reload Data Mode Register

This register selects the timing for generating a simultaneous rewrite control signal.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 264_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDM15	TAUDn RDM14	TAUDn RDM13	TAUDn RDM12	TAUDn RDM11	TAUDn RDM10	TAUDn RDM09	TAUDn RDM08	TAUDn RDM07	TAUDn RDM06	TAUDn RDM05	TAUDn RDM04	TAUDn RDM03	TAUDn RDM02	TAUDn RDM01	TAUDn RDM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.25 TAUDnRDM Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRDMm	These bits select the timing for generating a simultaneous rewrite trigger signal. 0: When the master channel counter starts to count 1: At the peak of cycle of triangular wave These bit settings are applied only when TAUDnRDE.TAUDnRDEm = 1 and TAUDnRDS.TAUDnRDSm = 0.

17.3.16 TAUDnRDC — TAUDn Channel Reload Data Control Register

This register specifies a channel which generates an INTTAUDnlm signal to trigger simultaneous rewrite.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + $26C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDC15	TAUDn RDC14	TAUDn RDC13	TAUDn RDC12	TAUDn RDC11	TAUDn RDC10	TAUDn RDC09	TAUDn RDC08	TAUDn RDC07	TAUDn RDC06	TAUDn RDC05	TAUDn RDC04	TAUDn RDC03	TAUDn RDC02	TAUDn RDC01	TAUDn RDC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.26 TAUDnRDC Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRDCm	These bits specify whether the channel generates a simultaneous rewrite trigger signal or not. 0: Not operate as a simultaneous rewrite trigger channel. 1: Operates as a simultaneous rewrite trigger channel. These bit settings are applied only when TAUDnRDE.TAUDnRDEm = 1 and TAUDnRDS.TAUDnRDSm = 1.

17.3.17 TAUDnRDT — TAUDn Channel Reload Data Trigger Register

This register triggers a simultaneous rewrite enabling state.

Access: Writable in 16-bit units. This value is always read as 0000_H.

Address: <TAUDn_base> + 44_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RDT15	TAUDn RDT14	TAUDn RDT13	TAUDn RDT12	TAUDn RDT11	TAUDn RDT10	TAUDn RDT09	TAUDn RDT08	TAUDn RDT07	TAUDn RDT06	TAUDn RDT05	TAUDn RDT04	TAUDn RDT03	TAUDn RDT02	TAUDn RDT01	TAUDn RDT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 17.27 TAUDnRDT Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRDTm	These bits are used to trigger a simultaneous rewrite enabling state. 0: No function 1: Triggers a simultaneous rewrite enabling state. The simultaneous rewrite pending flag (TAUDnRSFm) is set to 1. The system waits for a simultaneous rewrite trigger.

17.3.18 TAUDnRSF — TAUDn Channel Reload Status Register

This flag register indicates simultaneous rewrite status.

Access: Readable in 16-bit units.

Address: <TAUDn_base> + 48_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn RSF15	TAUDn RSF14	TAUDn RSF13	TAUDn RSF12	TAUDn RSF11	TAUDn RSF10	TAUDn RSF09	TAUDn RSF08	TAUDn RSF07	TAUDn RSF06	TAUDn RSF05	TAUDn RSF04	TAUDn RSF03	TAUDn RSF02	TAUDn RSF01	TAUDn RDT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.28 TAUDnRSF Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnRSFm	These bits indicate simultaneous rewrite status. 0: Indicates that a simultaneous rewrite trigger has started simultaneous rewrite. 1: Indicates that simultaneous rewrite is in the enabling state (TAUDnRDTm = 1) and that the system waits for a simultaneous rewrite trigger.

17.3.19 TAUDnTOE — TAUDn Channel Output Enable Register

This register enables/disables the independent channel output mode controlled by software.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + $5C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOE15	TAUDnTOE14	TAUDnTOE13	TAUDnTOE12	TAUDnTOE11	TAUDnTOE10	TAUDnTOE09	TAUDnTOE08	TAUDnTOE07	TAUDnTOE06	TAUDnTOE05	TAUDnTOE04	TAUDnTOE03	TAUDnTOE02	TAUDnTOE01	TAUDnTOE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.29 TAUDnTOE Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTOEm	These bits enable or disable the independent channel output function. 0: Disables the independent timer output function. 1: Enables the independent timer output function.

17.3.20 TAUDnTO — TAUDn Channel Output Register

This register specifies and reads a TAUDTTOUTm level.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 58_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTO15	TAUDnTO14	TAUDnTO13	TAUDnTO12	TAUDnTO11	TAUDnTO10	TAUDnTO09	TAUDnTO08	TAUDnTO07	TAUDnTO06	TAUDnTO05	TAUDnTO04	TAUDnTO03	TAUDnTO02	TAUDnTO01	TAUDnTO00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.30 TAUDnTO Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTOm	These bits specify and read a TAUDTTOUTm level. 0: Low level 1: High level Only TAUDnTOm bits for which Independent Channel Output function is disabled (TAUDnTOEm = 0) can be written.

17.3.21 TAUDnTOM — TAUDn Channel Output Mode Register

This register specifies the output mode of each channel.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 248_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TOM15	TAUDn TOM14	TAUDn TOM13	TAUDn TOM12	TAUDn TOM11	TAUDn TOM10	TAUDn TOM09	TAUDn TOM08	TAUDn TOM07	TAUDn TOM06	TAUDn TOM05	TAUDn TOM04	TAUDn TOM03	TAUDn TOM02	TAUDn TOM01	TAUDn TOM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.31 TAUDnTOM Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTOMm	These bits specify an output mode. 0: Independent channel operation 1: Synchronous channel operation As described in **Section 17.4.4, Channel Output Modes**, the output mode depends on the setting of each channel output control bit.

17.3.22 TAUDnTOC — TAUDn Channel Output Configuration Register

This register specifies the output mode of each channel in combination with TAUDnTOMm.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + $24C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOC15	TAUDnTOC14	TAUDnTOC13	TAUDnTOC12	TAUDnTOC11	TAUDnTOC10	TAUDnTOC09	TAUDnTOC08	TAUDnTOC07	TAUDnTOC06	TAUDnTOC05	TAUDnTOC04	TAUDnTOC03	TAUDnTOC02	TAUDnTOC01	TAUDnTOC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.32 TAUDnTOC Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTOCm	These bits specify an output mode. 0: Operating mode 1 1: Operating mode 2 As listed below, the output mode depends on the setting of TAUDnTOM.TAUDnTOMm.

TOMm	TOCm	Functional Description
0	0	Toggle mode: Toggle operation is conducted when INTTAUDnIm occurs.
0	1	Set/reset mode: Set when INTTAUDnIm occurs at the beginning of count operation, and reset when INTTAUDnIm is caused by detection of a match between TAUDnCNTm and TAUDnCDRm.
1	0	Synchronous channel operating mode 1: Set when INT occurs on master channels, and reset when INT occurs on slave channels.
1	1	Synchronous channel operating mode 2: Set when INTTAUDnIm occurs in count-down status, and reset when INTTAUDnIm occurs in count-up status.

17.3.23 TAUDnTOL — TAUDn Channel Output Active Level Register

This register specifies the output logic of channel output bit (TAUDnTO.TAUDnTOm).

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 040_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTOL15	TAUDnTOL14	TAUDnTOL13	TAUDnTOL12	TAUDnTOL11	TAUDnTOL10	TAUDnTOL09	TAUDnTOL08	TAUDnTOL07	TAUDnTOL06	TAUDnTOL05	TAUDnTOL04	TAUDnTOL03	TAUDnTOL02	TAUDnTOL01	TAUDnTOL00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.33 TAUDnTOL Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTOLm	Specifies the output logic of channel m output bit (TAUDnTO.TAUDnTOm). 0: Positive logic (active high) 1: Negative logic (active low) These bits apply in all channel output modes except independent channel output mode controlled by software and independent channel output mode 1.

17.3.24 TAUDnTDE — TAUDn Channel Dead Time Output Enable Register

This register enables/disables the dead time operation of every channel.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 250_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTDE15	TAUDnTDE14	TAUDnTDE13	TAUDnTDE12	TAUDnTDE11	TAUDnTDE10	TAUDnTDE09	TAUDnTDE08	TAUDnTDE07	TAUDnTDE06	TAUDnTDE05	TAUDnTDE04	TAUDnTDE03	TAUDnTDE02	TAUDnTDE01	TAUDnTDE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.34 TAUDnTDE Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTDEm	These bits enable or disable the dead time control operation of channel m. 0: Disables dead time operation 1: Enables dead time operation. The same setting should be made for both even and odd slave channels in pairs. These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm = 1

17.3.25 TAUDnTDM — TAUDn channel dead time output mode register

This register specifies the timing to add dead time during dead time output.

Access: Readable/writable in 16-bit units. Writable only while the counter is stopped (TAUDnTE.TAUDnTEm = 0).

Address: <TAUDn_base> + 254_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTDM15	TAUDnTDM14	TAUDnTDM13	TAUDnTDM12	TAUDnTDM11	TAUDnTDM10	TAUDnTDM09	TAUDnTDM08	TAUDnTDM07	TAUDnTDM06	TAUDnTDM05	TAUDnTDM04	TAUDnTDM03	TAUDnTDM02	TAUDnTDM01	TAUDnTDM00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.35 TAUDnTDM Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTDMm	These bits specify the timing to add dead time during dead time output. 0: When detecting the duty cycle of an upper even-numbered channel (duty dead time output). 1: When detecting the TIN input edge of a lower odd-numbered channel (one-phase dead time output). The same setting should be made for both even and odd slave channels in pairs. These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm, TAUDnTDE.TAUDnTDEm = 1

17.3.26 TAUDnTDL — TAUDn Channel Dead Time Output Level Register

This register selects a phase in which dead time is added.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 54_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TDL15	TAUDn TDL14	TAUDn TDL13	TAUDn TDL12	TAUDn TDL11	TAUDn TDL10	TAUDn TDL09	TAUDn TDL08	TAUDn TDL07	TAUDn TDL06	TAUDn TDL05	TAUDn TDL04	TAUDn TDL03	TAUDn TDL02	TAUDn TDL01	TAUDn TDL00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.36 TAUDnTDL Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTDLm	These bits select a phase in which dead time is added. 0: Normal phase 1: Reverse phase These bit settings are applied when: • TAUDnTOE.TAUDnTOEm, TAUDnTOM.TAUDnTOMm, TAUDnTOC.TAUDnTOCm, TAUDnTDE.TAUDnTDEm = 1

17.3.27 TAUDnTRE — TAUDn Channel Real-time Output Enable Register

This register enables/disables real-time output.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + 258_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TRE15	TAUDn TRE14	TAUDn TRE13	TAUDn TRE12	TAUDn TRE11	TAUDn TRE10	TAUDn TRE09	TAUDn TRE08	TAUDn TRE07	TAUDn TRE06	TAUDn TRE05	TAUDn TRE04	TAUDn TRE03	TAUDn TRE02	TAUDn TRE01	TAUDn TRE00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.37 TAUDnTRE Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTREm	These bits enable or disable real-time output of channel m. 0: Disables real-time output 1: Enables real-time output. These bit settings are applied only when TAUDnTOE.TAUDnTOEm = 1.These bit settings are applied only when TAUTnTOE.TAUDnTOEm = 1. When TAUDnTRE.TREm = 0, TAUDTTOUTm is not affected by real-time output. When TAUDnTRE.TREm = 1, TAUDTTOUTm outputs the value of real-time output bit TAUDnTRO.TAUDnTROm in response to a timer operation.

17.3.28 TAUDnTRC — TAUDn Channel Real-time Output Control Register

This register controls the real-time output trigger of each channel.

Access: Readable/writable in 16-bit units. Writable only while TAUDnTE.TAUDnTEm = 0.

Address: <TAUDn_base> + $25C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTRC15	TAUDnTRC14	TAUDnTRC13	TAUDnTRC12	TAUDnTRC11	TAUDnTRC10	TAUDnTRC09	TAUDnTRC08	TAUDnTRC07	TAUDnTRC06	TAUDnTRC05	TAUDnTRC04	TAUDnTRC03	TAUDnTRC02	TAUDnTRC01	TAUDnTRC00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.38 TAUDnTRC Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTRCm	These bits specify a channel on which the real-time output trigger for channel m is generated. 0: Next upper channel with this bit set to 1 1: Channel m These bit settings are applied only when TAUTnTRE.TAUDnTREm = 1.

17.3.29 TAUDnTRO — TAUDn Channel Real-time Output Register

This register sets a value which is output to TAUDTTOUTm.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + $4C_H$

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnTRO15	TAUDnTRO14	TAUDnTRO13	TAUDnTRO12	TAUDnTRO11	TAUDnTRO10	TAUDnTRO09	TAUDnTRO08	TAUDnTRO07	TAUDnTRO06	TAUDnTRO05	TAUDnTRO04	TAUDnTRO03	TAUDnTRO02	TAUDnTRO01	TAUDnTRO00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.39 TAUDnTRO Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTROm	These bits set a value which is output to TAUDTTOUTm. 0: Low level 1: High level TAUDnTROm value is not output to TAUDTTOUTm when TAUDnTRE.TAUDnTREm = 0, even if a real-time output trigger occurs.

17.3.30 TAUDnTME — TAUDn Channel Modulation Output Enable Register

This register enables/disables modulation output for timer output and real-time output.

Access: Readable/writable in 16-bit units.

Address: <TAUDn_base> + 50_H

Value after reset: 0000_H This register is initialized by any reset source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDn TME15	TAUDn TME14	TAUDn TME13	TAUDn TME12	TAUDn TME11	TAUDn TME10	TAUDn TME09	TAUDn TME08	TAUDn TME07	TAUDn TME06	TAUDn TME05	TAUDn TME04	TAUDn TME03	TAUDn TME02	TAUDn TME01	TAUDn TME00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.40 TAUDnTME Register Contents

Bit position	Bit name	Function
15 to 0	TAUDnTMEm	These bits enable or disable modulation output for timer output and real-time output of channel m. 0: Disables modulation 1: Enables modulation These bit settings are applied only when TAUDnTOE.TAUDnTOEm and TAUDnTRE.TAUDnTREm = 1.

17.4 Function

17.4.1 General Operating Procedure

The following lists the general operation procedure for the TAUDn:

After reset release, the operation of each channel is stopped. Clock supply is started and writing to each register is enabled. All circuits and registers of all channels are initialized. The control register of TAUDTTOUTm is also initialized and outputs a low level.

(1) Set the TAUDnTPS and TAUDnBRS registers to specify the clock frequency of CK0 to CK3.

(2) Configure the desired TAUDn function:
- Set the operation mode
- Set the channel output mode
- Set any other control bits

(3) Enable the counter by setting the TAUDnTS.TAUDnTSm bit to 1.
The counter starts to count immediately, or when an appropriate trigger is detected, depending on the bit settings.

(4) Stop the counter or perform a forced restart operation during count operation. The counter can be stopped by setting the TAUDnTT.TAUDnTTm bit to 1. The counter can be forcibly restarted by setting the TAUDnTS.TAUDnTSm bit to 1.

(5) Stop the function by setting the TAUDnTT.TAUDnTTm bit to 1.

NOTE

A detailed description of the required control bits and the operation of the individual functions is given below:

- **Section 17.4.9, Independent Channel Operation Functions**
- **Section 17.4.12, Synchronous Channel Operation Functions**

17.4.2 Concepts of Synchronous Channel Operation

The synchronous channel operation function is implemented using a combination of channel groups (consisted of master and slave channels). Several rules apply to the settings of channels. These rules are detailed in **Section 17.4.2.1, Rules of Synchronous Channel Operation**.

Two special features for synchronous channel operation are detailed in the following subchapters:

- **Section 17.4.2.2, Simultaneous Start and Stop of Synchronous Channel Counters**
- **Section 17.4.3, Simultaneous Rewrite**

17.4.2.1 Rules of Synchronous Channel Operation

Number of masters and slaves

- Only even-numbered channels (CH0, CH2, CH4, …) can be set as master channels. Any channel apart from CH0 can be set as a slave channel.
- Only channels lower than the master channel can be set as slave channels, and several slave channels can be set for one master channel.
  Example: If CH2 is a master channel, CH3 and the lower channels (CH4, CH5, …) can be set as slave channels.
- If multiple master channels are used, slave channels cannot cross the master channels.
  Example: If CH0 and CH4 are master channels, CH1 to CH3 can be set as slave channels for CH0, but CH5 to CH15 cannot.

Operation clock

- The same operation clock must be set for the slave channel and the master channel. This is achieved using the TAUDnCMORm.TAUDnCKS[1:0] bits of the slave and master channel.

The basic concepts of master/slave usage and operation clocks are illustrated in **Figure 17.3**.

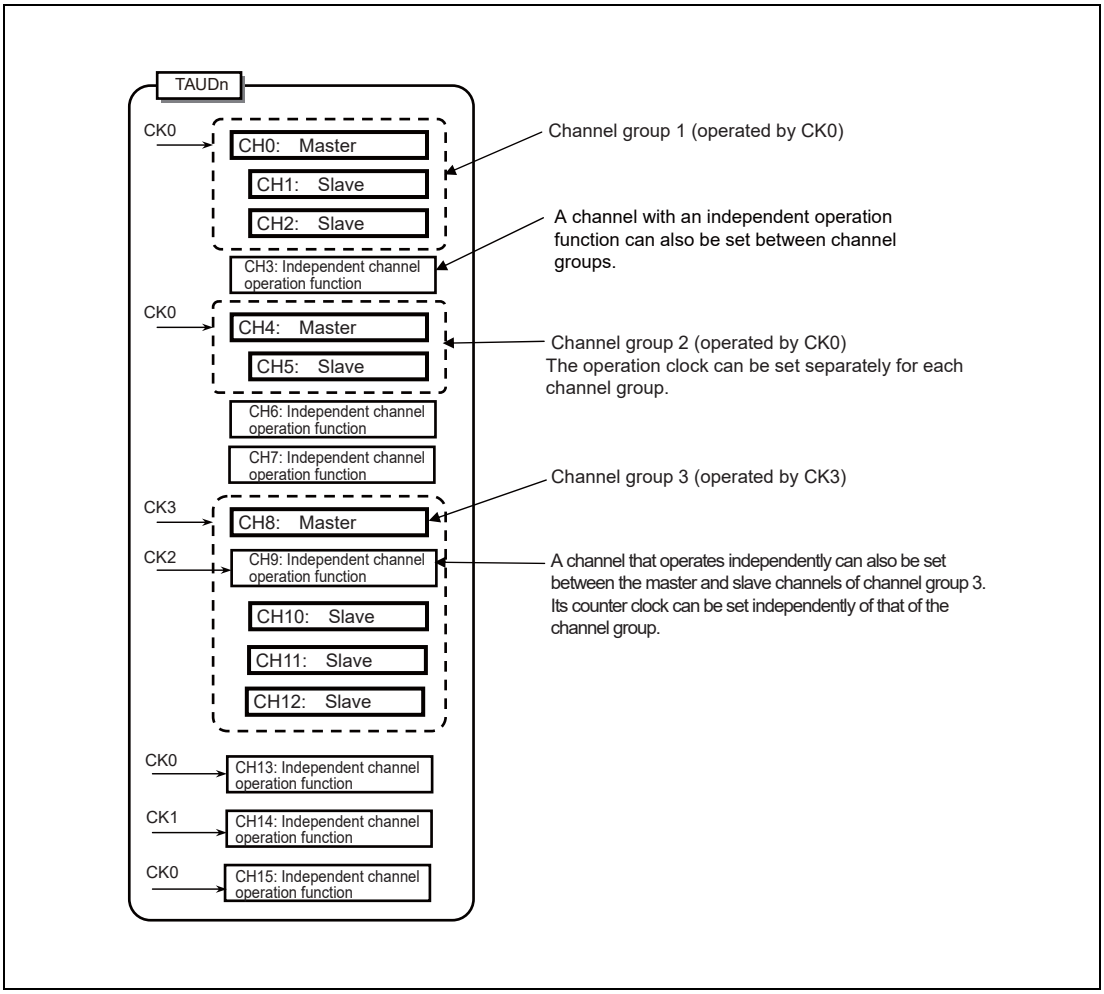


Figure 17.3 Grouping of Channels and Assignment of Count Clocks

Control trigger signal for master/salve channels

- Master channels can output control trigger signals to slave channels.
- Slave channels can use control trigger signals from master channels but cannot output control trigger signals for their own to lower channels.
- Master channels cannot use control trigger signals from upper master channels.

17.4.2.2 Simultaneous Start and Stop of Synchronous Channel Counters

Channels that are operated synchronously can be started and stopped simultaneously within the same unit and between the units.

(1) Simultaneous Start and Stop within the Same Unit

- To simultaneously start synchronized channels, the TAUDnTS.TAUDnTSm bits of the channels should be set at the same time.
- To simultaneously stop synchronized channels, the TAUDnTT.TAUDnTTm bits of the channels should be set at the same time.

Setting to the TAUDnTS.TAUDnTSm bits to 1 also sets the corresponding TAUDnTE.TAUDnTEm bits to 1, enabling counting. The count start timing depends on operating mode.

(2) Simultaneous Start between the Units

Counters in different units can also be started simultaneously if the corresponding counters are enabled before receiving the simultaneous trigger signal.

For details on how to make a simultaneous start between the units, see **Section 23.2.3.1, Simultaneous Start Trigger Function**.

17.4.3 Simultaneous Rewrite

17.4.3.1 Overview of Operations

Simultaneous rewrite describes the ability to change the compare/start value and the output logic of multiple channels at the same time.

The corresponding data and control registers (TAUDnCDRm and TAUDnTOLm) can nevertheless be written at any time. The new value does not affect the counter operation or the output signal until simultaneous rewrite is triggered.

Simultaneous rewrite can be triggered by:

- The counter on the master channel or upper channel (depending on the selected operation mode) reaching a certain value
- INTTAUDnIm being issued on the upper channel specified by TAUDnRDC.TAUDnRDCm

There are four methods for simultaneous rewrite. These are listed in **Table 17.41**, along with how to specify them and when they cause simultaneous rewrite to be triggered.

Table 17.41 Simultaneous Rewrite Methods and when They are Triggered

Method	Simultaneous Rewrite Triggered when	TAUDnRDE. TAUDnRDEm	TAUDnRDS. TAUDnRDSm	TAUDnRDM. TAUDnRDMm
—	No simultaneous rewrite	0	0	0
A	The master channel (re)starts counting	1	0	0
B	Counting is started in the master channel. The master channel starts counting down at the peak of triangular cycle of the corresponding slave channel.	1	0	1
C1	INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm	1	1	0/1
C2	INTTAUDnIm is generated on an upper channel specified by TAUDnRDC.TAUDnRDCm that in turn is triggered by an external signal	1	1	0/1

Table 17.42 lists which of these four methods is available for each channel operation function. For more information about the individual channel operation functions, see the corresponding sections in **Section 17.4.9, Independent Channel Operation Functions** and **Section 17.4.12, Synchronous Channel Operation Functions**.

Table 17.42 Channel Operation Functions and Methods They Use

Function	A	B	C1	C2
Simultaneous Rewrite Trigger Output Function Type 1			√	
PWM Output Function	√		√	
One-Shot Pulse Output Function	√			
Trigger Start PWM Output Function	√			√
Delay Pulse Output Function	√			
Triangle PWM Output Function		√	√	
Triangle PWM Output Function with Dead Time		√	√	
Interrupt Request Signals Culling Function	√	√	√	
AD Conversion Trigger Output Function Type 1	√		√	
AD Conversion Trigger Output Function Type 2		√	√	
Non-Complementary Modulation Output Function Type 1	√		√	
Non-Complementary Modulation Output Function Type 2		√	√	
Complementary Modulation Output Function		√	√	

Note: √: Available, (Blank): Unavailable

17.4.3.2 How to Control Simultaneous Rewrite

Figure 17.4 shows the general procedure for simultaneous rewrite. The three main blocks (initial settings, start and counter count operation, and simultaneous rewrite) are explained afterwards.

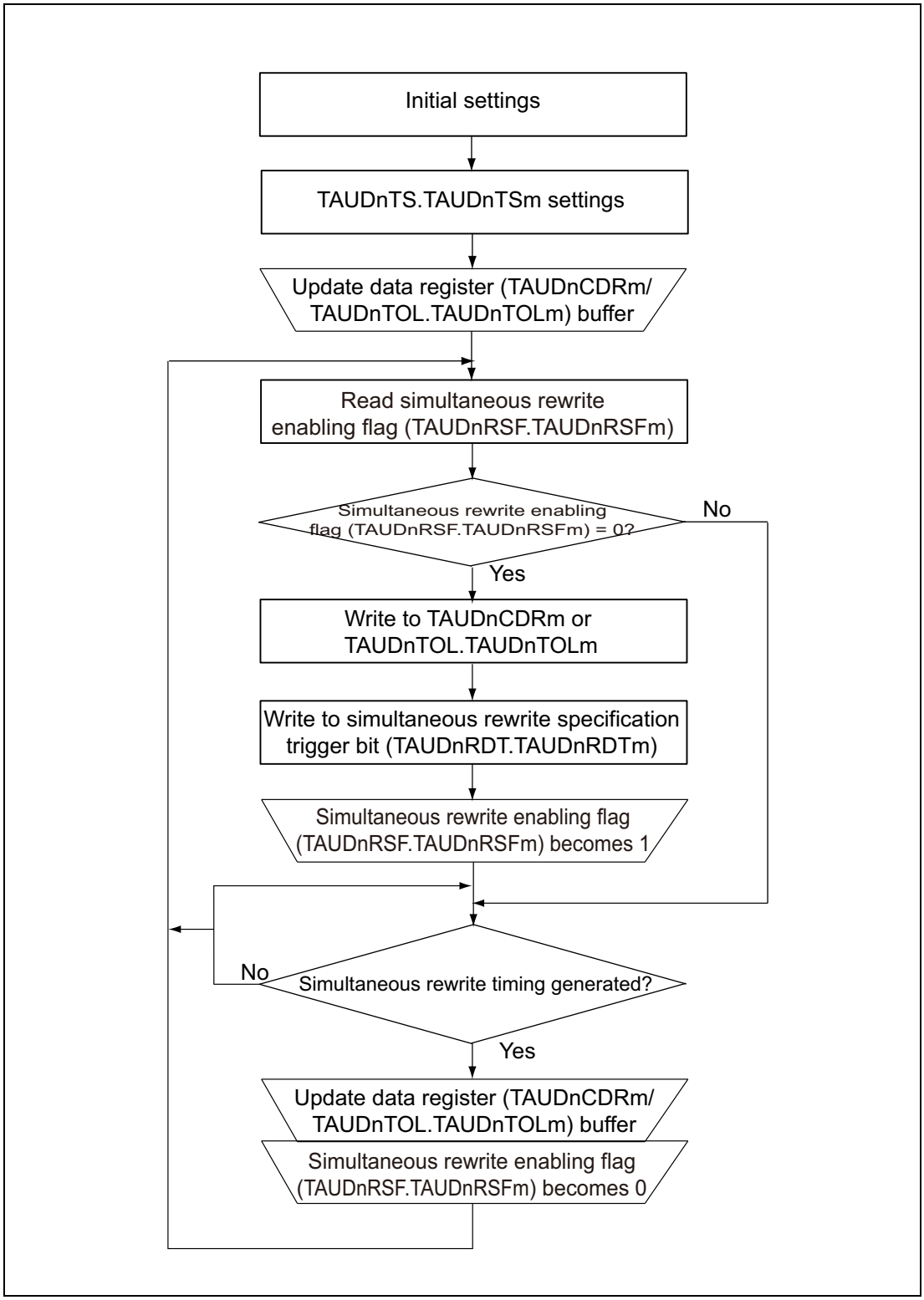


Figure 17.4 General Procedure for Simultaneous Rewrite

(1) Initial Settings

- To enable simultaneous rewrite in channel m, set TAUDnRDE.TAUDnRDEm = 1
- To select the type of simultaneous rewrite, set TAUDnRDM.TAUDnRDMm and TAUDnRDS.TAUDnRDSm according to the values listed in **Table 17.41, Simultaneous Rewrite Methods and when They are Triggered**.
- To select which upper channel is monitored for simultaneous rewrite triggers, use TAUDnRDC.TAUDnRDCm (prerequisite: TAUDnRDS.TAUDnRDSm is set in upper channel.)

(2) Start Counter and Count Operation

- To start all the TAUDnCNTm counters of the channel group, set the corresponding TAUDnTS.TAUDnTSm bits to1. The values of TAUDnTOL.TAUDnTOLm and the data registers (TAUDnCDRm) are loaded into the corresponding TAUDnTOL.TAUDnTOLm buffer (TAUDnTOL.TAUDnTOLm buf) and data buffer registers (TAUDnCDRm buf) and the counters start.
- Setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1 sets the reload flag (TAUDnRSF.TAUDnRSFm) to 1, enabling simultaneous rewrite. TAUDnRSF.TAUDnRSFm remains set to 1 until simultaneous rewrite is completed.
- When the specified trigger for simultaneous rewrite is detected, the TAUDnRSF.TAUDnRSFm bit is checked to see if simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1). If it is, simultaneous rewrite is carried out. Otherwise the simultaneous rewrite is not carried out and waits for the next trigger detection.

(3) Simultaneous Rewrite

- When simultaneous rewrite is enabled (TAUDnRSF.TAUDnRSFm = 1) and the simultaneous rewrite trigger is detected, the current values of the data registers are copied to their buffers. These values are then loaded into the corresponding counters and are applied the next time the counter starts or restarts.
- The TAUDnRSF.TAUDnRSFm bit is set to 0, and the system awaits the next simultaneous rewrite trigger.

17.4.3.3 Other General Rules of Simultaneous Rewrite

The following rules also apply:

- TAUDnRDE.TAUDnRDEm, TAUDnRDS.TAUDnRDSm, TAUDnRDM.TAUDnRDMm, and TAUDnRDC.TAUDnRDCm cannot be changed while the counter is in operation (TAUDnTE.TAUDnTEm = 1).
- TAUDnTOL.TAUDnTOLm can only be rewritten during operation with PWM output function or triangle PWM output function. For all other output functions, TAUDnTOL.TAUDnTOLm should be written before the counter starts. If it is rewritten while any other function is used, TAUDTTOUTm outputs an invalid wave.
- When an upper channel is used as a channel issuing the simultaneous rewrite trigger (TAUDnRDS.TAUDnRDSm = 1), the TAUDnRDC.TAUDnRDCm bit controls all the lower channels. This means that if the TAUDnRDC.TAUDnRDCm bits of CH2 and CH7 are set to 1 and the TAUDnRDC.TAUDnRDCm bits of other channels are set to 0, CH2 and CH7 serve as simultaneous rewrite trigger generation channels. CH2 controls the lower channels CH3 to CH6, and CH7 controls the lower channels CH8 to CH15.
- If simultaneous rewrite is enabled and an upper channel is selected for the simultaneous rewrite trigger (TAUDnRDE.TAUDnRDEm and TAUDnRDS.TAUDnRDSm = 1) but no upper channel is set (TAUDnRDC.TAUDnRDC[15:0] = 0), simultaneous rewrite cannot take place.

17.4.3.4 Types of Simultaneous Rewrite

In the following section, the four simultaneous rewrite methods are explained using timing diagrams.

(1) Simultaneous Rewrite when the Master Channel (Re)starts Counting (Method A)

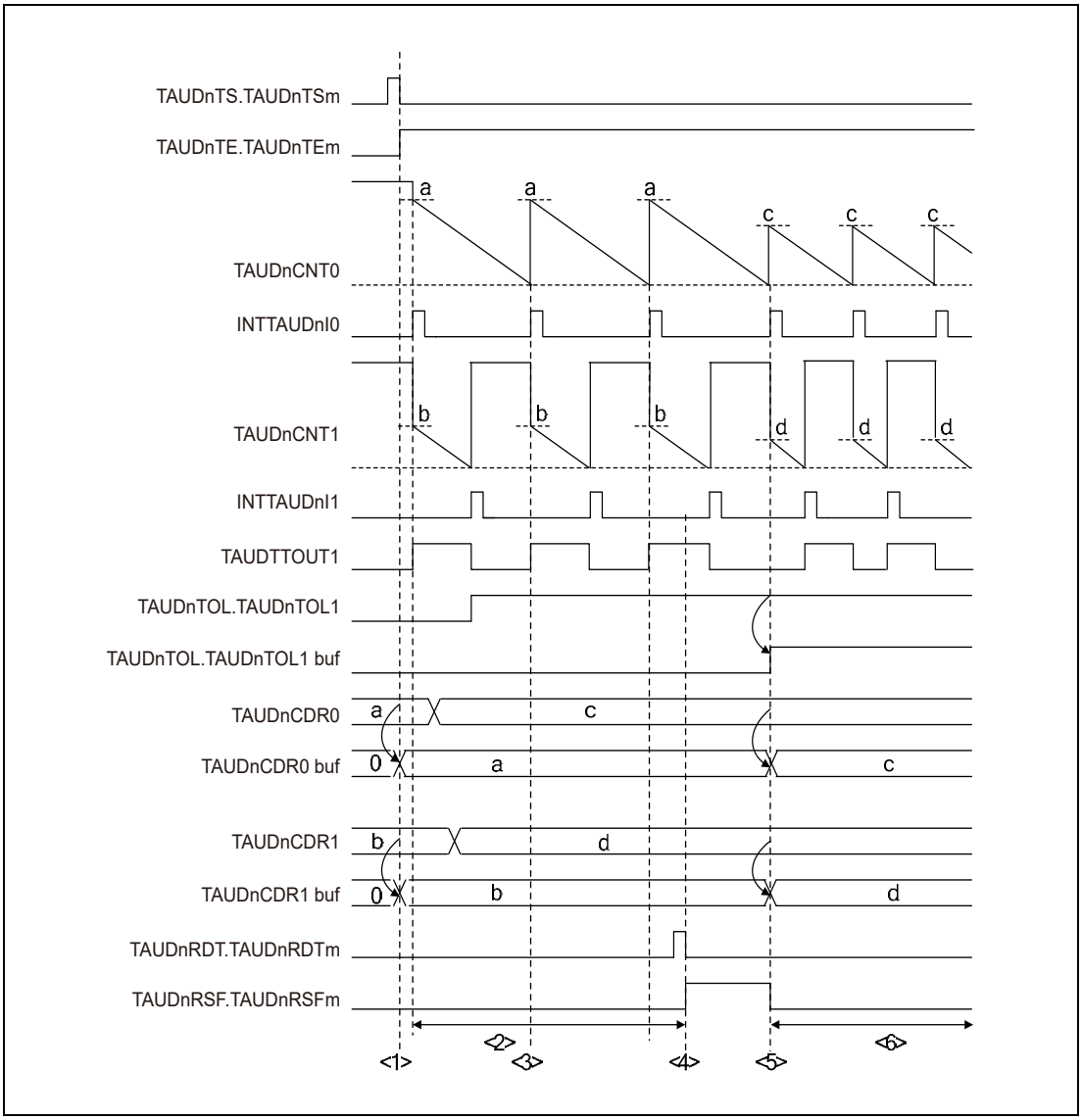


Figure 17.5 Simultaneous Rewrite when the Master Channel (Re)starts Counting

Setting

CH0 is the master channel, which counts down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method A is applied.

Description:

(1) When TAUDnTS.TAUDnTSm = 1 is set, the value of TAUDnCDRm is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm and TAUDnTOL.TAUDnTOLm registers can be written at any time.

(3) CH0 restarts counting, but simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0)

(4) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(5) Because simultaneous rewrite is enabled, it is triggered when CH0 restarts counting. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer and the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

(2) Simultaneous Rewrite at the Peak of a Triangular Cycle of Slave Channel (Method B)

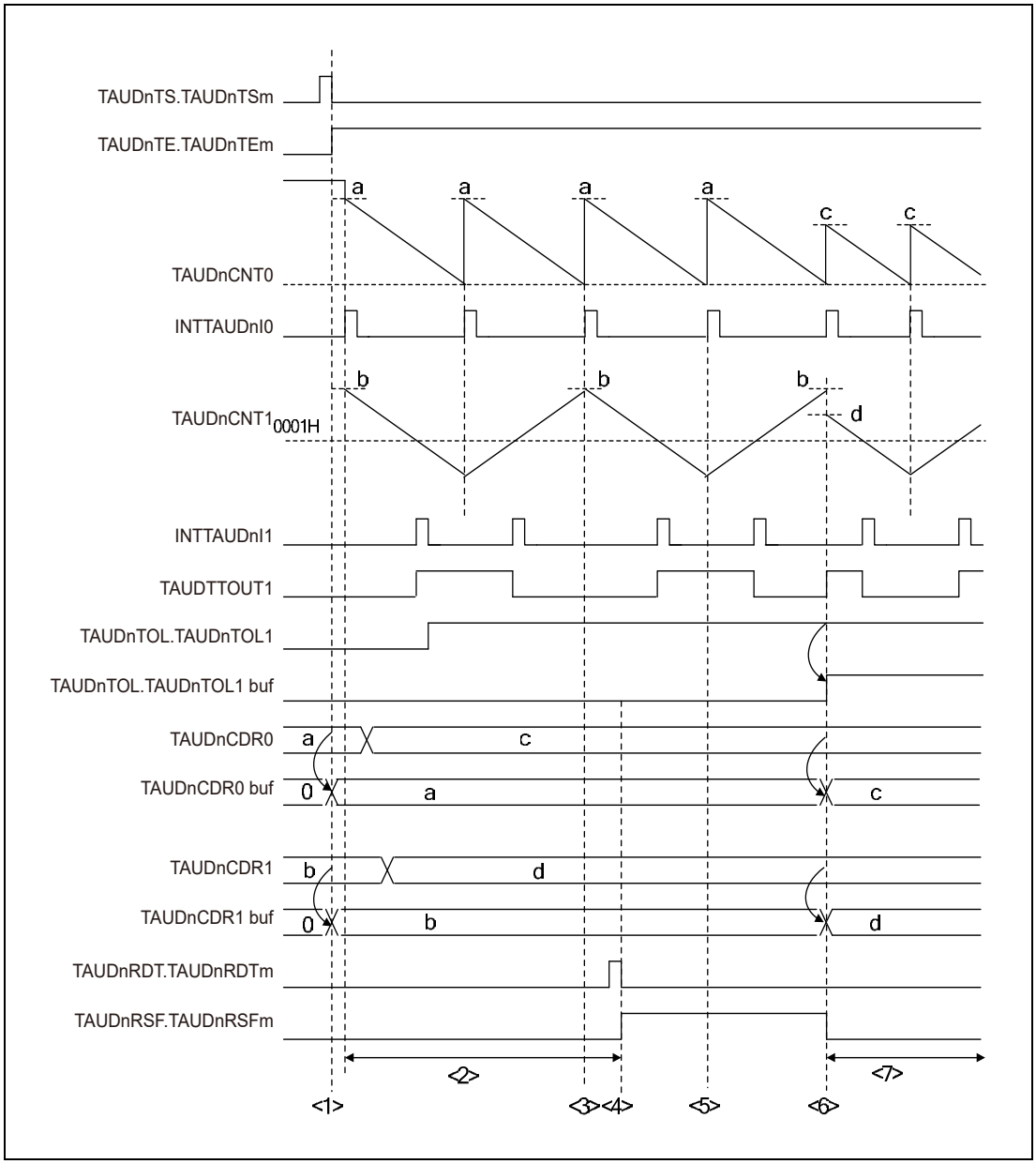


Figure 17.6 Simultaneous Rewrite at the Peak of a Triangular Cycle of Slave Channel

Setting

CH0 is the master channel which performs counting down, and CH1 represents an arbitrary slave channel. The simultaneous rewrite method B is applied.

Description:

(1) When TAUDnTS.TAUDnTSm = 1 is set, the value of TAUDnCDRm is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm and TAUDnTOL registers can be written at any time.

(3) Simultaneous rewrite does not occur because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

(4) The reload data trigger bit (TAUDnRDT.TAUDnRDTm) is set to 1 which sets the status flag (TAUDnRSF.TAUDnRSFm = 1), enabling simultaneous rewrite.

(5) Simultaneous rewrite does not take place at the bottom of the triangular cycle.

(6) Simultaneous rewrite takes place at the top of the triangular cycle. The TAUDnCDRm value is loaded into the TAUDnCDRm buffer, the TAUDnTOL.TAUDnTOLm value is loaded into the TAUDnTOL.TAUDnTOLm buffer.

(7) The counters count down and await the next simultaneous rewrite trigger. The values of TAUDnCDRm and TAUDnTOL.TAUDnTOLm can be changed again.

(3) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm (Method C1)

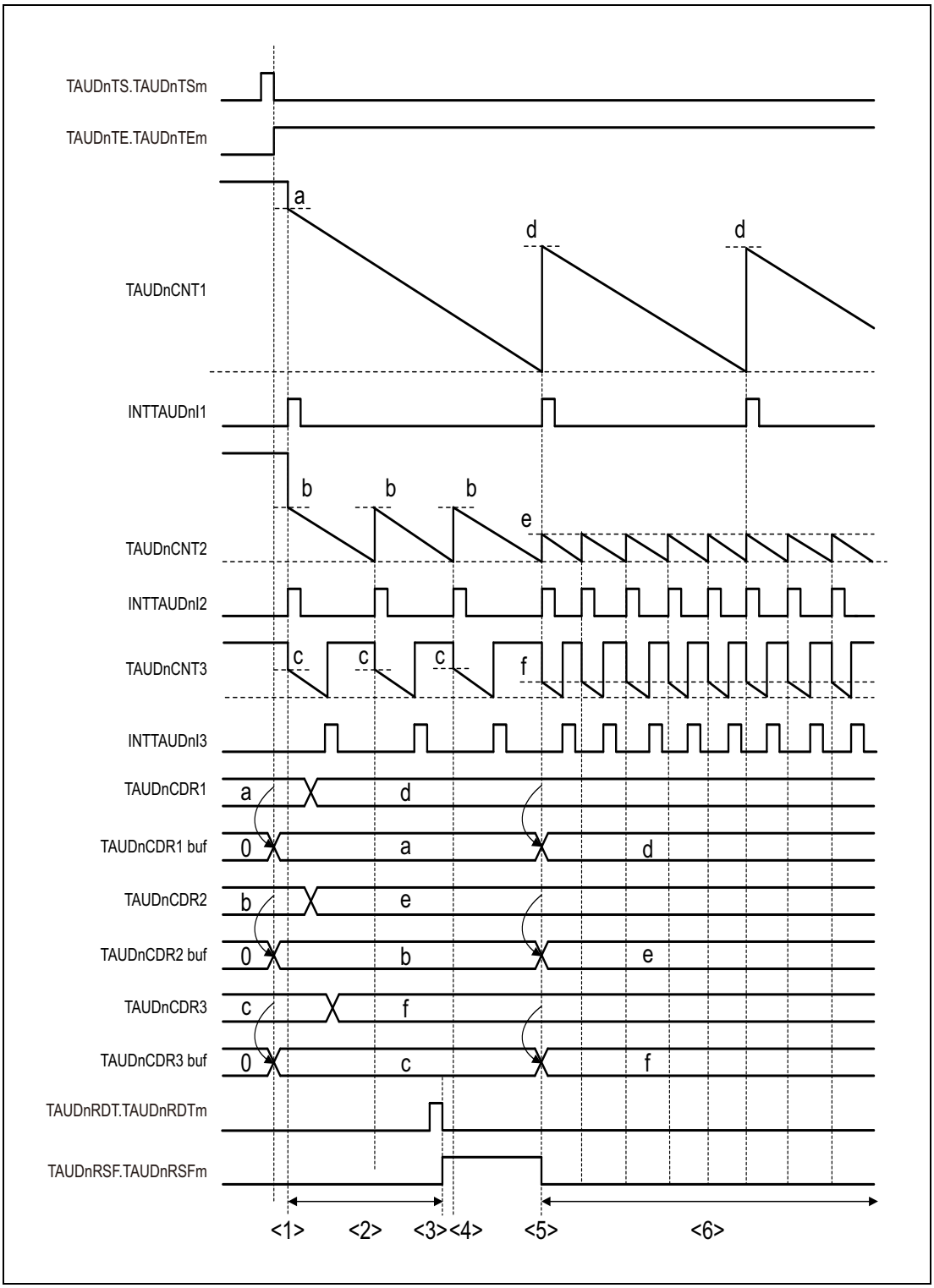


Figure 17.7 Simultaneous Rewrite When INTTAUDnIm Is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm

Setting

CH1 is an upper channel which performs counting down, CH2 is a master channel, and CH3 is the slave channel. The simultaneous rewrite method C1 is applied. The TAUDnRDC register specifies a channel which generates simultaneous rewrite triggers.

Description:

(1) When TAUDnTS.TAUDnTSm is set to 1, the TAUDnCDRm value is copied to the TAUDnCDRm buffer.

(2) The TAUDnCDRm register is always ready to write.

(3) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(4) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(5) Simultaneous rewrite is triggered by INT1 which is generated when counter 1 reaches 0000_H. The TAUDnCDRm values are loaded into the corresponding TAUDnCDRm buffers.

(6) The counter counts down and awaits the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be rechanged.

(4) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal (Method C2)

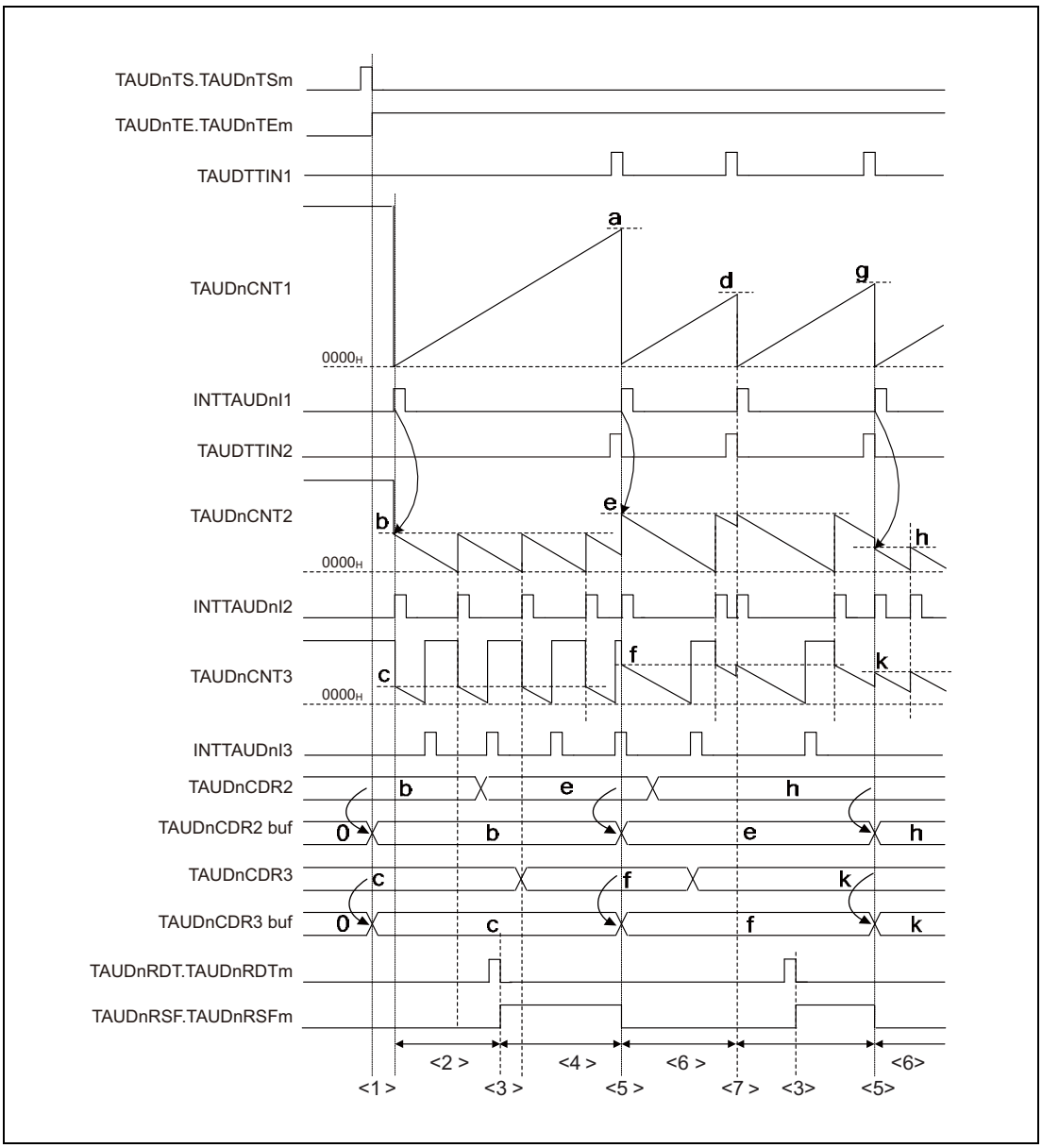


Figure 17.8 Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal

Setting

CH1 is an upper channel which performs counting up, CH2 is a master channel, and CH3 is the slave channel. The synchronous channel operation method C2 is applied. The TAUDnRDC register specifies which upper channel is monitored for an INTTAUDnIm trigger.

Description:

(1) When TAUDnTS.TAUDnTSm is set to 1, the TAUDnCDRm value is copied to the TAUDnCDRm buffer. However, as TAUDnCDR1 operates in capture mode, the TAUDnCDR1 value is not copied to the TAUDnCDR1 buffer.

(2) The TAUDnCDRm register is always ready to write.

(3) By setting the reload data trigger bit (TAUDnRDT.TAUDnRDTm) to 1, the status flag is set (TAUDnRSF.TAUDnRSFm = 1) to enable simultaneous rewrite.

(4) Simultaneous rewrite is triggered only by a CH1 interrupt. Therefore, simultaneous rewrite is not conducted even if enabled.

(5) Simultaneous rewrite is triggered by INT1 which is caused by external signal TIN1. The TAUDnCDRm values are written to the corresponding TAUDnCDRm buffers.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of the TAUDnCDRm registers can be changed again.

(7) An external signal occurs at TIN2 but simultaneous rewrite does not take place because it is disabled (TAUDnRSF.TAUDnRSFm = 0).

17.4.4 Channel Output Modes

The output of the TAUDTTOUTm pin can be controlled in two ways, the latter of which can be further split into individual modes.

- By software (TAUDnTOE.TAUDnTOEm = 0)
  When controlled by software, the value written in the output register bit (TAUDnTO.TAUDnTOm) is sent out of the output pin (TAUDTTOUTm).
- By TAUD signals (TAUDnTOE.TAUDnTOEm = 1)
  When controlled by TAUD signals, the output level of TAUDTTOUTm is set or reset or toggled by internal signals. The value of TAUDnTO.TAUDnTOm is updated accordingly to reflect the value of TAUDTTOUTm
  - Independently (TAUDnTOM.TAUDnTOMm = 0)
    In case of independent operation, the output of the TAUDTTOUTm pin is only affected by settings of channel m. Therefore, independent channel operation should be selected (TAUDnTOM.TAUDnTOMm = 0).
  - Synchronously (TAUDnTOM.TAUDnTOMm = 1)
    In case of synchronous operation, the output of the TAUDTTOUTm pin is affected by settings of channel m and those of other channels. Therefore, synchronous channel operation should be selected for all synchronized channels (TAUDnTOM.TAUDnTOMm = 1).

The TAUDnTO.TAUDnTOm bit can always be read to determine the current value of TAUDTTOUTm, regardless of whether the pin is controlled by software, operated independently, or operated synchronously.

Control bits

The settings of the control bits required to select a specific channel output mode are listed in **Table 17.43, Channel Output Modes**.

The channel output modes are described in details below.

- **Section 17.4.4.2, Channel Output Modes Controlled Independently by TAUDn Signals**
- **Section 17.4.4.3, Channel Output Modes Controlled Synchronously by TAUDn Signals**

Batch operation of TAUDnTOm bit

Whether a set value is reflected to the TAUDnTOm bit or not is controlled by the TAUDnTOE.TAUDnTOEm bit.

The TAUDnTOm setting is written only to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 0 when a write to the TAUDnTO register is attempted. No TAUDnTOm setting is reflected to the bit (channel) set with TAUDnTOE.TAUDnTOEm bit = 1.

NOTE

TAUDnTO.TAUDnTOm bit is placed so that its bit number corresponds to a channel number.

Output logic

Positive or negative logic of the output is specified by control bit TAUDnTOL.TAUDnTOLm.

The value of TAUDnTOL.TAUDnTOLm bit should be set before the counter is started. It can only be changed during operation with PWM output function or triangle PWM output function. Otherwise, changes to TAUDnTOL.TAUDnTOLm result in an invalid TAUDTTOUTm signal output.

See **Section 17.4.3, Simultaneous Rewrite**.

The various channel output modes and the channel output control bits are listed in **Table 17.43**.

Table 17.43 Channel Output Modes

Channel Output Mode	TAUDn TOE. TAUDn TOEm	TAUDn TOM. TAUDn TOMm	TAUDn TOC. TAUDn TOCm	TAUDn TDE. TAUDn TDEm	TAUDn TRE. TAUDn TREm	TAUDn TME. TAUDn TMEm	TAUDn TDM. TAUDn TDMm
By software							
Independent channel output mode controlled by software	0	X					
By TAUD signals, independently							
Independent channel output mode 1	1	0	0	0	0	0	0
with real-time output					1		
Independent channel output mode 2			1		0		
By TAUD signals, synchronously							
Synchronous channel output mode 1	1	1	0	0	0	0	0
with non-complementary modulation output					1	X	
Synchronous channel output mode 2			1	0	0	0	0
with dead time output				1			
with one-phase PWM output							1
with complementary modulation output					1	1	0
with non-complementary modulation output			1	0			

- All combinations not listed in this table are forbidden.
- Bits marked with an x can be set to any value.

NOTES

1. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTE = 1):
   - TAUDnTOE.TAUDnTOEm
   - TAUDnTOM.TAUDnTOMm
   - TAUDnTOC.TAUDnTOCm
   - TAUDnTDE.TAUDnTDEm
   - TAUDnTRE.TAUDnTREm
   - TAUDnTDM.TAUDnTDMm
2. The following bits cannot be changed during count operation (TAUDnTE.TAUDnTEm = 1) except in channel output modes with modulation output:
   - TAUDnTME.TAUDnTMEm
   - TAUDnTDL.TAUDnTDLm

17.4.4.1 General Procedures for Specifying a Channel Output Mode

This section describes the general procedures for specifying a TAUDTTOUTm channel output mode. The prerequisite is that timer output operation is disabled (TAUDnTOE.TAUDnTOEm = 0).

(1) Set TAUDnTO.TAUDnTOm to specify the initial level of the TAUDTTOUTm output.

(2) Set channel output mode according to **Table 17.43, Channel Output Modes**, and the output logic using the TAUDnTOL.TAUDnTOLm bit.

(3) Start the counter (TAUDnTS.TAUDnTSm = 1).

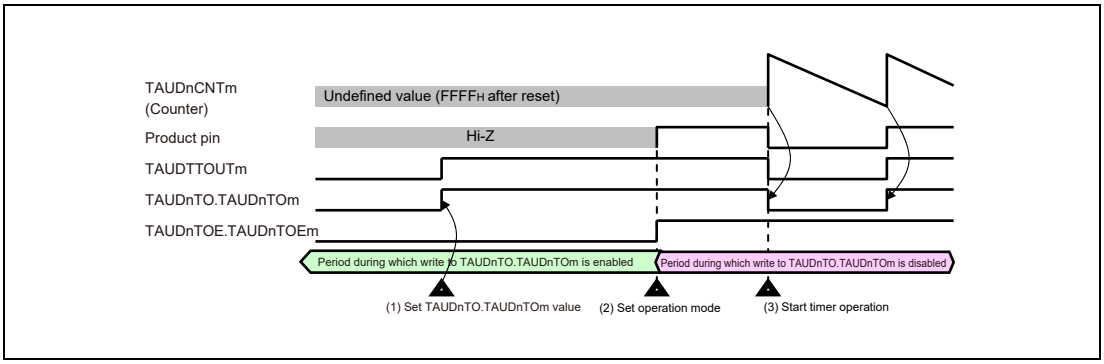


Figure 17.9 General Procedure for Specifying a TAUDTTOUTm Channel Output Mode

17.4.4.2 Channel Output Modes Controlled Independently by TAUDn Signals

This section lists the channel output modes that are controlled independently by TAUDn signals. The control bits used to specify a mode are listed in **Table 17.43, Channel Output Modes**.

(1) Independent Channel Output Mode 1

Set/reset conditions

In this output mode, TAUDTTOUTm toggles when INTTAUDnIm is detected. The value of TAUDnTOL.TAUDnTOLm is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 17.43, Channel Output Modes**

(2) Independent Channel Output Mode 1 with Real-Time Output

In this output mode, the value of TAUDnTRO.TAUDnTROm bit of the trigger channel is output to TAUDTTOUTm. The trigger channel is specified by setting the corresponding TAUDnTRC.TAUDnTRCm bit to 1. It controls all lower channels for which TAUDnTRC.TAUDnTRCm = 0.

Set/reset conditions

The value of TAUDnTRO.TAUDnTROm bit is sent to TAUDTTOUTm only when an INTTAUDnIm interrupt occurs on the trigger channel. The interrupt is generated either:

- at certain specified intervals or
- on detection of an effective TAUDTTINm input edge/counter start

The type of trigger is set using the TAUDnCMORm.TAUDnMD[4:1] bits.

Prerequisites

Both master and slave channels can be set as a trigger generation channel. A channel for which TAUDnTRC.TRCm is set to 1 serves as a trigger generation channel even if TAUDnTRE.TAUDnTREm is set to 0. If there is no channel for which TAUDnTRC.TAUDnTRCm is set to 1 or if TAUDnTRC.TAUDnTRC0 = 0, real-time output cannot take place.

This can be seen in **Figure 17.10**.

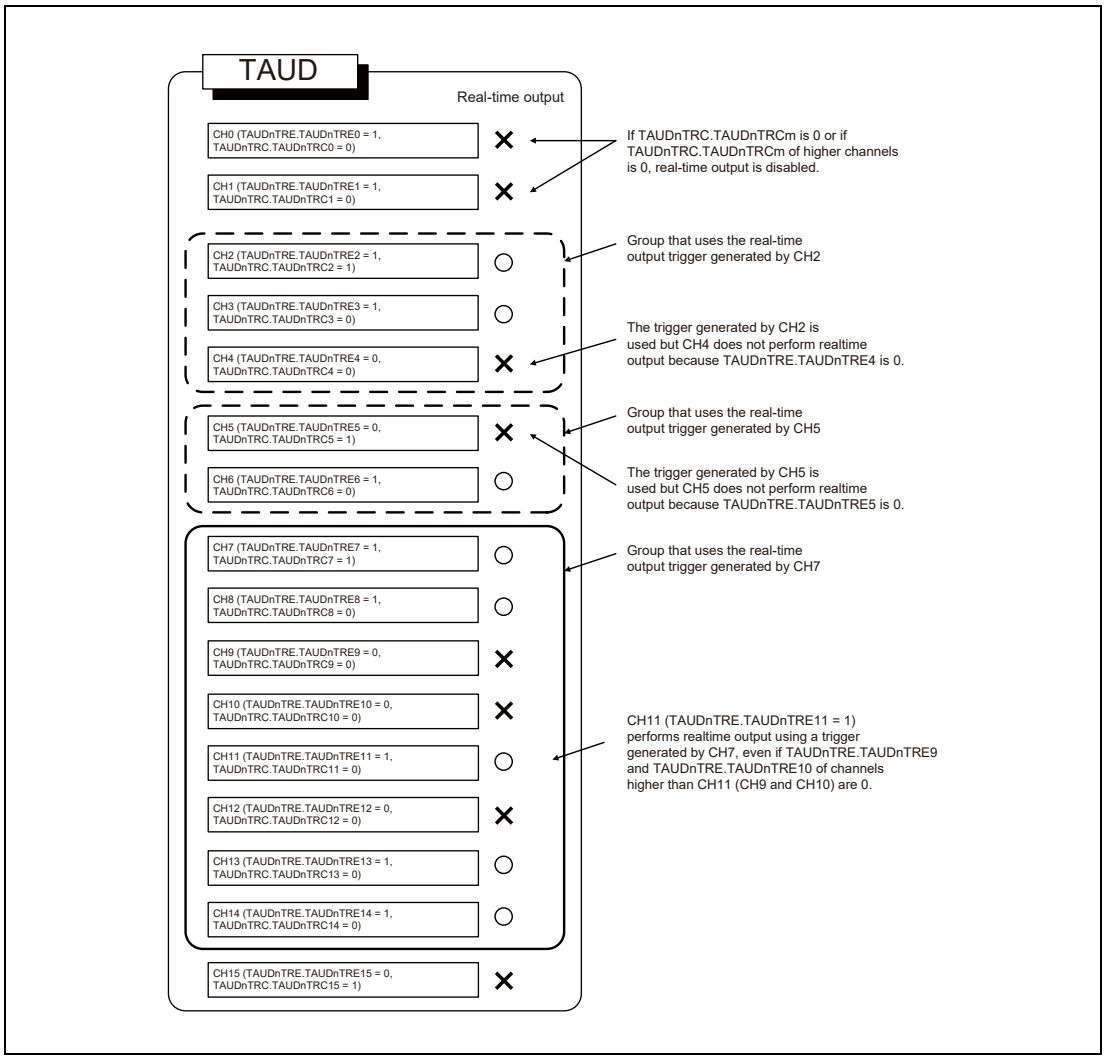


Figure 17.10 Real-Time Output

(3) Independent Channel Output Mode 2

Set/reset conditions

In this output mode, TAUDTTOUTm is set when INTTAUDnIm occurs at the time of count start, and reset when INTTAUDnIm occurs due to a match between TAUDnCNTm and TAUDnCDRm.

Prerequisites

There are no prerequisites other than those shown in **Table 17.43, Channel Output Modes**.

17.4.4.3 Channel Output Modes Controlled Synchronously by TAUDn Signals

This section lists the channel output modes that are controlled synchronously by TAUDn signals. The control bits used to specify a mode are listed in **Table 17.43, Channel Output Modes**.

(1) Synchronous Channel Output Mode 1

Set/reset conditions

In this output mode, INTTAUDnIm of master channel serves as a set signal and INTTAUDnIm of the slave channel as a reset signal. If INTTAUDnIm of master channel and INTTAUDnIm of the slave channel are generated at the same time, INTTAUDnIm of the slave channel (reset signal) has priority over INTTAUDnIm (set signal) of master channel, i.e., the master channel is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 17.43, Channel Output Modes**.

(2) Synchronous Channel Output Mode 1 with Non-Complementary Modulation Output

Set/reset conditions

In this output mode, TAUDTTOUTm outputs the result of an AND operation between the PWM output and the real-time output bit (TAUDnTRO.TAUDnTROm) of a channel.

The phase period to which the dead time is added is specified using the TAUDnTDL.TDLm bit; for positive phase set TAUDnTDL.TAUDnTDLm = 0 and for negative phase set TAUDnTDL.TAUDnTDLm = 1.

Prerequisites

A set of at least three channels is required to generate the PWM output. The master channel and slave channel 1 generate a period, and slave channel 2 generates the duty cycle. In typical applications, five more slave channels are also used that operate in the same manner as slave channel 2.

Only the PWM output and the real-time output bit of the same channel can be combined.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

- If TAUDnTME.TAUDnTMEm is changed, its new value is applied upon detection of INTTAUDnIm on the specified channel.
- If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on the master channel.

(3) Synchronous Channel Output Mode 2

In this output mode, the operating mode should be set to up/down count mode. The result is a triangle PWM wave at TAUDTTOUTm. For details, see **Section 17.4.12.7, Triangle PWM Output Function**.

Set/reset conditions

TAUDnCNTm of the slave channel counts down and up alternatively. When it passes 0001_H it generates an interrupt, causing TAUDTTOUTm to toggle.

Prerequisites

A set of two channels is required to generate the triangle PWM output. TAUDTTOUTm should be set to 0 before the function starts.

(4) Synchronous Channel Output Mode 2 with Dead Time Output

In this output mode, a dead time delay is added to TAUDTTOUTm. The set/reset conditions are shown in **Figure 17.11**.

Set/reset conditions

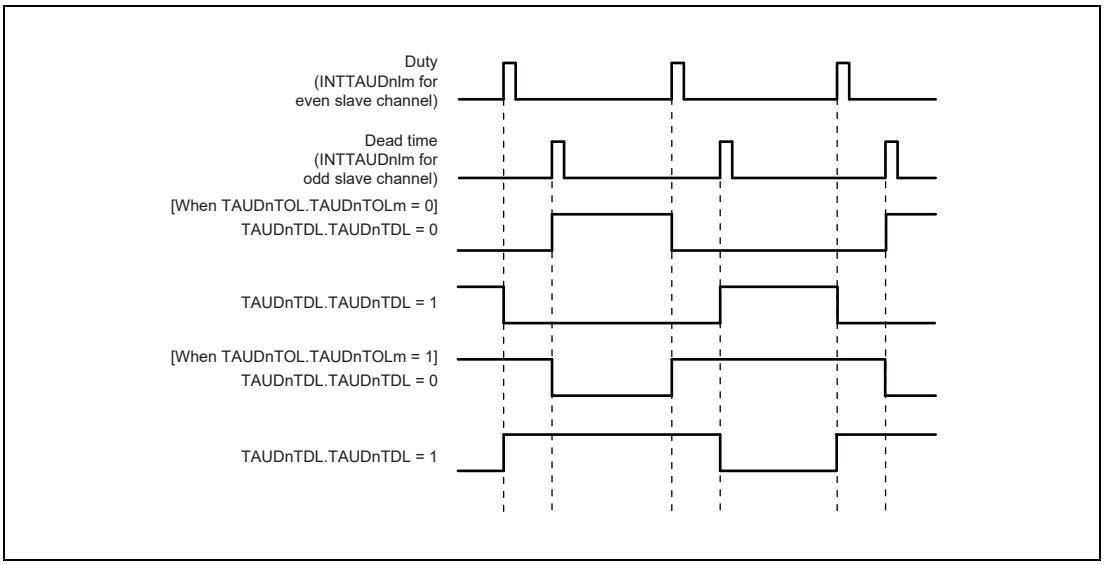


Figure 17.11 Set/Reset Conditions for Synchronous Channel Output Mode 2 with Dead Time Output

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges.

Prerequisites

Dead time control requires a set of three channels, each operating in the following modes:

- One master channel
  The master channel should be set to interval timer mode.
- One even slave channel
  The even slave channel should be set up/down count mode.
- One odd slave channel (even-numbered channel + 1)
  The odd slave channel should be set to one-count mode.

The values of the following bits should be the same for the odd-numbered channel and the even-numbered channel:

- TAUDnTOE.TAUDnTOEm
- TAUDnTME.TAUDnTMEm
- TAUDnTRE.TAUDnTREm
- TAUDnTOM.TAUDnTOMm

- TAUDnTOC.TAUDnTOCm
- TAUDnTDE.TAUDnTDEm
- TAUDnTDM.TAUDnTDMm

(5) Synchronous Channel Output Mode 2 with One-Phase PWM Output

In this output mode, a dead time delay is added to TAUDTTOUTm. The set/reset conditions are shown in **Figure 17.12**.

Set/reset conditions

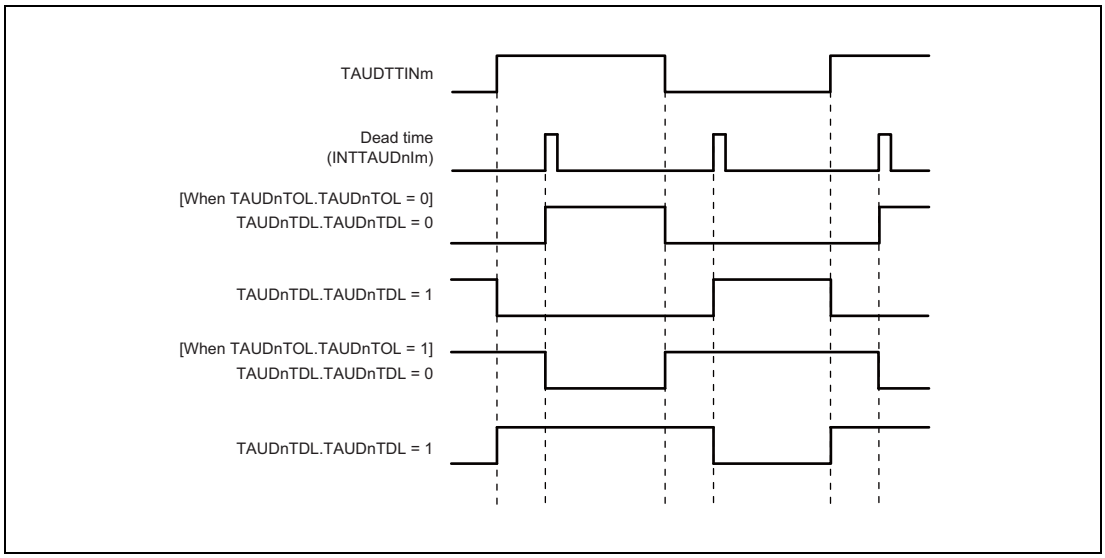


Figure 17.12 Set/Reset Conditions for Synchronous Channel Output Mode 2 with One-Phase PWM Output

With regard to the edge to which dead time is added, set TAUDnTDL.TAUDnTDLm = 0 for rising edges and TAUDnTDL.TAUDnTDLm = 1 for falling edges.

Prerequisites

One-phase PWM output control requires a set of two channels:

- One even slave channel
- One odd slave channel (even-numbered channel + 1)
  The odd slave channel should be set to one-count mode.

The values of the following bits should be the same for the odd-numbered channel and the even-numbered channel:

- TAUDnTOE.TAUDnTOEm
- TAUDnTME.TAUDnTMEm
- TAUDnTRE.TAUDnTREm
- TAUDnTOM.TAUDnTOMm
- TAUDnTOC.TAUDnTOCm
- TAUDnTDE.TAUDnTDEm
- TAUDnTDM.TAUDnTDMm

(6) Synchronous Channel Output Mode 2 with Complementary Modulation Output

Set/reset conditions

In this output mode, TAUDTTOUTm outputs a PWM signal, a high signal, or a low signal depending on the value of real-time output bit (TAUDnTRO.TAUDnTROm), the modulation output bit (TAUDnTME.TAUDnTMEm), and the output level bit (TAUDnTOL.TAUDnTOLm) of a pair of slave channels.

For details, see **Section 17.4.13.3, Complementary Modulation Output Function**.

Prerequisites

A set of at least four channels is required for this mode. The master channel and slave channel 1 generate a period, slave channel 2 generates a duty cycle, and slave channel 3 generates dead time. Slave channels 2 and 3 are a pair. In typical applications, 4 more channels are also used, which operates in the same manner as slave channels 2 and 3 respectively.

TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can only be changed during count operation.

- If TAUDnTME.TAUDnTMEm is changed during operation, its new value is applied upon detection of INTTAUDnIm at the specified channel.
- If TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm are changed, their new values are applied upon detection of INTTAUDnIm on an even slave channel.

(7) Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output

The difference from synchronous channel output mode 1 with non-complementary modulation output is the PWM wave shape.

Mode 1 has a rectangular wave while mode 2 has a triangular wave.

17.4.5 Start Timing in Each Operating Modes

This section describes the timing at which the counter starts after TAUDnTS.TAUDnTSm is set to 1 in each operating mode.

In all modes, the value of data register and whether or not an interrupt occurs depends on mode and register settings.

CAUTION

The count start timing described in this section is for your reference. Actually, the count start timing depends on the count clock timing.

17.4.5.1 Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode

The counter starts operating with the next count clock after TAUDnTS.TAUDnTSm is set to 1. The value of data register is also loaded when the counter starts.

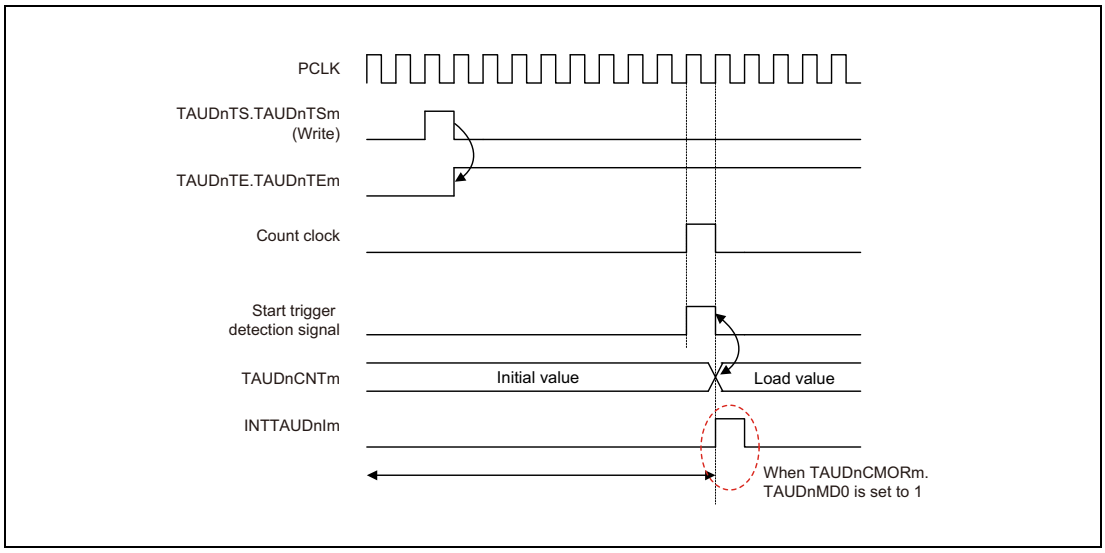


Figure 17.13 Start Timing in Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode

NOTE

Make sure to set TAUDnCMORm.TAUDnMD0 to 0 when using the up/down count mode.

17.4.5.2 Event Count Mode

The value of data register is loaded as soon as TAUDnTS.TAUDnTSm is set to 1. The counter also starts immediately. The value of data register decrements with subsequent count clocks.

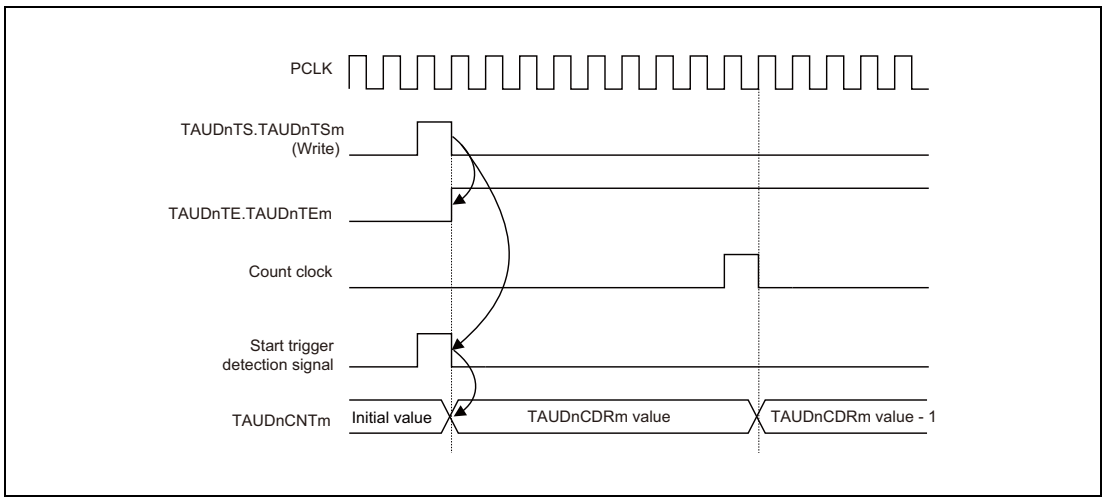


Figure 17.14 Start Timing in Event Count Mode

17.4.5.3 Other Operating Modes

In other operating modes, the counter operation start timing is triggered only upon detection of an effective edge of TAUDTTINm. Once the counter starts, the value of data register is also loaded. The count clock cycles, which is irrelevant to start of counter operation, determine the frequency with which all operations take place.

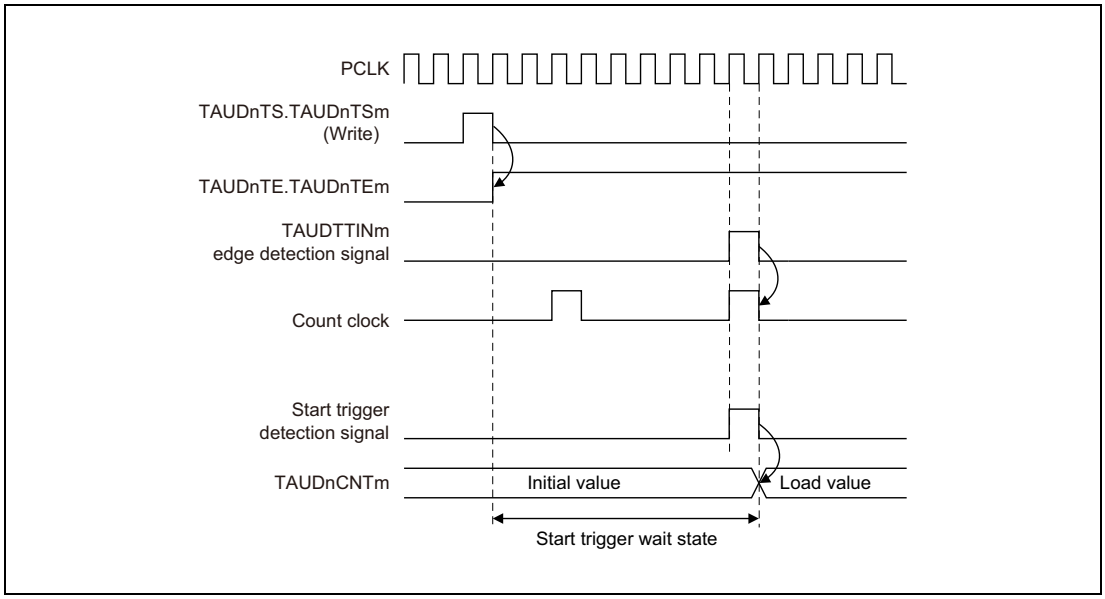


Figure 17.15 Start Timing in Other Operating Modes

17.4.6 TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts

When the counter starts, it is possible to specify whether an INTTAUDnIm is generated using the TAUDnCMOR.TAUDnMD0 bit. The effect of the bit depends on the selected mode, as shown in **Table 17.44**. The effects of INTTAUDnIm on TAUDTTOUTm depend on the selected channel operation function.

Table 17.44 Effect of TAUDnCMORm.TAUDnMD0 Bit on Generation of INTTAUDnIm when Counter is Triggered

Mode	TAUDnCMORm.TAUDnMD0 Bit	INTTAUDnIm Generated when Counter Starts
Interval timer mode Capture Mode Count Capture Mode	0	No
	1	Yes
Capture & One Count Mode Capture & Gate Count Mode Event Count Mode Up Down Count Mode	0	No
One Count Mode	0/1	No, regardless of setting of TAUDnCMORm.TAUDnMD0 bit.
Pulse One Count Mode	0/1	Yes, regardless of setting of TAUDnCMORm.TAUDnMD0 bit.

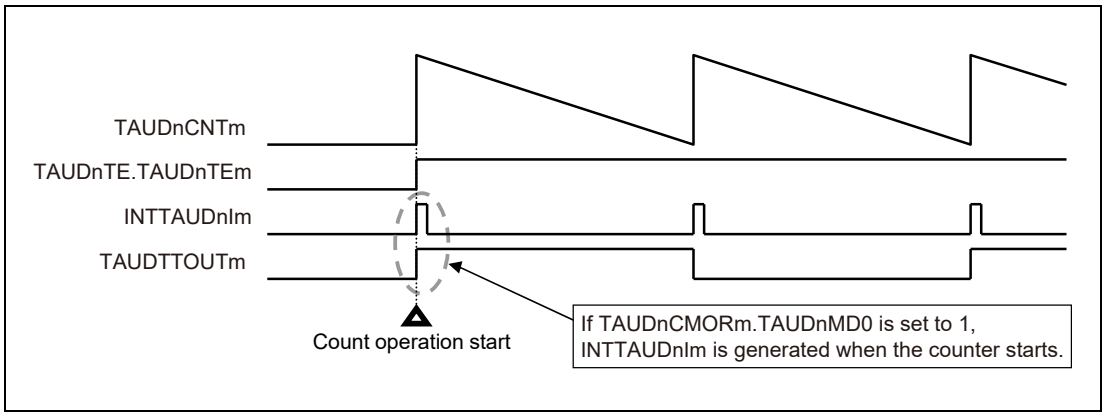


Figure 17.16 INTTAUDnIm Generated when Counter Starts

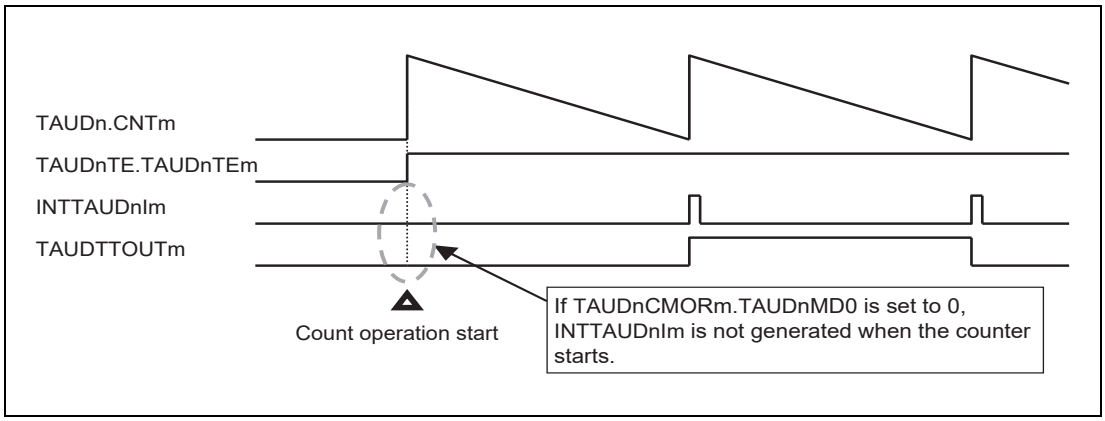


Figure 17.17 INTTAUDnIm not Generated when Counter Starts

17.4.7 Interrupt Generation upon Overflow

Certain independent functions that count up, overflow without generating an interrupt when they reach $FFFF_H$. This section describes how it is possible to generate an interrupt, by combining a channel operating in one of these modes with a channel in a different operation mode which counts down.

The appropriate operation mode for the second channel depends on the operation mode of the first channel. Nevertheless, the principle is the same for all combinations:

- Find an operation mode for the second channel that counts down in such a manner, that it reaches 0000_H at the same time as the first channel overflows (TAUDnCNTm = $FFFF_H$).
- Set TAUDnCDRm of the second channel to $FFFF_H$.
- The two channels must count at the same speed (i.e. they must have the same count clock).

Result:

The down-counter of the second channel reaches 0000_H at exactly the same time as the up-counter of the first channel overflows (TAUDnCNTm = $FFFF_H$). Thus the second channel generates the desired interrupt.

The following sections list the operating modes that count down that are required to match specific operating modes that count up, as well as example timing diagrams.

17.4.7.1 Count Capture Mode

Applies to

- TAUDTTINm Input Position Detection Function

Combine with

Interval Timer Mode

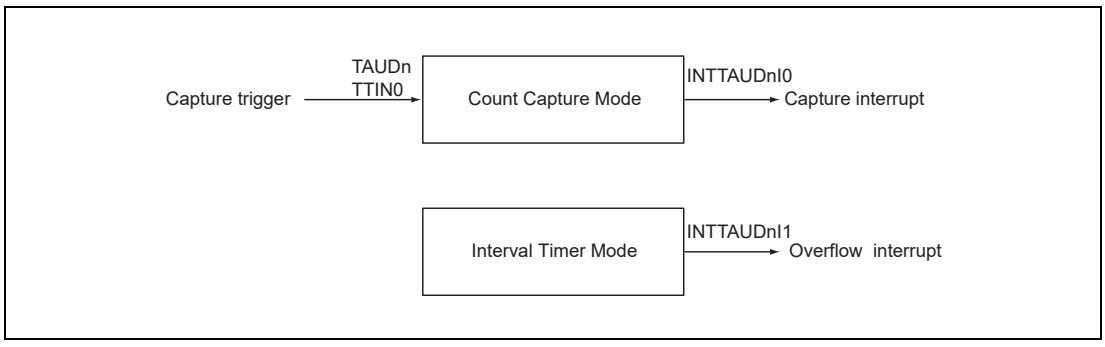


Figure 17.18 Combination of Count Capture Mode and Interval Timer Mode

Timing diagram

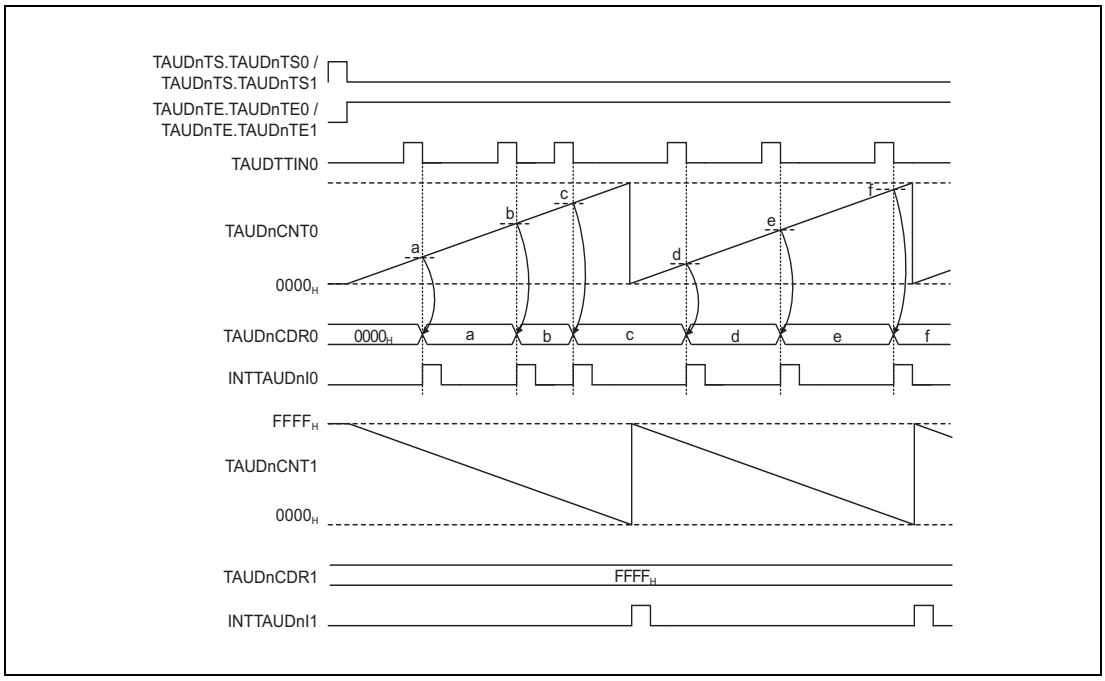


Figure 17.19 Interrupt Generation via Combination of Count Capture Mode and Interval Timer Mode

17.4.8 TAUDTTINm Edge Detection

Edge detection is based on the operation clock. This means that an edge can only be detected at the next rising edge of the operation clock. This can lead to a maximum delay of one operation clock cycle.

Figure 17.20 shows when edge detection takes place.

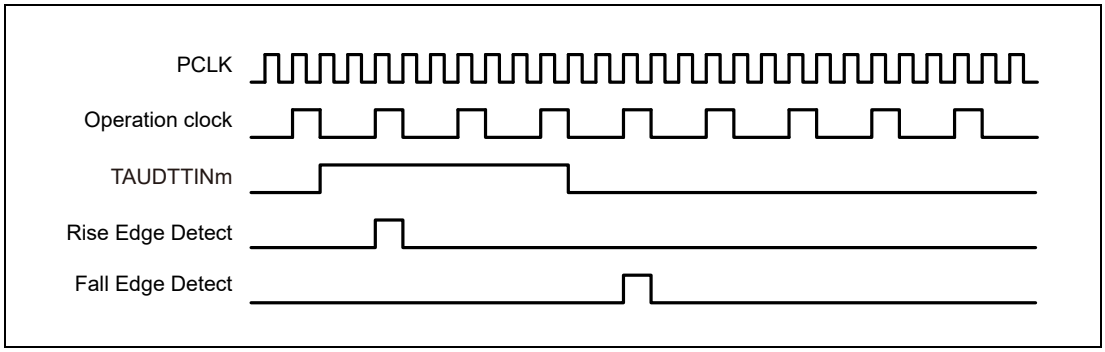


Figure 17.20 Basic Edge Detection Timing

Figure 17.20 shows an operation timing image. Actually, a noise filter or synchronization circuit which is located between the TAUDnlm pin and TAUDn causes a delay time.

17.4.9 Independent Channel Operation Functions

The following sections list the independent channel operation functions provided by the Timer Array Unit D. For a general overview of independent channel operation, see **Section 17.2, Overview**.

This section describes functions that generate interrupts at regular intervals or with a specified delay.

- **Section 17.4.9.1, Interval Timer Function**
- **Section 17.4.9.2, TAUDTTINm Input Interval Timer Function**
- **Section 17.4.9.3, Clock Divide Function**
- **Section 17.4.9.4, External Event Count Function**
- **Section 17.4.9.5, Delay Count Function**
- **Section 17.4.9.6, One-Pulse Output Function**
- **Section 17.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function**
- **Section 17.4.9.8, TAUDTTINm Input Signal Width Measurement Function**
- **Section 17.4.9.9, TAUDTTINm Input Position Detection Function**
- **Section 17.4.9.10, TAUDTTINm Input Period Count Detection Function**
- **Section 17.4.9.11, TAUDTTINm Input Pulse Interval Judgment Function**
- **Section 17.4.9.12, TAUDTTINm Input Signal Width Judgment Function**

17.4.9.1 Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUDnIm) at regular intervals. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operation mode must be set to Interval Timer Mode, refer to **Table 17.45, Contents of TAUDnCMORm Register for Interval Timer Function**.
- The channel output mode must be set to Independent Channel Output Mode 1, refer to **Section 17.4.4, Channel Output Modes**.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is written to TAUDnCNTm and the counter starts to count down from this value.

When the counter reaches 0000_H, INTTAUDnIm is generated and the TAUDTTOUTm signal toggles. TAUDnCNTm then reloads the TAUDnCDRm value and subsequently continues to operate.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1, which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm stop but retain their values. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. The counter can also be forcibly restarted (without stopping it first) by setting TAUDnTS.TAUDnTSm to 1 during operation.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUDTTOUTm does not toggle. This results in an inverted TAUDTTOUTm signal compared to when TAUDnCMORm.TAUDnMD0 is set to 1. For details refer to **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUDnIm cycle = count clock cycle × (TAUDnCDRm + 1)

TAUDTTOUTm square wave cycle = count clock cycle × (TAUDnCDRm + 1) × 2

(3) Block Diagram and General Timing Diagram

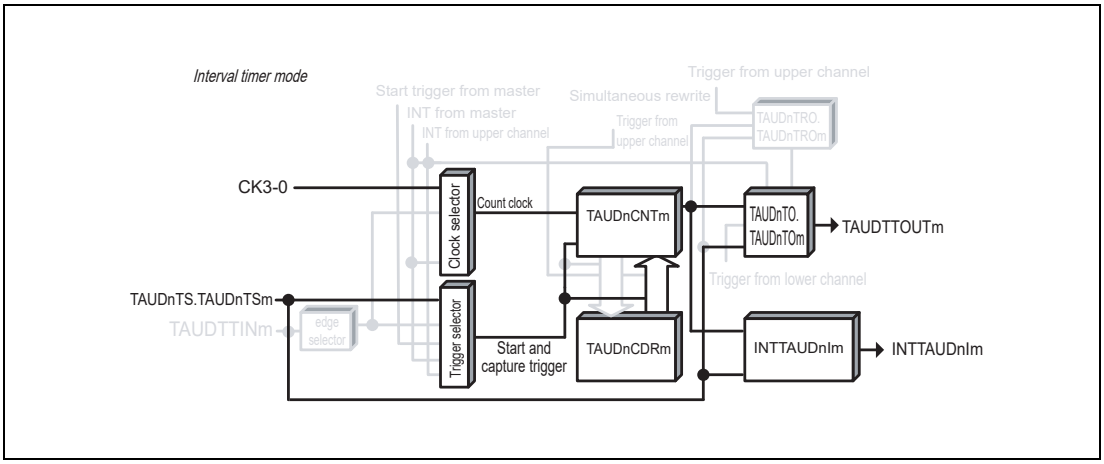


Figure 17.21 Block Diagram of Interval Timer Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 1)

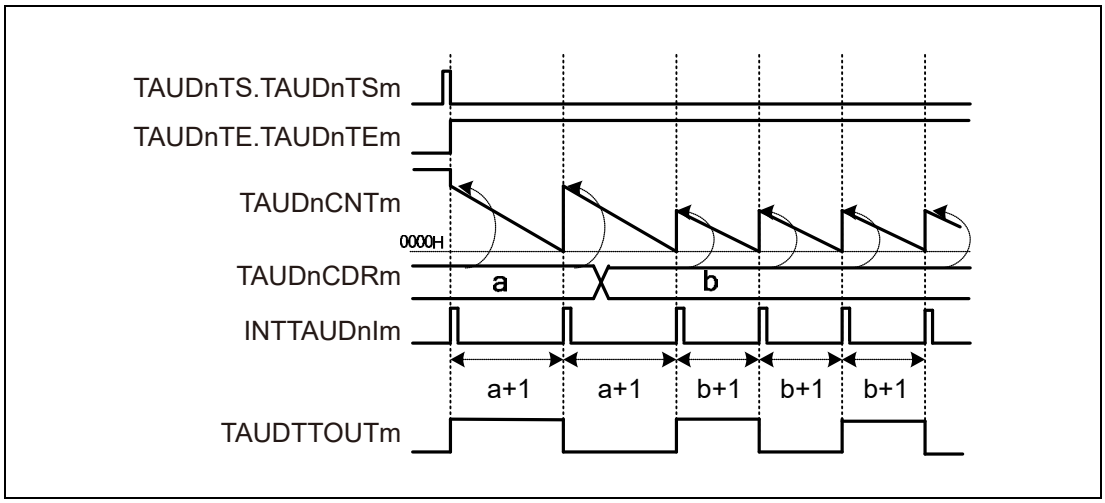


Figure 17.22 General Timing Diagram of Interval Timer Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.45 Contents of TAUDnCMORm Register for Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the sampling clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUDnCCS [1:0]	00: Uses the sampling clock as a counter clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Triggers the counter by software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.46 Contents of TAUDnCMURm Register for Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	Unused. Set to 00.

(c) Channel output mode

Table 17.47 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts. For details, see **Section 17.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the interval timer function. Therefore, these registers should be set to 0.

Table 17.48 Simultaneous Rewrite Settings for Interval Timer Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Interval Timer Function

Table 17.49 Operating Procedure for Interval Timer Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.45, Contents of TAUDnCMORm Register for Interval Timer Function**, and **Table 17.46, Contents of TAUDnCMURm Register for Interval Timer Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 17.47, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. The TAUDnCDRm value is loaded in TAUDnCNTm. When TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated and TAUDTTOUTm toggles.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated and TAUDTTOUTm toggles.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H, count clock = PCLK/2

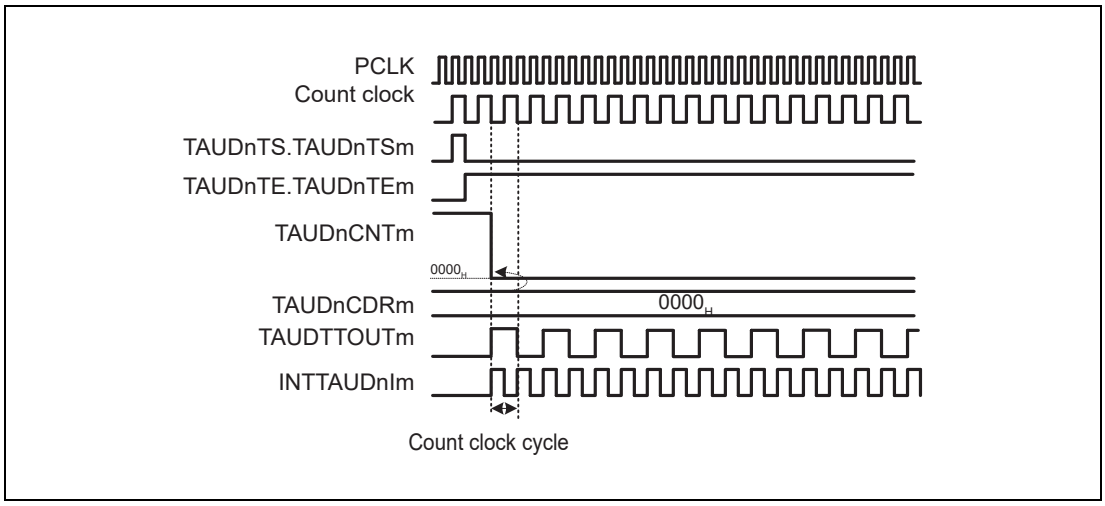


Figure 17.23 TAUDnCDRm = 0000_H, Count Clock = PCLK/2

- If TAUDnCDRm = 0000_H and the count clock = PCLK/2[1], the TAUDnCDRm value is loaded into TAUDnCNTm every count clock, meaning that TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated every count clock, resulting in TAUDTTOUTm toggling every count clock.

(b) TAUDnCDRm = 0000_H, count clock = PCLK

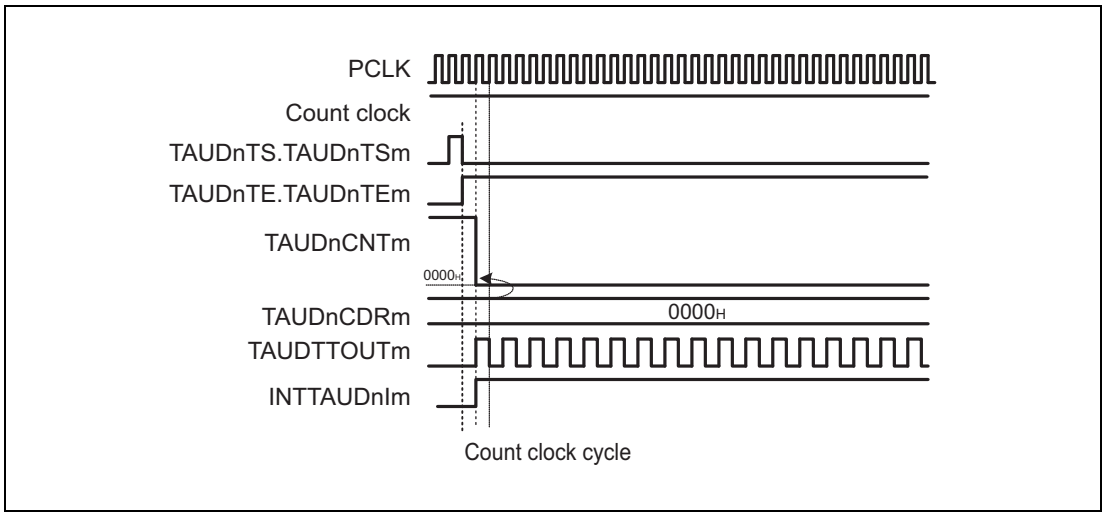


Figure 17.24 TAUDnCDRm = 0000_H, Count Clock = PCLK

- If TAUDnCDRm = 0000_H and the count clock = PCLK, the TAUDnCDRm value is loaded into TAUDnCNTm every PCLK clock, meaning that TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated continuously, resulting in TAUDTTOUTm toggling every PCLK clock.

(c) Operation stop and restart

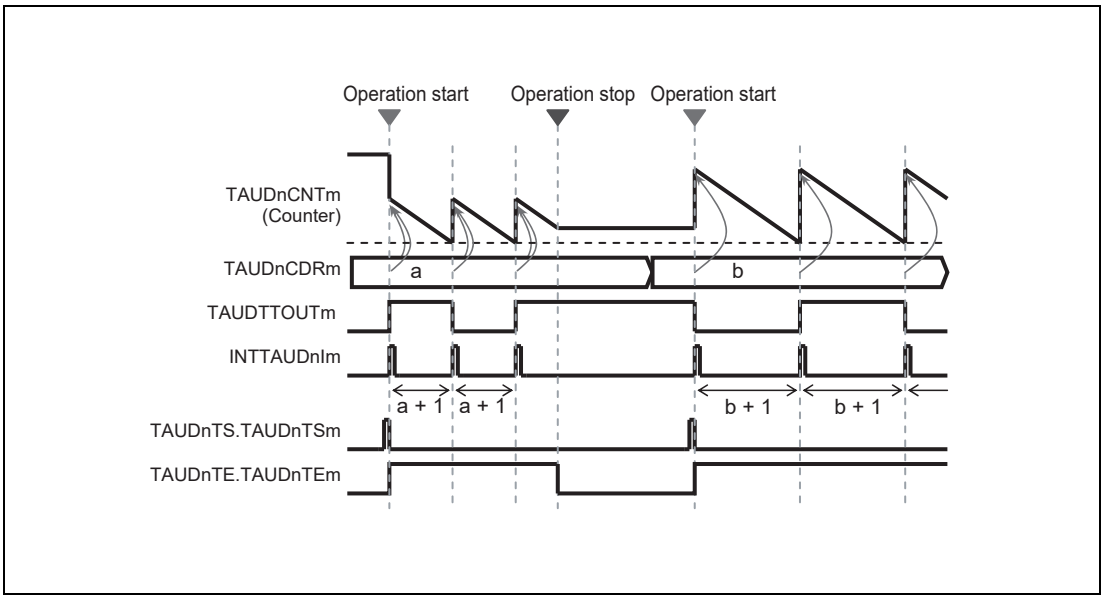


Figure 17.25 Operation Stop and Restart (TAUDnCMORm.TAUDnMD0 = 1)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm and TAUDTTOUTm stop but retain their values.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

(d) Forced restart (TAUDnCMORm.TAUDnMD0 = 1)

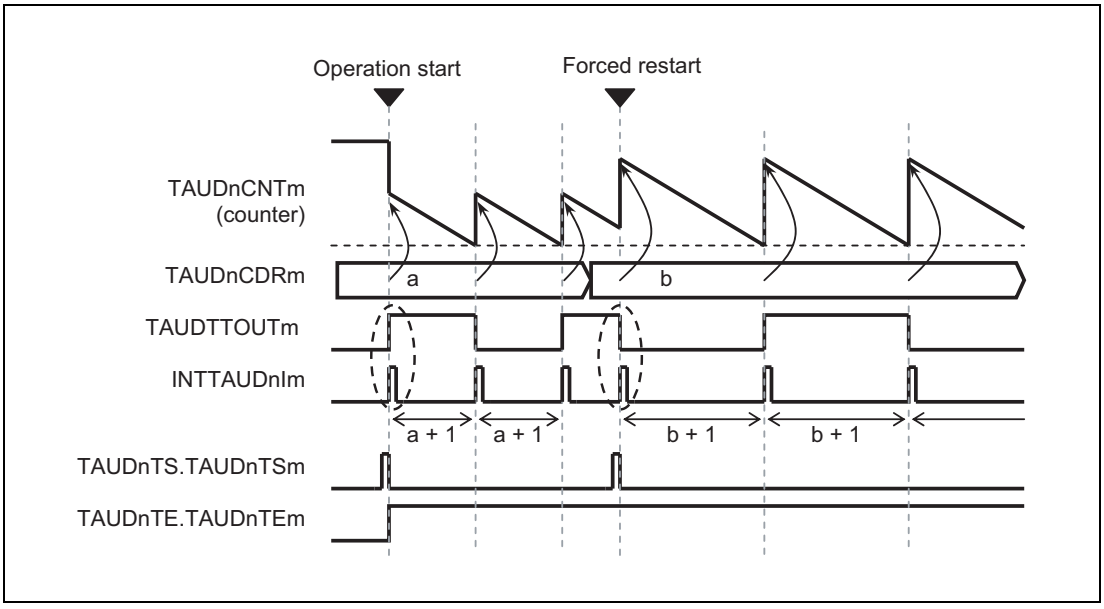


Figure 17.26 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 1)

- The counter can be forcibly restarted (without stopping it first) by setting TAUDnTS.TAUDnTSm to 1 during operation.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 1, the first interrupt after a start or restart is generated.

- When a forced restart is made, the TAUDnCDRm value is reflected to TAUDnCNTm and counting starts. Execute a forced restart to reflect the changed TAUDnCDRm value immediately.
- When a forced restart is made, an interrupt (INTTAUDnlm) is generated and TAUDTTOUTm is inverted.

(7) Forced restart (TAUDnCMORm.TAUDnMD0 = 0)

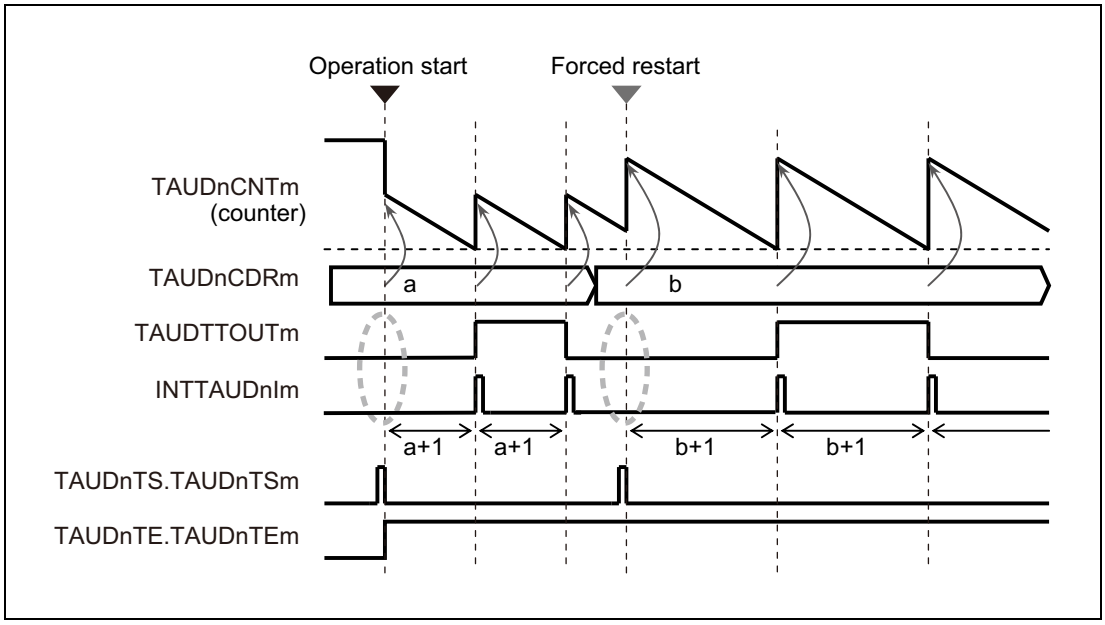


Figure 17.27 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 0)

- When a forced restart is made, an interrupt (INTTAUDnlm) is not generated and TAUDTTOUTm is not inverted.

17.4.9.2 TAUDTTINm Input Interval Timer Function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUDnIm) at regular intervals or when an effective TAUDTTINm input edge is detected. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode should be set to interval timer mode. See **Table 17.50, Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function**.
- The channel output mode should be set to independent channel output mode 1. See **Section 17.4.4, Channel Output Modes**.

Functional description

This function operates in an identical manner to the interval timer function (see **Section 17.4.9.1, Interval Timer Function**) except that this function is restarted by an effective TAUDTTINm input edge. The type of edge used as a trigger is specified using the TAUDnCMURm.TAUDnTIS[1:0] bits. Either rising edge, falling edge, or rising and falling edge can be selected.

(2) Equations

INTTAUDnIm cycle = count clock cycle × (TAUDnCDRm + 1)

TAUDTTOUTm square wave cycle = count clock cycle × (TAUDnCDRm + 1) × 2

(3) Block Diagram and General Timing Diagram

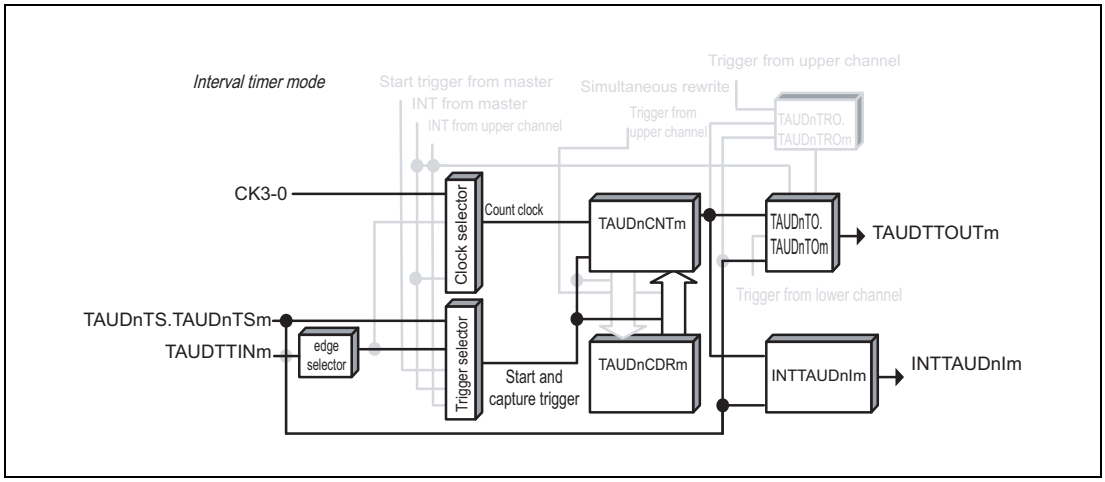


Figure 17.28 Block Diagram of TAUDTTINm Input Interval Timer Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 1)
- Rising edge detection (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

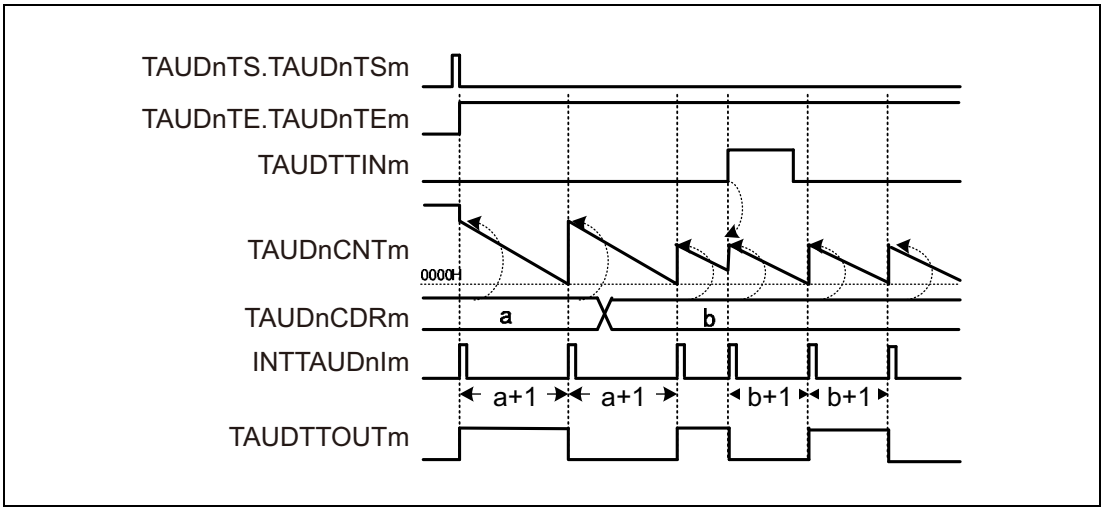


Figure 17.29 General Timing Diagram of TAUDTTINm Input Interval Timer Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.50 Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a counter clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.51 Contents of TAUDnCMURm Register for TAUDTTINm Input Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 17.52 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Setting prohibited

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts. For details, see **Section 17.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm Input Interval Timer Function. Therefore, these registers should be set to 0.

Table 17.53 Simultaneous Rewrite Settings for TAUDTTINm Input Interval Timer Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Interval Timer Function

Table 17.54 Operating Procedure for TAUDTTINm Input Interval Timer Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.50, Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function**, and **Table 17.51, Contents of TAUDnCMURm Register for TAUDTTINm Input Interval Timer Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 17.52, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. The TAUDnCDRm value is loaded in TAUDnCNTm. When TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated and TAUDTTOUTm toggles.
During Operation	The values of the TAUDnCMURm.TAUDnTIS[1:0] and the TAUDnCDRm register are changeable at any time. The TAUDnCNTm register can be read at all times. Detection of TAUDTTINm edge	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated and TAUDTTOUTm toggles. When an effective TAUDTTINm input edge is detected during count operation, the TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(6) Specific Timing Diagrams

The timing diagrams in **Section 17.4.9.1, Interval Timer Function** apply, and in addition the counter can also be restarted by an effective TAUDTTINm input edge.

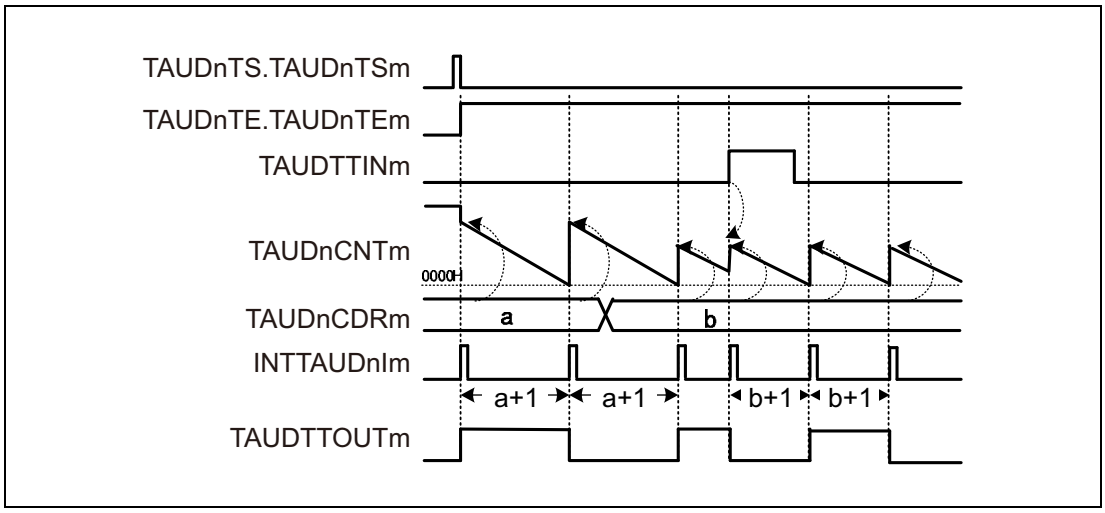


Figure 17.30 Counter Triggered by Rising TAUDTTINm Input Edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B), TAUDnCMORm.TAUDnMD0 = 1

- If an effective TAUDTTINm input edge is detected, an interrupt is generated which causes TAUDTTOUTm to toggle. In this example, the effective edge is a rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B).

17.4.9.3 Clock Divide Function

(1) Overview

Summary

This function is used as a frequency divider. The frequency of the input signal TAUDTTINm is divided by a factor related to TAUDnCDRm, and the resulting signal is output to TAUDTTOUTm.

Prerequisites

- TAUDTTINm should have a fixed frequency.
- The operating mode should be set to interval timer mode (see **Table 17.55, Contents of TAUDnCMORm Register for Clock Divide Function**).
- The channel output mode should be set to independent channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value, using TAUDTTINm as a count clock.

When the counter value reaches 0000_H, INTTAUDnIm occurs and TAUDTTOUTm signal is toggled. Then, TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently.

The value of TAUDnCDRm can be rewritten at any time. The changed value of TAUDnCDRm is applied when the counter starts to count down next time.

The counter can be stopped by setting TAUDnTT.TAUDnTTm = 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm and TAUDTTOUTm stop but retain their values. The function can be restarted by setting TAUDnTS.TAUDnTSm = 1. The counter can also be forcibly restarted without making a stop by setting TAUDnTS.TAUDnTSm = 1 during operation (forced restart).

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUDTTOUTm does not toggle. This results in an inverted TAUDTTOUTm signal compared to when TAUDnCMORm.TAUDnMD0 is set to 1. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

NOTE

TAUDTTINm input signals are sampled at the frequency of the operation clock set by TAUDnCMORm.TAUDnCKS[1:0] bits. Therefore, the TAUDTTOUTm output clock cycle has an error of ± 1 operation clock cycle.

(2) Equations

- When rising edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/[(TAUDnCDRm + 1) × 2]
- When falling edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/[(TAUDnCDRm + 1) × 2]
- When falling and rising edge detection is selected:
  TAUDTTOUTm frequency = TAUDTTINm frequency/(TAUDnCDRm + 1)

(3) Block Diagram and General Timing Diagram

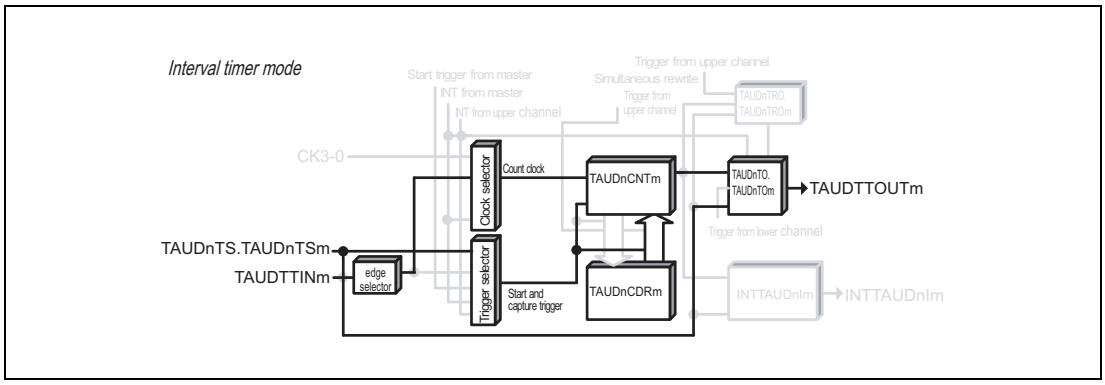


Figure 17.31 Block Diagram of Clock Divide Function

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

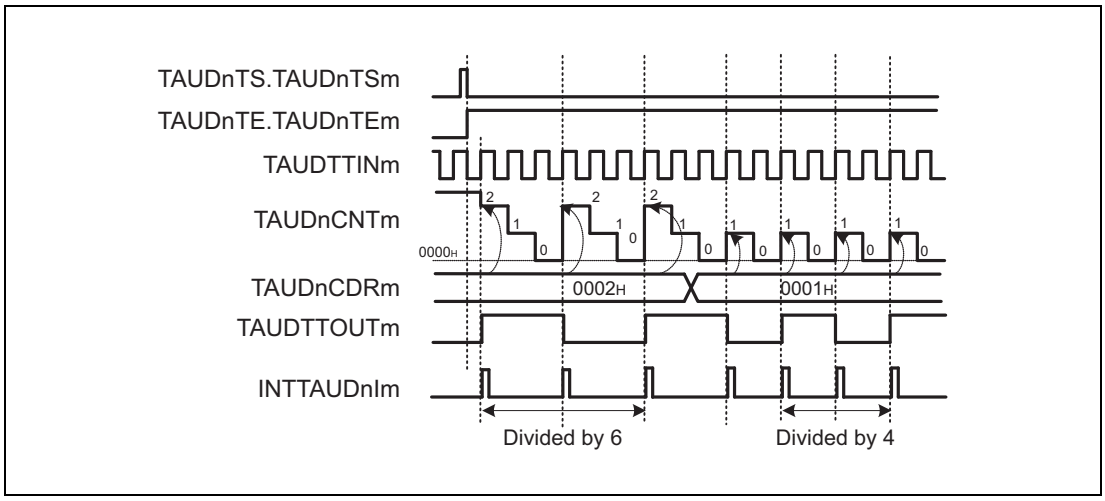


Figure 17.32 General Timing Diagram of Clock Divide Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.55 Contents of TAUDnCMORm Register for Clock Divide Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	01: An effective TAUDTTINm input edge is used as a count clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.56 Contents of TAUDnCMURm Register for Clock Divide Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 17.57 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.58 Simultaneous Rewrite Settings for Clock Divide Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Clock Divide Function

Table 17.59 Operating Procedure for Clock Divide Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.55, Contents of TAUDnCMORm Register for Clock Divide Function**, and **Table 17.56, Contents of TAUDnCMURm Register for Clock Divide Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 17.57, Control Bit Settings in Independent Channel Output Mode 1**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm loads TAUDnCDRm value. If TAUDnCMORm.TAUDnMD0 is set to 1, INTTAUDnIm occurs and TAUDTTOUTm is toggled.
During Operation	The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down each time TAUDTTINm input edge is detected. When the counter reaches 0000_H: • TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm is toggled. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H

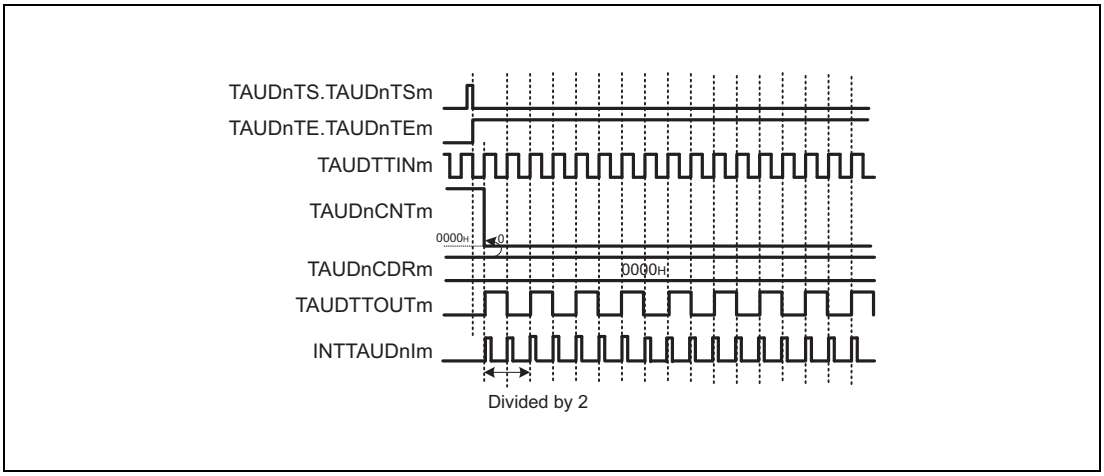


Figure 17.33 TAUDnCDRm = 0000_H, TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B

- If TAUDnCDRm is 0000_H, TAUDnCNTm is always 0000_H.
- INTTAUDnIm is generated every count clock, resulting in TAUDTTOUTm toggling every count clock

Figure 17.33 shows an operation timing example. Actually, there is a delay from TINm detection until TOUTm output because of the delay time of a noise filter or synchronization circuit placed between the TAUDnlm pin and TAUDn.

(b) Restart

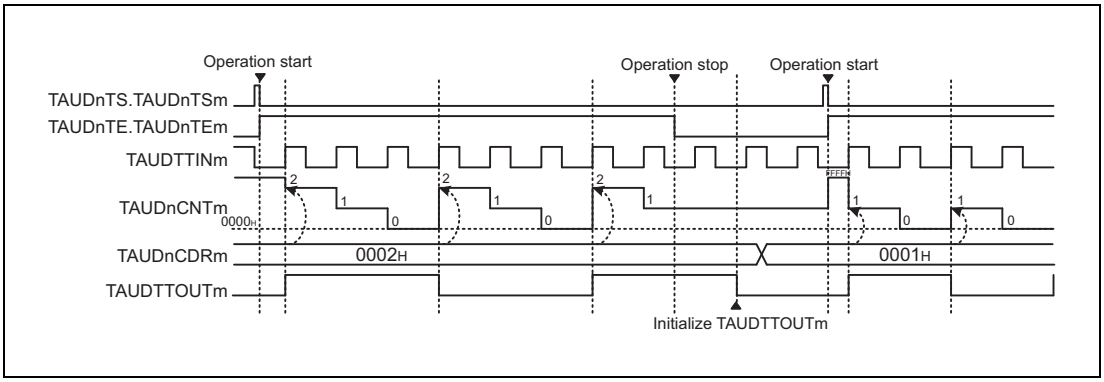


Figure 17.34 Restart (TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

To reset the value of TAUDTTOUTm:

- Set TAUDnTOE.TAUDnTOEm = 0 when the counter is stopped (TAUDnTE.TAUDnTEm = 0).
- Then, write either 0 or 1 to TAUDTO.TAUDnTOm to set the new start value of TAUDTTOUTm.

(c) Forced restart

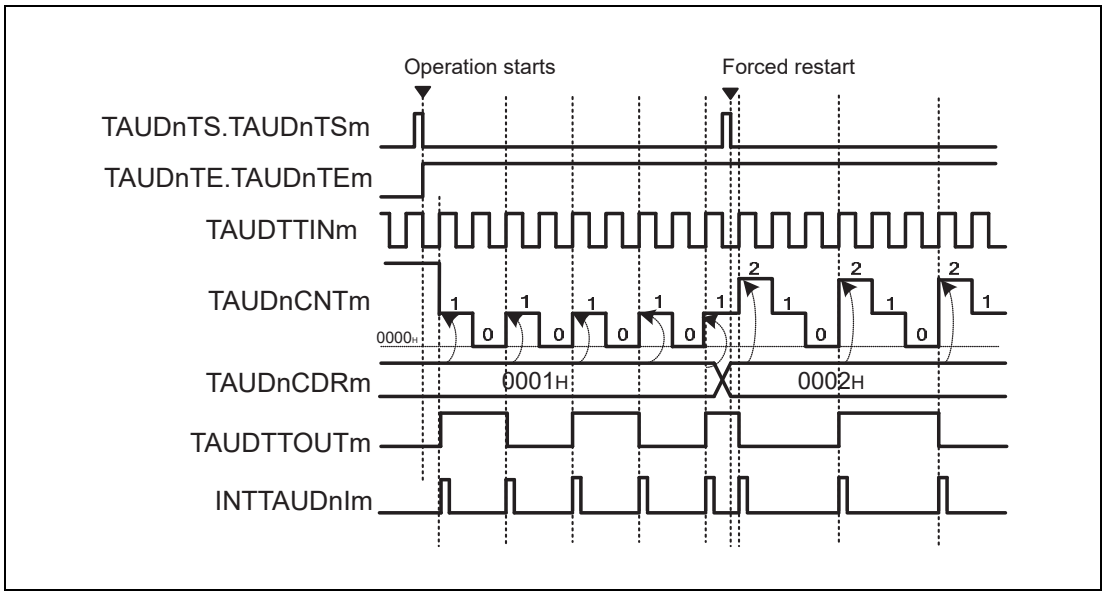


Figure 17.35 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

- The counter can be forcibly restarted (without stopping it first) by setting TAUDnTS.TSm = 1 during operation.
- The value of TAUDnCDRm is written to TAUDnCNTm and the count operation restarts.
- TAUDTTOUTm restarts at the same level as before the forced restart.

17.4.9.4 External Event Count Function

(1) Overview

Summary

This function is used as an event timer, which generates an interrupt (INTTAUDnIm) when a specific number of TAUDTTINm input pulses has occurred.

Prerequisites

- The operating mode should be set to the event count mode (see **Table 17.60, Contents of TAUDnCMORm Register for External Event Count Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When the counter starts, the current value of TAUDnCDRm is loaded into TAUDnCNTm.

When an effective TAUDTTINm input edge is detected, the value of TAUDnCNTm decrements by 1. TAUDnCNTm retains this value until an effective TAUDTTINm input edge is detected or the counter is restarted.

When the effective edge is detected for the (TAUDnCDRm + 1) times, INTTAUDnIm is generated. Then, TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. The counter can also be restarted without stopping it first (forced restart) by setting TAUDnTS.TAUDnTSm to 1 during operation.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

An edge type used as a trigger is specified by TAUDnCMURm.TAUDnTIS[1:0] bits.

- When TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges are counted.
- When TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges are counted.
- When TAUDnCMURm.TAUDnTIS[1:0] = 10_B, both edges are counted.

(2) Equations

Number of effective edges detected before INTTAUDnIm generation = TAUDnCDRm + 1

(3) Block Diagram and General Timing Diagram

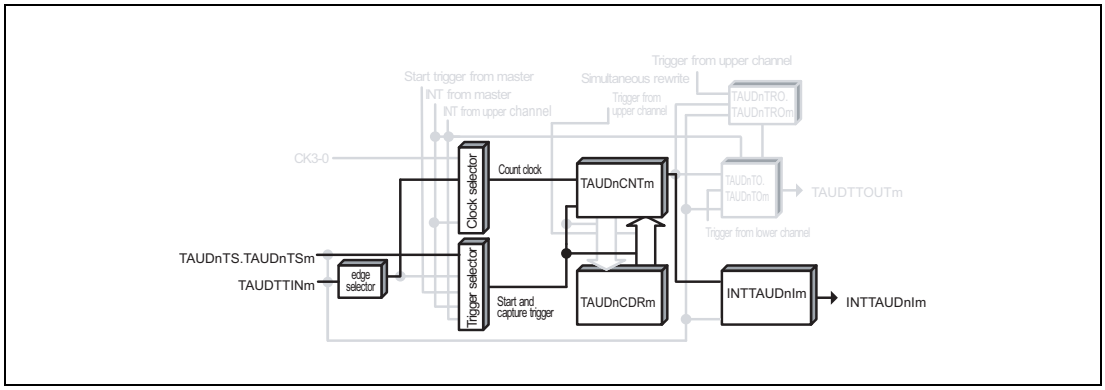


Figure 17.36 Block Diagram of External Event Count Function

The following settings apply to the general timing diagram.

- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

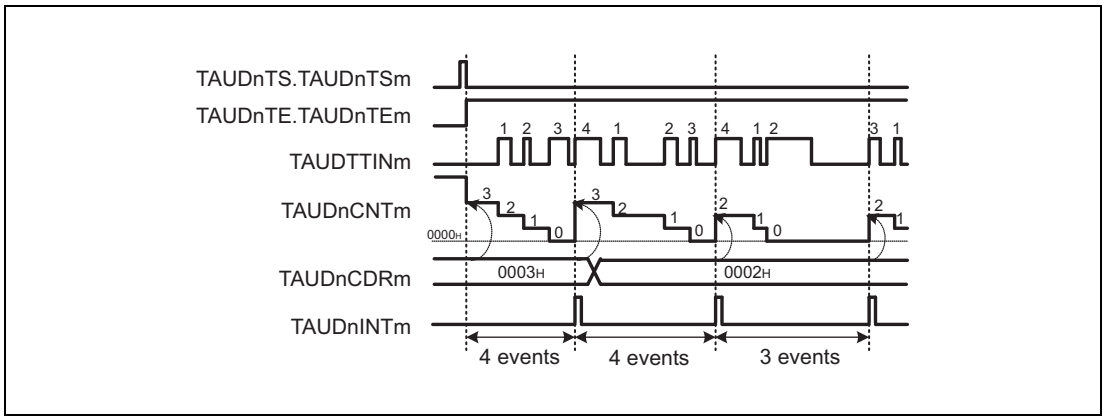


Figure 17.37 General Timing Diagram of External Event Count Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.60 Contents of TAUDnCMORm Register for External Event Count Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	01: An effective TAUDTTINm input edge is used as a count clock.
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.61 Contents of TAUDnCMURm Register for External Event Count Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

The channel output mode is not used by this function. However, it can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.62 Simultaneous Rewrite Settings for External Event Count Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for External Event Count Function

Table 17.63 Operating Procedure for External Event Count Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.60, Contents of TAUDnCMORm Register for External Event Count Function**, and **Table 17.61, Contents of TAUDnCMURm Register for External Event Count Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm loads TAUDnCDRm value and waits for TAUDTTINm input edge detection.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm counts down each time TAUDTTINm input edge is detected. When the counter reaches 0000_H. When effective edges are detected (TAUDnCDRm + 1) times: • TAUDnCDRm value is loaded in TAUDnCNTm and count operation continues. • INTTAUDnIm is generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

(6) Specific Timing Diagrams

(a) TAUDnCDRm = 0000_H

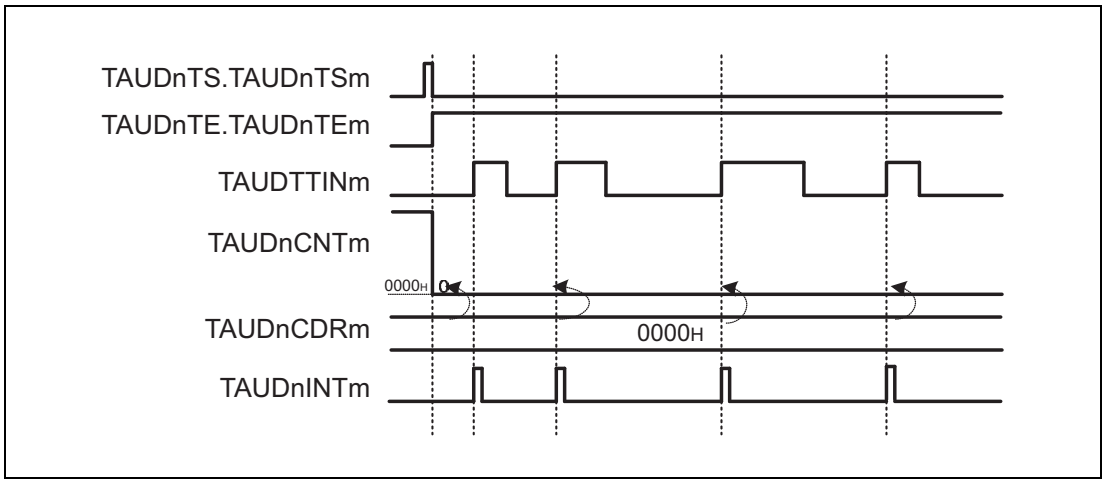


Figure 17.38 TAUDnCDRm = 0000_H, TAUDnCMURm.TAUDnTIS[1:0] = 01_B

- If TAUDnCDRm = 0000_H, 0000_H is loaded into TAUDnCNTm each time an effective TAUDTTINm input edge is detected.
  In other words, INTTAUDnlm occurs each time an effective TAUDTTINm input edge is detected.

(b) Operation stop and restart

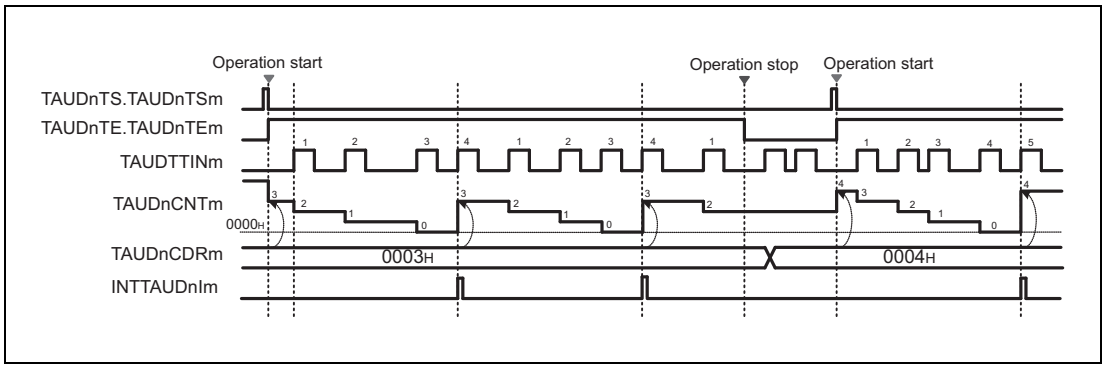


Figure 17.39 Operation Stop and Restart (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value. TAUDTTINm continues and TAUDnCNTm ignores the effective edge.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm loads the TAUDnCDRm value and restarts count operation.

(c) Forced restart

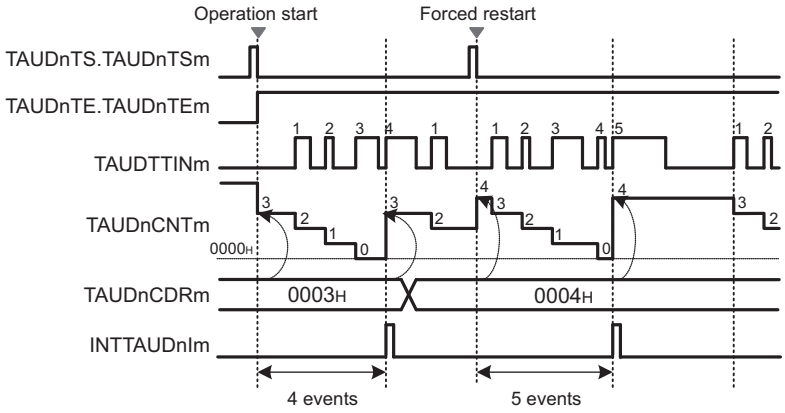


Figure 17.40 Forced Restart Operation (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)

Once a forced restart is made, the changed value of TAUDnCDRm is applied to TAUDnCNTm.

- The counter can be restarted without making a stop by setting TAUDnTS.TAUDnTSm to 1 during operation.
- The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter awaits the next effective TAUDTTINm input edge.

17.4.9.5 Delay Count Function

(1) Overview

Summary

This function generates interrupts (INTTAUDnIm), which have a defined delay to the TAUDTTINm input signal. TAUDTTINm input signal pulses that occur within the delay period are ignored.

Prerequisites

- The operating mode should be set to one-count mode. See **Table 17.64, Contents of TAUDnCMORm Register for Delay Count Function**.
- TAUDTTOUTm is not used with this function.
- Trigger detection should be disabled during counting (TAUDnCMORn.TAUDnMD0 = 0).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input start edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value.

When the counter reaches 0000_H, an interrupt is generated. The counter returns to $FFFF_H$ and awaits the next effective TAUDTTINm input edge.

When the counter is counting down, further TAUDTTINm input signals are ignored, i.e., the counter does not reset.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The type of edge used as a trigger is specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, rising and falling edges trigger the counter.

(2) Equations

Delay between TAUDTTINm and INTTAUDnIm = count clock cycle × (TAUDnCDRm + 1)

(3) Block Diagram and General Timing Diagram

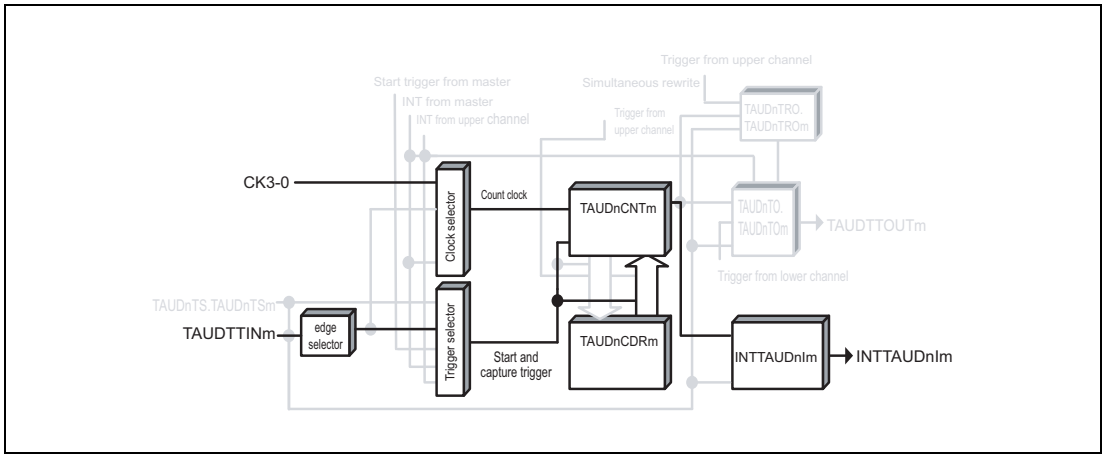


Figure 17.41 Block Diagram of Delay Count Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

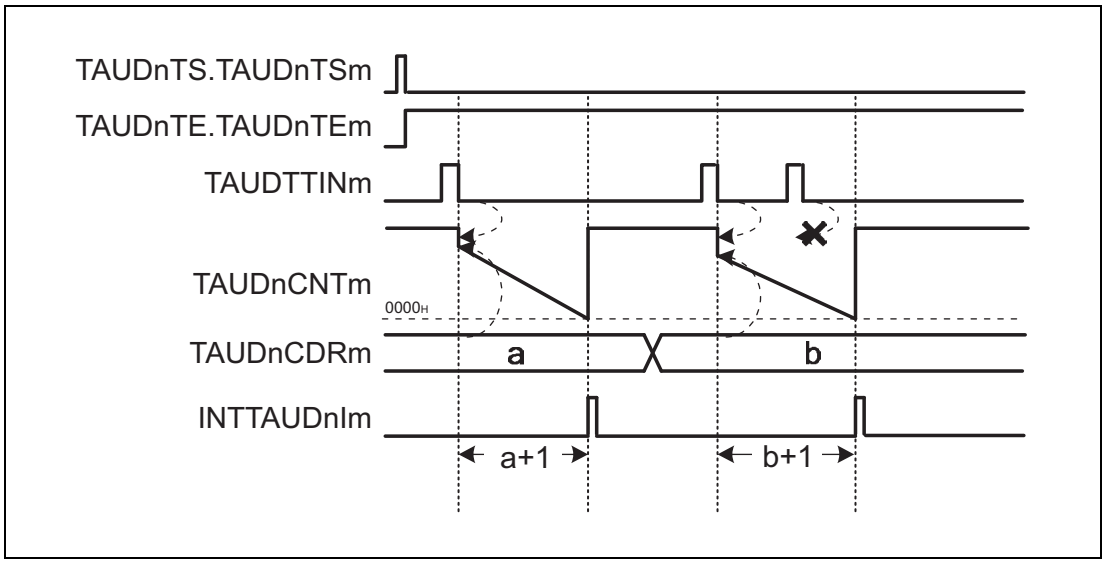


Figure 17.42 General Timing Diagram of Delay Count Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.64 Contents of TAUDnCMORm Register for Delay Count Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	0: Disables a start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.65 Contents of TAUDnCMURm Register for Delay Count Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.66 Simultaneous Rewrite Settings for Delay Count Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for Delay Count Function

Table 17.67 Operating Procedure for Delay Count Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.64, Contents of TAUDnCMORm Register for Delay Count Function**, and **Table 17.65, Contents of TAUDnCMURm Register for Delay Count Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, the TAUDnCDRm value is loaded in TAUDnCNTm.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at all times.	TAUDnCNTm counts down. When the counter reaches 0000_H. INTTAUDnIm is generated. TAUDnCNTm stops counting, returns $FFFF_H$, and waits for a trigger. If a trigger occurs while TAUDnCNTm is counting, the trigger is ignored. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its value.

Restart

17.4.9.6 One-Pulse Output Function

(1) Overview

Summary

This function generates an interrupt (INTTAUDnIm) when an effective TAUDTTINm input edge is detected and at a defined interval afterward. TAUDTTINm input signal pulses that occur within the defined interval are ignored. When an interrupt is generated, the TAUDTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode should be set to pulse one-count mode (see **Table 17.68, Contents of TAUDnCMORm Register for One-Pulse Output Function**).
- The channel output mode should be set to independent channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- Trigger detection should be disabled during counting (TAUDnCMORn.TAUDnMD0 = 0).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value. An interrupt is generated and TAUDTTOUTm toggles.

When the counter reaches 0001_H, an interrupt is generated and TAUDTTOUTm is set to the inactive level. The counter stops at 0000_H and awaits the next effective TAUDTTINm input edge.

When the counter is counting down, further TAUDTTINm input signals are ignored, i.e., the counter does not reset.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The type of edge used as a trigger is specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 00_B, falling edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 01_B, rising edges trigger the counter.
- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, rising and falling edges trigger the counter.

(2) Equations

Interval between TAUDTTINm and INTTAUDnIm = TAUDTTOUTm (timer output) width = count clock cycle × TAUDnCDRm

(3) Block Diagram and General Timing Diagram

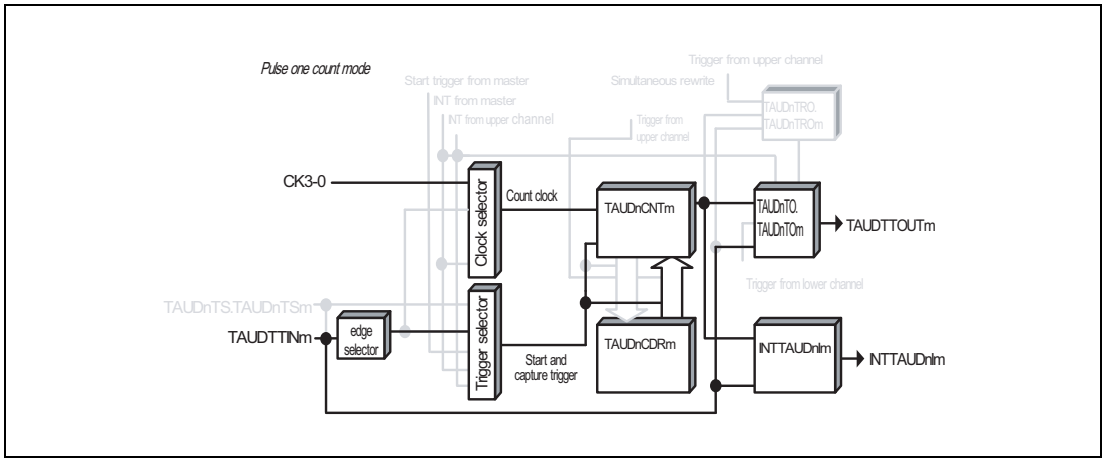


Figure 17.43 Block Diagram of One-Pulse Output Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

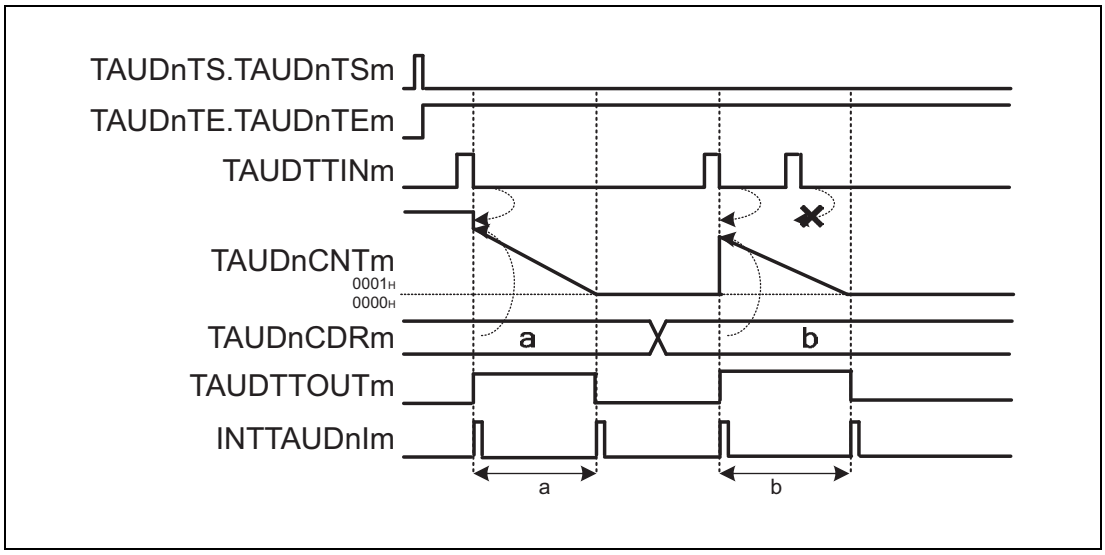


Figure 17.44 General Timing Diagram of One-Pulse Output Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.68 Contents of TAUDnCMORm Register for One-Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	0: Disables a start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.69 Contents of TAUDnCMURm Register for One-Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

Table 17.70 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode controlled by software.
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Set/reset mode
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

NOTE

The channel output mode can also be set to channel output mode controlled by software by setting TAUDnTOE.TAUDnTOEm = 0. TAUDTTOUTm can then be controlled independently of the interrupts. For details, see **Table 17.43, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the One-Pulse Output Function. Therefore, these registers should be set to 0.

Table 17.71 Simultaneous Rewrite Settings for One-Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for One-Pulse Output Function

Table 17.72 Operating Procedure for One-Pulse Output Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.68, Contents of TAUDnCMORm Register for One-Pulse Output Function**, and **Table 17.69, Contents of TAUDnCMURm Register for One-Pulse Output Function**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 17.70, Control Bit Settings in Independent Channel Output Mode 2**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, TAUDnCNTm loads the TAUDnCDRm value.
During Operation	The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at all times.	INTTAUDnIm is generated when TAUDnCNTm starts and TAUDTTOUTm is set to its active level. TAUDnCNTm counts down. When the counter reaches 0001_H: • INTTAUDnIm is generated. • TAUDTTOUTm is set to its inactive level. TAUDnCNTm stops counting and waits for a trigger. If a trigger occurs while TAUDnCNTm is counting, the trigger is ignored.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

17.4.9.7 TAUDTTINm Input Pulse Interval Measurement Function

(1) Overview

Summary

This function captures the count value and uses this value and the overflow bit TAUDnCSRm.TAUDnOVF to measure the interval of the TAUDTTINm input signal.

Prerequisites

- The operating mode should be set to capture mode. See **Table 17.74, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function**.
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected, the value of TAUDnCNTm is captured, transferred to TAUDnCDRm, and an interrupt INTTAUDnIm is generated. The counter resets to 0000_H and subsequently continues to operate.

If the counter reaches $FFFF_H$ before an effective TAUDTTINm edge is detected, it overflow. The counter is reset to 0000_H and subsequently continues to operate. The values transferred to TAUDnCDRm and TAUDnCSRm.TAUDnOVF respectively depend on the values of bits TAUDnCMORm.TAUDnCOS[1:0].

Table 17.73 Effects of Overflow

TAUDnCMORm.COS[1:0]	When Overflow Occurs		When an Effective TAUDTTINm Input is Detected	
	TAUDnCDRm	TAUDnCSRm.TAUDnOVF	TAUDnCDRm, TAUDnCNTm	TAUDnCSRm.TAUDnOVF
00	Unchanged	0	TAUDnCNTm loaded into TAUDnCDRm	1
01		1		
10	Set to $FFFF_H$	0	TAUDnCNTm set to 0, TAUDnCDRm unchanged	Unchanged
11		1		

When TAUDnCMORm.TAUDnCOS[0] = 1, the overflow bit (TAUDnCSRm.TAUDnOVF) can be cleared only by setting TAUDnCSCm.TAUDnCLOV = 1.

The combination of the value of TAUDnCDRm and TAUDnCSRm.TAUDnOVF can be used to deduce the interval of the TAUDTTINm signal. However, if an overflow occurs multiple times before an effective TAUDTTINm input is detected, the overflow bit TAUDnCSRm.TAUDnOVF cannot indicate the occurrence of multiple overflows.

The function can be stopped by setting TAUDnTT.TAUDnTTm = 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm stops but retains its value. While the function is stopped, effective TAUDTTINm input edge detection and TAUDnCNTm capture are not performed.

The counter is reset to 0000_H and subsequently continues to operate.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

NOTE

When TAUDnCMORm.TAUDnCOS[1:0] = 10_B or 11_B, the value of TAUDnCNTm is not loaded into TAUDnCDRm when the first effective TAUDTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUDTTINm input pulse interval = count clock cycle × [(TAUDnCSRm.TAUDnOV F × ($FFFF_H$ + 1)) + TAUDnCDRm capture value + 1]

(3) Block Diagram and General Timing Diagram

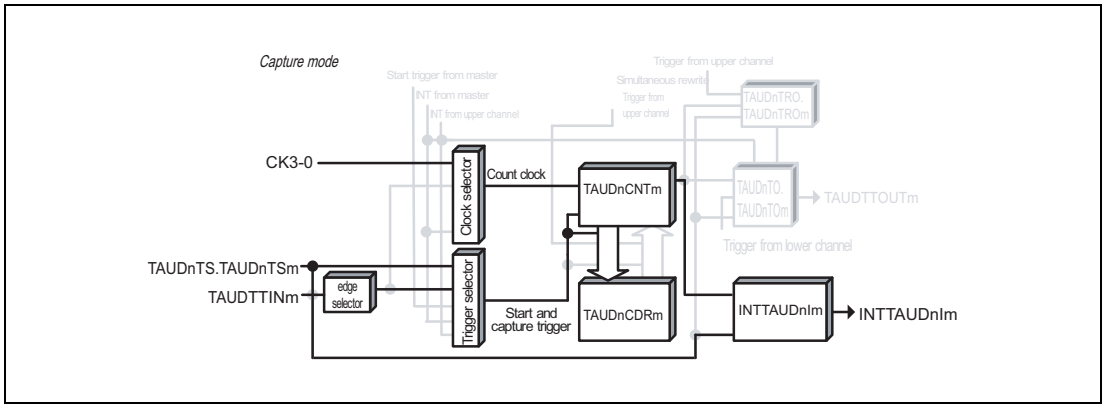


Figure 17.45 Block Diagram of TAUDTTINm Input Pulse Interval Measurement Function

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation (TAUDnCMORm.TAUDnMD0 = 0)
- Falling edge detection (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)
- When an effective TAUDTTINm input is detected after an overflow, TAUDnCDRm is changed and TAUDnCSRm.TAUDnOVF is set to 1 (TAUDnCMORm.TAUDnCOS[1:0] = 00_B)

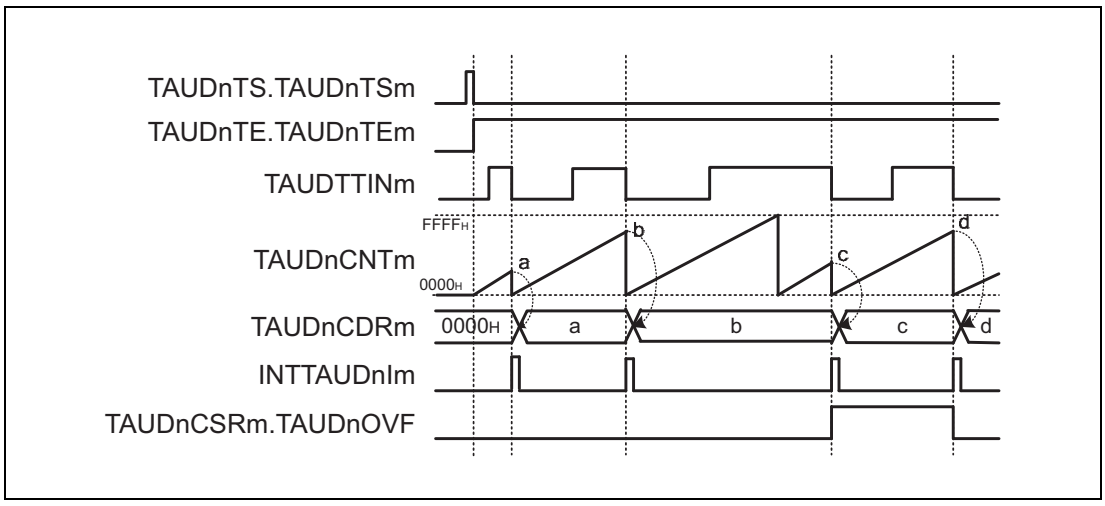


Figure 17.46 General Timing Diagram of TAUDTTINm Input Pulse Interval Measurement Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.74 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is the external capture trigger.
7, 6	TAUDnCOS [1:0]	See **Table 17.73, Effects of Overflow**.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.75 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm input pulse interval measurement function. Therefore, these registers should be set to 0.

Table 17.76 Simultaneous Rewrite Settings for TAUDTTINm Input Pulse Interval Measurement Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Pulse Interval Measurement Function

Table 17.77 Operating Procedure for TAUDTTINm Input Pulse Interval Measurement Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.74, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function**, and **Table 17.75, Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Measurement Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm is cleared to 0000_H. INTTAUDnIm is generated when TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	Detection of TAUDTTINm edge The TAUDnCMURm.TAUDnTIS[1:0] bits can be changed at any time. The TAUDnCDRm and TAUDnCSRm registers can be read at any time. TAUDnCSCm.TAUDnCLOV can be written to 1. (TAUDnCSRm.TAUDnOVF bit is cleared to 0.)	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its value to TAUDnCDRm, and returns to 0000_H. • INTTAUDnIm is then generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and both it and TAUDnCSRm.TAUDnOVF retain their current values.

Restart

(6) Specific Timing Diagrams: Overflow Operation

(a) TAUDnCMORm.TAUDnCOS[1:0] = 00_B

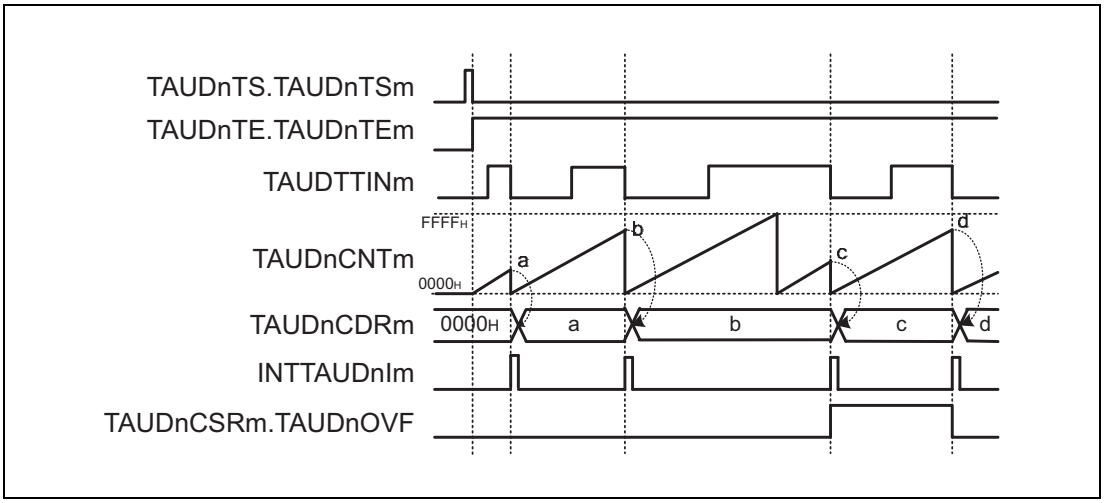


Figure 17.47 TAUDnCMORm.TAUDnCOS[1:0] = 00_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm and TAUDnCSRm.TAUDnOVF is set to1.
- Upon detection of the next effective TAUDTTINm input edge with no overflow occurring, TAUDnCSRm.TAUDnOVF is cleared to 0.

(b) TAUDnCMORm.TAUDnCOS[1:0] = 01_B

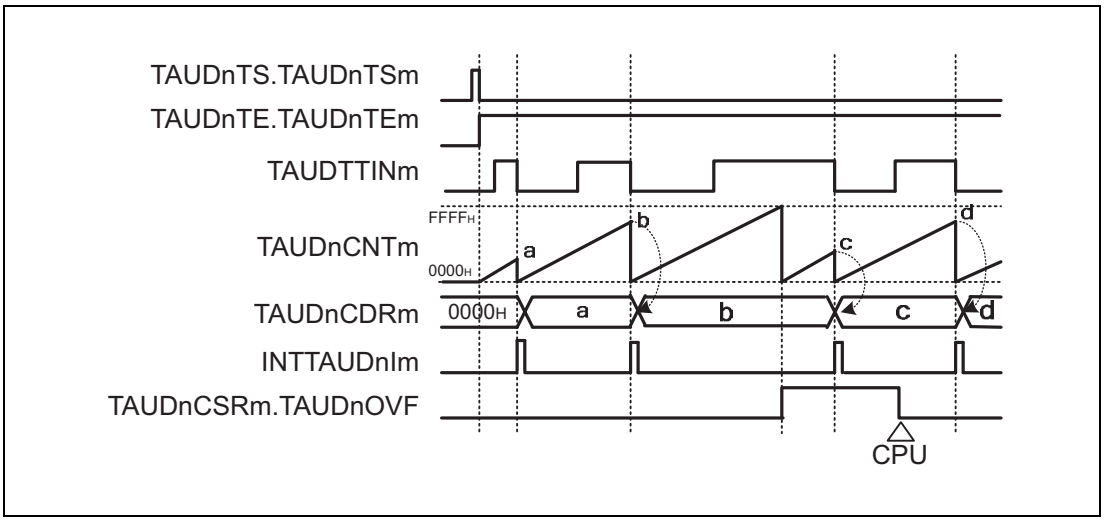


Figure 17.48 TAUDnCMORm.TAUDnCOS[1:0] = 01_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF is set to1.

- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm.
- TAUDnCSRm.TAUDnOVF is only cleared by a CPU command (by setting TAUDnCSCm.TAUDnCLOV bit to 1).

(c) TAUDnCMORm.TAUDnCOS[1:0] = 10_B

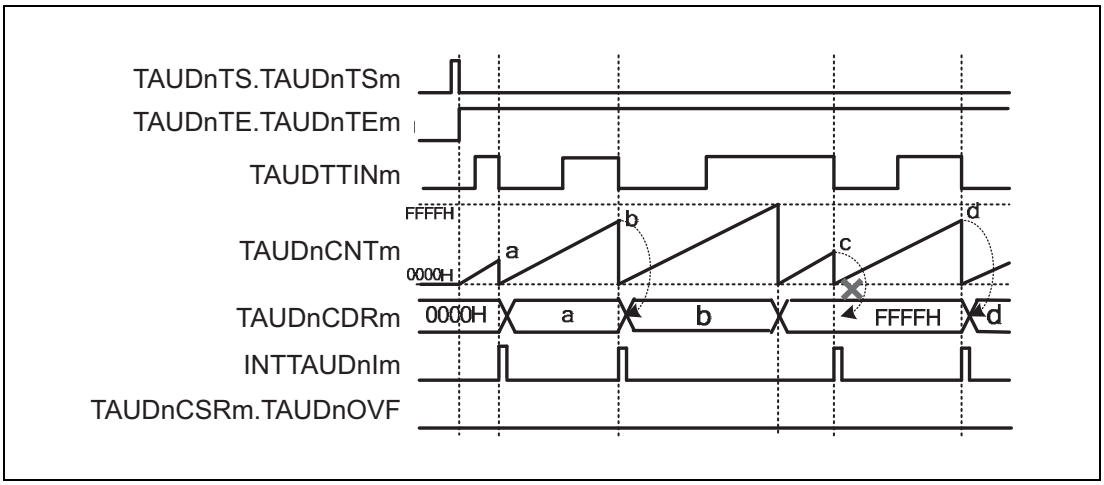


Figure 17.49 TAUDnCMORm.TAUDnCOS[1:0] = 10_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm is reset to 0, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.

(d) TAUDnCMORm.TAUDnCOS[1:0] = 11_B

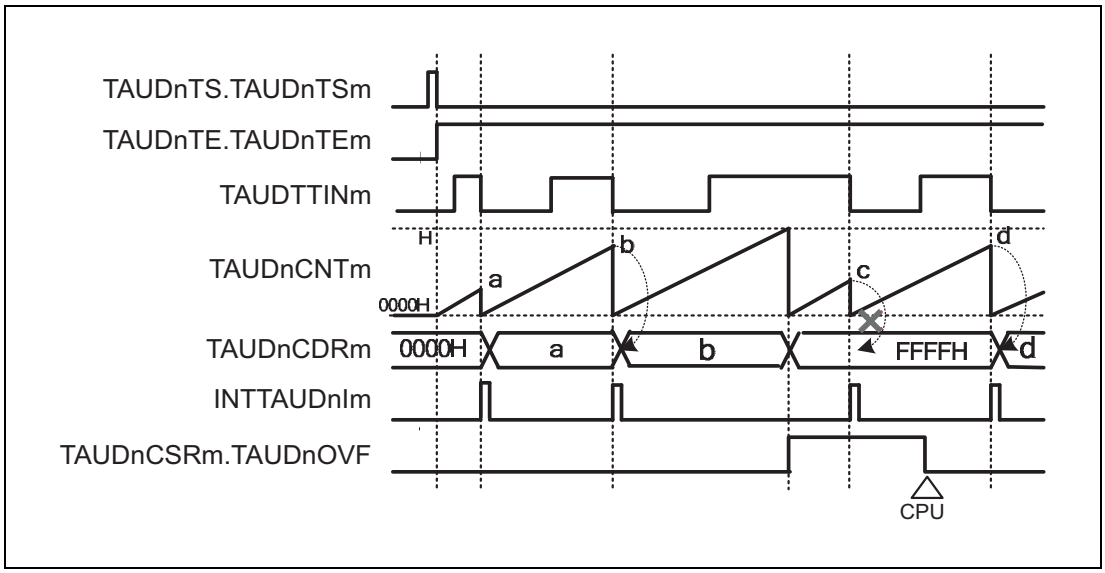


Figure 17.50 TAUDnCMORm.TAUDnCOS[1:0] = 11_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm is reset to 0, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.
- TAUDnCSRm.TAUDnOVF is cleared by setting TAUDnCSCm.TAUDnCLOV to 1.

17.4.9.8 TAUDTTINm Input Signal Width Measurement Function

(1) Overview

Summary

This function measures the width of a TAUDTTINm signal, by starting the count at one edge of TAUDTTINm and capturing the count value at the other edge.

Prerequisites

- The operating mode should be set to capture and one-count mode. See **Table 17.79, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function**.
- TAUDTTOUTm is not used with this function.
- TAUDnCMORm.TAUDnMD0 should be set to 0.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When an effective TAUDTTINm start edge is detected, the counter TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm stop edge is detected, the value of TAUDnCNTm is captured, transferred to TAUDnCDRm, and an interrupt INTTAUDnIm is generated. The counter retains its value (TAUDnCDRm +1) and awaits the next effective TAUDTTINm input start edge.

If the counter reaches $FFFF_H$ before an effective TAUDTTINm stop edge is detected, it overflows. The counter is reset to 0000_H and subsequently continues to operate. The values transferred to TAUDnCDRm and TAUDnCSRm.TAUDnOVF respectively depend on the values of bits TAUDnCMORm.TAUDnCOS[1:0].

Table 17.78 Effects or Overflow

TAUDnCMORm.COS[1:0]	When Overflow Occurs		When an Effective TAUDTTINm Input Stop Edge is Detected	
	TAUDnCDRm	TAUDnCSRm.TAUDnOVF	TAUDnCDRm, TAUDnCNTm	TAUDnCSRm.TAUDnOVF
00	Unchanged	0	TAUDnCNTm loaded into TAUDnCDRm	1
01		1		
10	Set to $FFFF_H$	0	TAUDnCNTm stops counting TAUDnCDRm unchanged	Unchanged
11		1		

When TAUDnCMORm.TAUDnCOS[0] = 1, overflow bit TAUDnCSRm.TAUDnOVF can be cleared only by setting TAUDnCSCm.TAUDnCLOV to 1.

The combination of the value of TAUDnCDRm and TAUDnCSRm.TAUDnOVF can be used to deduce the width of the TAUDTTINm signal. However, if an overflow occurs multiple times before an effective TAUDTTINm input is detected, overflow bit TAUDnCSRm.TAUDnOVF cannot indicate the occurrence of multiple overflows.

This function cannot be forcibly restarted.

NOTE

When TAUDnCMORm.COS[1:0] = 10_B or 11_B, the value of TAUDnCNTm is not loaded to TAUDnCDRm when the first effective TAUDTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUDTTINm input signal width = count clock cycle × [(TAUDnCSRm.TAUDnOVF × ($FFFF_H$ + 1)) + TAUDnCDRm capture value + 1]

(3) Block Diagram and General Timing Diagram

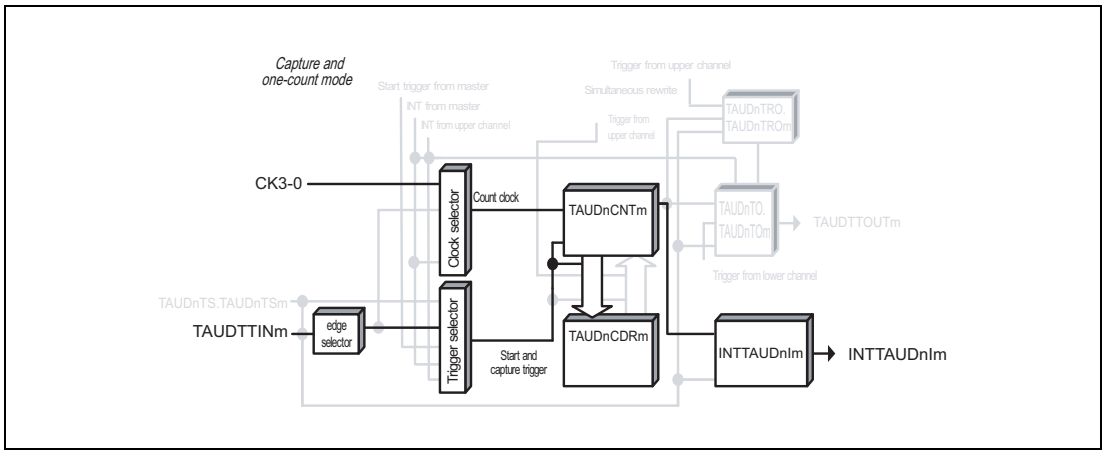


Figure 17.51 Block Diagram of TAUDTTINm Input Signal Width Measurement Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)
- When an effective TAUDTTINm input is detected after an overflow, TAUDnCDRm is changed and TAUDnCSRm.TAUDnOVF is set to 1. (TAUDnCMORm.TAUDnCOS[1:0] = 00_B)

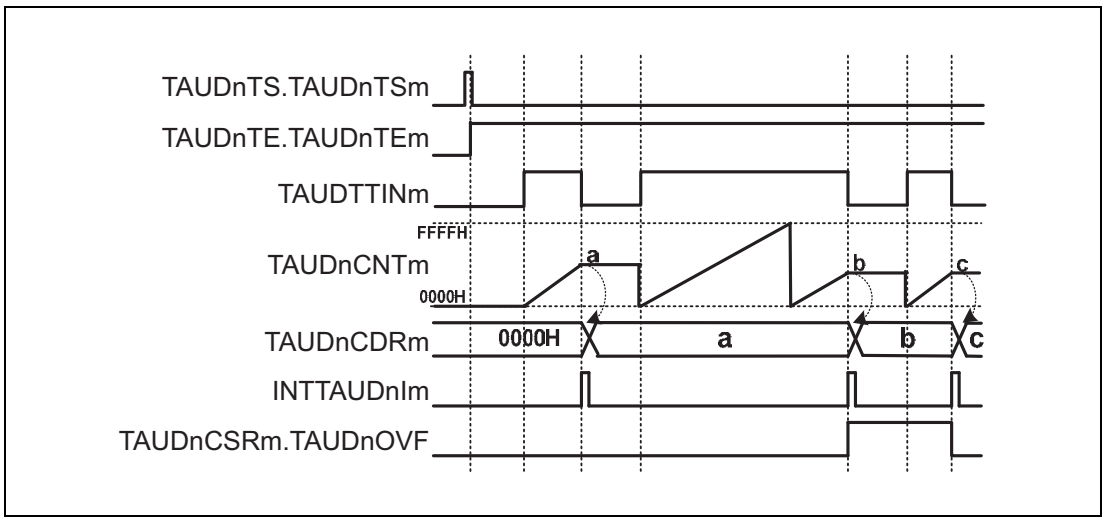


Figure 17.52 General Timing Diagram of TAUDTTINm Input Signal Width Measurement Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.79 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS [1:0]	See **Table 17.78, Effects or Overflow**.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0110: Capture and one-count mode
0	TAUDnMD0	0: Disables the start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.80 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the TAUDTTINm input signal width measurement function. Therefore, these registers should be set to 0.

Table 17.81 Simultaneous Rewrite Settings for TAUDTTINm Input Signal Width Measurement Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Signal Width Measurement Function

Table 17.82 Operating Procedure for TAUDTTINm Input Signal Width Measurement Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.79, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function**, and **Table 17.80, Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Measurement Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a TAUDTTINm start edge is detected, TAUDnCNTm starts to count up.
During Operation	Detection of TAUDTTINm edge TAUDnCDRm, TAUDnCNTm, and TAUDnCSRm registers can be read at any time. The TAUDnCSCm.TAUDnCLOV bit can be set to 1.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its value to TAUDnCDRm, and retains its value. INTTAUDnIm is then generated. Counting stops at the “value that transferred to TAUDnCDRm + 1” and TUAnCNTm waits for detection of the TAUDTTINm start edge. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and both it and TAUDnCSRm.TAUDnOVF retain their current values.

Restart

(6) Specific Timing Diagrams: Overflow Operation

(a) TAUDnCMORm.TAUDnCOS[1:0] = 00_B

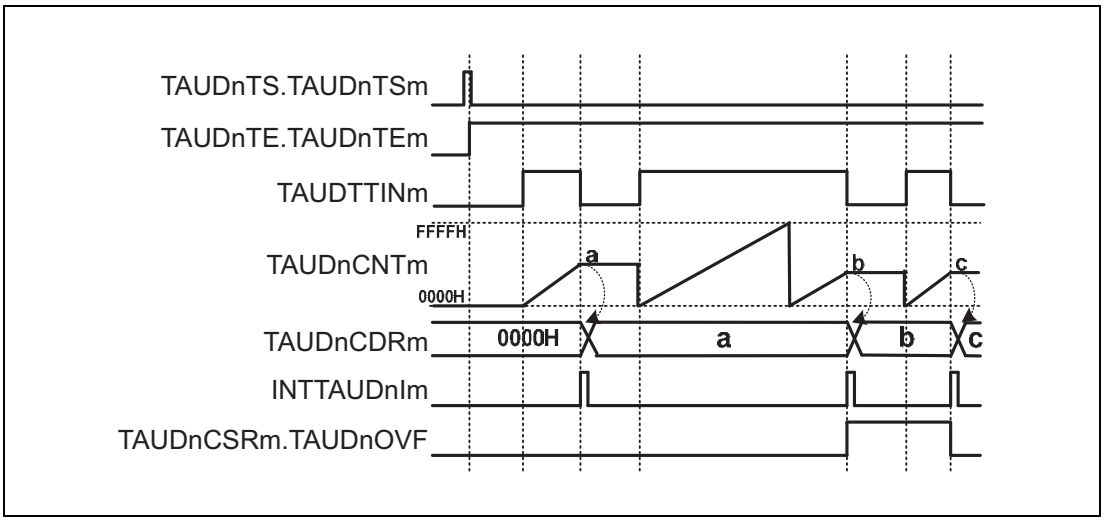


Figure 17.53 TAUDnCMORm.TAUDnCOS[1:0] = 00_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge with no overflow occurring, TAUDnCSRm.TAUDnOVF is cleared to 0.

(b) TAUDnCMORm.TAUDnCOS[1:0] = 01_B

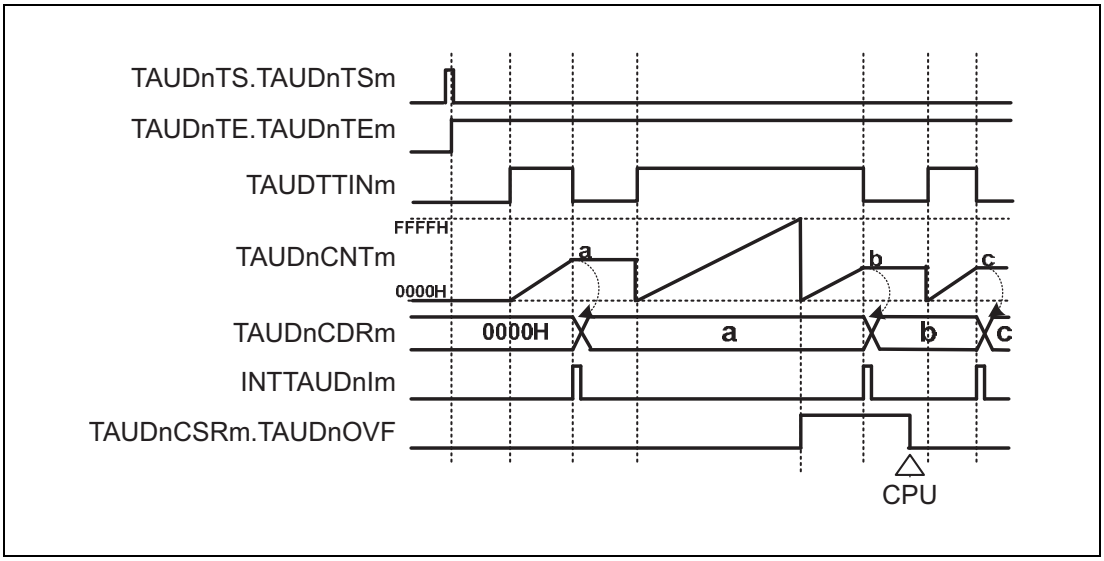


Figure 17.54 TAUDnCMORm.TAUDnCOS[1:0] = 01_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUDnCDRm remains unchanged and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, the value of TAUDnCNTm is loaded into TAUDnCDRm.
- TAUDnCSRm.TAUDnOVF is only cleared by a CPU command. (by setting TAUDnCSCm.TAUDnCLOV bit to 1.)

(c) TAUDnCMORm.TAUDnCOS[1:0] = 10_B

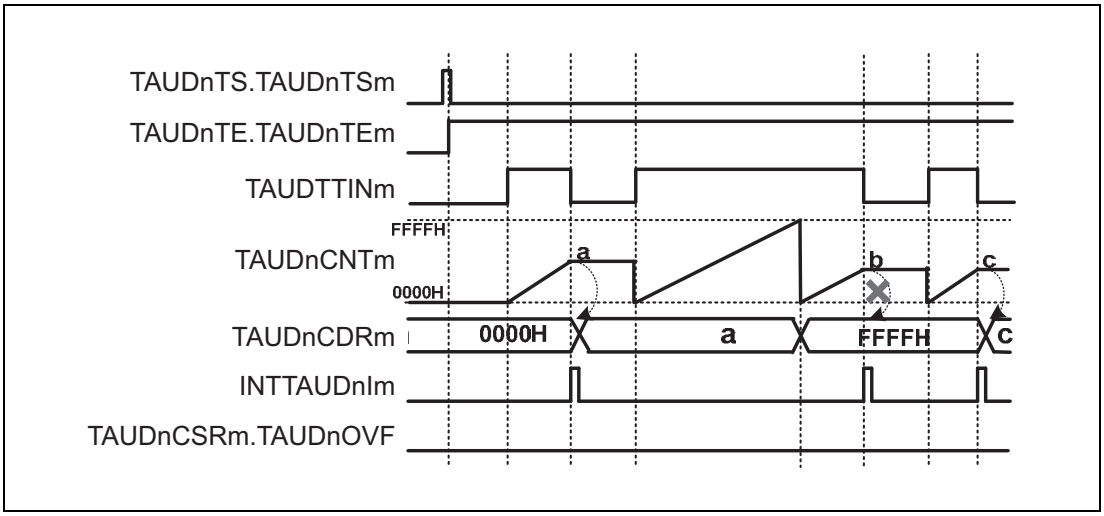


Figure 17.55 TAUDnCMORm.TAUDnCOS[1:0] = 10_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF remains = 0.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm stops counting, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.

(d) TAUDnCMORm.TAUDnCOS[1:0] = 11_B

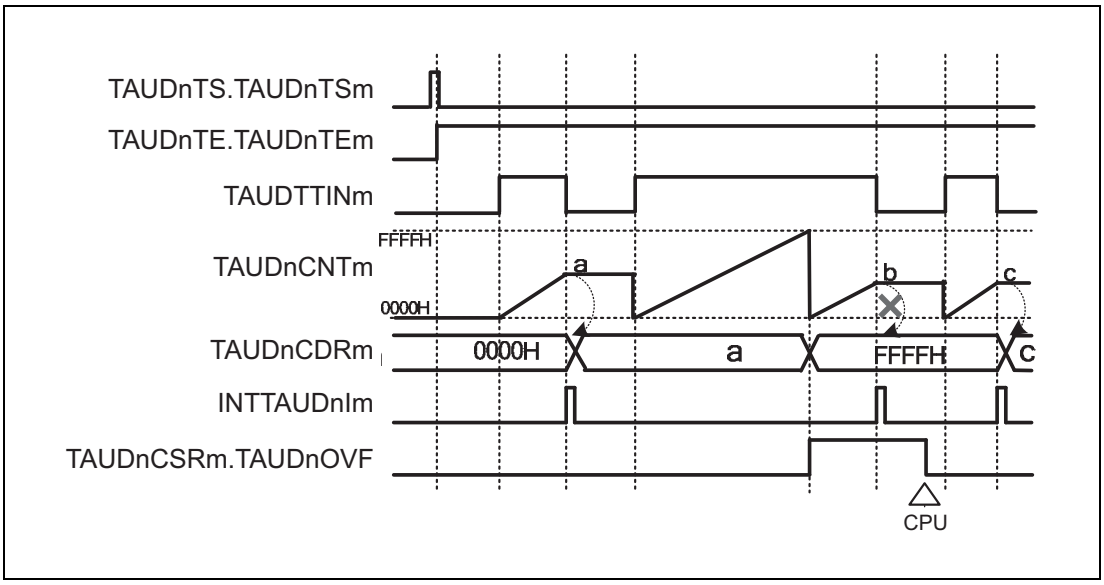


Figure 17.56 TAUDnCMORm.TAUDnCOS[1:0] = 11_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B

- When an overflow occurs, TAUDnCDRm is set to $FFFF_H$ and TAUDnCSRm.TAUDnOVF is set to 1.
- Upon detection of the next effective TAUDTTINm input edge, TAUDnCNTm stops counting, but TAUDnCDRm and TAUDnCSRm.TAUDnOVF remain unchanged.
- Thus, the next effective TAUDTTINm input edge after the overflow is ignored.
- TAUDnCSRm.TAUDnOVF is cleared by setting TAUDnCSCm.TAUDnCLOV to 1.

17.4.9.9 TAUDTTINm Input Position Detection Function

(1) Overview

Summary

This function measures the input signal duration by capturing the count value at the effective edge of TAUDTTINm.

Prerequisites

- The operating mode should be set to count capture mode (see **Table 17.83, Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter starts counting from 0000_H. When an effective TAUDTTINm input edge is detected, the current value of TAUDnCNTm is loaded into TAUDnCDRm and an interrupt (INTTAUDnlm) is generated. The count operation continues.

When the counter reaches $FFFF_H$, the counter restarts to count from 0000_H.

NOTE

The input TAUDTTINm is sampled at the frequency of the operation clock, specified by the TAUDnCMORm.TAUDnCKS[1:0] bits. As a result, the output cycle of TAUDTTOUTm has an error of ± 1 operation clock cycle.

Conditions

If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt does not occur at the beginning of operation or after restart. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnlm Generation when Counter Starts or Restarts**.

(2) Equations

Functional duration at a TAUDTTINm input pulse =

count clock cycle × (TAUDnCDRm capture value + 1)

(3) Block Diagram and General Timing Diagram

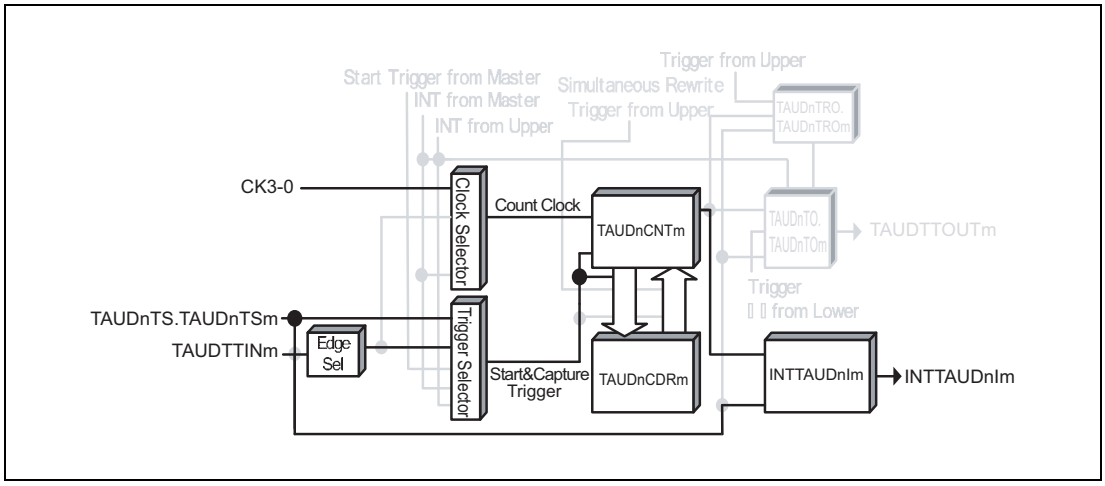


Figure 17.57 Block Diagram of TAUDTTINm Input Position Detection Function

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

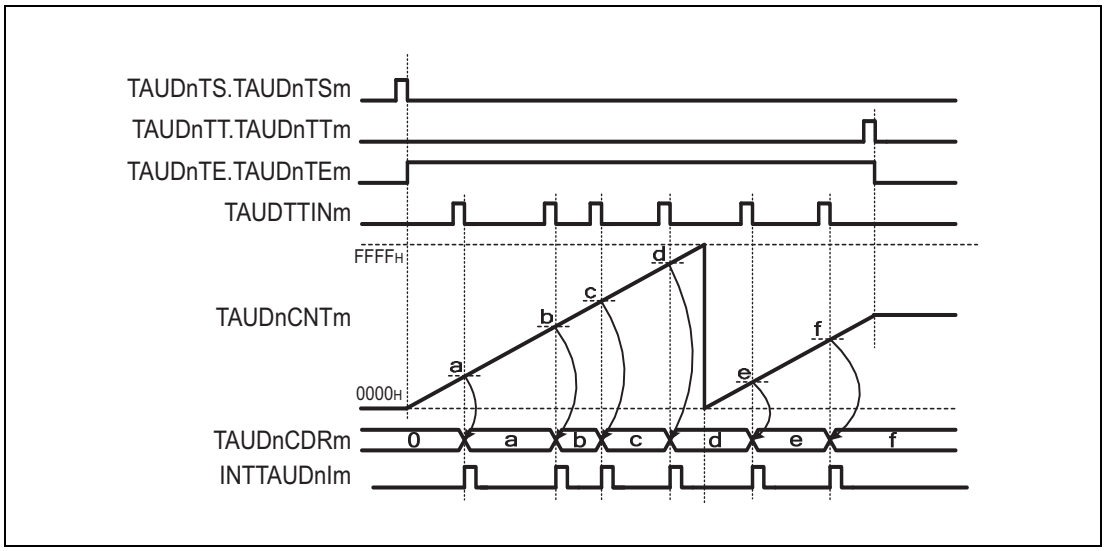


Figure 17.58 General Timing Diagram of TAUDTTINm Input Position Detection Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.83 Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as an external capture trigger.
7, 6	TAUDnCOS [1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1011: Count capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.84 Contents of TAUDnCMURm Register for TAUDTTINm Input Position Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

The channel output mode is not used by this function. However, it can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.85 Simultaneous Rewrite Settings for TAUDTTINm Input Position Detection Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Position Detection Function

Table 17.86 Operating Procedure for TAUDTTINm Input Position Detection Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.83, Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function**, and **Table 17.84, Contents of TAUDnCMURm Register for TAUDTTINm Input Position Detection Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	The TAUDnCMURm.TAUDnTIS[1:0] bits can be changed at any time. The TAUDnCDRm and TAUDnCSRm registers can be read at any time.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm edge is detected: • TAUDnCNTm transfers (captures) its own value to TAUDnCDRm. • INTTAUDnIm occurs. • The counter is not cleared to 0000_H and TAUDnCNTm continues counting. Afterwards, this procedure is repeated. If TAUDnCNTm reaches $FFFF_H$, the counter restarts from 0000_H.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm retains their current values.

Restart

(6) Specific Timing Diagrams

(a) Operation stop and restart

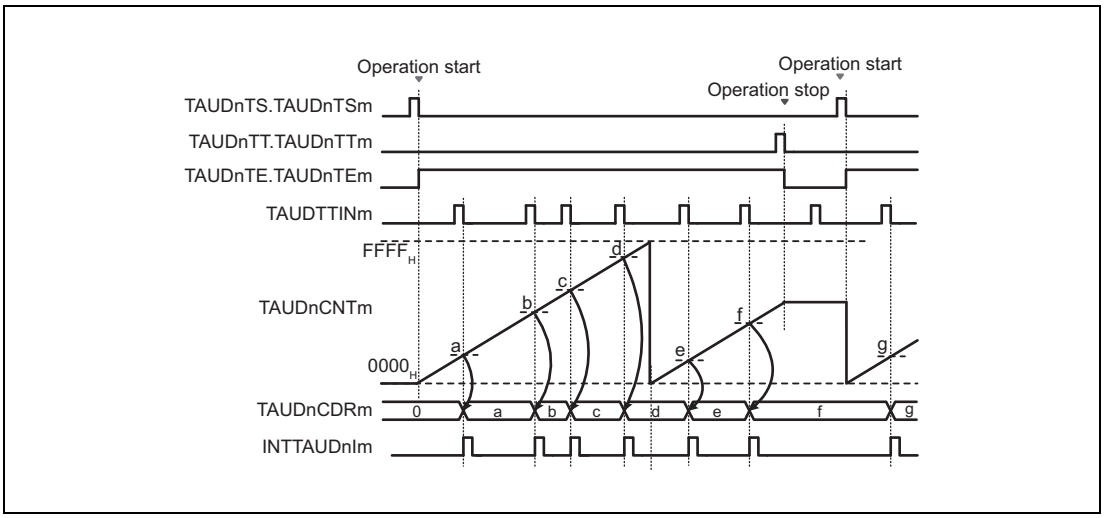


Figure 17.59 Operation Stop and Restart (TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

- The counter can stop operating by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value.
- If the counter stops operating, effective TAUDTTINm input edges are ignored.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm restarts to count from 0000_H.

17.4.9.10 TAUDTTINm Input Period Count Detection Function

(1) Overview

Summary

This function measures the cumulative width of a TAUDTTINm input signal.

Prerequisites

- The operating mode should be set to capture and gate count mode (see **Table 17.87, Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The counter awaits an effective TAUDTTINm input edge.

When an effective TAUDTTINm input start edge is detected, the counter starts to count from 0000_H.

When an effective TAUDTTINm input stop edge is detected, the current TAUDnCNTm value is loaded into TAUDnCDRm and an interrupt (INTTAUDnIm) is generated. The counter stops and retains its value (TAUDnCDRm + 1) until the next effective TAUDTTINm input start edge is detected.

When the next effective TAUDTTINm input start edge is detected, the counter restarts to count from the value retained when stopped.

If the counter reaches $FFFF_H$, the counter restarts to count from 0000_H.

NOTE

TAUDTTINm input signal is sampled at the frequency of an operation clock set by the TAUDnCMORm.TAUDnCKS[1:0] bits.
As this function is to measure the TAUDTTINm input signal width, setting TAUDnTS.TAUDnTSm to 1 is disabled while TAUDnTE.TAUDnTEm = 1.

Conditions

The effective start and stop edges are specified by the TAUDnCMURm.TAUDnTIS[1:0] bits:

- If TAUDnCMURm.TAUDnTIS[1:0] = 10_B, the TAUDTTINm input low period is measured. The start trigger is a falling edge and the stop trigger is a rising edge.
- If TAUDnCMURm.TAUDnTIS[1:0] = 11_B, the TAUDTTINm input high period is measured. The start trigger is a rising edge and the stop trigger is a falling edge.

(2) Equations

Cumulative TAUDTTINm input width = count clock cycle × (TAUDnCDRm capture value + 1)

(3) Block Diagram and General Timing Diagram

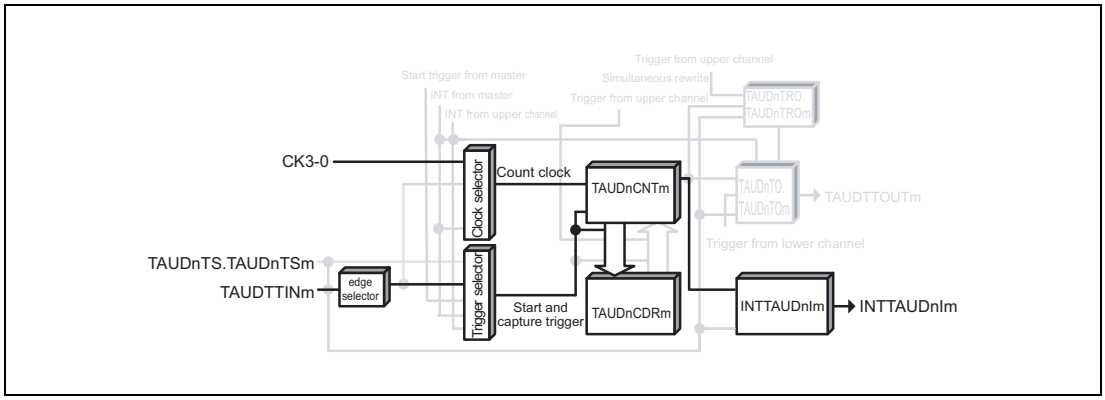


Figure 17.60 Block Diagram of TAUDTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

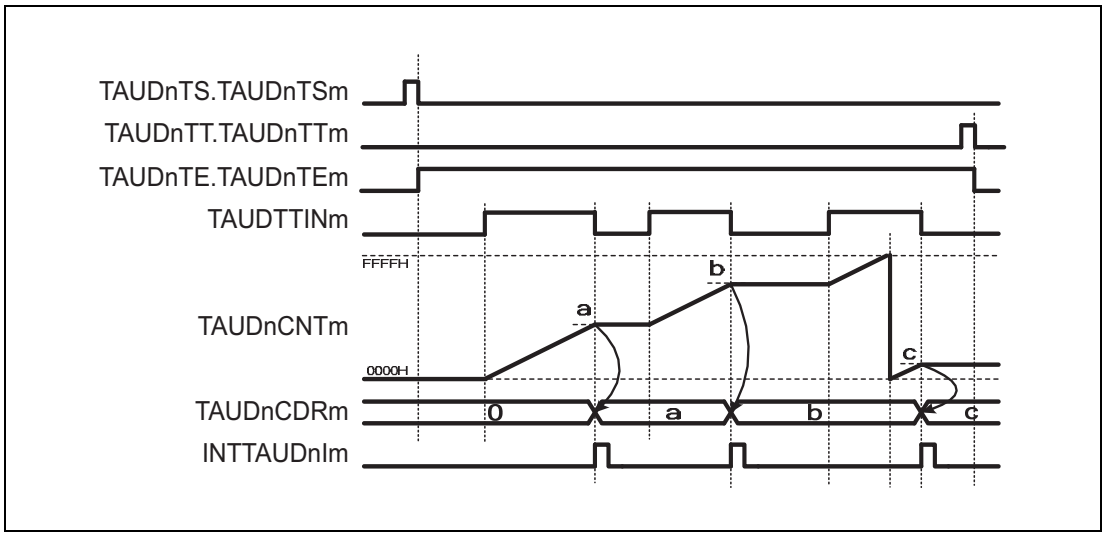


Figure 17.61 General Timing Diagram of TAUDTTINm Input Period Count Detection Function

(4) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.87 Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS [1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1101: Capture and gate count mode
0	TAUDnMD0	0: Disables the start trigger during operation.

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.88 Contents of TAUDnCMURm Register for TAUDTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.89 Simultaneous Rewrite Settings for TAUDTTINm Input Period Count Detection Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Operating Procedure for TAUDTTINm Input Period Count Detection Function

Table 17.90 Operating Procedure for TAUDTTINm Input Period Count Detection Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.87, Contents of TAUDnCMORm Register for TAUDTTINm Input Period Count Detection Function**, and **Table 17.88, Contents of TAUDnCMURm Register for TAUDTTINm Input Period Count Detection Function**. The TAUDnCDRm register functions as a capture register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When a start edge is detected, TAUDnCNTm is cleared to 0000_H and starts counting up.
During Operation	Detection of TAUDTTINm edge The TAUDnCDRm, TAUDnCNTm, and TAUDnCSRm registers can be read at any time. The TAUDnCSCm.TAUDnCLOV bit can be set to 1.	When a TAUDTTINm start edge (rising edge for high width measurement, falling edge for low width measurement) is detected, TAUDnCNTm starts counting up from the stop value. When TAUDnCNTm detects a stop edge (falling edge for high width measurement, rising edge for low width measurement), it transfers the value to TAUDnCDRm and INTTAUDnIm is generated. Counting stops at the “value transferred to TAUDnCDRm + 1” and TAUDnCNTm waits for detection of the TAUDTTINm start edge. When TAUDnCNTm reaches $FFFF_H$, the counter restarts count operation from 0000_H. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm retains its current values.

Restart

(6) Specific Timing Diagrams

(a) Operation stop and restart

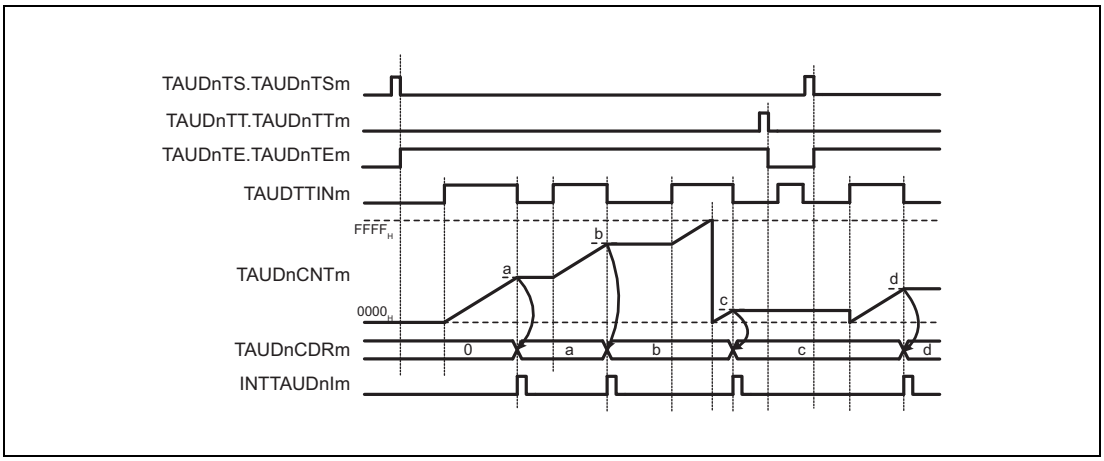


Figure 17.62 Operation Stop and Restart (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm stops and retains its current value.
- If the counter is stopped, effective TAUDTTINm input edges are ignored.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1. TAUDnCNTm restarts to count from 0000_H.

17.4.9.11 TAUDTTINm Input Pulse Interval Judgment Function

(1) Overview

Summary

This function outputs the result of a comparison between the count value (TAUDnCNTm) and the value in the channel data register (TAUDnCDRm) when a TAUDTTINm input pulse occurs. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true.

Prerequisites

- The operating mode should be set to judge mode. See **Table 17.91, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function**.
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value.

When an effective TAUDTTINm edge is detected or TAUDnTS.TAUDnTSm is set to 1, the function compares the current values of TAUDnCNTm and TAUDnCDRm. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true. TAUDnCNTm reloads the value of TAUDnCDRm and subsequently continues to operate, regardless of the result of the comparison.

If the counter reaches 0000_H before an effective TAUDTTINm edge is detected, TAUDnCNTm overflows and is set to $FFFF_H$. It then continues to count down.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

The TAUDnCMORm.TAUDnMD0 bit specifies the type of comparison:

- If TAUDnCMORm.TAUDnMD0 = 0, INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm.
- If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm.

(2) Block Diagram and General Timing Diagram

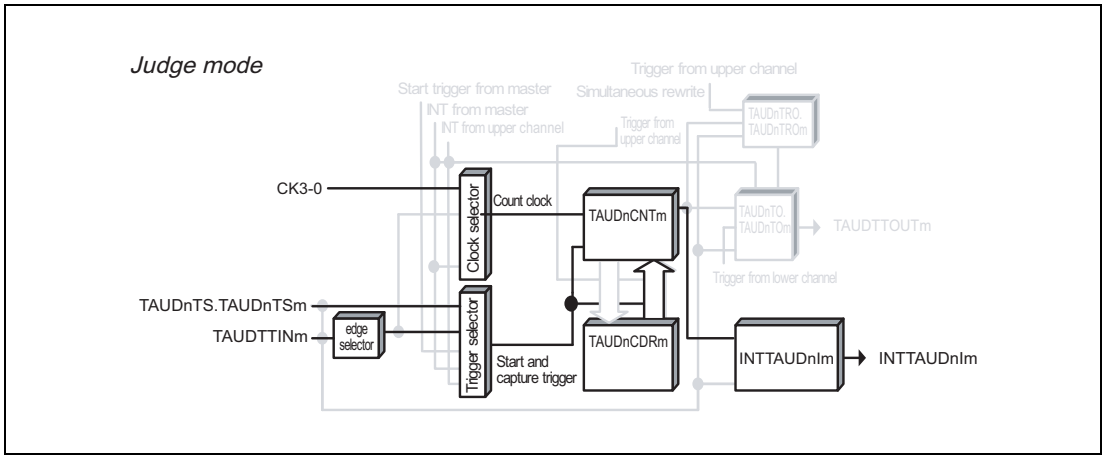


Figure 17.63 Block Diagram of TAUDTTINm Input Pulse Interval Judgment Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

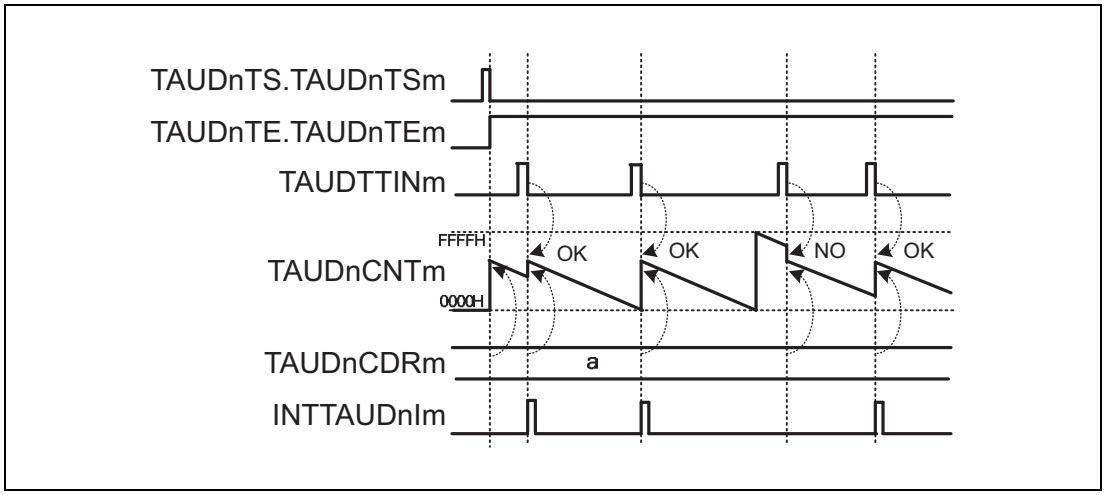


Figure 17.64 General Timing Diagram of TAUDTTINm Input Pulse Interval Judgment Function

(3) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.91 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0001: Judge mode
0	TAUDnMD0	0: INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm 1: INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.92 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Judgment Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.93 Simultaneous Rewrite Settings for TAUDTTINm Input Pulse Interval Judgment Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Operating Procedure for TAUDTTINm Input Pulse Interval Judgment Function

Table 17.94 Operating Procedure for TAUDTTINm Input Pulse Interval Judgment Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.91, Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function**, and **Table 17.92, Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Judgment Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	When TAUDnCMORm.TAUDnMD0 = 0: If TAUDnCNTm ≤ TAUDnCDRm when a TAUDTTINm input edge is detected, INTTAUDnIm is generated. When TAUDnCMORm.TAUDnMD0 = 1: If TAUDnCNTm > TAUDnCDRm when a TAUDTTINm input edge is detected, INTTAUDnIm is generated. If a TAUDTTINm input edge is detected, then TAUDnCNTm starts to count down from the value of TAUDnCDRm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

17.4.9.12 TAUDTTINm Input Signal Width Judgment Function

(1) Overview

Summary

This function compares the count value (TAUDnCNTm) for the high or low level width of a TAUDTTINm input signal and the TAUDnCDRm value, and outputs the judgment result from the interrupt request signal INTTAUDnIm.

Prerequisites

- The operating mode should be set to judge and one-count mode (see **Table 17.95, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function**).
- TAUDTTOUTm is not used with this function.

Functional description

The counter is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm = 1, enabling count operation. When an effective TAUDTTINm input start edge is detected, the current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value.

When an effective TAUDTTINm stop edge is detected, the function compares the current values of TAUDnCNTm and TAUDnCDRm. An interrupt request signal INTTAUDnIm is generated if the result of the comparison is true. The counter TAUDnCNTm retains its value until the next effective TAUDTTINm start edge is detected, regardless of the result of the comparison.

If the counter reaches 0000_H before an effective TAUDTTINm stop edge is detected, TAUDnCNTm overflows and is set to $FFFF_H$. The counter then continues to count down.

The value of TAUDnCDRm can be rewritten at any time, and the changed value of TAUDnCDRm is applied the next time the counter starts to count down.

Conditions

- The TAUDnCMORm.TAUDnMD0 bit specifies the type of comparison:
  - If TAUDnCMORm.TAUDnMD0 = 0, INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm.
  - If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm.
- The TAUDnCMURm.TAUDnTIS[1:0] bits specify a type of width measurement:
  - For high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B), TAUDTTINm rising edge is used as a start edge and TAUDTTINm falling edge as a stop edge.
  - For low width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 10_B), TAUDTTINm falling edge is used as a start edge and TAUDTTINm rising edge as a stop edge.
- This function cannot make a forced restart.

(2) Block Diagram and General Timing Diagram

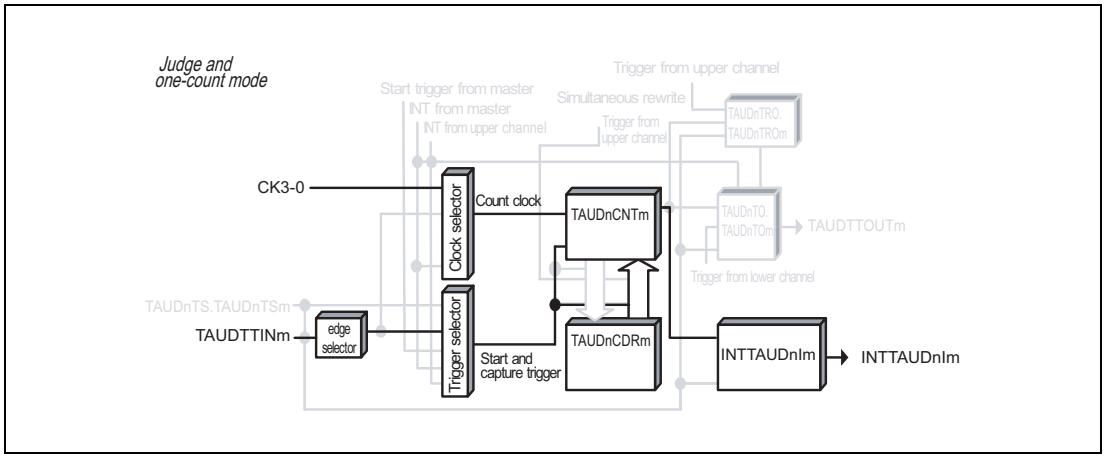


Figure 17.65 Block Diagram of TAUDTTINm Input Signal Width Judgment Function

The following settings apply to the general timing diagram.

- INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm (TAUDnCMORm.TAUDnMD0 = 0)
- Effective TAUDTTINm start edge = rising edge, effective TAUDTTINm stop edge = falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

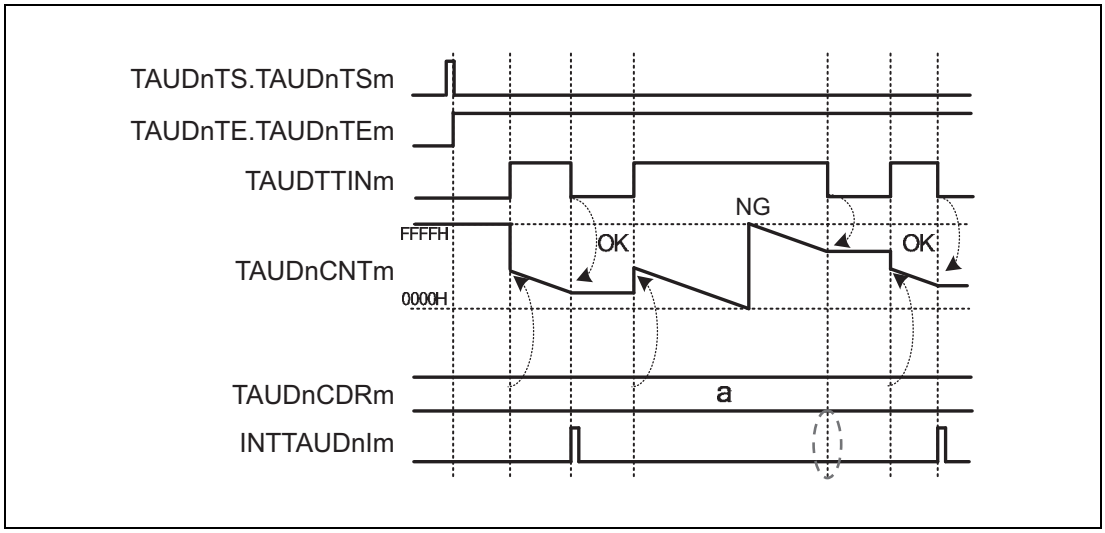


Figure 17.66 General Timing Diagram of TAUDTTINm Input Signal Width Judgment Function

(3) Register Settings

(a) TAUDnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.95 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	010: Effective edge of the TAUDTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0111: Judge and one-count mode
0	TAUDnMD0	0: INTTAUDnIm is generated when TAUDnCNTm ≤ TAUDnCDRm 1: INTTAUDnIm is generated when TAUDnCNTm > TAUDnCDRm

(b) TAUDnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.96 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Judgment Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.97 Simultaneous Rewrite Settings for TAUDTTINm Input Signal Width Judgment Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Operating Procedure for TAUDTTINm Input Signal Width Judgment Function

Table 17.98 Operating Procedure for TAUDTTINm Input Signal Width Judgment Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers as described in **Table 17.95, Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function**, and **Table 17.96, Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Judgment Function**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and TAUDnCNTm waits for detection of the TAUDTTINm start edge.
During Operation	Detection of TAUDTTINm edge The value of TAUDnCDRm is changeable at any time. The TAUDnCNTm register can be read at any time.	Upon detection of a TAUDTTINm start edge, TAUDnCNTm starts count down from the value of TAUDnCDRm. When TAUDnCMORm.TAUDnMD0 = 0: If TAUDnCNTm ≤ TAUDnCDRm when a TAUDTTINm input stop edge is detected, INTTAUDnIm is generated. When TAUDnCMORm.TAUDnMD0 = 1: If TAUDnCNTm > TAUDnCDRm when a TAUDTTINm input stop edge is detected, INTTAUDnIm is generated. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

17.4.10 Independent Channel Real-Time Functions

This section describes functions that output the value of the TAUDnTRO.TROm bit in real time:

- **Section 17.4.10.1, Real-Time Output Function Type 1**
- **Section 17.4.10.2, Real-Time Output Function Type 2**

17.4.10.1 Real-Time Output Function Type 1

(1) Overview

Summary

This function outputs a value of the TAUDnTRO.TAUDnTROm bit from TAUDTTOUTm when a specified channel generates an interrupt (INTTAUDnIm). In this function, the interrupt is generated at certain specified intervals.

The upper channel is a channel which generates a real-time output trigger (TAUDnTRC.TAUDnTRCm = 1), and the lower channel is a channel which makes a real-time output in response to the upper channel trigger (TAUDnTRC.TAUDnTRCm = 0).

Prerequisites

- Channels should use the TAUDTTOUTm control of other channels.
- The operating mode for the upper channel should be set to interval timer mode (see **Table 17.99, Contents of TAUDnCMORm Register for Real-Time Output Function Type 1**).
- Any operating mode can be set for lower channels.
- The channel output mode for all the channels should be set to independent channel output mode 1 with real-time output (see **Section 17.4.4, Channel Output Modes**).
- Real-time output should be enabled for the upper channel (TAUDnTRE.TAUDnTREm = 1).

Functional description

The counter of the upper channel is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation. The current value of the data register of the upper channel (TAUDnCDRm) is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value.

When the counter of the upper channel reaches 0000_H, INTTAUDnIm is generated and TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) of every channel (only channels with TAUDnTRE.TAUDnTREm = 1). TAUDnCNTm then reloads the TAUDnCDRm value to continue operation subsequently.

The TAUDTTOUTm signal changes only when an interrupt is generated, and when its value is different to current value of TAUDnTRO.TAUDnTROm at the moment that the interrupt is generated.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnTRC.TAUDnTRCm to 1 for the corresponding channel. The TAUDnTRC.TAUDnTRCm bit should be 0 for all other channels.
- If real-time output of a lower channel is disabled (TAUDnTRE.TAUDnTREm = 0) or if the channel itself is used as a rewrite trigger (TAUDnTRC.TAUDnTRCm = 1), the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in that channel.
- If real-time output of a lower channel is enabled (TAUDnTRE.TAUDnTREm = 1) and TAUDnTRC.TAUDnTRCm = 0, the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in the upper channel.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not output. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUDnIm generation cycle = count clock cycle × (TAUDnCDRm value + 1)

(3) Block Diagram and General Timing Diagram

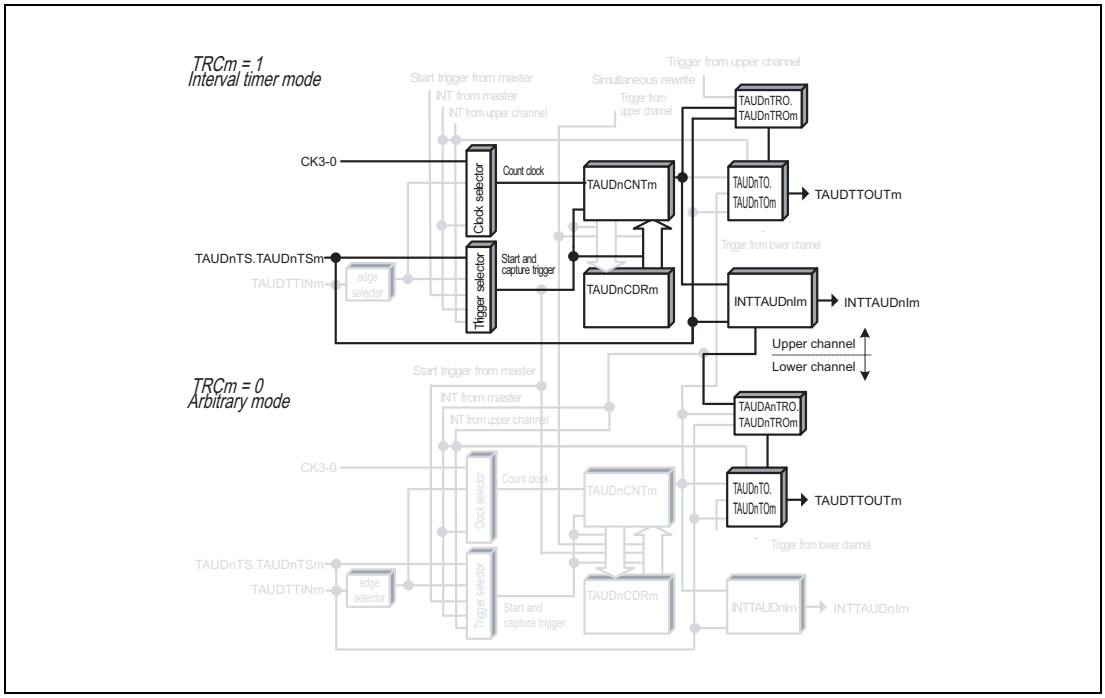


Figure 17.67 Block Diagram of Real-Time Output Function Type 1

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

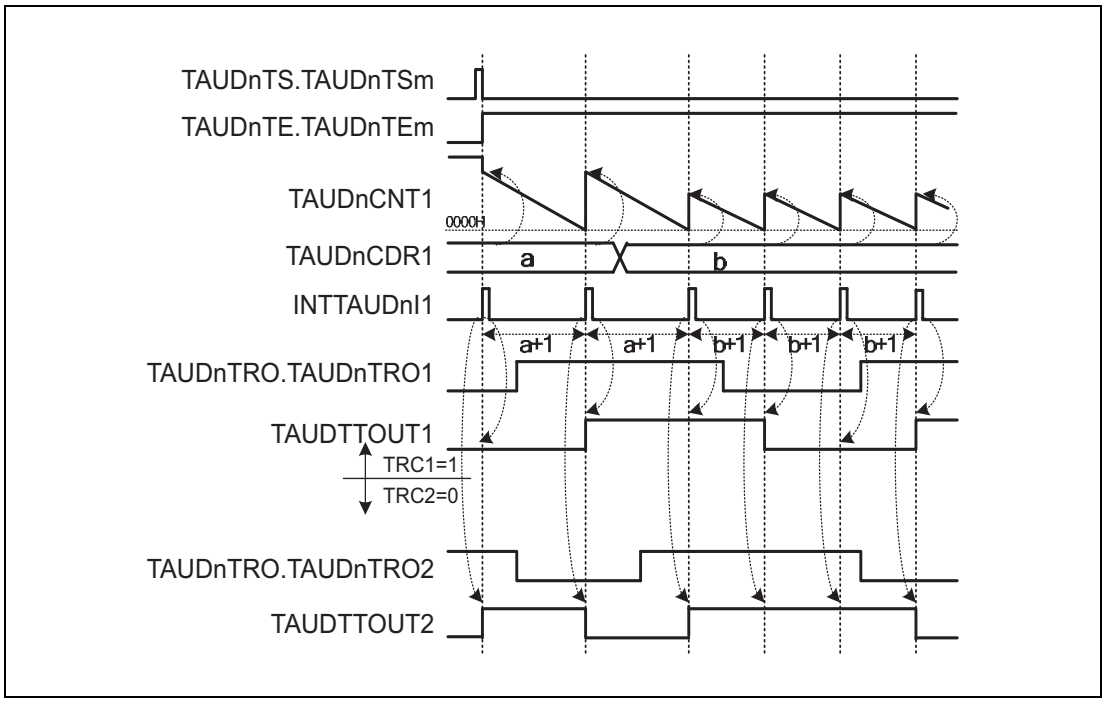


Figure 17.68 General Timing Diagram of Real-Time Output Function Type 1

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.99 Contents of TAUDnCMORm Register for Real-Time Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnSTS[2:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.100 Contents of TAUDnCMURm Register for Real-Time Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for upper channels

Table 17.101 Control Bit Settings in Independent Channel Output Mode 1 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	1: Channel m generates a unique real-time output trigger.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for upper channels

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the real-time output function type 1. Therefore, these registers should be set to 0.

Table 17.102 Simultaneous Rewrite Settings for Real-Time Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

The TAUDnCMORm register for lower channels is available for any setting.

(b) TAUDnCMURm for lower channels

The TAUDnCMURm register for lower channels is available for any setting.

(c) Channel output mode for lower channels

Table 17.103 Control Bit Settings for the Lower Channels in Independent Channel Output Mode 1 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers for lower channels can be set arbitrarily.

(6) Operating Procedure for Real-Time Output Function Type 1

Table 17.104 Operating Procedure for Real-Time Output Function Type 1

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for upper channels as described in **Table 17.99, Contents of TAUDnCMORm Register for Real-Time Output Function Type 1**, and **Table 17.100, Contents of TAUDnCMURm Register for Real-Time Output Function Type 1**. Set TAUDnCMORm and TAUDnCMURm registers for lower channels as described in **Section (5), Register Settings for Lower Channels**. Set the value of TAUDnCDRm register (only channels with TAUDnTRC.TAUDnTRCm = 1) Set channel output mode by setting the control bits as described in **Table 17.101, Control Bit Settings in Independent Channel Output Mode 1 with Real-Time Output**. Set channel output mode by setting the control bits as described in **Table 17.103, Control Bit Settings for the Lower Channels in Independent Channel Output Mode 1 with Real-Time Output**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm = 1 on the channel with TAUDnTRC.TAUDnTRCm set to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	[Channel with TAUDnTRC.TAUDnTRCm set to 1] TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm.TAUDnCMORm. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	TAUDnCDRm and TAUDnTRO.TAUDnTROm can be changed at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm outputs the current value of the real-time output bit TAUDnTRO.TAUDnTROm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. Both TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

(7) Specific Timing Diagrams

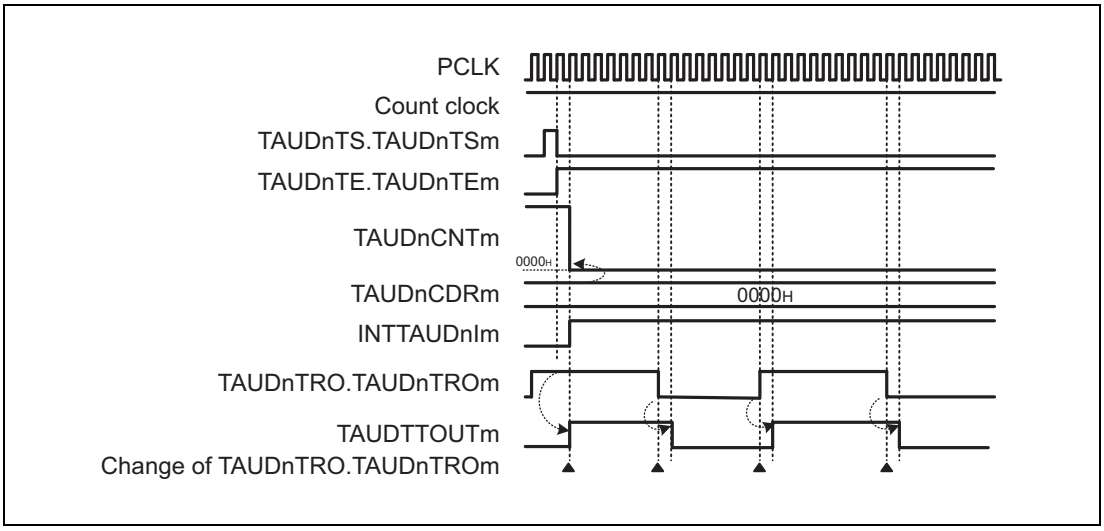


Figure 17.69 TAUDnCDRm = 0000_H, TAUDnCMORm.TAUDnMD0 = 1

- The value of TAUDTTOUTm changes according to the setting of TAUDnTRO.TAUDnTROm with a delay of one PCLK cycle.

17.4.10.2 Real-Time Output Function Type 2

(1) Overview

Summary

This function outputs the value of TAUDnTRO.TAUDnTROm bit from TAUDTTOUTm when a specified channel generates an interrupt (INTTAUDnIm). In this function, the interrupt is generated when an effective TAUDTTINm input edge is detected or the function starts.

The upper channel is a channel which generates a real-time output trigger (TAUDnTRC.TAUDnTRCm = 1), and the lower channel is a channel which makes a real-time output in response to the upper channel trigger (TAUDnTRC.TAUDnTRCm = 0).

Prerequisites

- Channels should use the TAUDTTOUTm control of the other channels
- The operating mode for the upper channel should be set to interval timer mode (see **Table 17.105, Contents of TAUDnCMORm Register for Real-Time Output Function Type 2**).
- Any operating mode can be set for lower channels.
- The channel output mode for all the channels should be set to independent channel output mode 1 with real-time output (see **Section 17.4.4, Channel Output Modes**).
- Real-time output should be enabled for the upper channel (TAUDnTRE.TREm = 1).

Functional description

The counter for upper channels is started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm to 1, enabling count operation. The counter starts to count up.

When an effective TAUDTTINm input edge is generated on one of upper channels, an interrupt occurs and TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) of every channel (only channels with TAUDnTRE.TAUDnTREm = 1).

The TAUDTTOUTm signal changes only when an interrupt is generated, and when TAUDTTOUTm value is different to the current value of TAUDnTRO.TAUDnTROm during the occurrence of the interrupt.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnTRC.TRCm to 1 for the corresponding channel. The TAUDnTRC.TRCm bit should be 0 for all other channels.
- If real-time output of a lower channel is disabled (TAUDnTRE.TAUDnTREm = 0) or if the channel itself is used as a rewrite trigger (TAUDnTRC.TAUDnTRCm = 1), the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in that channel.
- If real-time output of a lower channel is enabled (TAUDnTRE.TAUDnTREm = 1) and TAUDnTRC.TAUDnTRCm = 0, the value of that channel's TAUDnTRO.TAUDnTROm bit is output when INTTAUDnIm is generated in the upper channel.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not output. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Block Diagram and General Timing Diagram

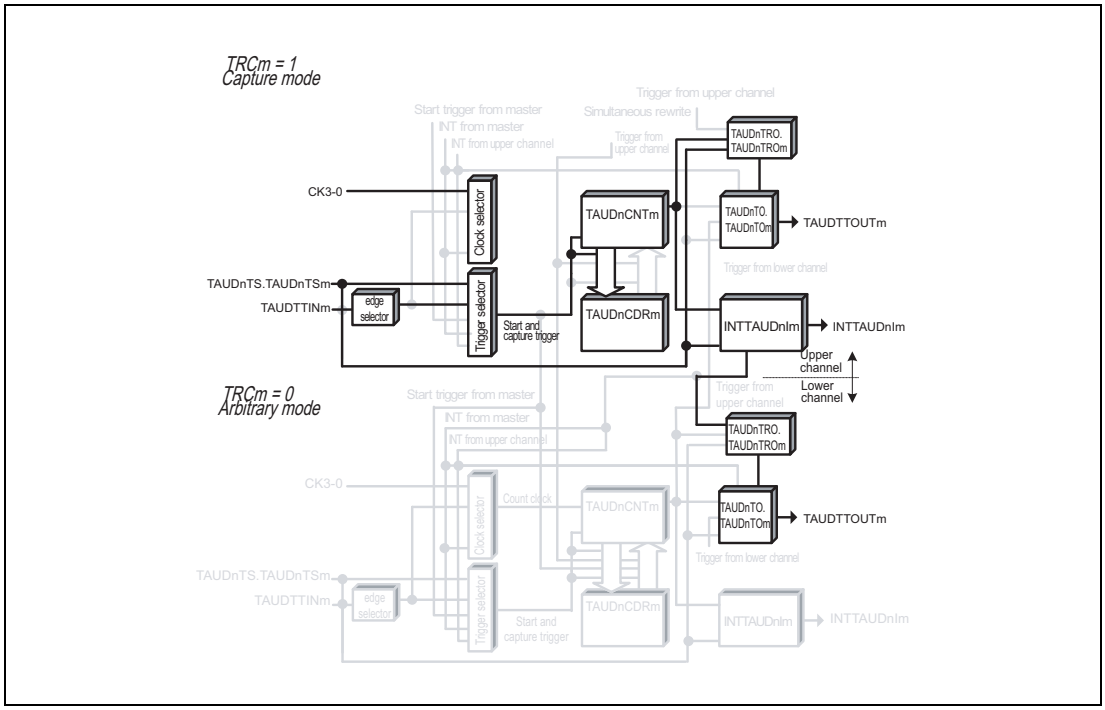


Figure 17.70 Block Diagram of Real-Time Output Function Type 2

The following settings apply to the general timing diagram.

- INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)

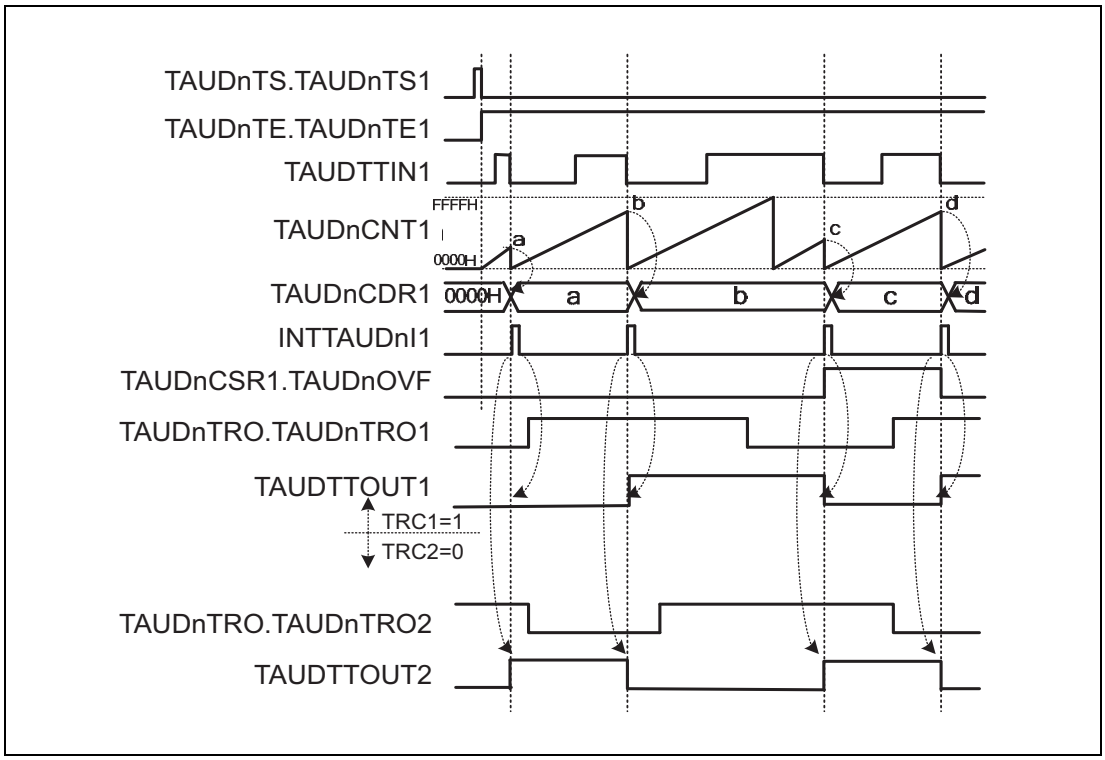


Figure 17.71 General Timing Diagram of Real-Time Output Function Type 2

(3) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.105 Contents of TAUDnCMORm Register for Real-Time Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.106 Contents of TAUDnCMURm Register for Real-Time Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for upper channels

Table 17.107 Control Bit Settings in Independent Channel Output Mode 2 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	1: Channel m generates a unique real-time output trigger.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for upper channels

The simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with the real-time output function type 2. Therefore, these registers should be set to 0.

Table 17.108 Simultaneous Rewrite Settings for Real-Time Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

The TAUDnCMORm register for lower channels is available for any setting.

(b) TAUDnCMURm for lower channels

The TAUDnCMURm register for lower channels is available for any setting.

(c) Channel output mode for lower channels

Table 17.109 Control Bit Settings for Lower Channels in Independent Channel Output Mode 2 with Real-Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode.
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation.
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output 1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers for lower channels can be set arbitrarily.

(5) Operating Procedure for Real-Time Output Function Type 2

Table 17.110 Operating Procedure for Real-Time Output Function Type 2

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for upper channels as described in **Table 17.105, Contents of TAUDnCMORm Register for Real-Time Output Function Type 2**, and **Table 17.106, Contents of TAUDnCMURm Register for Real-Time Output Function Type 2**. Set TAUDnCMORm and TAUDnCMURm registers for the lower channel as described in **Section (4), Register Settings for Lower Channels**. The TAUDnCDRm register functions as a capture register. (channels with TAUDnTRC.TAUDnTRCm = 1) Set channel output mode by setting the control bits as described in **Table 17.107, Control Bit Settings in Independent Channel Output Mode 2 with Real-Time Output**. Set channel output mode by setting the control bits as described in **Table 17.109, Control Bit Settings for Lower Channels in Independent Channel Output Mode 2 with Real-Time Output**.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm = 1 on the channel with TAUDnTRC.TAUDnTRCm set to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	[Channel with TAUDnTRC.TAUDnTRCm set to 1] TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCNTm is cleared to 0000_H. If TAUDnCMORm.TAUDnMD0 is 1, INTTAUDnIm occurs.
During Operation	TAUDnTRO.TAUDnTROm can be changed at any time.	TAUDnCNTm starts to count up from 0000_H. When an effective TAUDTTINm input edge is detected: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. • TAUDTTOUTm outputs the current value of the real-time output bit TAUDnTRO.TAUDnTROm. TAUDTTOUTm outputs the current value of real-time output bit TAUDnTRO.TAUDnTROm. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm, TAUDnCSRm.TAUDnOVF, and TAUDTTOUTm retain their current values.

Restart

(6) Specific Timing Diagrams

(a) Operation start and stop

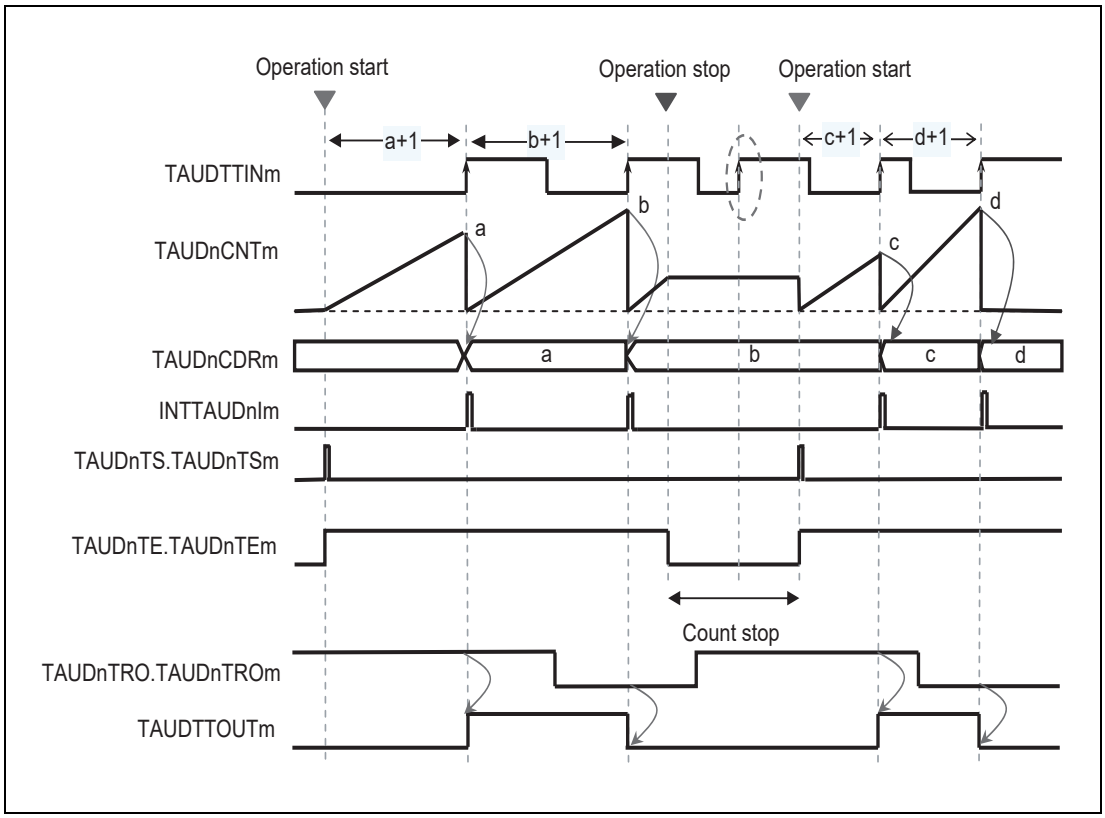


Figure 17.72 Operation Start and Stop (TAUDnCMORm.TAUDnMD0 = 0)

- When TAUDnTS.TAUDnTSm is set to 1, the counter starts counting up.
- When an effective input edge is detected, the current value of the counter is written to the data register (TAUDnCDRm) and an interrupt is generated.
- TAUDTTOUTm outputs the current value of the real-time output bit (TAUDnTRO.TAUDnTROm) and the counter resets and starts to count up again.
- The TAUDTTOUTm signal only changes when an interrupt is generated, and then only when its value is different to current value of TAUDnTRO.TAUDnTROm at the moment that the interrupt is generated.
- If the counter is stopped (TAUDnTE.TAUDnTEm = 0), effective input edges are ignored and no interrupt is generated.

17.4.11 Independent Channel Simultaneous Rewrite Functions

This section describes functions that carry out simultaneous rewrite:

- **Section 17.4.11.1, Simultaneous Rewrite Trigger Generation Function Type 1**

17.4.11.1 Simultaneous Rewrite Trigger Generation Function Type 1

(1) Overview

Summary

This function generates an interrupt on a specific channel that can be used by lower channels as a simultaneous rewrite trigger. The interrupt is generated at regular intervals.

The upper channel is a channel which generates a simultaneous rewrite trigger (TAUDnRDC.TAUDnRDCm = 1), and the lower channel is a channel which makes a simultaneous rewrite in response to the upper channel trigger (TAUDnRDC.TAUDnRDCm = 0).

Prerequisites

- Two or more channels lower than the channel used as upper channel are enabled for simultaneous rewrite (TAUDnRDE.RDEm = 1).
- The operating mode for the upper channel should be set to interval timer mode (see **Table 17.111, Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1**).
- For the operating mode that can be set for lower channels, see **Table 17.42, Channel Operation Functions and Methods They Use**.
- TAUDTTOUTm is not used for any channel in this function.

Functional description

The counter operation is enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) for upper and lower channels to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of the data register buffer for upper channels (TAUDnCDRm buf) is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value. The counter for lower channels start to count according to the selected operating mode.

Once the counter reaches 0000_H, an interrupt occurs on the channel. The current value of the corresponding TAUDnCDRm buffer is loaded into TAUDnCNTm to continue operation subsequently.

If the channel where an interrupt occurs is specified as a trigger channel for simultaneous rewrite (TAUDnRDC.RDCm = 1) and is an upper channel, simultaneous rewrite takes place on all lower channels in which simultaneous rewrite is currently possible (TAUDnRSF.RSFm = 1).

The values of the data registers are copied to the corresponding data register buffers. Each time a counter starts to count down, it reads the value in the data register buffer and counts down from this value.

The value of a data register can be changed at any time, but it is only transferred to the corresponding data register buffer when simultaneous rewrite occurs.

Conditions

- The channel which is monitored for INTTAUDnIm occurrence is specified by setting TAUDnRDC.TAUDnRDCm = 1 for the corresponding channel. The TAUDnRDC.TAUDnRDCm bit should be 0 for all other channels in which simultaneous rewrite should take place.
- If the TAUDnCMORm.TAUDnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 17.4.6, TAUDTTOUTm Output and INTTAUDnIm Generation when Counter Starts or Restarts**.

(2) Equations

Simultaneous rewrite trigger generation cycle = count clock cycle × (TAUDnCDRm + 1)

To control simultaneous rewrite, the following condition should be satisfied:

[PWM]

TAUDnCDRm = [(value of TAUDnCDRm of master channel subject to simultaneous rewrite + 1) × number of interrupts] – 1

[Triangle PWM]

TAUDnCDRm = [(value of TAUDnCDRm of master channel subject to simultaneous rewrite + 1) × 2 × number of interrupts] – 1

That is, the ratio of TAUDnCDRm + 1 and TAUDnCDRm_master + 1 should be an integer. This integer corresponds to the number of interrupts.

For triangle PWM, remember that the cycle doubles.

(3) Block Diagram and General Timing Diagram

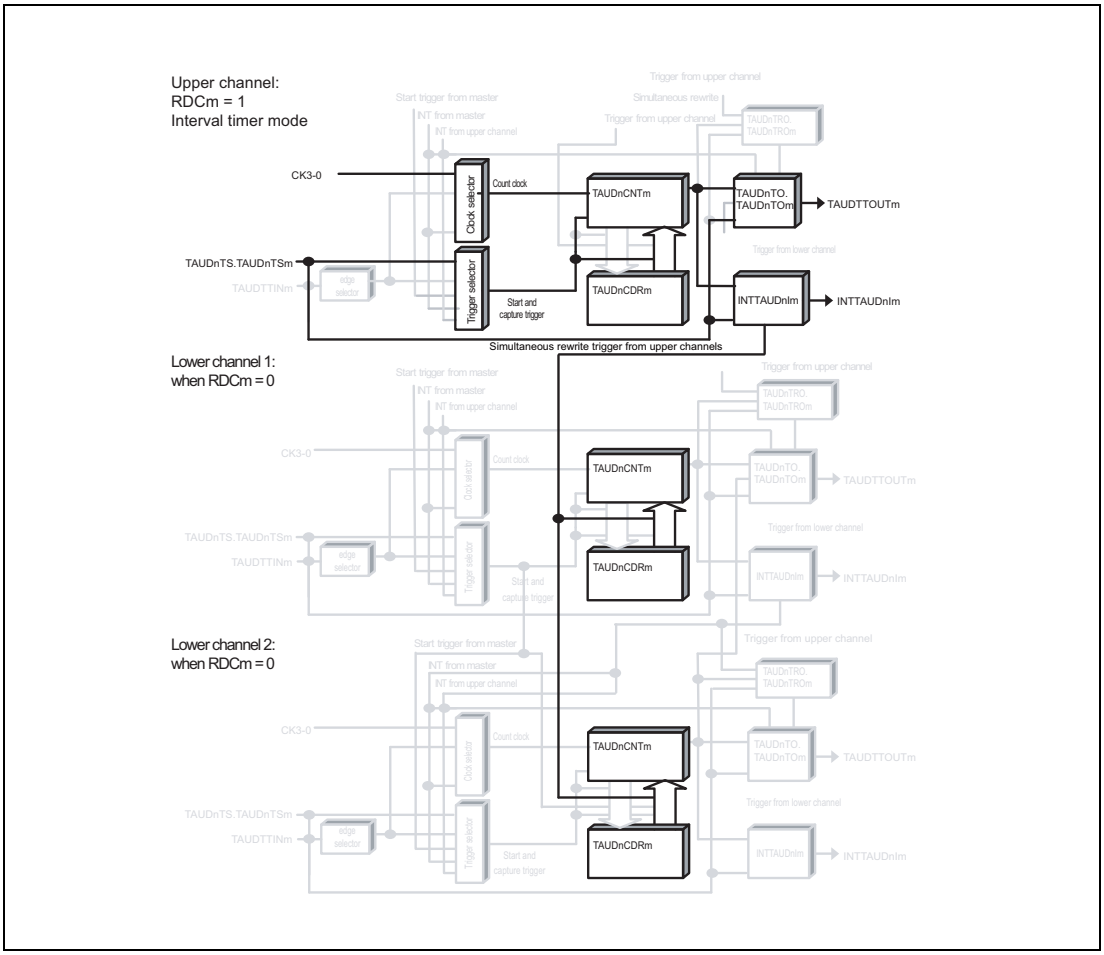


Figure 17.73 Block Diagram of Simultaneous Rewrite Trigger Generation Function Type 1

The following settings apply to the general timing diagram.

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

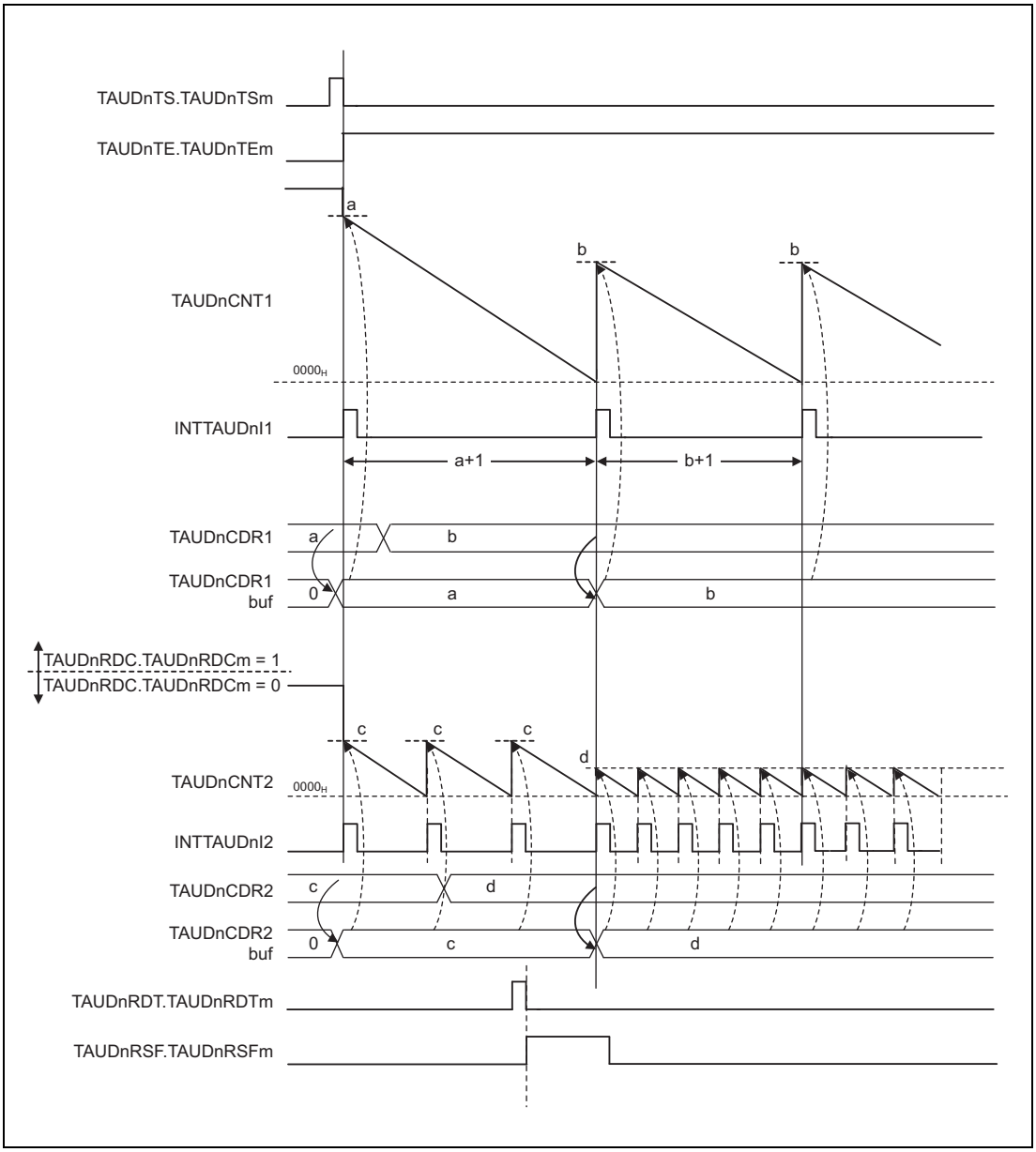


Figure 17.74 General Timing Diagram of Simultaneous Rewrite Trigger Generation Function Type 1

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.111 Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for upper channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.112 Contents of TAUDnCMURm Register for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for upper channels

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite for upper channels

Table 17.113 Simultaneous Rewrite Settings for Simultaneous Rewrite Trigger Generation Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	1: Selects one of upper channels as simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Loads a simultaneous rewrite control signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	1: Monitors INTTAUDnIm signal which triggers a simultaneous rewrite on the channel.

(5) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

TAUDnCMORm register for lower channels must follow the TAUDnCMORm register settings in the operating mode which can be set (see **Table 17.42, Channel Operation Functions and Methods They Use**).

(b) TAUDnCMURm for lower channels

TAUDnCMURm register for lower channels must follow the TAUDnCMURm register settings in the operating mode which can be set (see **Table 17.42, Channel Operation Functions and Methods They Use**).

(c) Channel output mode for lower channels

Output can be made according to the operating mode setting (master/slave) for lower channels.
As for the available function for simultaneous rewrite trigger generation function type 1, see **Table 17.42, Channel Operation Functions and Methods They Use**.

(d) Simultaneous rewrite for lower channels

Table 17.114 Simultaneous Rewrite Settings for Lower Channels in Simultaneous Rewrite Trigger Generation Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	1: Selects one of upper channels as simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Loads a simultaneous rewrite control signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Simultaneous Rewrite Trigger Generation Function Type 1

Table 17.115 Operating Procedure for Simultaneous Rewrite Trigger Generation Function Type 1

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for the upper channel as described in **Table 17.111, Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1**, and **Table 17.112, Contents of TAUDnCMURm Register for Simultaneous Rewrite Trigger Generation Function Type 1**. Set TAUDnCMORm and TAUDnCMURm registers for lower channels as described in **Section (5), Register Settings for Lower Channels**. Set the value of TAUDnCDRm register.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm to 1. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is set to 1 and the counter starts. TAUDnCDRm value is loaded into TAUDnCNTm. If TAUDnCMORm.TAUDnMD0 = 1, INTTAUDnIm occurs.
During Operation	TAUDnRDT.TAUDnRDTm and TAUDnCDR.CDRm is changeable. TAUDnRSF.TAUDnRSFm can be always read.	TAUDnCNTm counts down. When the counter reaches 0000_H: • The TAUDnCDRm value is loaded in TAUDnCNTm again and count operation continues. • INTTAUDnIm is generated. If INTAUDnIm occurs on the channel where TAUDnRDC.TAUDnRDCm is set to 1, simultaneous rewrite is controlled. Afterwards, this procedure is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm to 1. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops and retains its current value.

Restart

17.4.12 Synchronous Channel Operation Functions

This section lists all the synchronous channel operation functions provided by the timer array unit D. For a general overview of synchronous channel operation, see **Section 17.2, Overview**

This section describes functions that generate PWM signals at regular intervals.

- **Section 17.4.12.1, PWM Output Function**
- **Section 17.4.12.2, One-Shot Pulse Output Function**
- **Section 17.4.12.3, Trigger Start PWM Output Function**
- **Section 17.4.12.4, Delay Pulse Output Function**
- **Section 17.4.12.5, Offset Trigger Output Function**
- **Section 17.4.12.6, A/D Conversion Trigger Output Function Type 1**
- **Section 17.4.12.7, Triangle PWM Output Function**
- **Section 17.4.12.8, Triangle PWM Output Function with Dead Time**
- **Section 17.4.12.9, A/D Conversion Trigger Output Function Type 2**
- **Section 17.4.12.10, Interrupt Request Signals Culling Function**
- **Section 17.4.12.11, One-Phase PWM Output Function**

17.4.12.1 PWM Output Function

(1) Overview

Summary

This function generates multiple PWM outputs by using a master and multiple slave channels. It enables the pulse cycle (frequency) and the duty cycle of the TAUDTTOUTm to be set. The pulse cycle is set in the master channel. The duty cycle is set in the slave channel.

Prerequisites

- Two channels
- The operating mode for the master channel should be set to interval timer mode (see **Table 17.116, Contents of TAUDnCMORm Register for Master Channels of PWM Output Function**).
- The operating mode for the slave channels should be set to one count mode (see **Table 17.119, Contents of TAUDnCMORm Register for Slave Channels of PWM Output Function**).
- TAUDTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channels should be set to Synchronous Channel Output Mode 1 (see **Section 17.4.4, Channel Output Modes**).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDmCNT, and the counter starts counting down from the TAUDnCDRm value. If an INTTAUDnlm occurs on the master channel and TAUDTTOUTm (slave) is set/reset, PWM output is made.

- Master channel:
  When the master channel counter reaches 0000_H and the pulse cycle time has passed, INTTAUDnIm occurs. The counter loads TAUDnCDRm value into TAUDnCNTm and counts down.
- Slave channels:
  When INTTAUDnIm occurs on the master channel, the counter operation of the slave channel is triggered. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave) and the counter starts counting down from the TAUDnCDRm value. TAUDTTOUTm signal is set to the active level.
  When the counter reaches to 0000_H (duty time has elapsed), INTTAUDnIm occurs and a TAUDTTOUTm signal is set to an inactive level. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm (start of the next pulse cycle) of the master channel.

The counter can stop operating by setting the TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = (TAUDnCDRm (slave)/(TAUDnCDRm (master) + 1) × 100

– Duty cycle = 0%

  TAUDnCDRm (slave) = 0000_H

– Duty cycle = 100%

  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

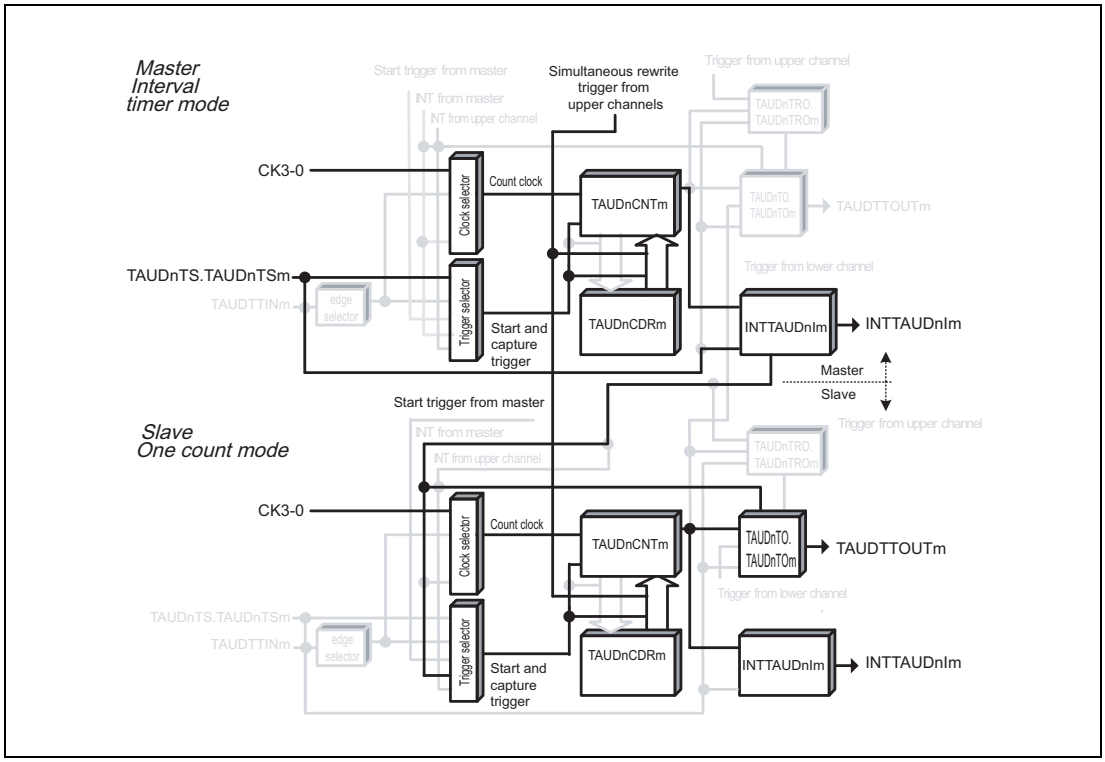


Figure 17.75 Block Diagram of PWM Output Function

The following settings apply to the general timing diagram.

- Slave channels: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

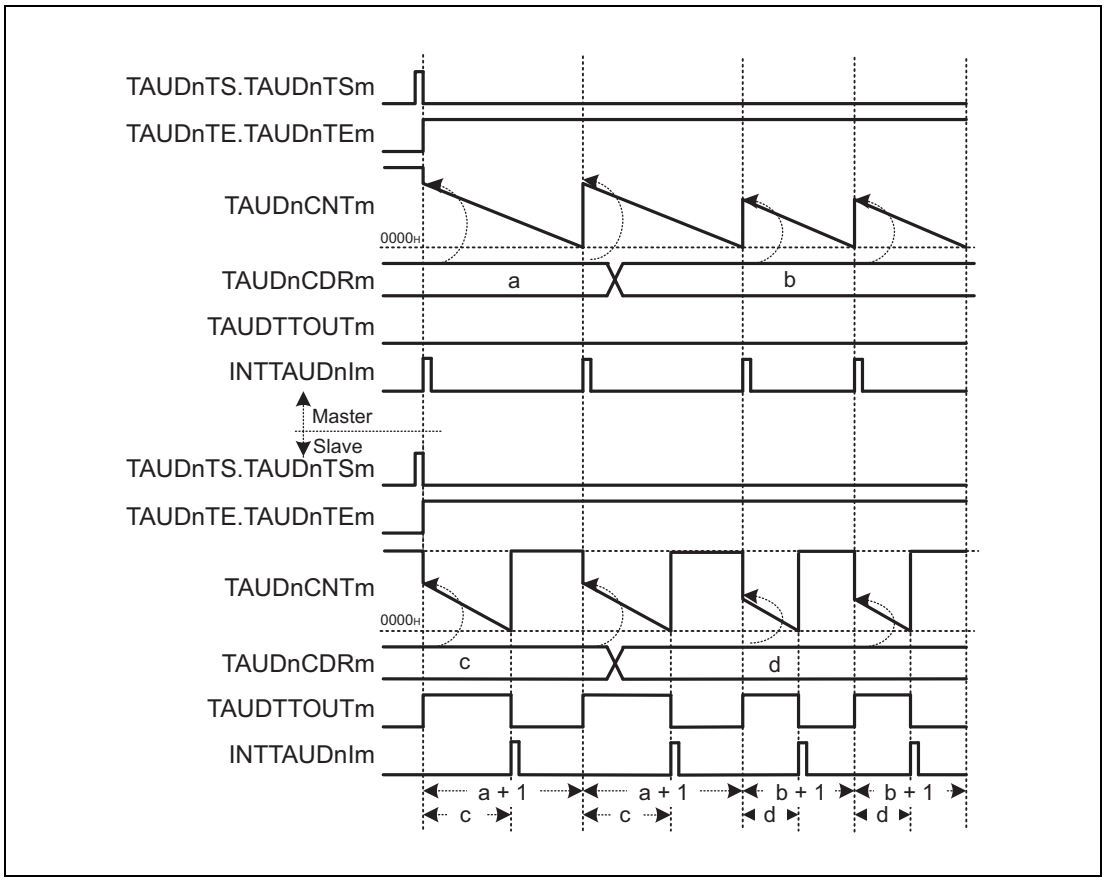


Figure 17.76 General Timing Diagram of PWM Output Function

NOTE

The interval between the slave channel starting to count and an interrupt being generated is the value of corresponding TAUDnCDRm, whereas for the master channel the interval is the value of the corresponding TAUDnCDRm + 1.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.116 Contents of TAUDnCMORm Register for Master Channels of PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.117 Contents of TAUDnCMURm Register for Master Channels of PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

The channel output mode is not used with this function. However, this mode can be used with another function or in independent channel output mode controlled by software.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.118 Simultaneous Rewrite Settings for Master Channels of the PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

Use with TAUDnRDS.TAUDnRDSm bit = 1 requires an upper channel higher than the master channel that operates with the **Section 17.4.11.1, Simultaneous Rewrite Trigger Generation Function Type 1**.

Conduct operation settings under the following conditions:

- Simultaneous rewrite trigger output function type 1 setting channel: TAUDnRDCm = 1, TAUDnRDS = 1
  TAUDnCDR settings for this channel are as follows:
  = ((TAUDnCDR setting for the master channel targeted for simultaneous rewrite + 1) × interrupt count) - 1
- Master channels: TAUDnRDCm = 0, TAUDnRDS = 1
- Slave channels: TAUDnRDCm = 0, TAUDnRDS = 1

If TAUDnCDRm (slave) setting > TAUDnCDRm (master) setting + 1, the duty value (which exceeds 100%) is aggregated to be 100% output.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.119 Contents of TAUDnCMORm Register for Slave Channels of PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is valid.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.120 Contents of TAUDnCMURm Register for Slave Channels of PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 17.121 Control Bit Settings in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.122 Simultaneous Rewrite Settings for Slave Channels of PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for PWM Output Function

Table 17.123 Operating Procedure for PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave) is set.
During operation	TAUDnCDRm can be changed at any time. TAUDnTOL.TAUDnTOLm can be changed. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel loads TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is loaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is loaded into TAUDnCNTm (slave) to perform counting down. • TAUDTTOUTm (slave) is set to the active level. If TAUDnCNTm (slave) reaches 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set to an inactive level. In addition, the counter of slave channel stops.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

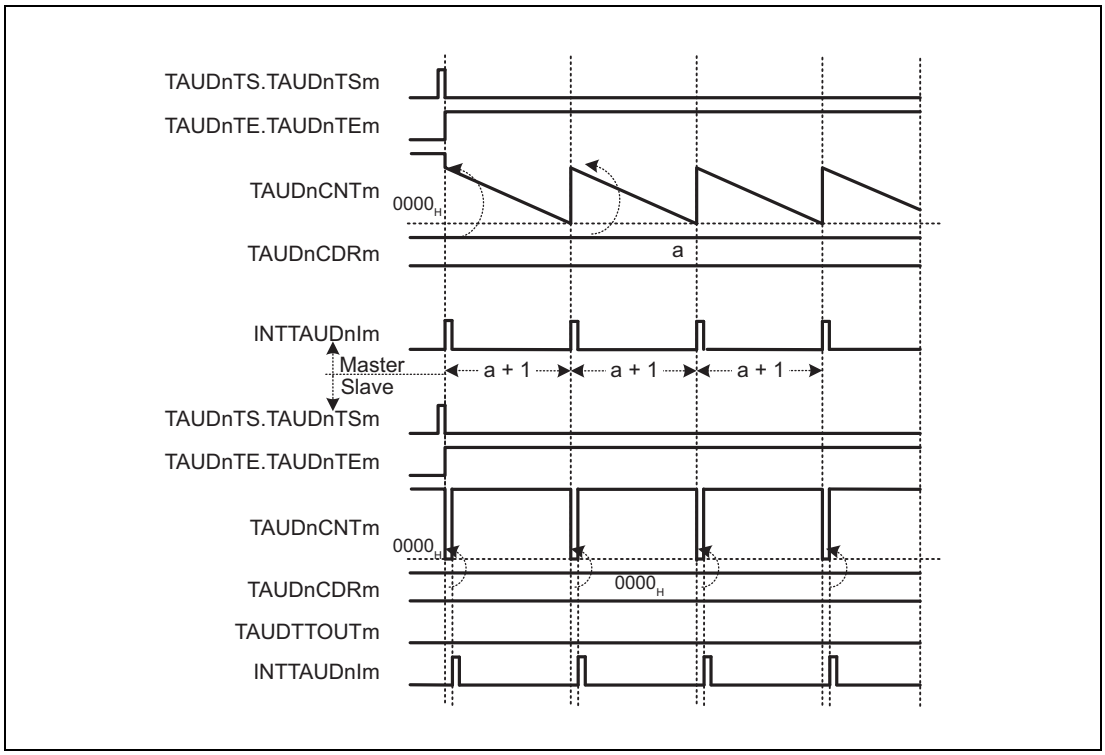


Figure 17.77 TAUDnCDRm (Slave) = 0000_H, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- Every time the master channel generates an interrupt (INTTAUDnIm), 0000_H is loaded in to TAUDnCNTm (slave). Therefore, TAUDnCNTm (slave) cannot start to count and TAUDTTOUTm remains inactive.
- TAUDnCDRm value is loaded into TAUDnCNTm (slave) to generate an interrupt.

(b) Duty cycle = 100%

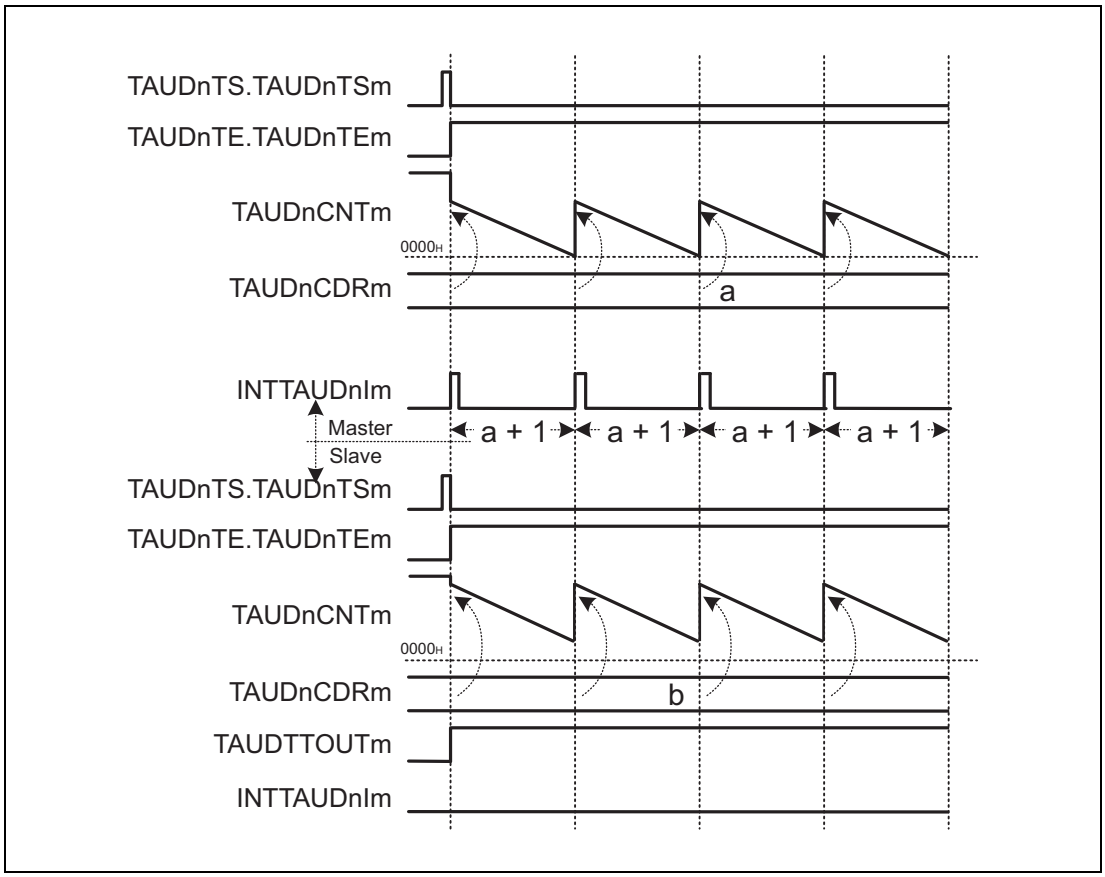


Figure 17.78 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1 Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- If TAUDnCDRm (slave) value is greater than TAUDnCDRm (master) value, the slave channel counter does not reach 0000_H and consequently, no interrupt occurs. TAUDTTOUTm remains active.

(c) Operation stop and restart

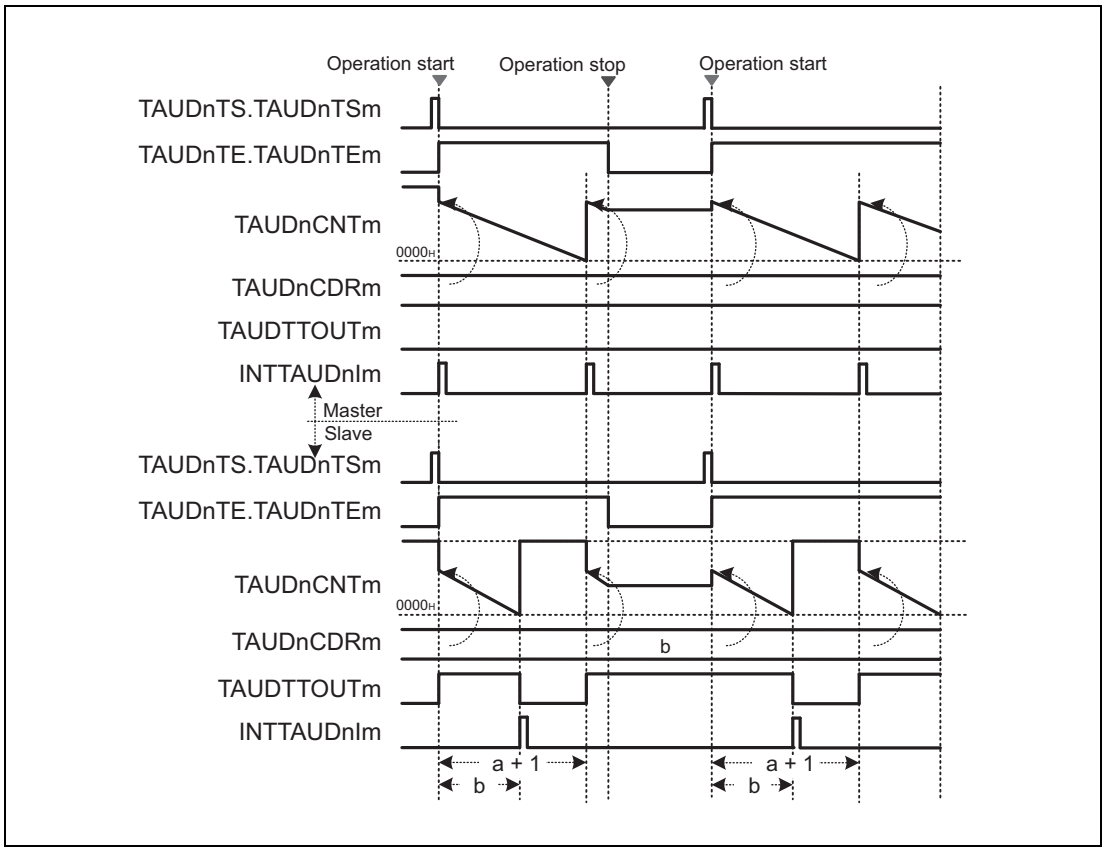


Figure 17.79 Operation Stop and Restart
Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)

- The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0.
- TAUDnCNTm and TAUDTTOUTm of all channels stop and the current values are retained. No interrupts are generated.
- The counter can be restarted by setting TAUDnTS.TAUDnTSm of master and slave channels to 1. TAUDnCNTm of master and slave channels reload the current values of TAUDnCDRm and start to count down from these values.

17.4.12.2 One-Shot Pulse Output Function

(1) Overview

Summary

This function outputs a signal pulse with a specific pulse width and delay time (both defined relative to an external input signal pulse or software trigger) by using a master and a slave channel. The delay time is specified using the master channel. The pulse width is specified using the slave channel.

Prerequisites

- Two channels
- The operating mode for the master channel should be set to one-count mode (see **Table 17.124, Contents of TAUDnCMORm Register for Master Channels of One-Shot Pulse Output Function**).
- The operating mode for slave channels should be set to pulse one-count mode (see **Table 17.127, Contents of TAUDnCMORm Register for Slave Channels of One-Shot Pulse Output Function**).
- TAUDTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channel should be set to synchronous channel output mode 2 (see **Section 17.4.4, Channel Output Modes**).
- TAUDTTINm (master) has to be detected while TAUDnCNTm (master) and TAUDnCNTm (slave) await a trigger. Furthermore, the slave is only triggered by an interrupt from the master channel and not by TAUDTTINm (slave).
- If only a software trigger is to be used, do not select the alternative pin function TAUDTTINm.

Functional description

The counters are enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm, enabling count operation.

- Master channel:
  When the next effective TAUDTTINm input edge or a software trigger (TAUDnTS.TAUDnTSm = 1 (m: master channel number) when TAUDnTE.TAUDnTEm = 1) is detected, the current value of TAUDnCDRm is loaded into TAUDnCNTm. The counter starts to count down from this value. If TAUDnCMORm.TAUDnMD0 = 0, a trigger (TAUDTTINm) which is detected within the delay time is ignored.
  When the counter of master channel reaches 0000_H, INTTAUDnIm is generated. The counter is reset to $FFFF_H$ and waits for the next effective TAUDTTINm input edge or a software trigger (TAUDnTS.TAUDnTSm = 1 (m: master channel number) when TAUDnTE.TAUDnTEm = 1).
- Slave channels:
  INTTAUDnIm generated on master channel triggers the counter operation of slave channel. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave). The counter starts counting down from this value. An interrupt occurs and the TAUDTTOUTm signal is set.
  When the counter reaches 0001_H, INTTAUDnIm is generated and TAUDTTOUTm signal is reset. The counter stops at 0000_H and waits for the next INTTAUDnIm of master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Setting TAUDnTS.TAUDnTSm to 1 while counting allows the counter to restart counting of master channel without making a stop (forced restart).

Conditions

- If TAUDnCMORn.TAUDnMD0 of master channel is set to 0, TAUDTTINm input edges detected during counting are ignored.
- Simultaneous rewrite can be used with this function. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Delay from trigger input to pulse output = (TAUDnCDRm (master) + 1) × count clock cycle

Pulse width = (TAUDnCDRm (slave)) × count clock cycle

(3) Block Diagram and General Timing Diagram

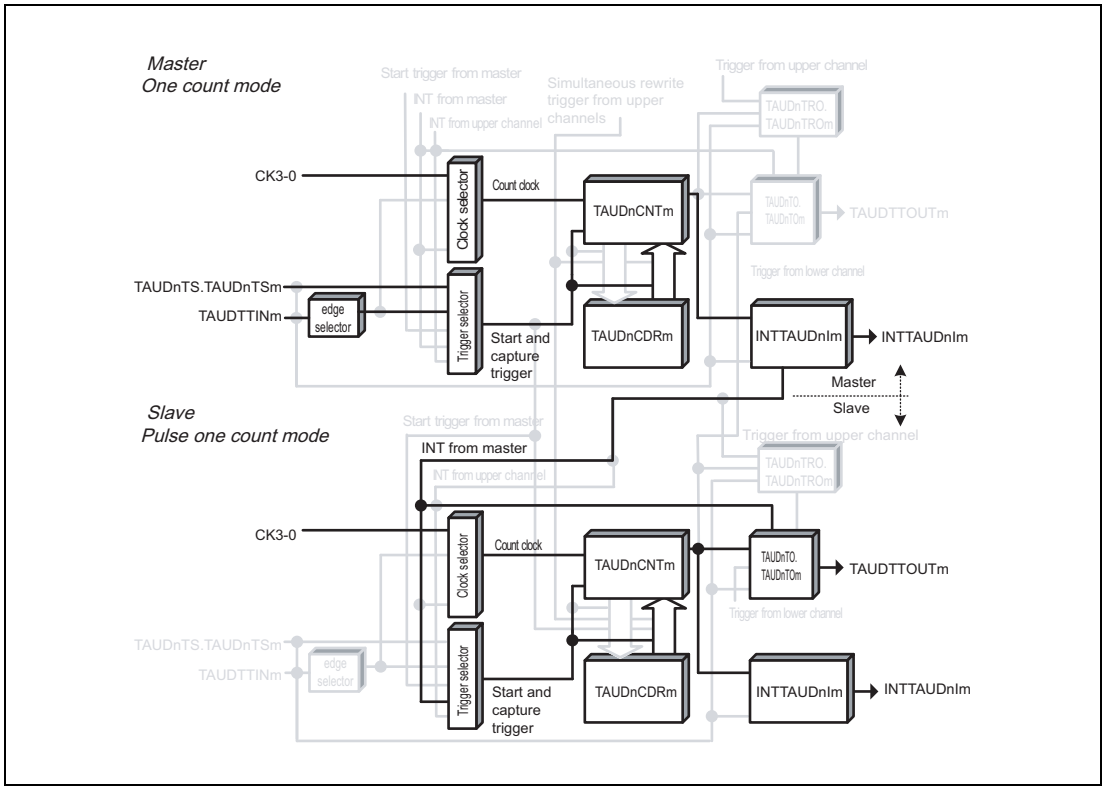


Figure 17.80 Block Diagram of One-Shot Pulse Output Function

The settings in **Figure 17.81, General Timing Diagram of One-Shot Pulse Output Function (in the Case of an External Input Signal)** and **Figure 17.82, General Timing Diagram of One-Shot Pulse Output Function (in the Case of a Software Trigger)** are as follows.

- Start trigger detection is disabled during counting (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

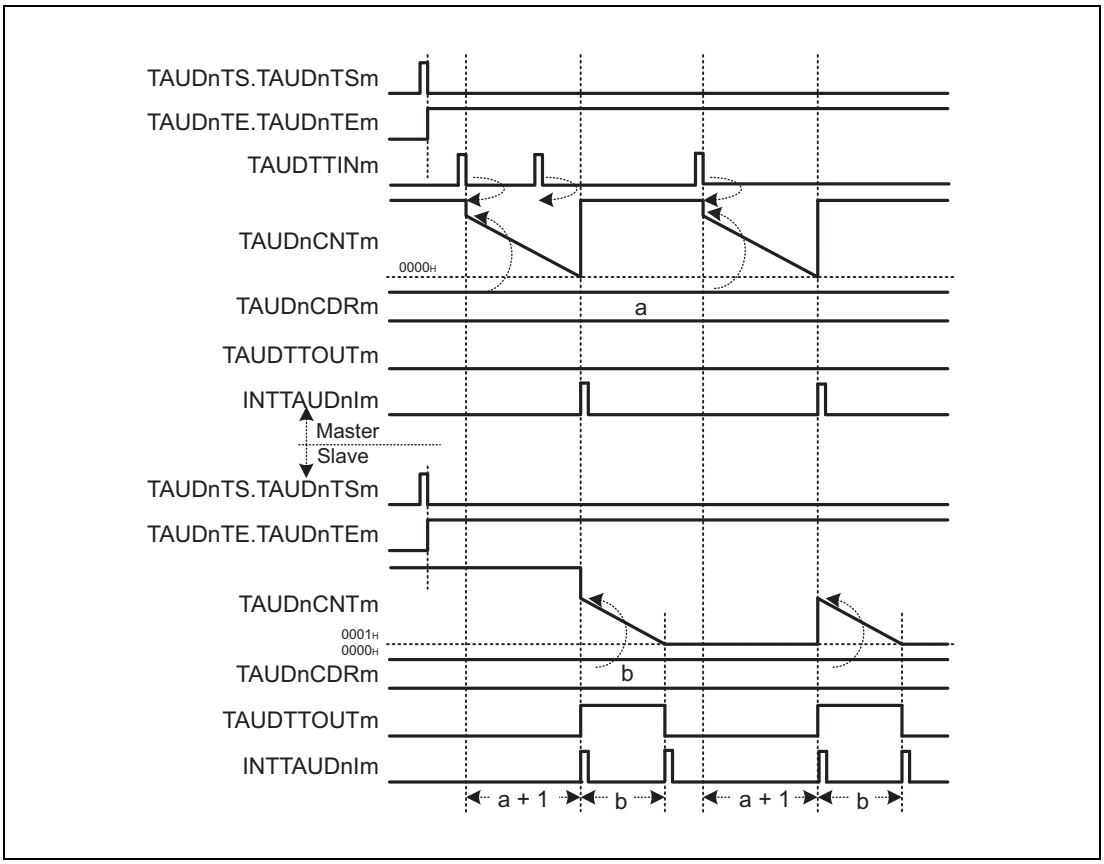


Figure 17.81 General Timing Diagram of One-Shot Pulse Output Function (in the Case of an External Input Signal)

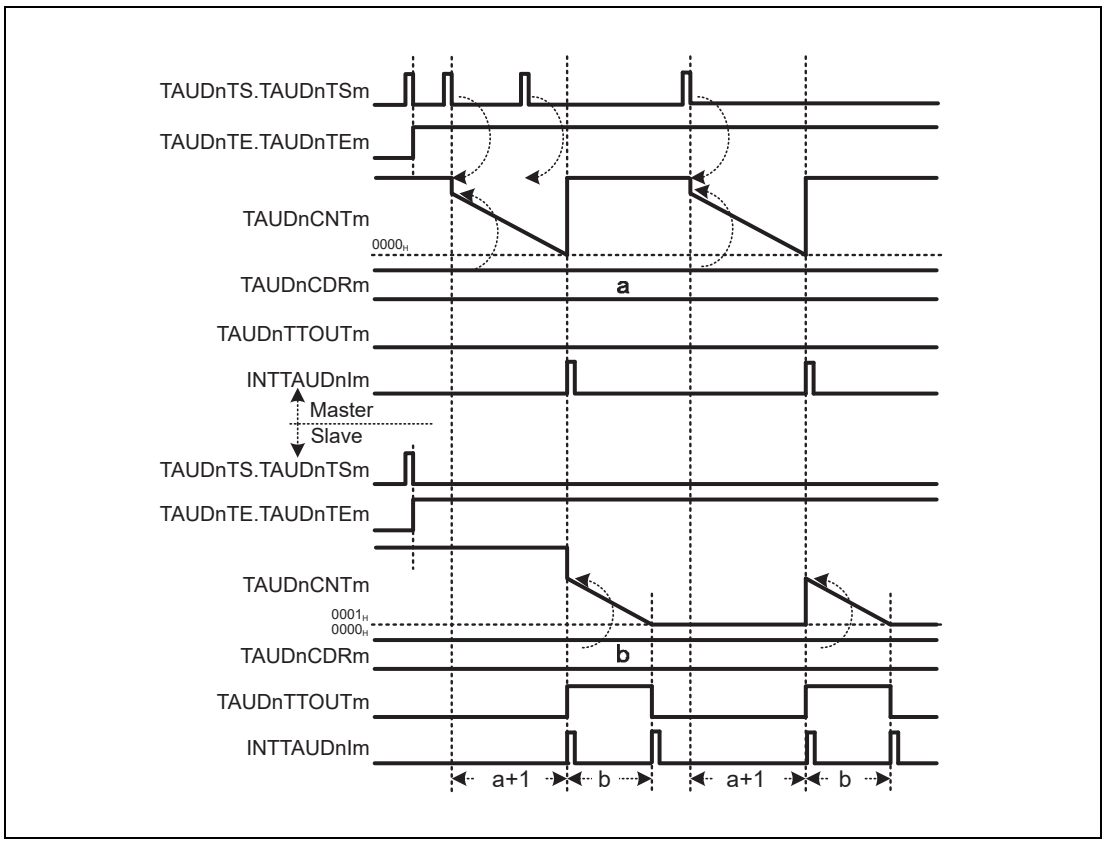


Figure 17.82 General Timing Diagram of One-Shot Pulse Output Function (in the Case of a Software Trigger)

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.124 Contents of TAUDnCMORm Register for Master Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	0: Disables detection of start trigger during count operation. 1: Enables start trigger detection while counting. TAUDnMD0 bit of master and slave channels should have the same value.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.125 Contents of TAUDnCMURm Register for Master Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function. However, this mode can be used with another function or in independent channel output mode controlled by software.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.126 Simultaneous Rewrite Settings for Master Channels of One-Shot Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.127 Contents of TAUDnCMORm Register for Slave Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	0: Disables detection of start trigger during count operation. 1: Enables start trigger detection while counting. TAUDnMD0 bit of master and slave channels should have the same value.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.128 Contents of TAUDnCMURm Register for Slave Channels of One-Shot Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for the slave channel

Table 17.129 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.130 Simultaneous Rewrite Settings for Slave Channels of One-Shot Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for One-Shot Pulse Output Function

Table 17.131 Operating Procedure for One-Shot Pulse Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the master channel awaits a TAUDTTINm input.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	When an effective TAUDTTINm input edge is detected, TAUDnCDRm value of master channel is loaded into TAUDnCNTm to start countdown. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) is reset to $FFFF_H$ and waits for the next valid TAUDTTINm input edge. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to start countdown. • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set. When TAUDnCNTm (slave) reaches 0001_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is reset. If TAUDTTINm input is detected on the master channel during count operation and TAUDnCMORm.TAUDnMD0 = 0, the input is ignored.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) TAUDnCDRm (master) = 0000_H

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

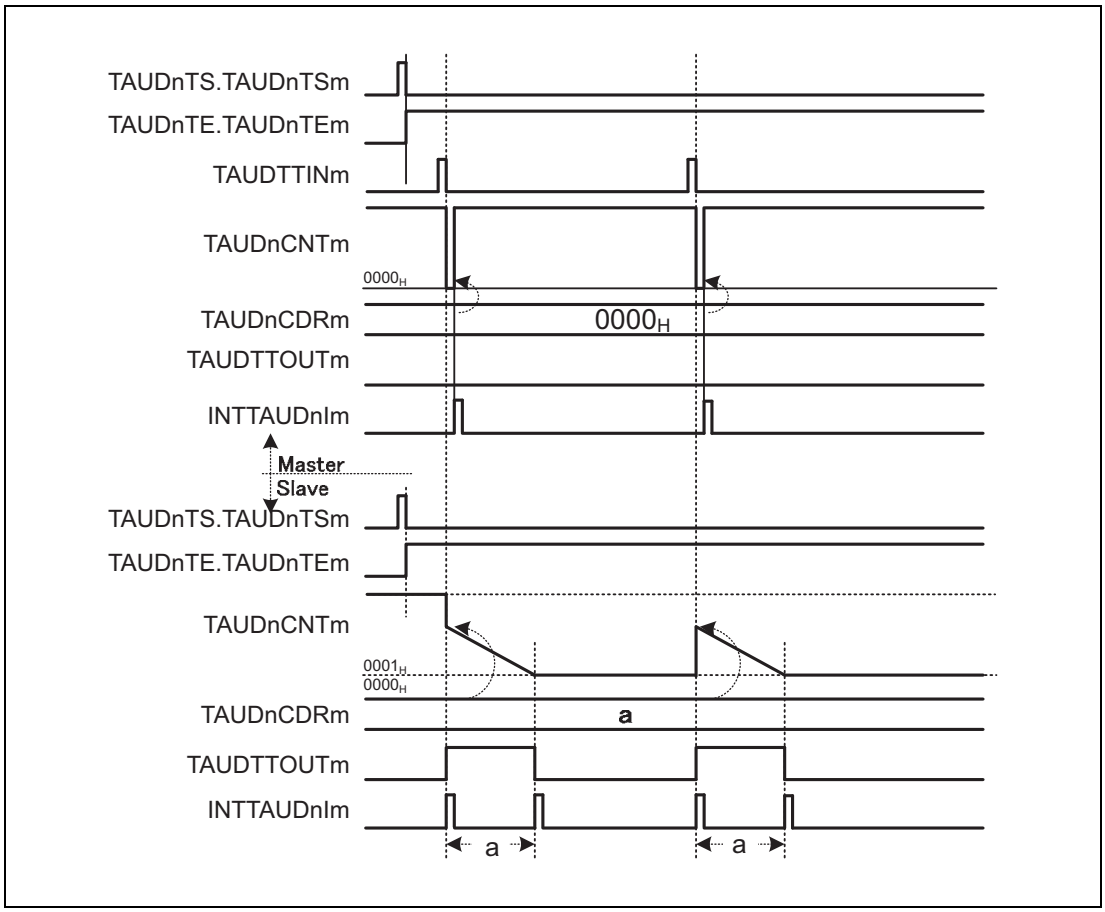


Figure 17.83 TAUDnCDRm (Master) = 0000_H

- When an effective TAUDTTINm input edge is detected, the value 0000_H is written to TAUDnCNTm (master). The counter is set to 0000_H for one count and returns to $FFFF_H$. Thus the slave channel starts to count down one count clock later to TAUDTTINm (master).

(b) TAUDnCDRm (slave) = 0000_H

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

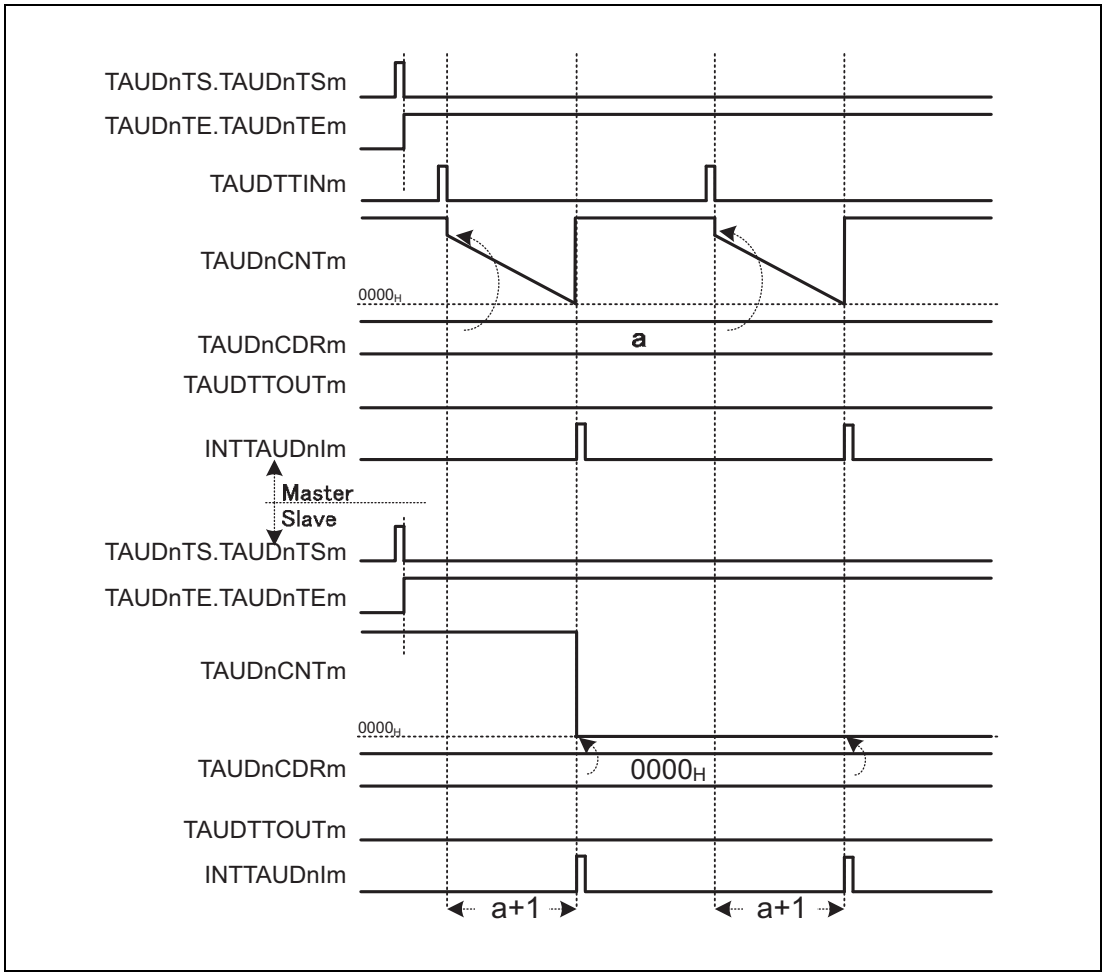


Figure 17.84 TAUDnCDRm (Slave) = 0000_H

- TAUDTTOUTm remains at not active state, because the pulse width is zero.

(c) TAUDnCMORm.TAUDnMD0 = 1

The following settings apply to this diagram:

- Enables start trigger detection while counting. (TAUDnCMORm.TAUDnMD0 = 1)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

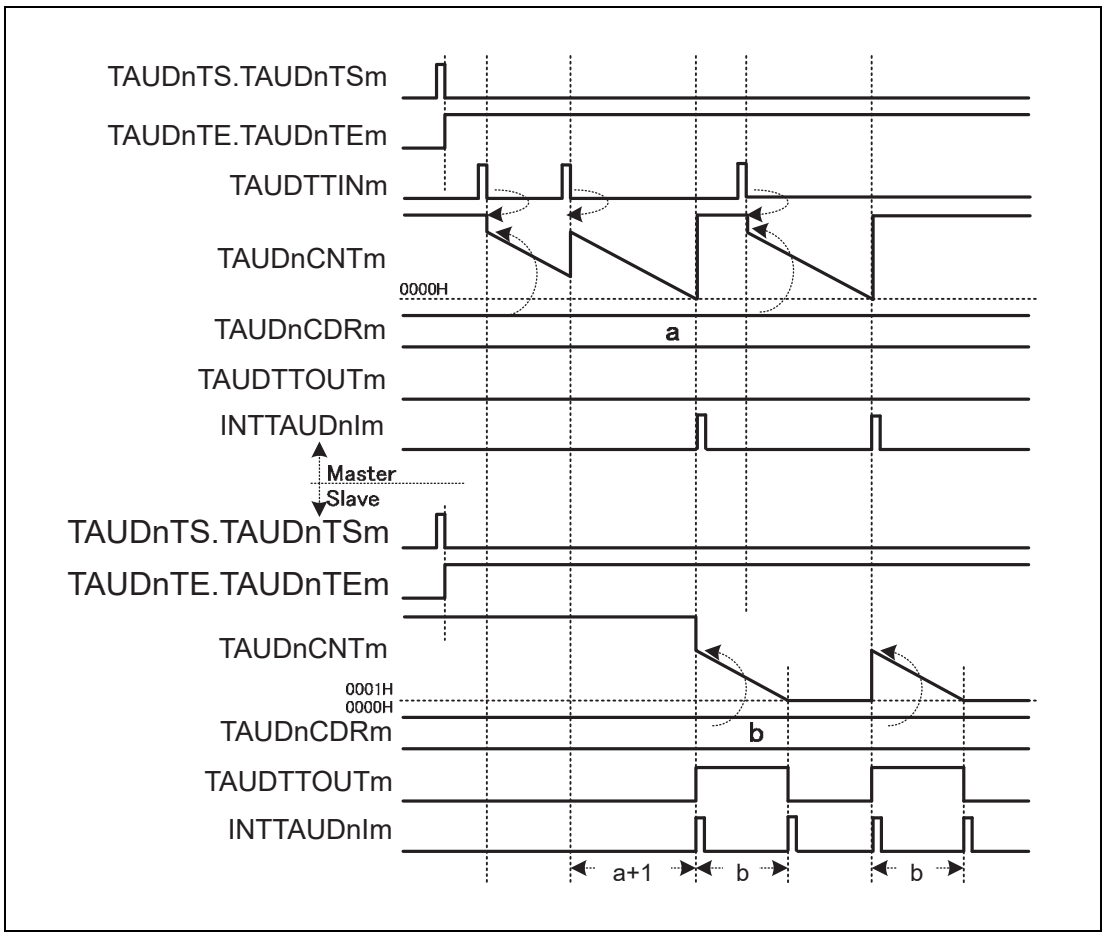


Figure 17.85 TAUDnCMORm.TAUDnMD0 = 1

- If an effective TAUDTTINm input edge is detected while the counter of the master channel counts down, TAUDnCNTm reloads the value of TAUDnCDRm. The counter restarts to count down. This means the delay is extended by the value of TAUDnCNTm at the time an effective TAUDTTINm input edge is detected.

(d) Restarting the master channel while the slave channel is counting

The following settings apply to this diagram:

- Disables detection of start trigger during count operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

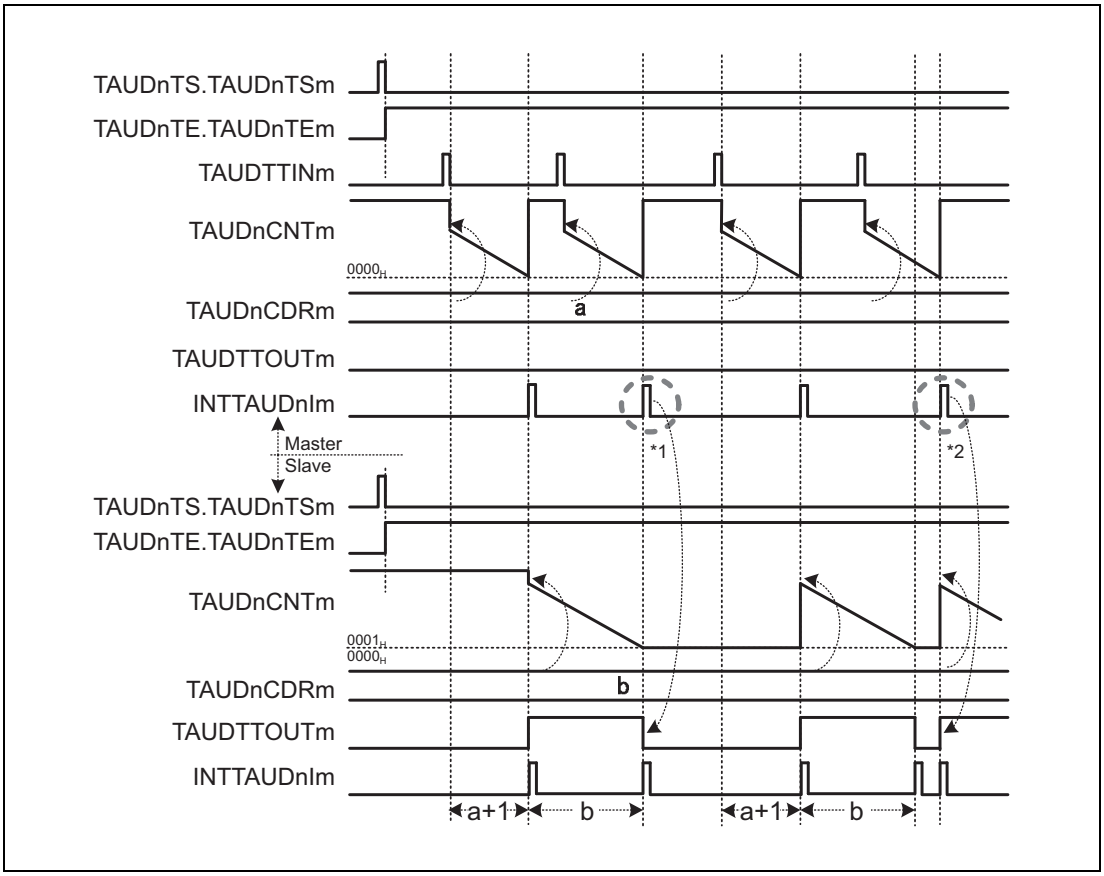


Figure 17.86 Interval of TAUDTTINm ≤ delay time + pulse width + 1

- If the master channel generates an interrupt before the counter of the slave channel has reached 0001_H or exactly when 0001_H is reached (*1), the interrupt (master) is ignored.
- If an interrupt of the master channel occurs when the counter of the slave channel awaits the next trigger, the value of TAUDnCDRm (slave) is reloaded. An interrupt is generated and TAUDTTOUTm toggles. If TAUDnCNTm (master) has started to count down while the TAUDnCNTm (slave) is still counting (*2), TAUDTTOUTm is not output with the expected delay time.
- To generate the correct one-shot pulse, the start trigger for the master channel must be detected while the master and slave channels are waiting for the start trigger, and not while they are counting.

17.4.12.3 Trigger Start PWM Output Function

(1) Overview

Summary

This function generates a PWM output using a master and a slave channel. It enables the pulse cycle (frequency) and the duty of the TAUDTTOUTm to be set. The pulse cycle is specified using the master channel. The duty is specified using the slave channel. The Trigger Start PWM Output Function is identical to the PWM Output Function except that the master channel of this function can be reset by an effective TAUDTTINm input edge.

Prerequisites

- Two channels
- The operation mode of the master channel must be set to Interval Timer Mode (see **Table 17.132, Contents of the TAUDnCMORm Register for the Master Channel of the Trigger Start PWM Output Function**).
- The operation mode of the slave channel must be set to One-Count Mode (see **Table 17.135, Contents of the TAUDnCMORm Register for Slave Channels of the Trigger Start PWM Output Function**.
- The channel output mode of the slave channel must be set to Synchronous Channel Output Mode 1 (see **Section 17.4.4, Channel Output Modes**).
- TAUDTTOUTm is not used with the master channel of this function.

Functional description

The counters (master and slave) are enabled by setting the channel trigger bits (TAUDnTS.TAUDnTSm) to 1. This in turn sets TAUDnTE.TAUDnTEm, enabling counting. The current value of TAUDnCDRm is loaded to TAUDnCNTm, and the counter starts to count down from this value. INTTAUDnIm is generated on the master channel, and a PWM output is realized by setting and resetting TAUDTTOUTm (slave).

- Master channel:
  The current value of TAUDnCDRm is loaded to the counter (TAUDnCNTm), INTTAUDnIm is generated and the counter starts to count down from this value.
  When the counter reaches 0000_H and the pulse cycle time has elapsed, INTTAUDnIm is generated and the counters (master and slave) reload the current TAUDnCDRm values.
  If an effective TAUDTTINm input edge is detected, the counter of the master channel reloads the current TAUDnCDRm value, restarts counting down and generates an interrupt.
- Slave channel:
  When the slave detects an interrupt from the master channel, it starts to count down from the current value of TAUDnCDRm. The TAUDTTOUTm signal is set to the active level.
  When the counter reaches 0000_H (duty time has elapsed), INTTAUDnIm is generated and the TAUDTTOUTm signal is reset. The counter returns to $FFFF_H$ and awaits the next INTTAUDnIm of the master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1 for the master and slave channel, which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channel stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = [TAUDnCDRm (slave) / (TAUDnCDRm (master) + 1)] × 100

– Duty cycle = 0%

TAUDnCDRm (slave) = 0000_H

– Duty cycle = 100%

TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

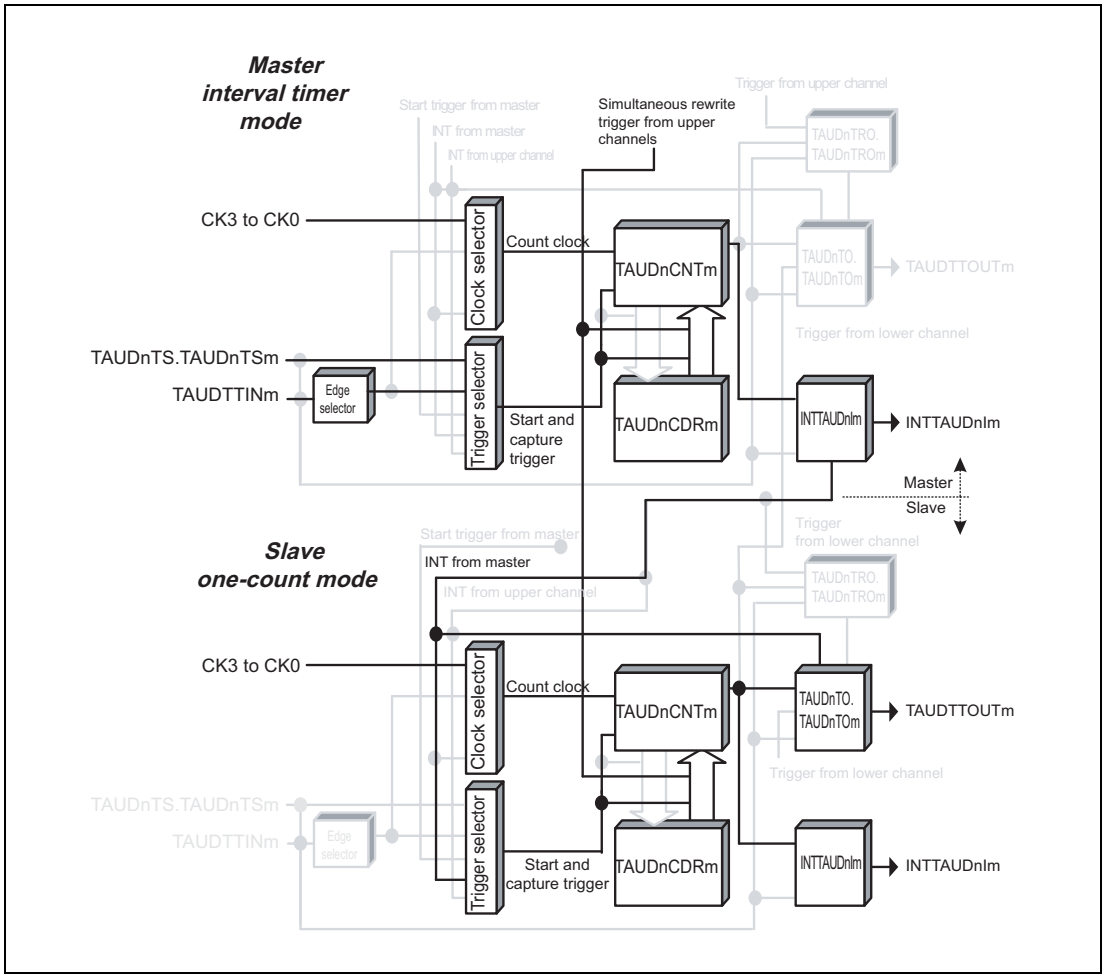


Figure 17.87 Block Diagram for Trigger Start PWM Output Function

The following settings apply to the general timing diagram.

- Detection of rising edge (TAUDnCMURm.TAUDnTIS[1:0] = 01_B)
- Positive logic (TAUDnTOL.TAUDnTOLm (slave) = 0)

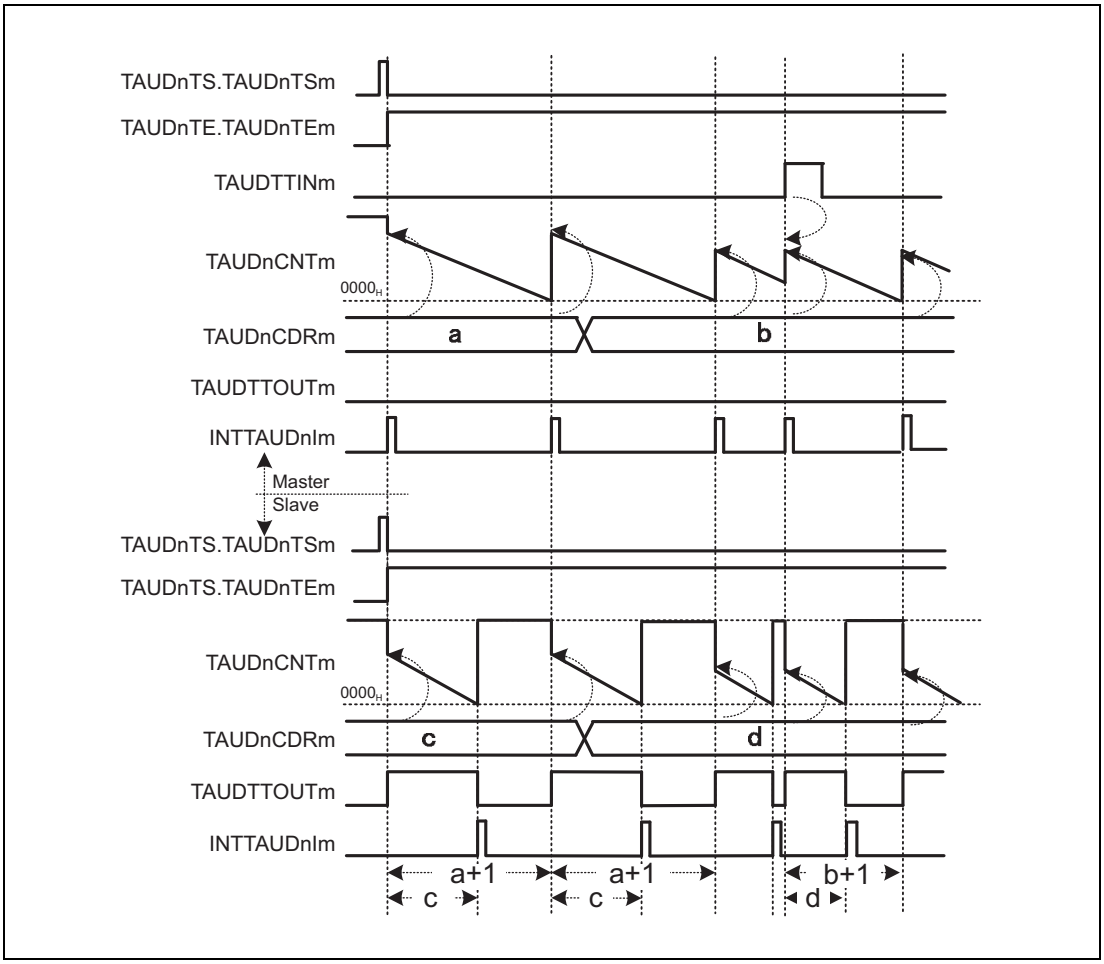


Figure 17.88 General Timing Diagram for Trigger Start PWM Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.132 Contents of the TAUDnCMORm Register for the Master Channel of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bits of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock.
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.133 Contents of the TAUDnCMURm Register for Master Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges 11: Setting prohibited

(c) Channel output mode for master channels

This function does not use channel output mode.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.134 Simultaneous Rewrite Settings for Master Channels of the Trigger Start PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.135 Contents of the TAUDnCMORm Register for Slave Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bits of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock.
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is valid. The value of the TAUDnMD[0] bit of the master and slave channel must be identical.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.136 Contents of the TAUDnCMURm Register for Slave Channels of the Trigger Start PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 17.137 Control Bit Settings in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode.
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation.
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation.
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TAUDnTDEm = 0), set these bits to 0.
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output.
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TAUDnTREm = 0), set these bits to 0.
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel.
TAUDnTME.TAUDnTMEm	0: Disables modulation.

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.138 Simultaneous Rewrite Settings for Slave Channels of the Trigger Start PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for the Trigger Start PWM Output Function

Table 17.139 Operating Procedure for the Trigger Start PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set the TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channels loads the TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • The TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue counting. • The TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to start counting down. • TAUDTTOUTm (slave) is set. When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set to an inactive level. In addition, the counter of slave channel stops. If a TAUDTTINm input is detected on the master channel while the counter is counting down: • TAUDnCNTm (master and slave) reloads the TAUDnCDRm value and counts down. • INTTAUDnIm (master) occurs. • TAUDTTOUTm (slave) is set to the active level.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

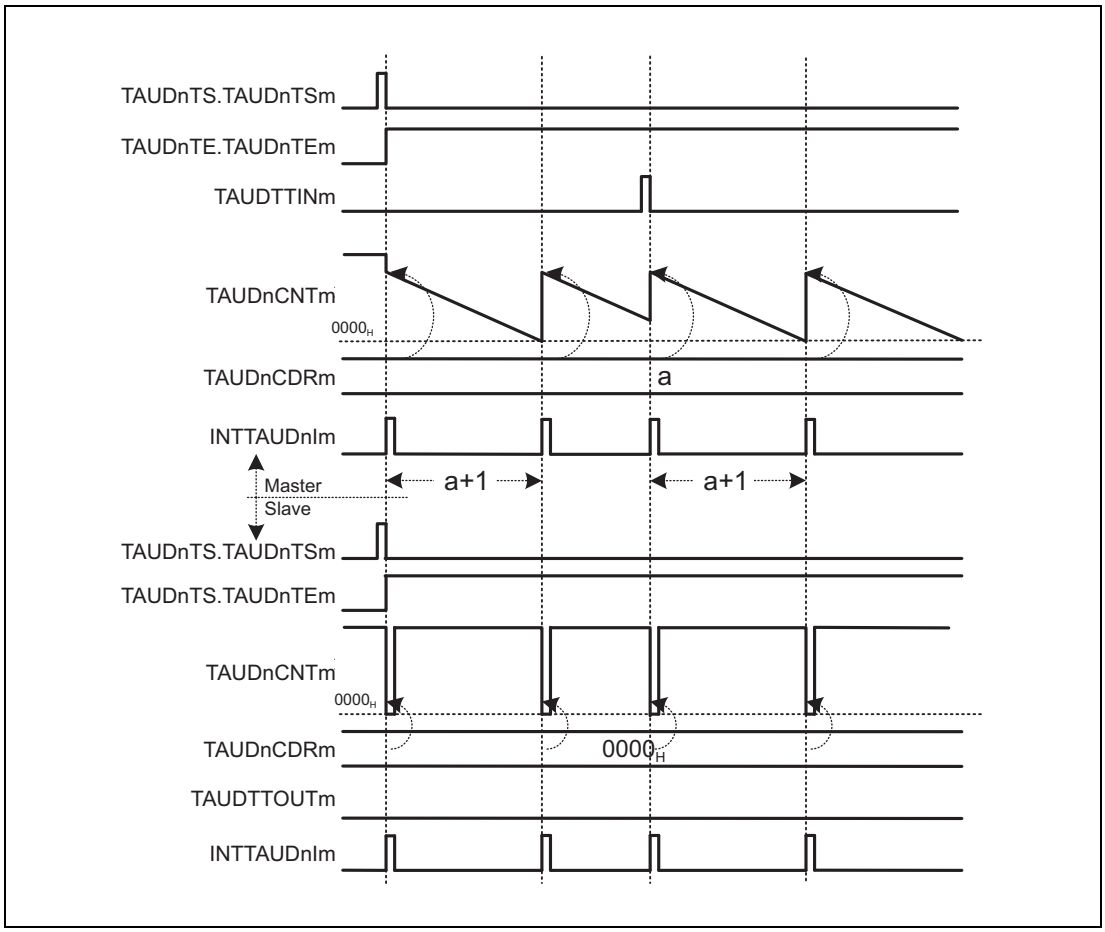


Figure 17.89 TAUDnCDRm (Slave) = 0000_H, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0) Detection of Falling Edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

- Every time the master channel generates an interrupt (INTTAUDnIm), 0000_H is written to TAUDnCNTm (slave). Therefore, TAUDnCNTm (slave) cannot start to count and TAUDTTOUTm remains inactive.
- TAUDnCNTm (slave) generates an interrupt every time the value of TAUDnCDRm is reloaded. The detection of an effective TAUDTTINm input edge has no effect on TAUDTTOUTm (slave).

(b) Duty cycle = 100%

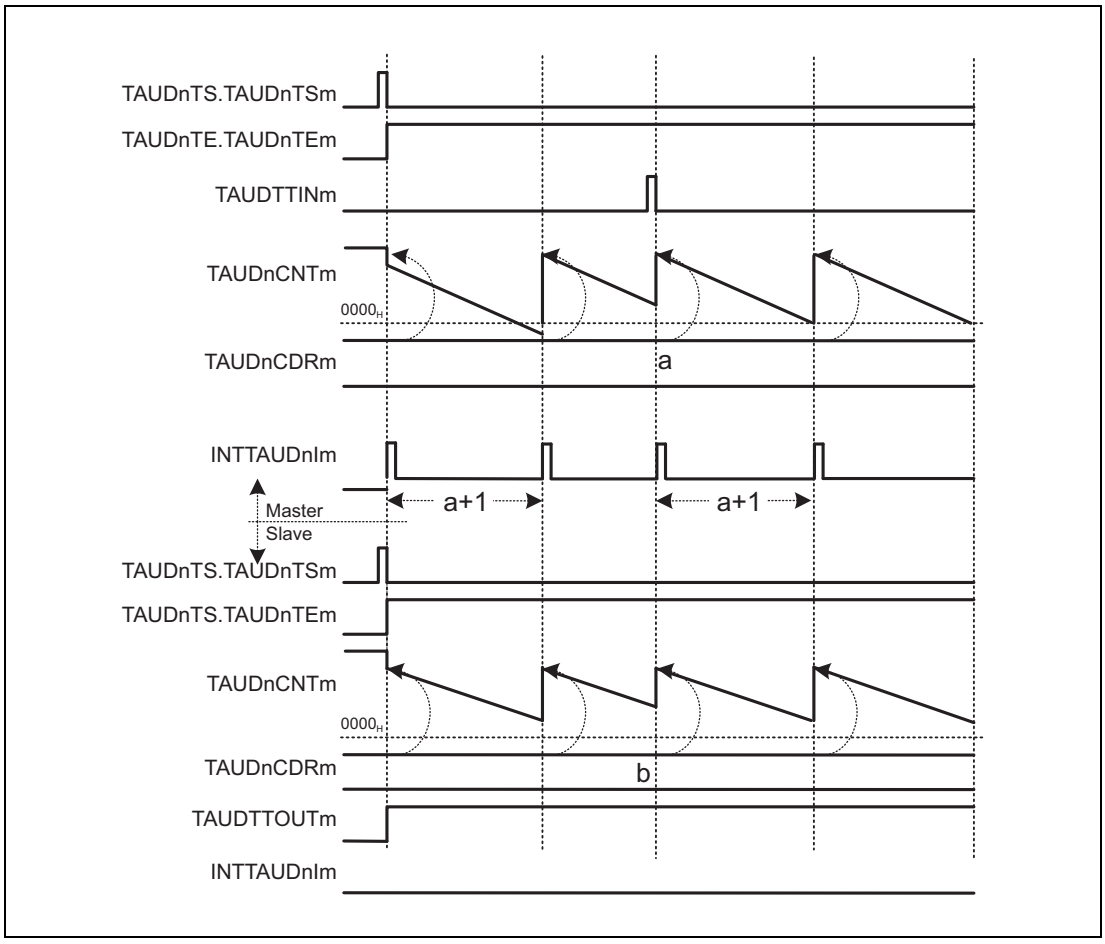


Figure 17.90 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1, Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0) Detection of Falling Edge (TAUDnCMURm.TIS[1:0] = 00_B)

- If the value of TAUDnCDRm (slave) is higher than the value of TAUDnCDRm (master), the counter of the slave channels cannot reach 0000_H and cannot generate interrupts.
  The TAUDTTOUTm remains in the active state.
  The detection of an effective TAUDTTINm input edge has no effect on TAUDTTOUTm (slave).

(c) TAUDTTINm detection and active slave counter

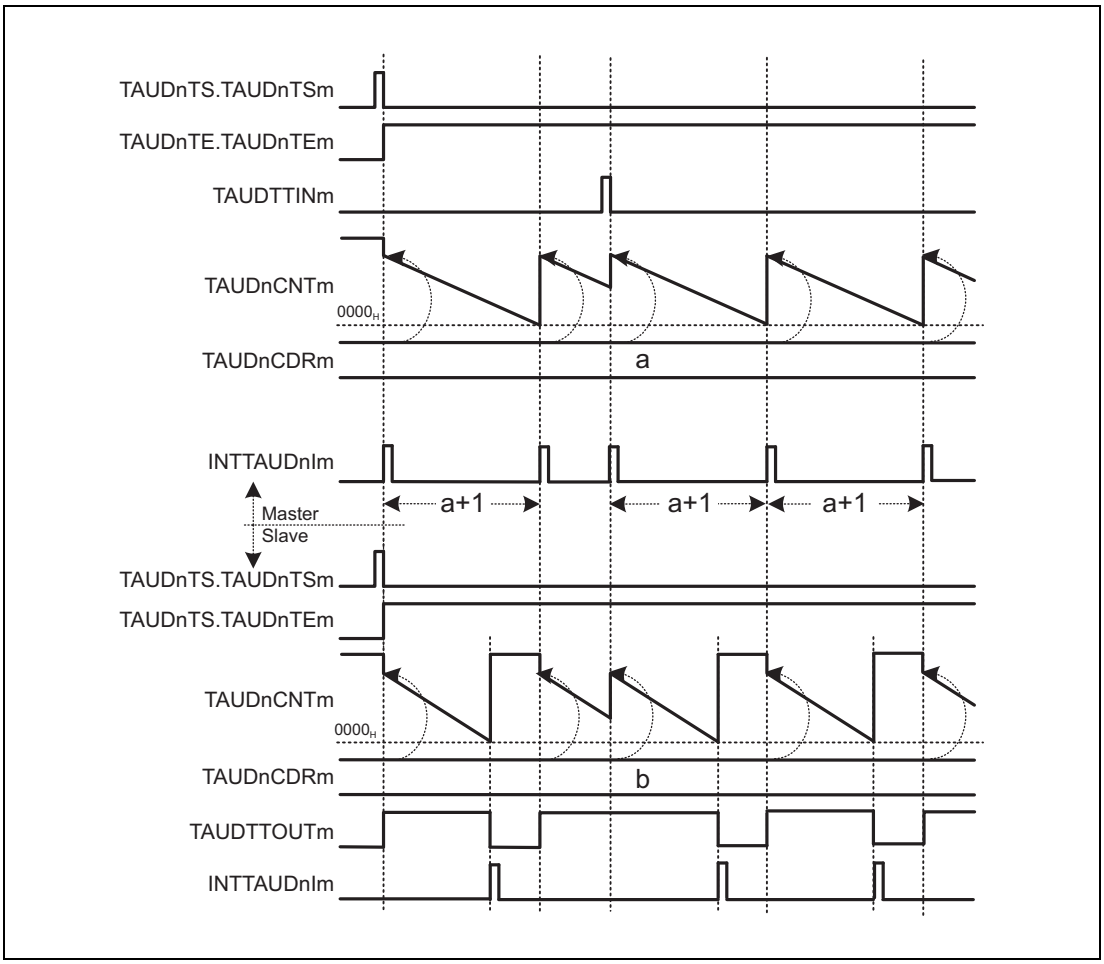


**Figure 17.91 Positive Logic (TAUDnTOL.TAUDnTOLm (Slave) = 0)
Detection of Falling Edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)**

- If TAUDnCNTm (slave) reloads the value of TAUDnCDRm (slave) while it is still counting down, TAUDTTOUTm cannot toggle and extends the duty.
  The duty does not correspond to the value of the slave's data register.

17.4.12.4 Delay Pulse Output Function

(1) Overview

Summary

This function outputs two signals. The pulse width and pulse cycle of the reference signal are defined using the master channel and slave channel 1. Slave channels 2 and 3 output the reference signal with a specified delay. The delay signal is identical to the reference signal, but delayed by the amount specified on slave channel 2.

The signal values are specified in the following way:

- The pulse cycle is specified using the master channel.
- The duty cycle of the reference signal is specified using slave channel 1. The duty cycle of the delay signal is specified using slave channel 3.
- The delay is specified on slave channel 2.

Prerequisites

- Four channels
- The operating mode for the master channel should be set to interval timer mode (see **Table 17.140, Contents of TAUDnCMORm Register for Master Channels of Delay Pulse Output Function**).
- The operating mode for slave channels 1 and 2 should be set to one-count mode (see **Table 17.143, Contents of TAUDnCMORm Register for Slave Channel 1 of Delay Pulse Output Function**).
- The operating mode for slave channel 3 should be set to pulse one-count mode (see **Table 17.147, Contents of TAUDnCMORm Register for Slave Channel 2 of Delay Pulse Output Function**).
- TAUDTTOUTm is not used with the master channel and slave channel 2.
- The channel output mode for slave channel 1 should be set to synchronous channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- The channel output mode for slave channel 3 should be set to independent channel output mode 2 (see **Section 17.4.4, Channel Output Modes**).

Functional description

The counters of the channel group are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm to 1, enabling count operation.

- Master channel:
  The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from this value. INTTAUDnIm is generated on the master channel.
  When the counter value of master channel reaches 0000_H and pulse cycle time has elapsed, INTTAUDnIm is generated. The TAUDnCDRm value is reloaded into the counter to perform count down.
- Slave channels 1 and 2:
  Slave channels 1 and 2 start to count down from the current TAUDnCDRm value when detecting an interrupt from the master channel. TAUDTTOUTm signal (slave 1) is set.

– Slave channel 1:
When the counter of slave channel 1 reaches 0000_H (duty time has elapsed), INTTAUDnlm is generated and TAUDTTOUTm signal is reset. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm of master channel.

– Slave channel 2:
When the counter of slave channel 2 reaches 0000_H and delay time has elapsed, INTTAUDnIm is generated. The counter is reset to $FFFF_H$ and waits for the next INTTAUDnlm of master channel.
Generating INTTAUDnIm (slave channel 2) triggers the counter of slave channel 3.

- Slave channel 3:
When slave channel 3 detects an interrupt from slave channel 2, its counter starts counting down from the current value of TAUDnCDRm. INTTAUDnIm is generated and the TAUDTTOUTm signal (slave channel 3) is set.
When the counter of slave channel 3 reaches 0001_H, INTTAUDnIn is generated and the TAUDTTOUTm signal is reset.
The delayed PWM pulse is output from slave channel 3.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but their values are retained. The counter can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

Duty width 1 = (TAUDnCDRm (slave 1)) × count clock cycle

Delay width = (TAUDnCDRm (slave 2) + 1) × count clock cycle

Duty width 2 = (TAUDnCDRm (slave 3)) × count clock cycle

However, the delay width shall be set within the following range:

$0000_H \leq$ TAUDnCDRm (slave 2) < TAUDnCDRm (master)

NOTES

1. The waveform of TAUDTTOUTm (slave 3) becomes the waveform made by delaying the waveform of TAUDTTOUTm (slave 1) by the quantity generated by slave 2. It is impossible to make a delay longer than the pulse cycle.
2. If TAUDnINTm of slave 2 is generated while slave 3 is counting, slave 3 restarts operation. Therefore, the waveform of TAUDTTOUTm (slave 3) is retained on the active level. In this case, TAUDTTOUTm (Slave-CH-3) cannot output the waveform generated by delaying the basic pulse of TAUDTTOUTm (Slave-CH-1).

(3) Block Diagram and General Timing Diagram

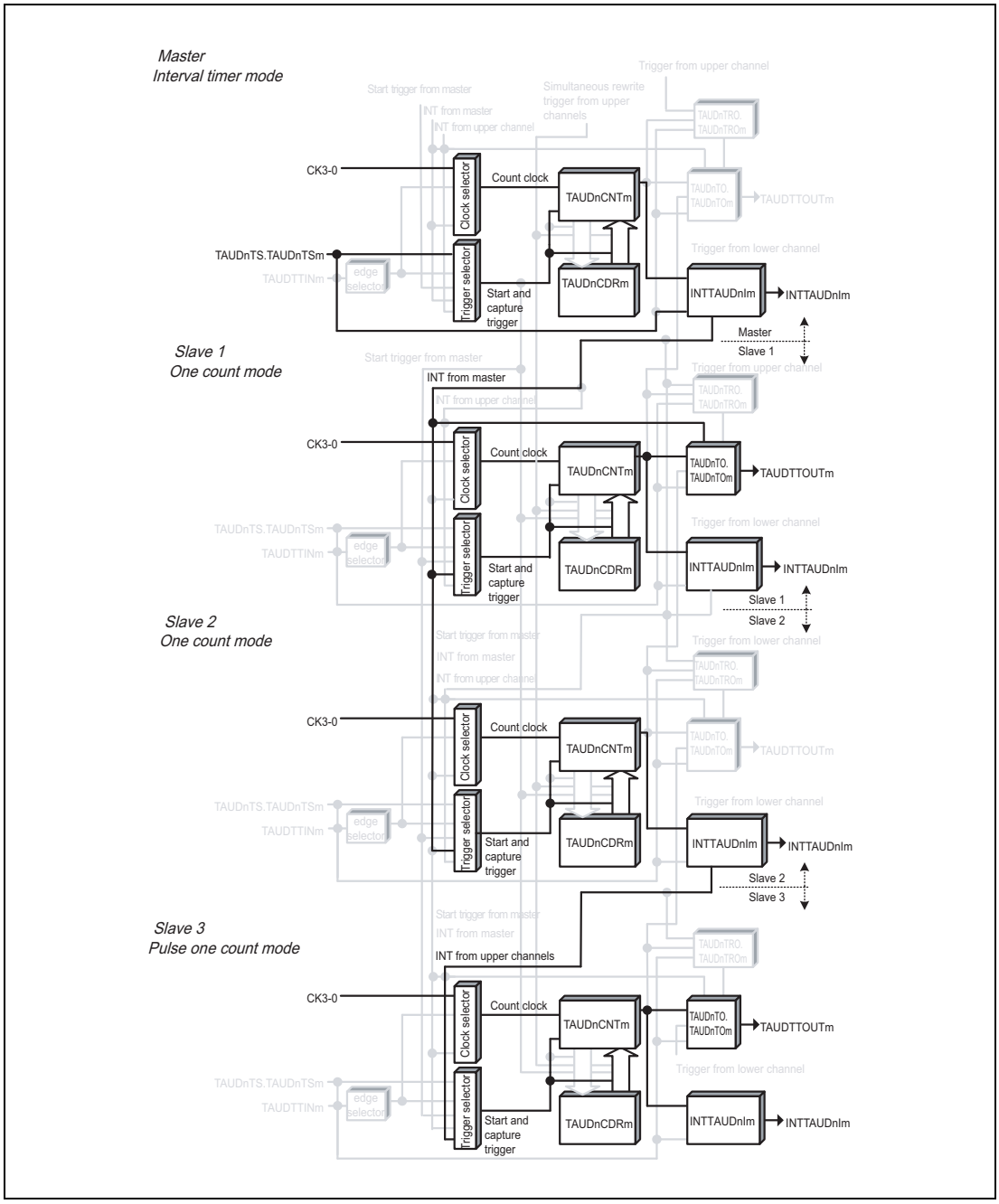


Figure 17.92 Block Diagram of Delay Pulse Output Function

The following settings apply to the general timing diagram.

- Slave channel 1: Positive logic (TAUDnTOL.TAUDnTOLm = 0)
- Slave channel 3: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

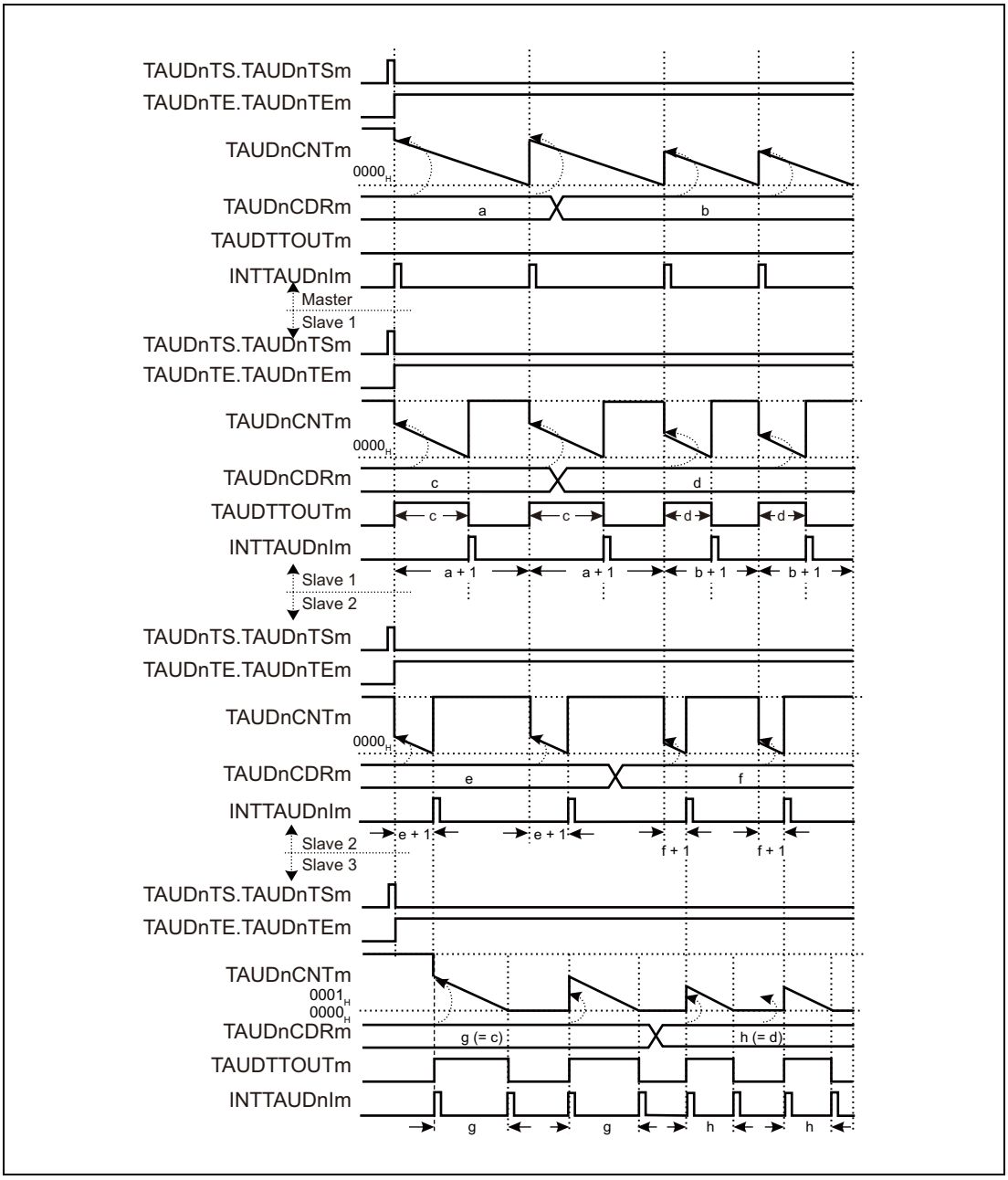


Figure 17.93 General Timing Diagram of Delay Pulse Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.140 Contents of TAUDnCMORm Register for Master Channels of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.141 Contents of TAUDnCMURm Register for Master Channels of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used for master channels with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.142 Simultaneous Rewrite Settings for Master Channels of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.143 Contents of TAUDnCMORm Register for Slave Channel 1 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.144 Contents of TAUDnCMURm Register for Slave Channel 1 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 1

Table 17.145 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	0: Disables the operation as a real-time output trigger channel
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.146 Simultaneous Rewrite Settings for Slave Channel 1 of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Register Settings for Slave Channel 2

(a) TAUDnCMORm for slave channel 2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDnMAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDnMD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.147 Contents of TAUDnCMORm Register for Slave Channel 2 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.148 Contents of TAUDnCMURm Register for Slave Channel 2 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 2

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for slave channel 2

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.149 Simultaneous Rewrite Settings for Slave Channel 2 of Delay Pulse Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(7) Register Settings for Slave Channel 3

(a) TAUDnCMORm for slave channel 3

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.150 Contents of TAUDnCMORm Register for Slave Channel 3 of Delay Pulse Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	101: INTTAUDnIm of upper channel (m - 1) is a start trigger regardless of master setting.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1010: Pulse one-count mode
0	TAUDnMD0	1: Valid start trigger during operation

(b) TAUDnCMURm for slave channel 3

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.151 Contents of TAUDnCMURm Register for Slave Channel 3 of Delay Pulse Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 3

Table 17.152 Control Bit Settings in Independent Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channel 3

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.153 Simultaneous Rewrite Settings for Slave channel 3 of Delay Pulse Output Function

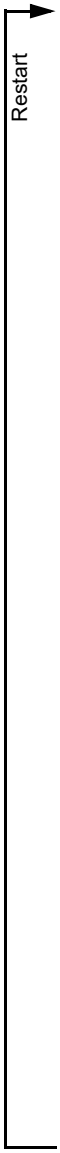
Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Master channel is simultaneous rewrite control channel.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(8) Operating Procedure for Delay Pulse Output Function

Table 17.154 Operating Procedure for Delay Pulse Output Function (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channel 2: Set TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (6), Register Settings for Slave Channel 2**. Slave channel 3: set the TAUDnCMORm and TAUDnCMURm registers and the channel output mode as described in **Section (7), Register Settings for Slave Channel 3**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.

Table 17.154 Operating Procedure for Delay Pulse Output Function (2/2)



	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master channel and slave channels 1 and 2 start. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave channel 1) is set.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel and slave channels 1 and 2 load TAUDnCDRm value and count down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 1/2) to start countdown. • TAUDTTOUTm (slave 1) is set. When TAUDnCNTm (slave 1) reaches 0000_H: • INTTAUDnIm (slave 1) occurs. • TAUDTTOUTm (slave 1) is reset. When TAUDnCNTm (slave 2) reaches 0000_H: • INTTAUDnIm (slave 2) occurs. • TAUDTTOUTm (slave 3) is set. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 3) to start a countdown operation. When TAUDnCNTm (slave 3) reaches 0000_H: • INTTAUDnIm (slave 3) occurs. • TAUDTTOUTm (slave 3) is reset.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

(9) Specific Timing Diagrams

(a) Duty cycle (slave 3) = 100%

The following values apply to **Figure 17.94**:

- TAUDnCDRm (master) = $000A_H$
- TAUDnCDRm (slave 1) = $000B_H$
- TAUDnCDRm (slave 2) = 0000_H
- TAUDnCDRm (slave 3) = $000B_H$

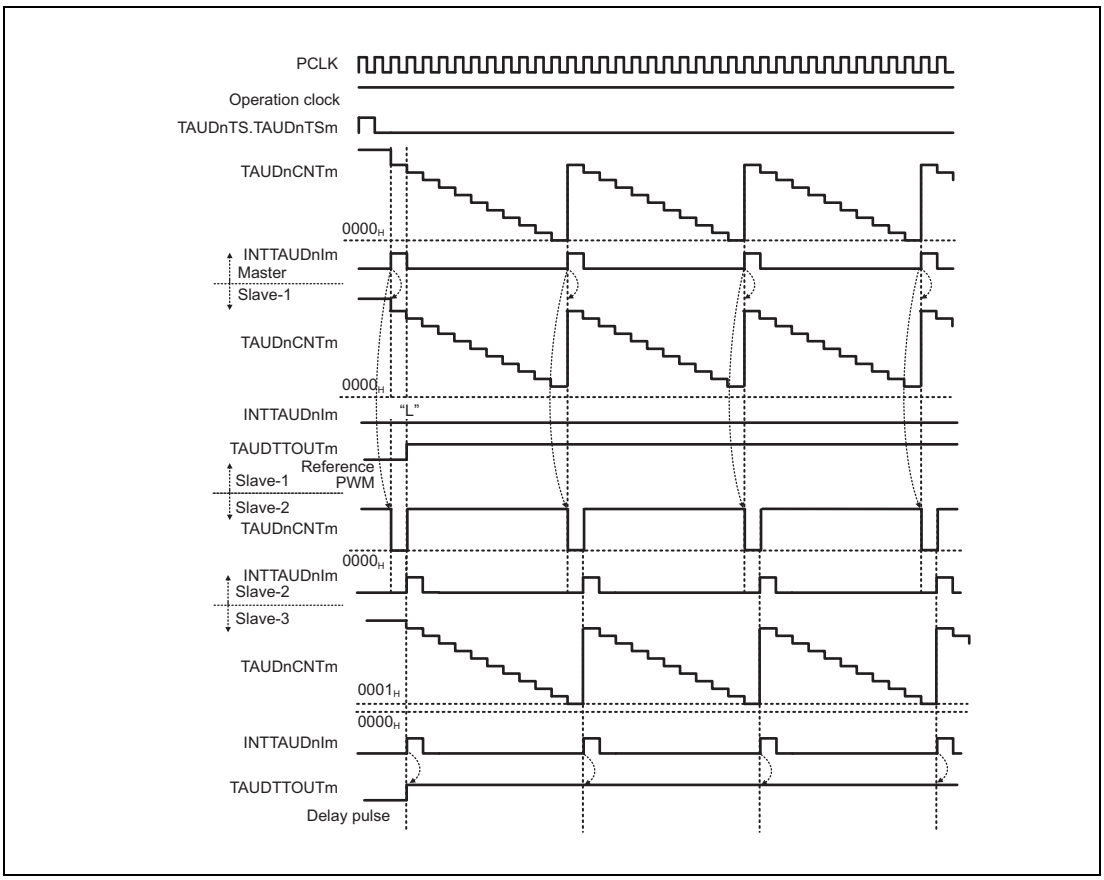


Figure 17.94 Duty Cycle (slave 3) = 100%

- If the value of TAUDnCDRm (slave 1 and 3) is higher than the value of TAUDnCDRm (master), the counter of slave channel 1 cannot reach 0000_H and cannot generate interrupts. TAUDTTOUTm of channels 1 and 3 remain in the active state.

(b) TAUDTTOUTm (slave 1) = TAUDTTOUTm (slave 3)

The following values apply to **Figure 17.95**.

- TAUDnCDRm (master) = $000A_H$
- TAUDnCDRm (slave 1) = 0005_H
- TAUDnCDRm (slave 2) = 0000_H
- TAUDnCDRm (slave 3) = 0005_H

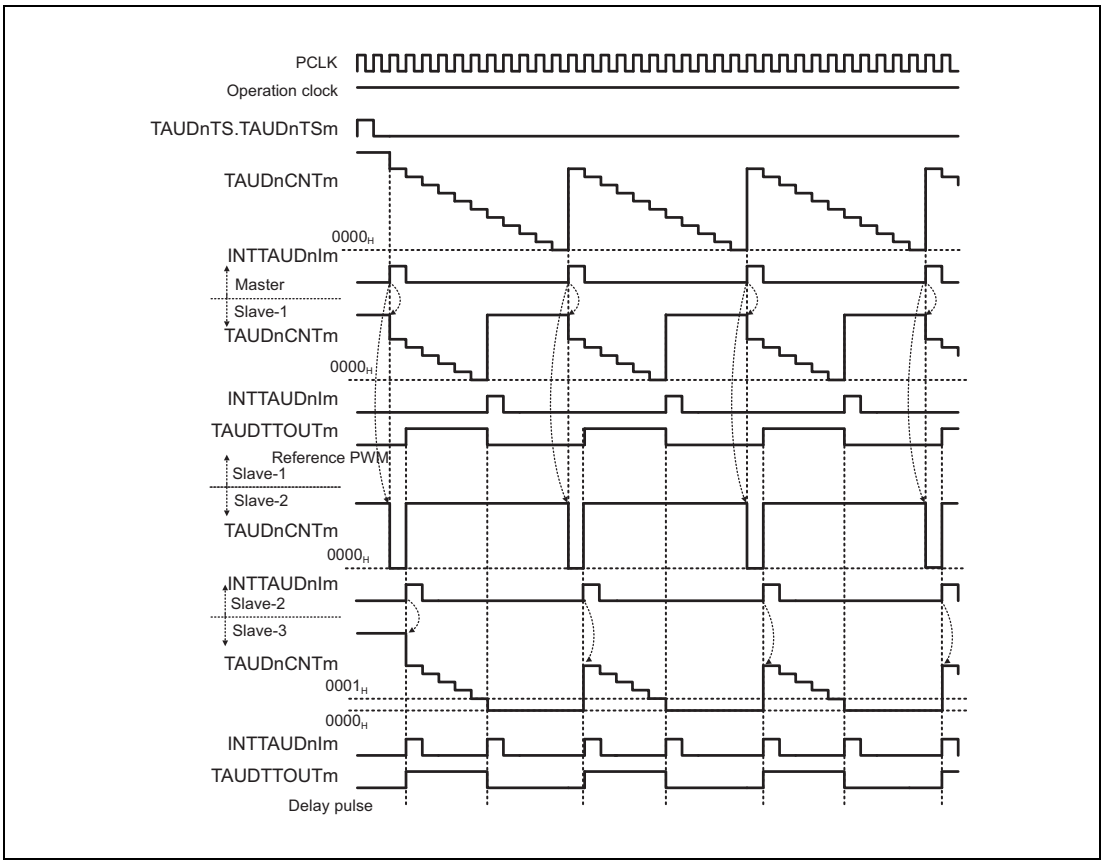


Figure 17.95 TAUDTTOUTm (Slave 1) = TAUDTTOUTm (Slave 3)

- If TAUDnCDRm (slave 2) = 0000_H, the counter of slave channel 3 starts counting one count clock later than the counter of slave channel 1. The reference pulse and the delay pulse are output with a delay of one clock count.

17.4.12.5 Offset Trigger Output Function

(1) Overview

Summary

This function generates a PWM output using a master channel and a slave channel, enabling the pulse width (duration) of the TAUDTTOUTm to be set. The pulse cycle is set by detecting an effective input edge of master channel. The pulse width is specified on the slave channel.

Prerequisites

- Two channels
- The operating mode for the master channel should be set to capture mode (see **Table 17.155, Contents of TAUDnCMORm Register for Master Channels of Offset Trigger Output Function**).
- The operating mode for slave channels should be set to one-count mode (see **Table 17.158, Contents of TAUDnCMORm Register for Slave Channels of Offset Trigger Output Function**).
- The output mode for slave channels should be set to synchronous channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- TAUDTTOUTm is not used with the master channel of this function.

Functional description

The counter can be started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This makes TAUDnTE.TAUDnTEm = 1, enabling the counter to count up. The master channel counter (TAUDnCNTm) starts to count up from 0000_H.

- Master channel:
  When an effective TAUDTTINm input edge is detected, the current value of the counter (TAUDnCNTm) is loaded into the data register of master channel (TAUDnCDRm). INTTAUDnIm is generated and the counter restarts to count up from 0000_H.
- Slave channels:
  The INTTAUDnIm of master channel sets the TAUDTTOUTm (slave) signal and triggers the counter of the slave channel. The current value of TAUDnCDRm (slave) is loaded into TAUDnCNTm (slave) and the counter starts to count down from this value.
  When the counter reaches 0000_H (duty time has elapsed), INTTAUDnIm is generated and TAUDTTOUTm signal is reset. The counter returns to $FFFF_H$ and awaits the next INTTAUDnIm of master channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

(2) Equations

Pulse width = (TAUDnCDRm (slave) + 1) × count clock cycle

Duty cycle [%] = [TAUDnCDRm (slave)/TAUDnCDRm (master) + 1)] × 100

– Duty cycle = 0%
  TAUDnCDRm (slave) = 0000_H
– Duty cycle = 100%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

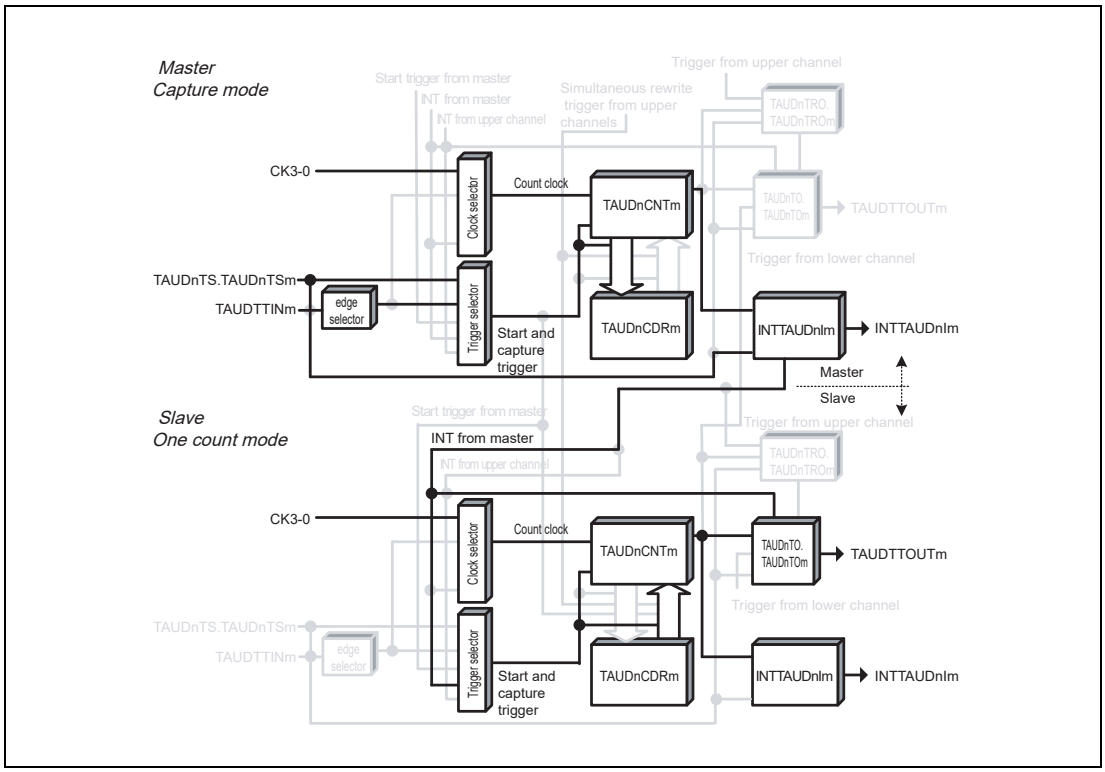


Figure 17.96 Block Diagram of Offset Trigger Output Function

The following settings apply to the general timing diagram.

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

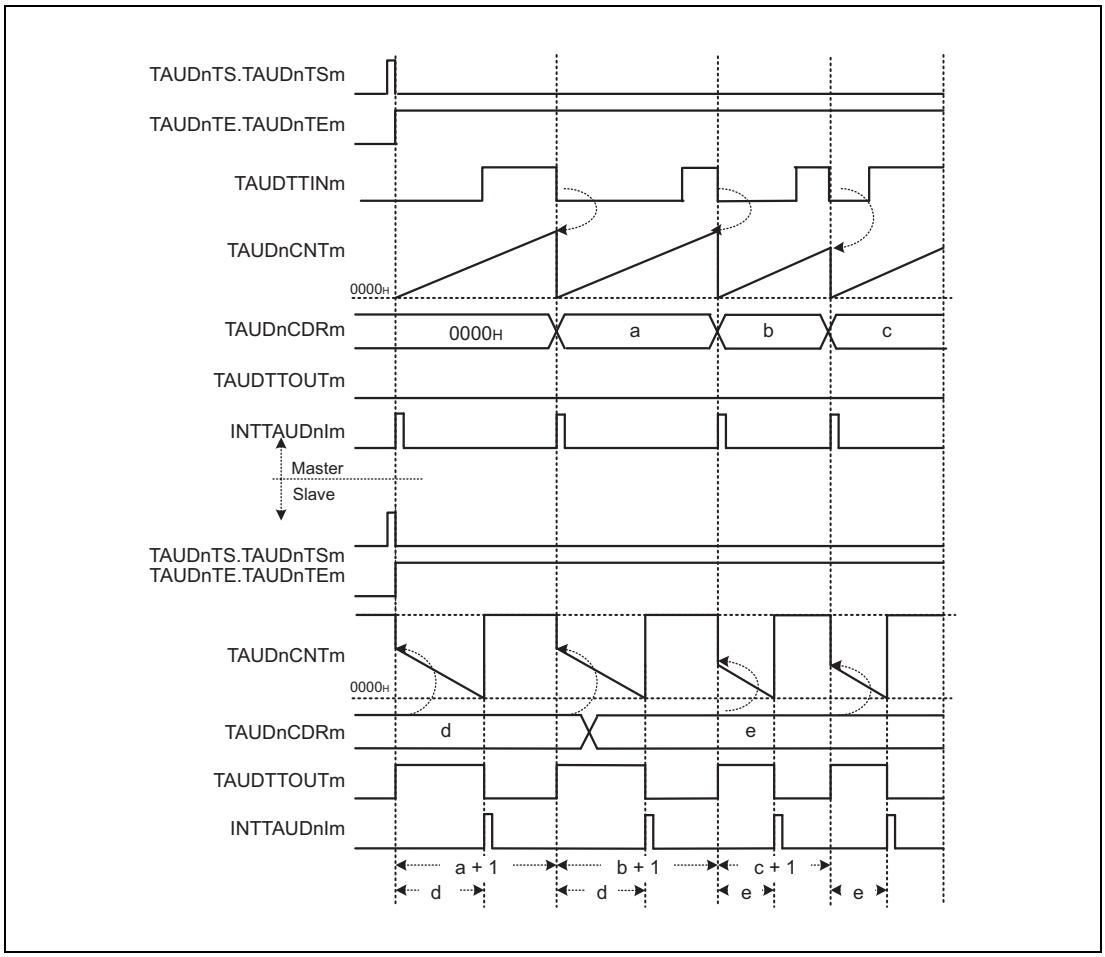


Figure 17.97 General Timing Diagram of Offset Trigger Output Function

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.155 Contents of TAUDnCMORm Register for Master Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	001: An effective TAUDTTINm input edge signal is used as the start trigger
7, 6	TAUDnCOS [1:0]	11: Capture register is updated upon detection of an effective TAUDTTINm input edge or when a counter overflow occurs: – Detection of an effective TAUDTTINm input edge: The counter value is written into TAUDnCDRm. – Occurrence of overflow: $FFFF_H$ is written into TAUDnCDRm. An effective TAUDTTINm input edge to be detected next is ignored. TAUDnCSRm.TAUDnOVF is set when a counter overflow occurs, and cleared by setting TAUDnCSCm.TAUDnCLOV = 1.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0010: Capture mode
0	TAUDnMD0	1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.156 Contents of TAUDnCMURm Register for Master Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Detection of falling edge 01: Detection of rising edge 10: Detection of rising and falling edges

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.157 Simultaneous Rewrite Settings for Master Channels of Offset Trigger Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.158 Contents of TAUDnCMORm Register for Slave Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.159 Contents of TAUDnCMURm Register for Slave Channels of Offset Trigger Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 17.160 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.161 Simultaneous Rewrite Settings for Slave Channels of Offset Trigger Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(6) Operating Procedure for Offset Trigger Output Function

Table 17.162 Operating Procedure for Offset Trigger Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and channel output mode as described in **Section (5), Register Settings for Slave Channels**. The TAUDnCDRm register of master channel functions as a capture register. Set the value of TAUDnCDRm register of slave channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start: • TAUDnCNTm (master) counts up. • TAUDnCDRm value is loaded into TAUDnCNTm (slave) to perform count down. INTTAUDnIm is generated on the master channel and TAUDTTOUTm (slave) is set.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCSCm.TAUDnCLOV can be set to 1. TAUDnCDRm of slave channel can be changed after the generation of INTTAUDnIm (master). TAUDnCNT.TAUDnCNTm and TAUDnCSRm can be read at any time.	When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is reset. When TAUDTTINm input edge is detected on the master channel: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) is reset to 0000_H and then continues count operation subsequently. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave) to perform count down. • TAUDTTOUTm (slave) is set.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to this diagram:

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

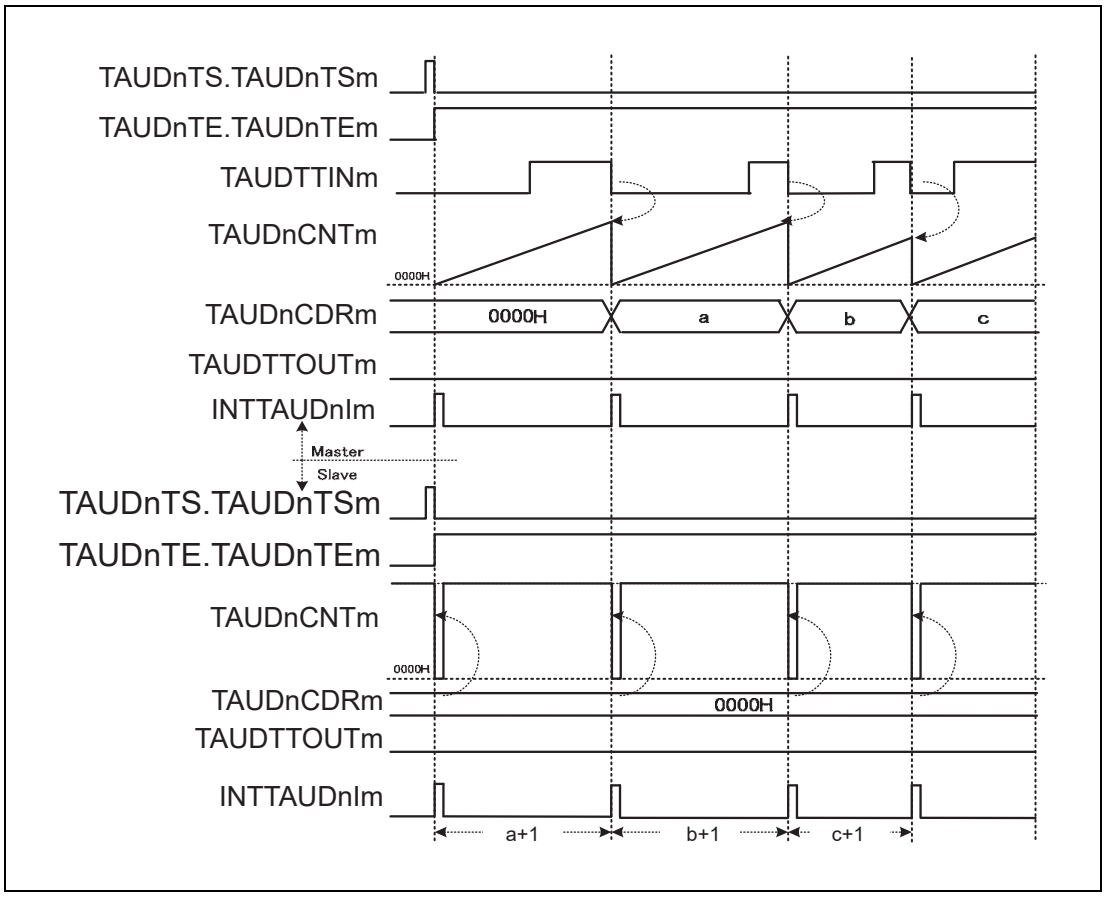


Figure 17.98 TAUDnCDRm (Slave) = 0000_H

- TAUDTTOUTm (slave) is set. When TAUDnCDRm (slave) = 0000_H, 0000_H is written to TAUDnCNTm every time the master channel generates an interrupt (INTTAUDnIm), and TAUDnCNTm cannot start to count. The TAUDTTOUTm remains at not active state.
- TAUDnCNTm (slave) generates an interrupt every time the value of TAUDnCDRm is reloaded. The slave and the master channel generate interrupts in the same cycle.

(b) Duty cycle = 100%

The following settings apply to this diagram:

- Detection of falling edge (TAUDnCMURm.TAUDnTIS[1:0] = 00_B)

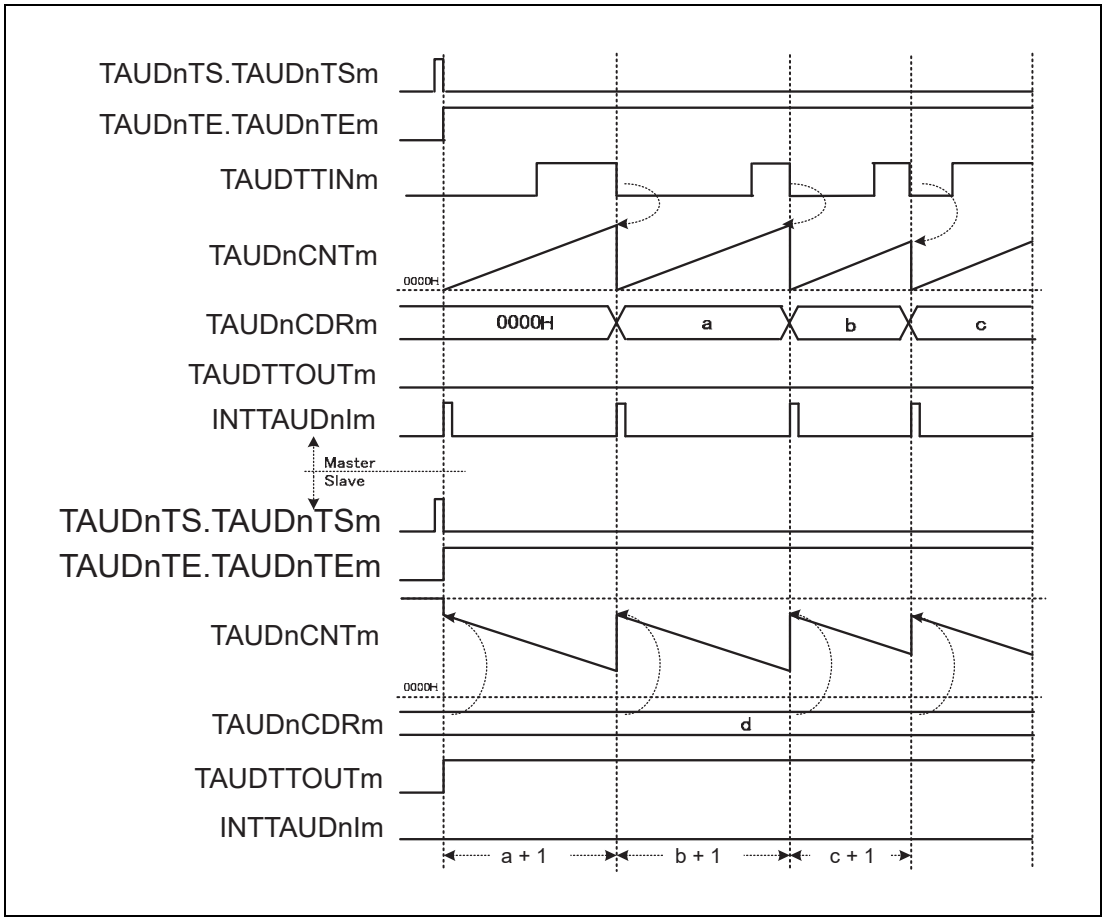


Figure 17.99 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1

- If the value TAUDnCDRm (slave) is higher than the interval of effective input edges, the counter of the slave channel cannot reach 0000_H and cannot generate interrupts. The TAUDTTOUTm remains at active state.

17.4.12.6 A/D Conversion Trigger Output Function Type 1

(1) Overview

Summary

This function is identical to **Section 17.4.12.1, PWM Output Function**, except that TAUDTTOUTm is not output.

This is achieved by setting the channel output mode for the slave to independent channel output mode controlled by software.

(2) Block Diagram and General Timing Diagram

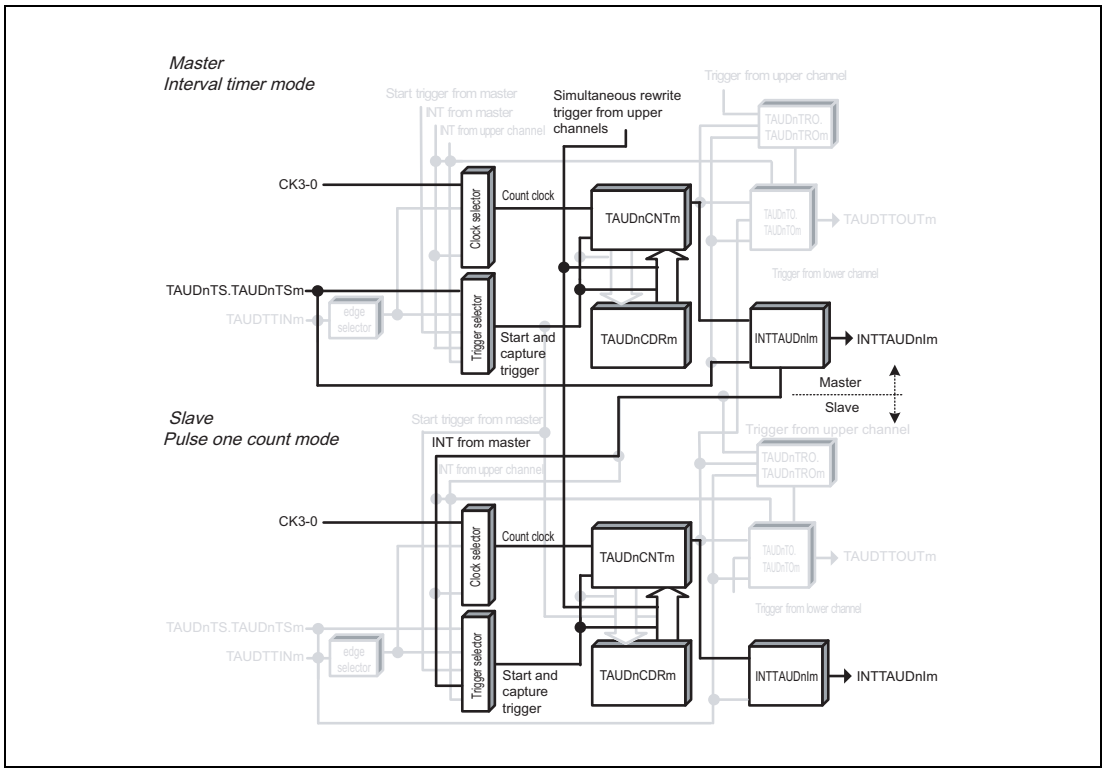


Figure 17.100 Block Diagram of A/D Conversion Trigger Output Function Type 1

The following settings apply to the general timing diagram.

- Slave channels: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

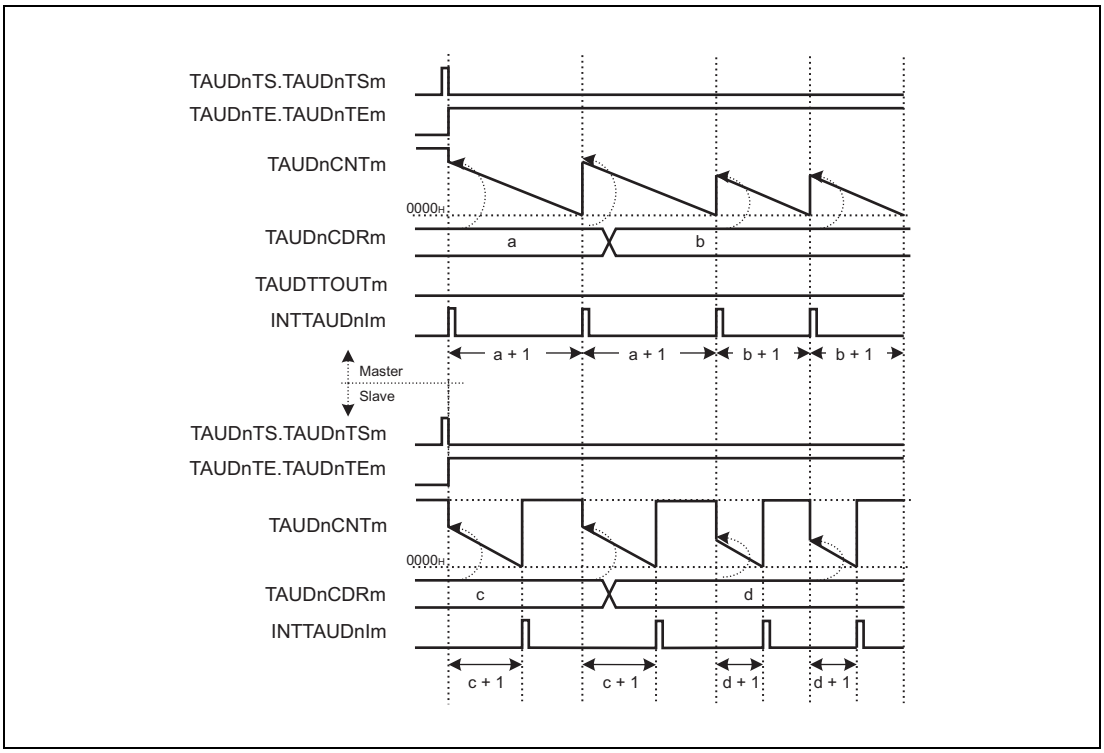


Figure 17.101 General Timing Diagram of A/D Conversion Trigger Output Function Type 1

17.4.12.7 Triangle PWM Output Function

(1) Overview

Summary

This function generates multiple triangle PWM outputs by using a master and one or more slave channels. It enables the pulse cycle (frequency) and the duty cycle of TAUDTTOUTm to be set using the master and slave channels respectively.

The master channel generates a carrier cycle. The first cycle of master channel controls the down status and the second cycle controls the up status of the slave counter.

Prerequisites

- Two channels
- The operating mode for master channels should be set to interval timer mode (see **Table 17.163, Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function**).
- The operating mode for slave channels should be set up/down count mode (see **Table 17.167, Contents of TAUDnCMORm Register for Slave Channels of Triangle PWM Output Function**).
- The channel output mode for master channels should be set to independent channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- The channel output mode for slave channels should be set to synchronous channel output mode 2 (see **Section 17.4.4, Channel Output Modes**).
- The following settings allows TAUDTTOUTm signal to be at high level during the down status of a carrier cycle.
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 0, TAUDnTO.TAUDnTOm should be set to 1 while TAUDnTOE.TAUDnTOEm is set to 0 (recommended setting).
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 1, TAUDnTO.TAUDnTOm should be set to 0 while TAUDnTOE.TAUDnTOEm is set to 0.

Functional description

The counters are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1 for every channel. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation.

The current values of TAUDnCDRm (master and slave) are loaded into TAUDnCNTm (master and slave) and the counters start counting down from these values. When the TAUDnCMORm.TAUDnMD0 bit of master channel is set to 1, an interrupt is generated and TAUDTTOUTm signal of master toggles.

- Master channel:
  When the counter of master channel reaches 0000_H (pulse cycle time has elapsed), INTTAUDnIm is generated and the TAUDTTOUTm signal toggles. TAUDnCNTm then reloads the TAUDnCDRm value and counts down.

- Slave channels:
  The INTTAUDnIm of master channel triggers the counter of the slave channel:
  - If the slave counter is counting down, the count direction changes.
  - If the slave counter is counting up, TAUDnCDRm value is reloaded and the counter starts to count down.

  When the counter of the slave channel reaches 0001_H while counting up or down, INTTAUDnIm is generated and the TAUDTTOUTm (slave) signal is set/reset.
  The counter continues count-up/-down and waits for the next INTTAUDnlm of master channel.
  Setting TAUDnTOL.TAUDnTOLm allows TAUDTTOUTm signal switching between normal phase and reverse phase during operation.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm = 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values.

Conditions

This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) $< FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

Duty cycle [%] =

[(TAUDnCDRm (master) + 1 - TAUDnCDRm (slave))/(TAUDnCDRm (master) + 1)] × 100

- Duty cycle = 100%
  TAUDnCDRm (slave) = 0000_H
- Duty cycle = 0%
  TAUDnCDRm (slave) ≥ TAUDnCDRm (master) + 1

(3) Block Diagram and General Timing Diagram

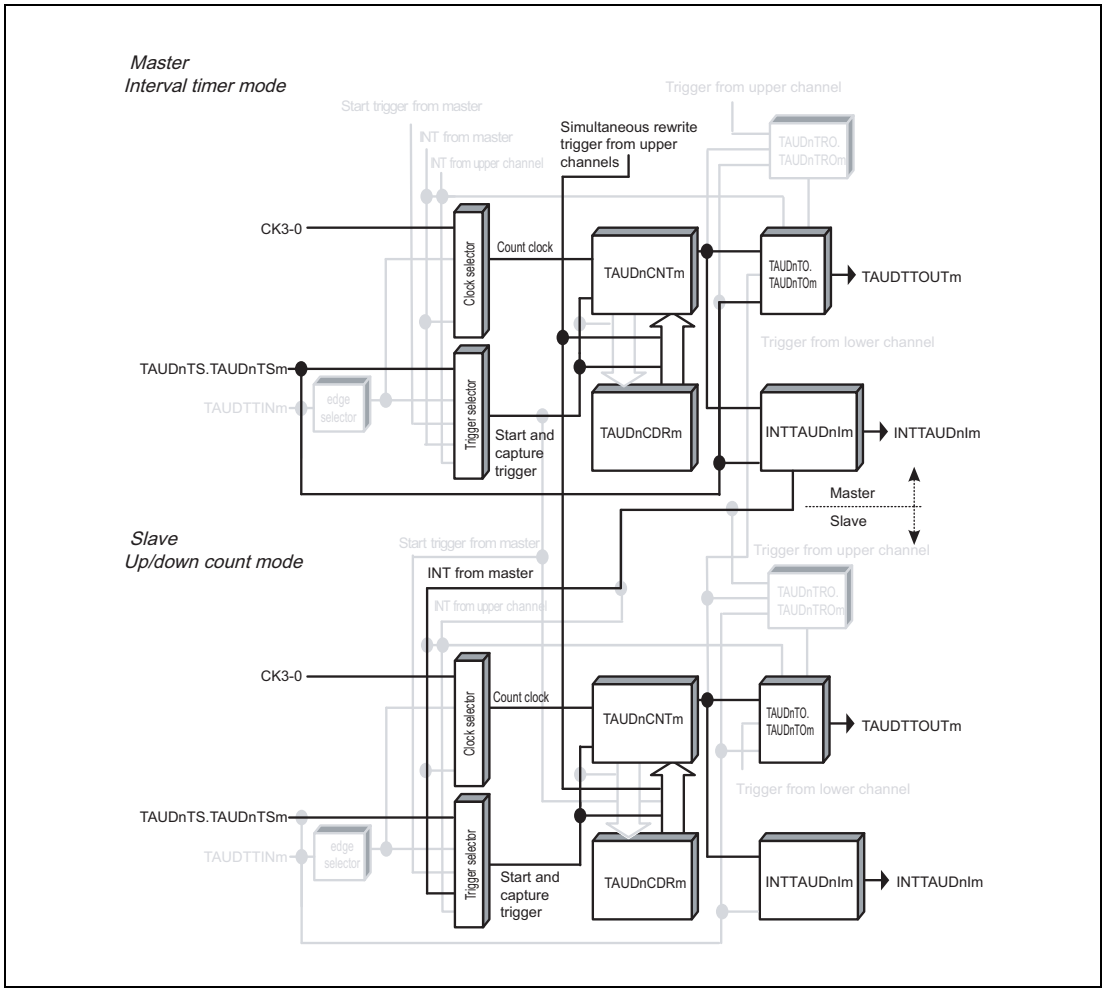


Figure 17.102 Block Diagram of Triangle PWM Output Function

The following settings apply to the general timing diagram.

- Master channel
  - INTTAUDnIm generated at the beginning of operation.
    (TAUDnCMORm.TAUDnMD0 = 1)

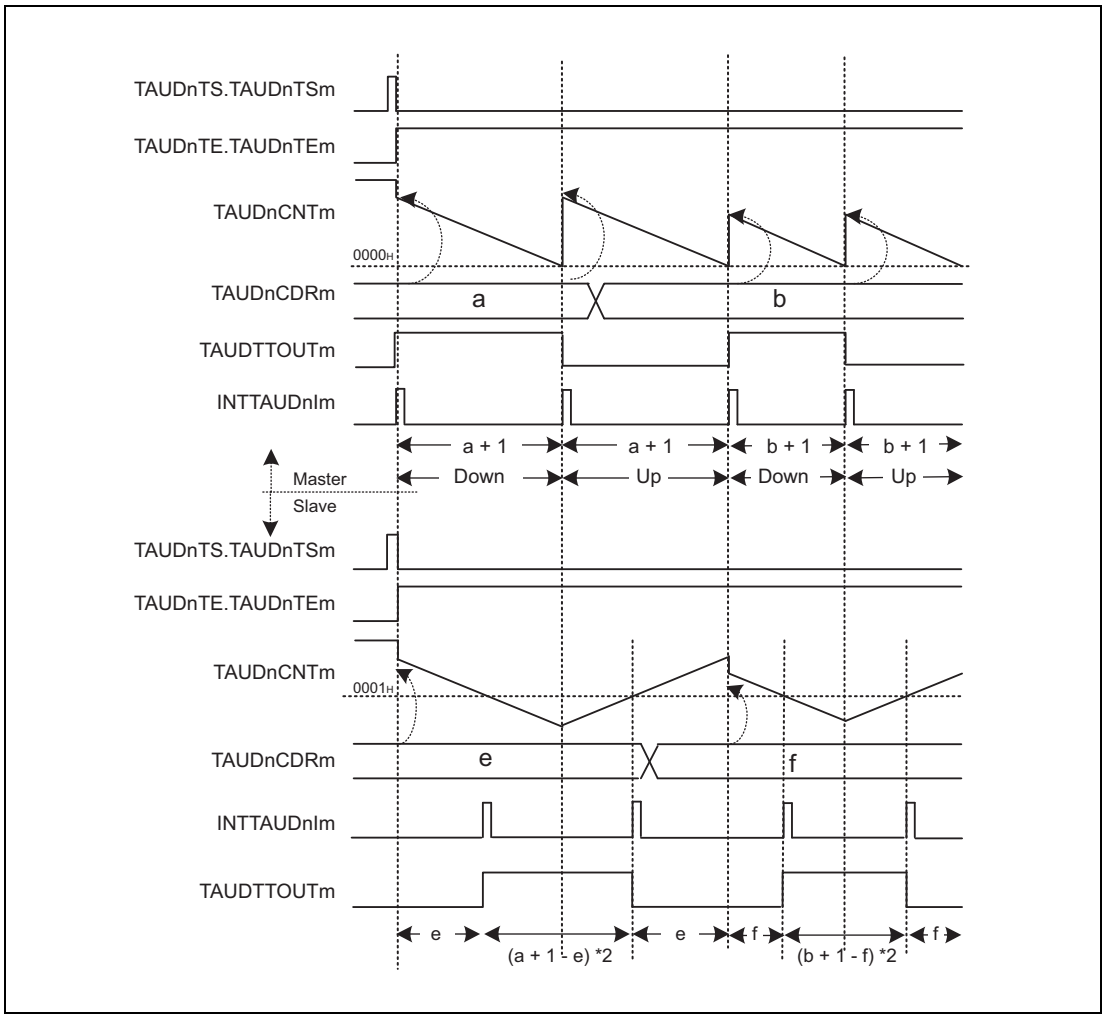


Figure 17.103 General Timing Diagram of Triangle PWM Output Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.163 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated to toggle TAUDTTOUTm at the beginning of an operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.164 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 17.165 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.166 Simultaneous Rewrite Settings for Master Channels of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.167 Contents of TAUDnCMORm Register for Slave Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.168 Contents of TAUDnCMURm Register for Slave Channels of Triangle PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels

Table 17.169 Control Bit Settings in Synchronous Channel Output Mode 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.170 Simultaneous Rewrite Settings for Slave Channels of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Triangle PWM Output Function

Table 17.171 Operating Procedure for Triangle PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm (master) is generated on the master channel if TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master and slave channels is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDTTOUTm (master) is toggled. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave) or counting is started in opposite direction. When TAUDnCNTm of slave channel reaches 0001_H: • INTTAUDnIm (slave) occurs. • TAUDTTOUTm (slave) is set in the count-down status or reset in count-up status.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnCDRm = a = 5_H
- Slave channels:
  - TAUDnCDRm = 6_H

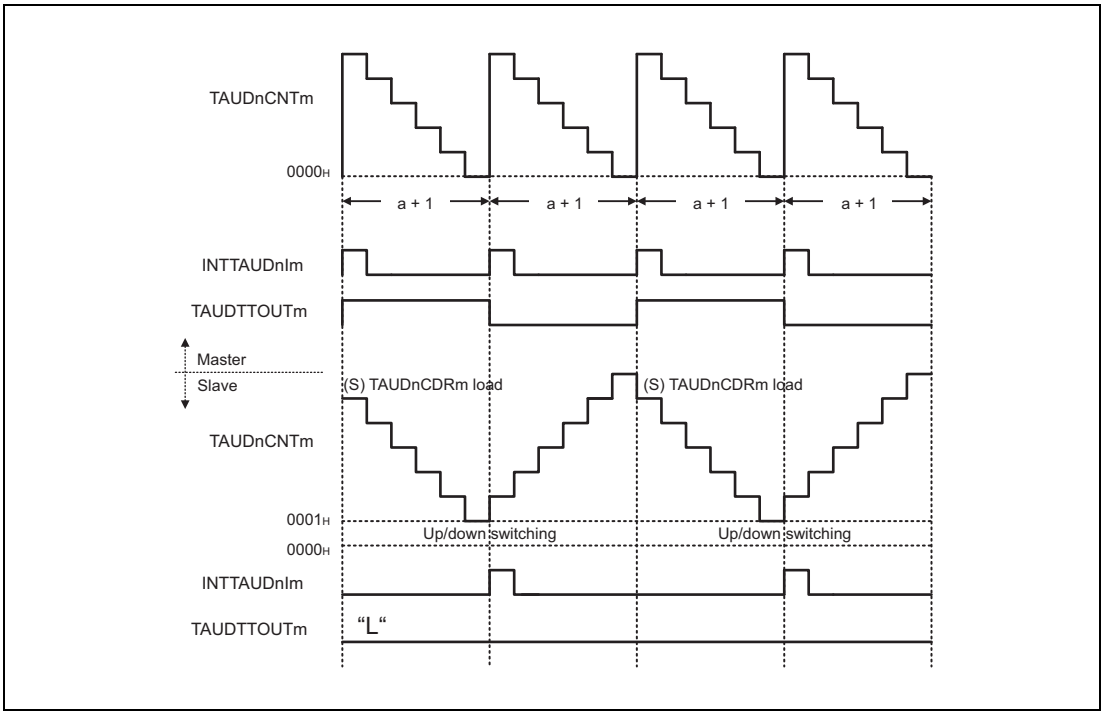


Figure 17.104 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1

- If the TAUDnCDRm (slave) value is greater than the TAUDnCDRm (master) value + 1, INTTAUDnIm of slave channel is not generated while counting down. TAUDTTOUTm remains low because there is no set signal to be detected.

(b) Duty cycle = 100%

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnCDRm = a = 5_H
- Slave channels:
  - TAUDnCDRm = 0_H

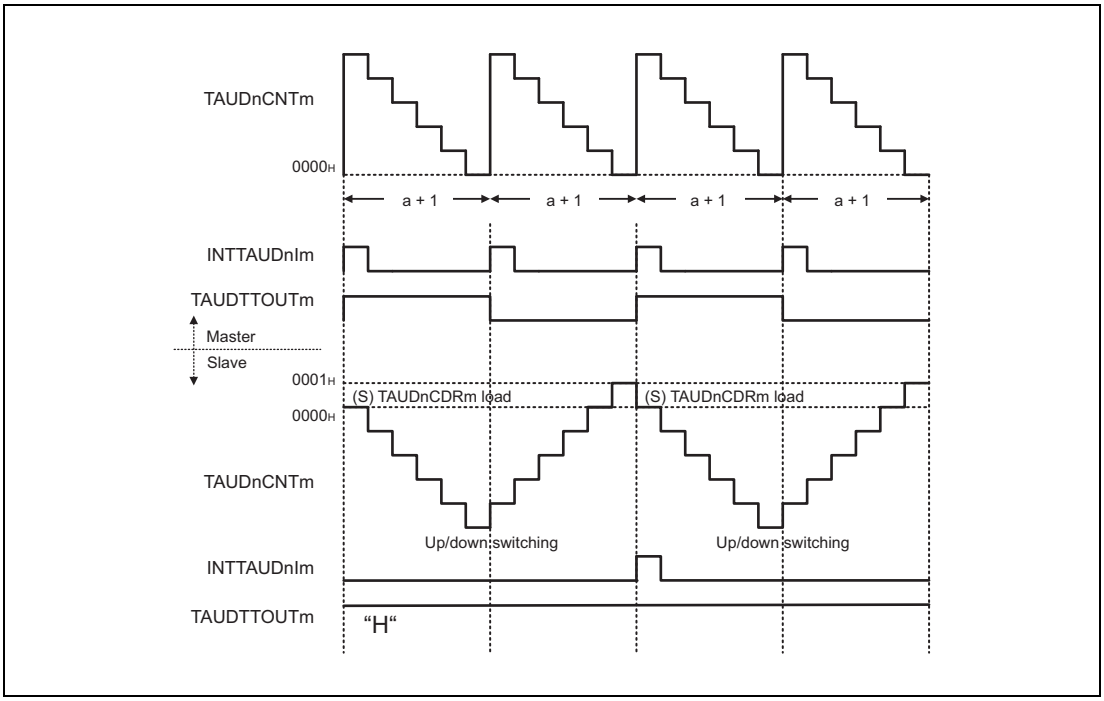


Figure 17.105 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm (slave) = 0000_H, INTTAUDnIm of slave channel is not generated while counting up. TAUDTTOUTm remains high because there is no reset signal to be detected.

17.4.12.8 Triangle PWM Output Function with Dead Time

(1) Overview

Summary

This function generates multiple triangle PWM outputs with a predefined dead time added by using a master and two or more slave channels. The resulting PWM signals are output via TAUDTTOUTm of the slave channels 2 and 3, enabling the pulse cycle (frequency) and the duty cycle of TAUDTTOUTm to be set using the master and slave channels.

Carrier cycles are generated on master channel. The first pulse controls the down status of slave counter and the second one controls the up status.

An interrupt on slave 2 causes TAUDTTOUTm of slave channels to be set/reset. Depending on the settings of TAUDnTDL.TAUDnTDLm, delay time is added to positive or negative logic side of the signal (i.e., whether TAUDTTOUTm is set/reset immediately or after dead time has elapsed). The duration of the dead time is specified by slave channel 3.

Prerequisites

- Three channels. For slave channels 2 and 3, select even-numbered channel CH (a) and odd-numbered channel CH (a + 1).
- The operating mode for master channels should be set to interval timer mode (see **Table 17.173, Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function with Dead Time**).
- Slave channel 1 is not used for this function. This ensures that slave channel 2 is an even-numbered channel, and slave channel 3 is an odd-numbered channel. Slave channel 1 can be used as a separate timer (independent function).
- The operating mode for slave channel 2 should be set to up/down count mode (see **Table 17.177, Contents of TAUDnCMORm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time**). Slave channel 2 should be an even-numbered channel.
- The operating mode for slave channel 3 should be set to one-count mode (see **Table 17.181, Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time**). Slave channel 3 should be an odd-numbered channel.
- The channel output mode for master channels should be set to independent channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- The output mode for slave channels 2 and 3 should be set to synchronous channel output mode 2 with dead time output (see **Section 17.4.4, Channel Output Modes**).
- The following settings make a TAUDTTOUTm signal at high level during the down status of the carrier cycle:
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 0, TAUDnTO.TAUDnTOm should be set to 1 while TAUDnTOE.TAUDnTOEm is set to 0. (recommended setting)
  - If TAUDnCMORm.TAUDnMD0 (master) bit is set to 1, TAUDnTO.TAUDnTOm should be set to 0 while TAUDnTOE.TAUDnTOEm is set to 0.

NOTE

The triangle PWM output function with dead time does not use slave channel 1.
Slave channel 1 can be used as a separate timer (independent function).

Functional description

The counter starts by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This makes TAUDnTE.TAUDnTEm = 1, enabling count operation. The current value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value. If TAUDnCMORm.TAUDnMD0 bit of master channel is set to 1, an interrupt is generated and the master's TAUDTTOUTm signal is toggled.

- Master channel:
  When the counter of master channel reaches 0000_H, an INTTAUDnIm is generated and the TAUDTTOUTm signal is toggled. The TAUDnCDRm value is reloaded to continue countdown.
- Slave channel 2:
  If INTTAUDnIm is generated on the master channel, the counter of slave channel 2 is triggered.
  - If the slave counter is counting down, the counting direction changes.
  - If the slave counter is counting up, the TAUDnCDRm value is reloaded and the counter starts to count down.

  The counter continues to count down/up and waits for the next INTTAUDnlm of master channel. When the counter value of slave channel 2 reaches 0001_H, INTTAUDnIm is generated.
- Slave channel 3:
  If INTTAUDnIm is generated on slave channel 2, the counter of slave channel 3 is triggered. The current value of TAUDnCDRm (slave 3) is loaded into TAUDnCNTm (slave 3) and the counter starts to count down from the TAUDnCDRm value.
  When the counter reaches 0000_H, INTTAUDnIm occurs. The counter returns to $FFFF_H$ and waits for the next INTTAUDnlm of slave channel 2.

As described in **Table 17.172, Operation of TAUDTTOUTm upon Occurrence of an Interrupt on Slave Channel 2**, the set/reset timing (right after occurrence of an interrupt or after dead time has elapsed) depends on the TAUDnTDL.TAUDnTDLm setting of the corresponding channel.

The setting of TAUDnTOL.TAUDnTOLm also determines whether a high level signal (TAUDnTOL.TAUDnTOLm = 0) or a low level signal (TAUDnTOL.TAUDnTOLm = 1) is output from the corresponding channel.

The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values.

TAUDTTOUTm can be 100% output by setting the TAUDnCDRm value of slave channel 2 to 0000_H.

Conditions

This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.

TAUDnTOL.TAUDnTOLm and TAUDnTDL.TAUDnTDLm should be set before start of count operation. Slave channels 2 and 3 should have the opposite settings of TAUDnTDL.TAUDnTDLm.

Table 17.172 Operation of TAUDTTOUTm upon Occurrence of an Interrupt on Slave Channel 2

TAUDnTDL. TAUDnTDLm	Count Direction of Slave Channel 2 upon Occurrence of Interrupt	TAUDTTOUTm Set/Reset Timing
0	Down	Set after elapse of dead time
	Up	Reset right after interrupt occurs
1	Down	Set right after interrupt occurs
	Up	Reset after elapse of dead time

(2) Equations

Pulse cycle = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) < $FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

PWM signal width (normal phase) = [(TAUDnCDRm (master) + 1 - TAUDnCDRm (slave 2)) × 2 – (TAUDnCDRm (slave 3) + 1)] × count clock cycle

PWM signal width (reverse phase) = [(TAUDnCDRm (master) + 1 -TAUDnCDRm (slave 2)) × 2 + (TAUDnCDRm (slave 3) + 1)] × count clock cycle

(3) Block Diagram and General Timing Diagram

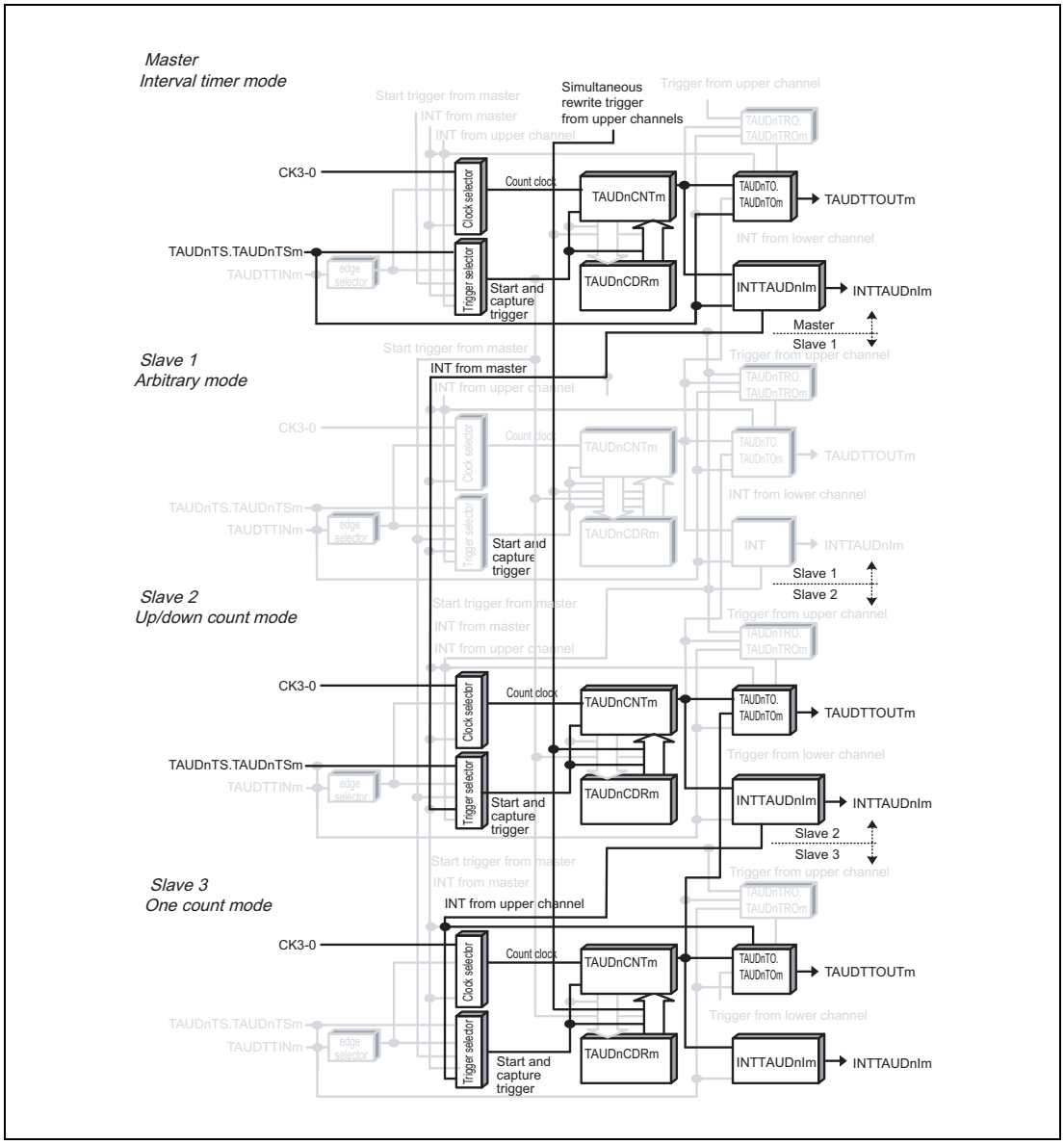


Figure 17.106 Block Diagram of Triangle PWM Output Function with Dead Time

The following settings apply to the general timing diagram.

- Master channel:
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)
- Slave channel 2:
  - INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
  - TAUDnTDL.TAUDnTDLm = 0
  - Positive logic (TAUDnTOL.TAUDnTOLm = 0)

- Slave channel 3:
  - Enables start trigger detection during counting. (TAUDnCMORm.TAUDnMD0 = 1)
  - TAUDnTDL.TAUDnTDLm = 1
  - Positive logic (TAUDnTOL.TAUDnTOLm = 0)

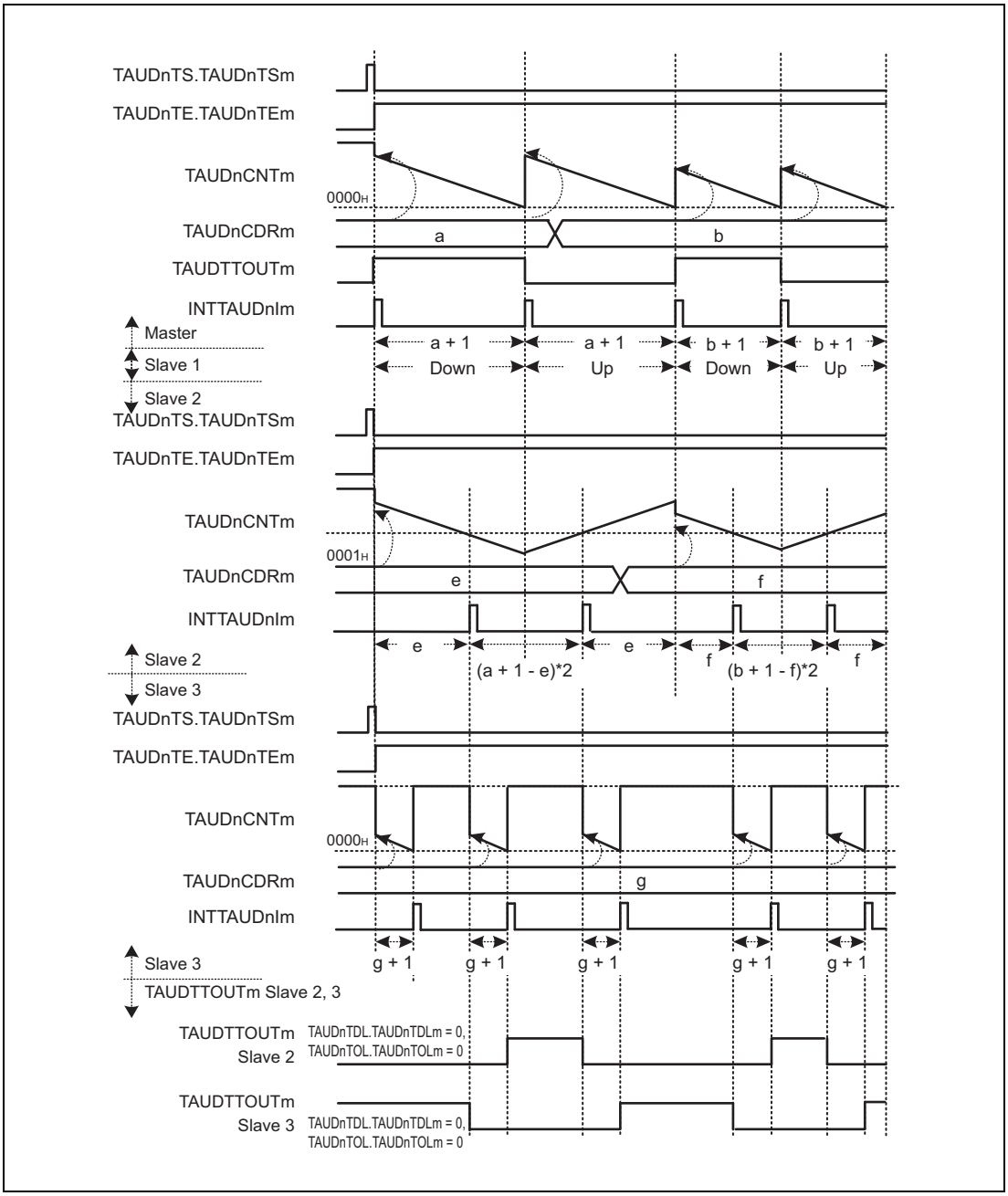


Figure 17.107 General Timing Diagram of Triangle PWM Output Function with Dead Time

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.173 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated and TAUDTTOUTm is not toggled at the beginning of operation. 1: INTTAUDnIm is generated and TAUDTTOUTm is toggled at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.174 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 17.175 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.176 Simultaneous Rewrite Setting for Master Channels of Triangle PWM Output Function with Dead Time

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 2

(a) TAUDnCMORm for slave channel 2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.177 Contents of TAUDnCMORm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channel 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.178 Contents of TAUDnCMURm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 2

Table 17.179 Control Bit Settings in Synchronous Channel Output Mode 2 with Dead Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from odd-numbered channels.

(d) Simultaneous rewrite for slave channel 2

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.180 Simultaneous Rewrite Settings for Slave Channel 2 of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Register Settings for Slave Channel 3

(a) TAUDnCMORm for slave channel 3

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.181 Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	110: Dead time output signal of the TAUDTTOUTm generation unit
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for slave channel 3

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.182 Contents of TAUDnCMURm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channel 3

Table 17.183 Control Bit Settings in Synchronous Channel Output Mode 2 with Dead Time Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel operation
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from even-numbered channels.

(d) Simultaneous rewrite for slave channel 3

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.184 Simultaneous Rewrite Settings for Slave Channel 3 of Triangle PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(7) Operating Procedure for Triangle PWM Output Function with Dead Time

Table 17.185 Operating Procedure for Triangle PWM Output Function with Dead Time

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 2: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 2**. Slave channel 3: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (6), Register Settings for Slave Channel 3**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm (master) is generated on the master channel if TAUDnCMORm.TAUDnMD0 is set to 1.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channel 2 is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCDRm value is reloaded into TAUDnCNTm (master) to continue count operation. • TAUDnCDRm value is reloaded into TAUDnCNTm (slave 2) or counting is started in opposite direction. When TAUDnCNTm of slave channel 2 reaches 0001_H: • IINTTAUDnIm (slave 2) is generated. • TAUDnCDRm value of slave channel 3 is loaded into TAUDnCNTm perform counting down. When TAUDnCNTm of slave channel 3 reaches 0000_H: • INTTAUDnIm is generated.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

(a) Duty cycle = 0%

The following settings apply to the general timing diagram.

- Slave channel 2:
  - Positive logic (TAUDnTDL.TAUDnTDLm = 0)
- Slave channel 3:
  - Negative logic (TAUDnTDL.TAUDnTDLm = 1)

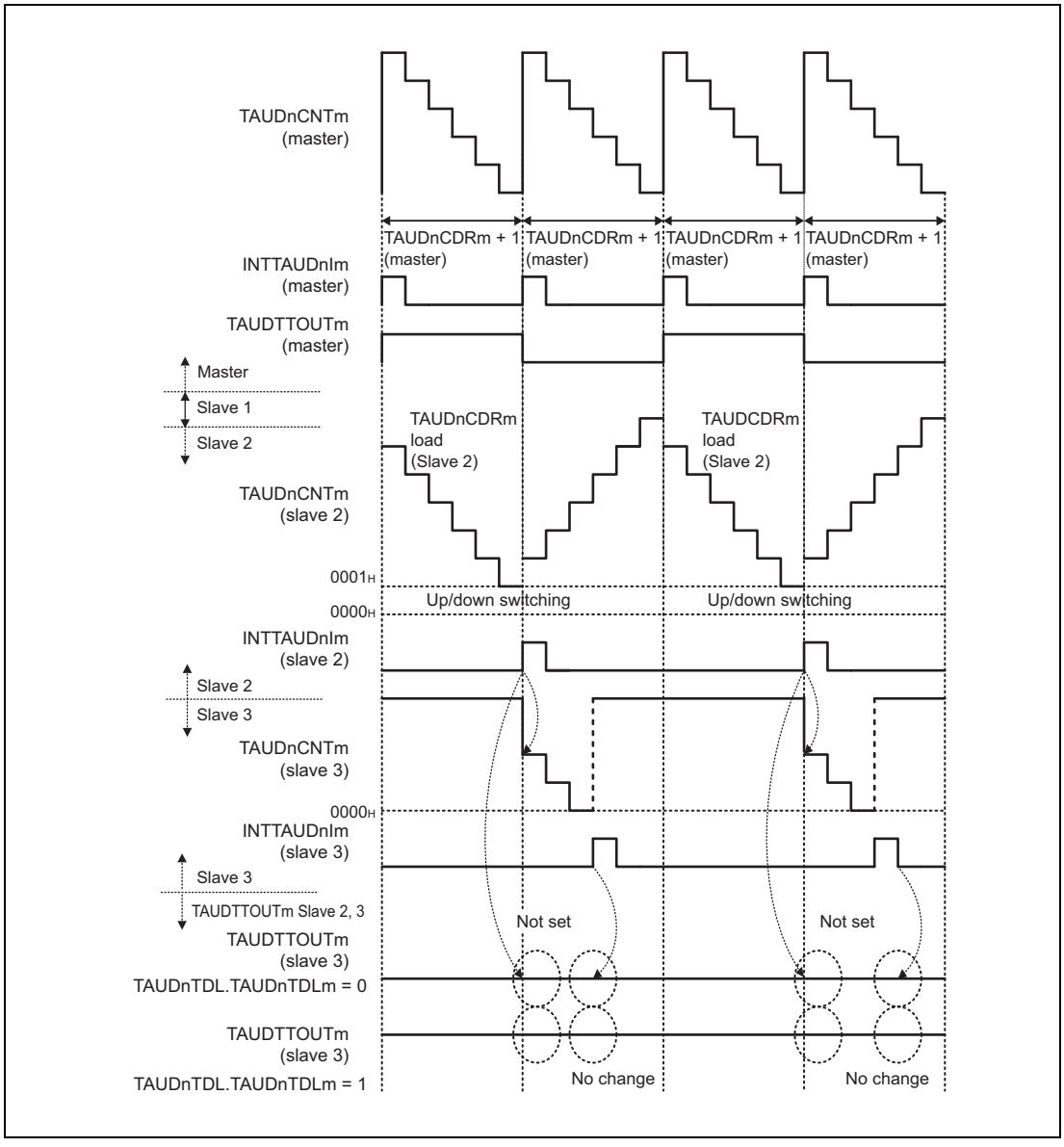


Figure 17.108 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1

- If TAUDnCDRm (slave 2) is greater than TAUDnCDRm (master), the counter of slave channel does not reach 0000_H while counting down. Therefore, TAUDTTOUTm signal is not set/reset and remains initial. This signal becomes a reset signal because an interrupt occurs on slave channel 2 during count-up operation.

(b) Duty cycle = 100%

The following settings apply to the general timing diagram.

- Slave channel 2:
  - Positive logic (TAUDnTDL.TAUDnTDLm = 0)
- Slave channel 3:
  - Negative logic (TAUDnTDL.TAUDnTDLm = 1)

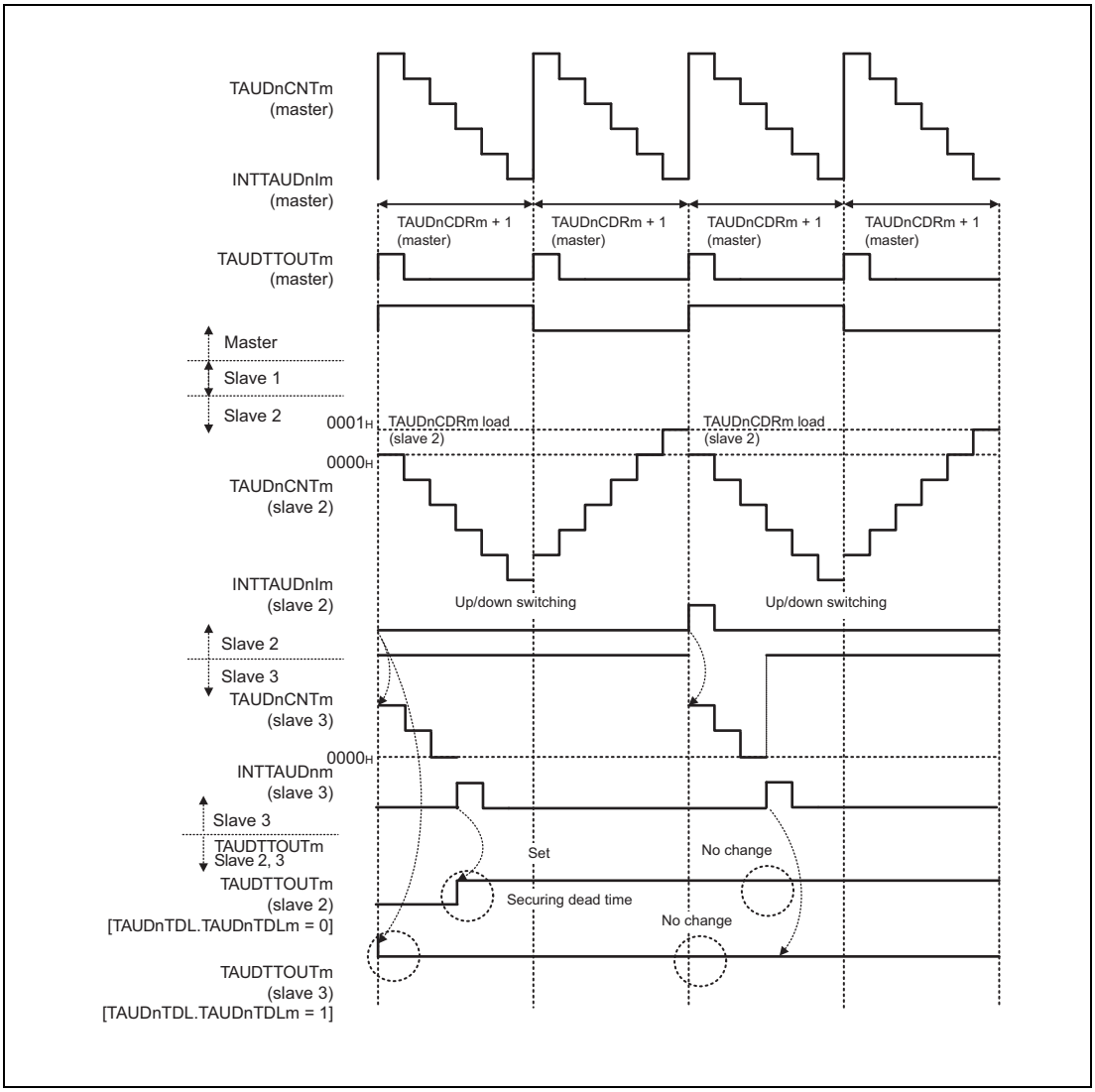


Figure 17.109 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm (slave 2) = 0000_H, the slave channel counter does not reach 0001_H while counting up. Therefore, no INTTAUDnlm occurs during count-up operation.
  - The set conditions for a channel with TAUDnTDL.TAUDnTDLm = 0 are met after elapse of dead time. TAUDTTOUTm is left in a newly set state even if a set/reset is made because no reset conditions are satisfied on such a channel.
  - Slave channel 3 in the above diagram is set when the counter starts. However, TAUDTTOUTm is left in an initial state on the slave channel with TAUDnTDL.TDLm = 1 because no reset conditions are satisfied on that channel.

17.4.12.9 A/D Conversion Trigger Output Function Type 2

(1) Overview

Summary

This function is identical to **Section 17.4.12.7, Triangle PWM Output Function**, except that TAUDTTOUTm is not output.

This function is enabled by setting channel output mode for the slave to independent channel output mode controlled by software.

(2) Block Diagram and General Timing Diagram

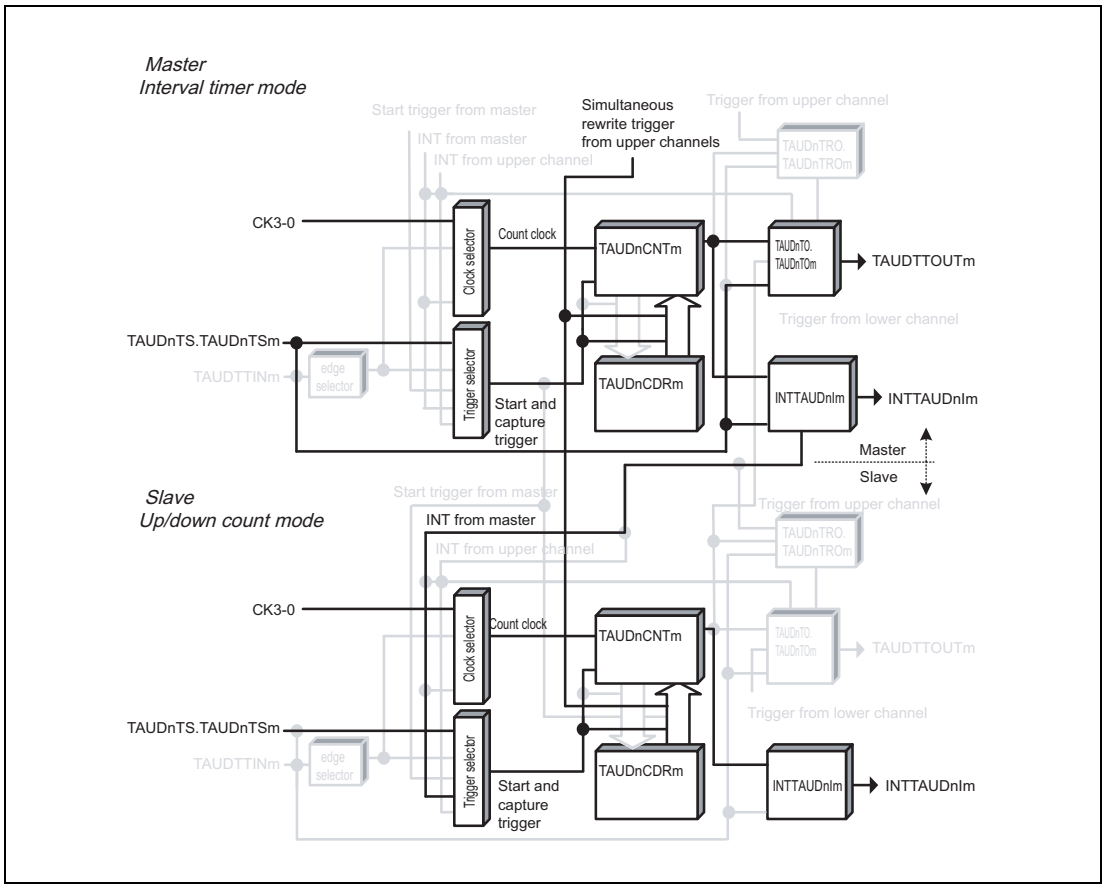


Figure 17.110 Block Diagram of A/D Conversion Trigger Output Function Type 2

The following settings apply to the general timing diagram.

- Master channel
  - INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

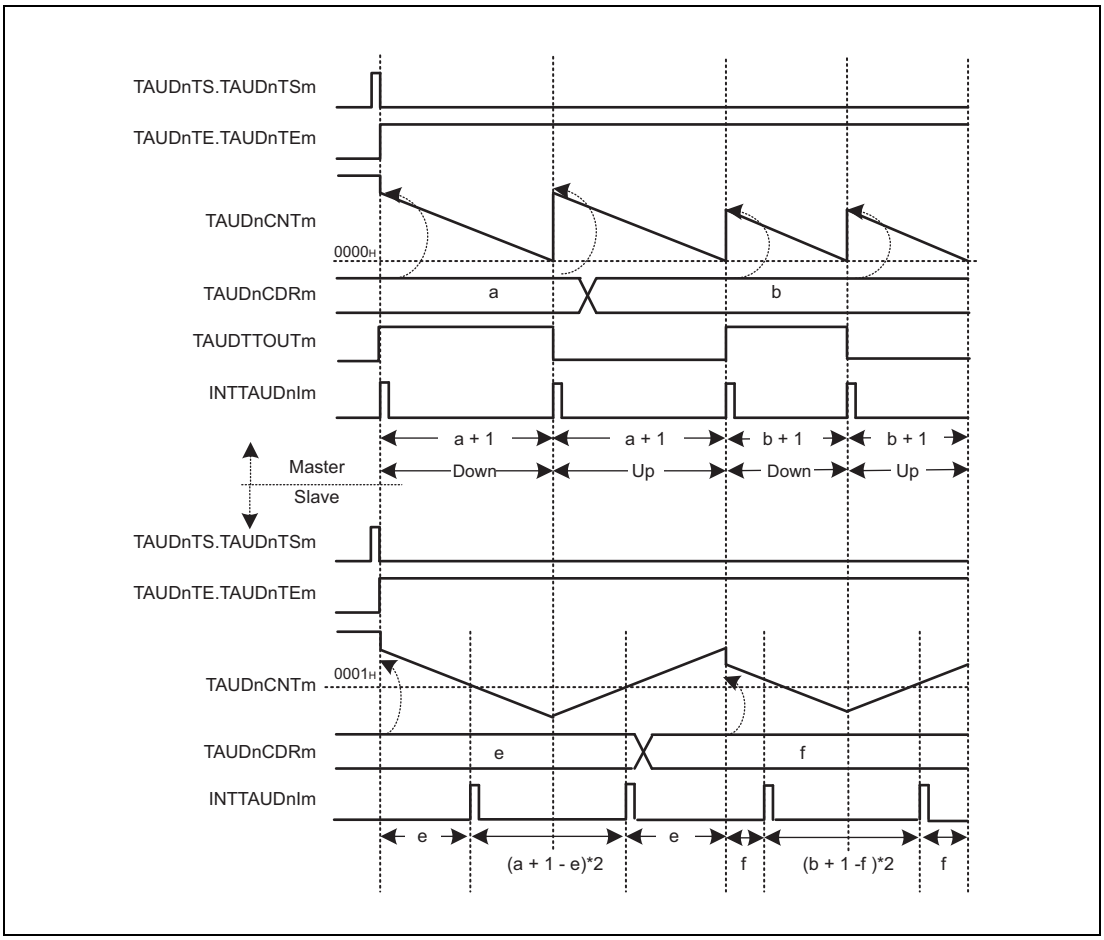


Figure 17.111 General Timing Diagram of A/D Conversion Trigger Output Function Type 2

17.4.12.10 Interrupt Request Signals Culling Function

(1) Overview

Summary

This function divides the number of interrupts of the master channel by a specified value using a slave channel.

The interrupt request signals Culling function is a sub function of the following functions:

- PWM Output Function (see **Section 17.4.12.1, PWM Output Function**)
- Triangle PWM Output Function (see **Section 17.4.12.7, Triangle PWM Output Function**)
- Triangle PWM Output Function with Dead Time
  (see **Section 17.4.12.8, Triangle PWM Output Function with Dead Time**)

Prerequisites

- Two channels
- The operation mode of the master channel must be set to interval timer mode (see **Table 17.186, Contents of TAUDnCMORm Register for Master Channels of Interrupt Request Signals Culling Function**).
- The operation mode of the slave channel must be set to Event Count Mode (see **Table 17.189, Contents of TAUDnCMORm Register for Slave Channels of Interrupt Request Signals Culling Function**).
- TAUDTTOUTm is not used for the master or slave channel of this function

Functional description

The counters (master and slave) are started by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1 for both channels. This in turn sets TAUDnTE.TAUDnTEm, enabling count operation. The current value of the data register of the master channel and slave channel (TAUDnCDRm) are written to the counter (TAUDnCNTm).

- Master channel:
  When the counter of the master channel reaches 0000_H, INTTAUDnIm is generated and TAUDnCDRm value is reloaded to TAUDnCNTm.
- Slave channel:
  Every time the master channel generates an INTTAUDnIm, the counter of the slave channel reduces by one. When the counter reaches 0000_H, it awaits the next interrupt from the master channel. This causes TAUDnCNTm (slave) to reload the value of TAUDnCDRm, and an INTTAUDnIm is generated.

Forced restart is not possible for this function. The counter can be stopped by setting TAUDnTT.TAUDnTTm to 1 for the master and slave channel(s), which in turn sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm of master and slave channel(s) stop but retain their values.

Conditions

This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Interrupt division operator = TAUDnCDRm (slave channel)

– One INTTAUDnlm is generated for the INTTAUDnlm count of the master channel defined by TAUDnCDRm (slave channel) + 1.

(3) Block Diagram and General Timing Diagram

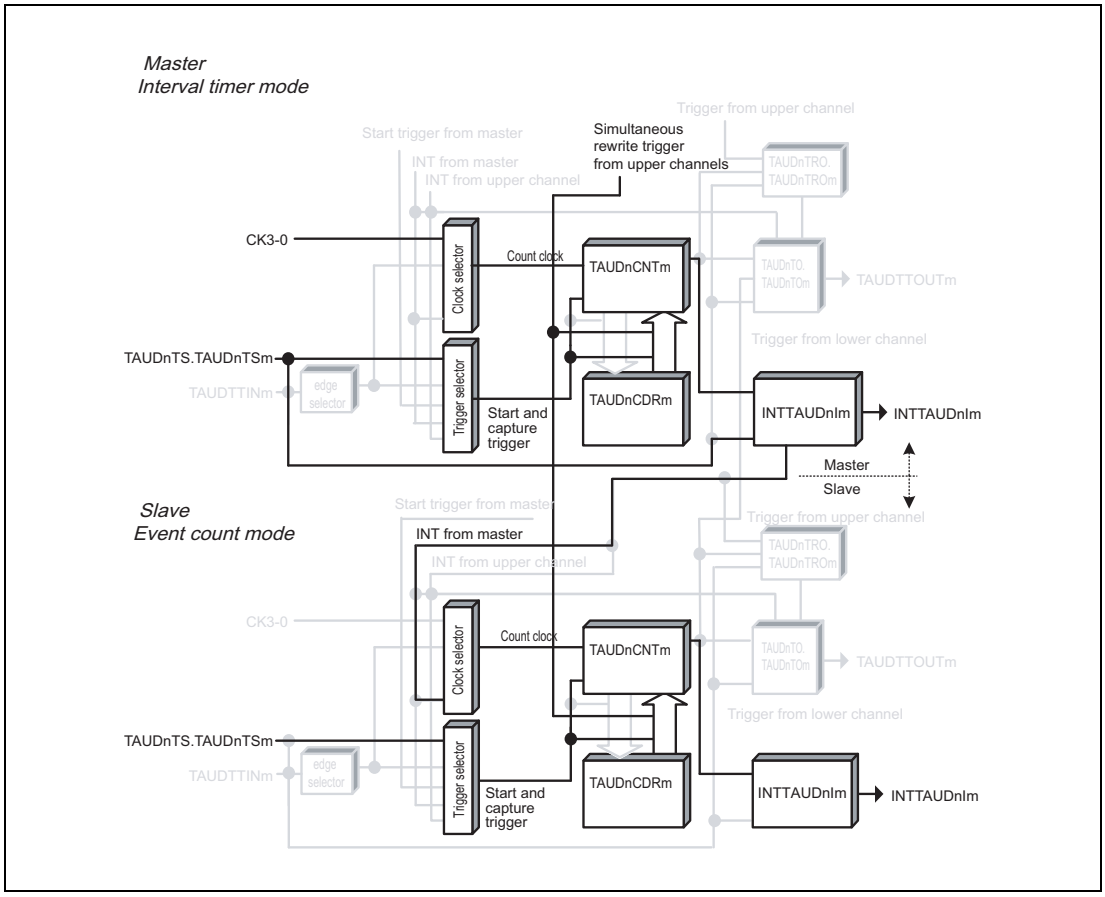


Figure 17.112 Block Diagram of Interrupt Request Signals Culling Function

The following settings apply to the general timing diagram.

Master channel:

- INTTAUDnIm generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 1)

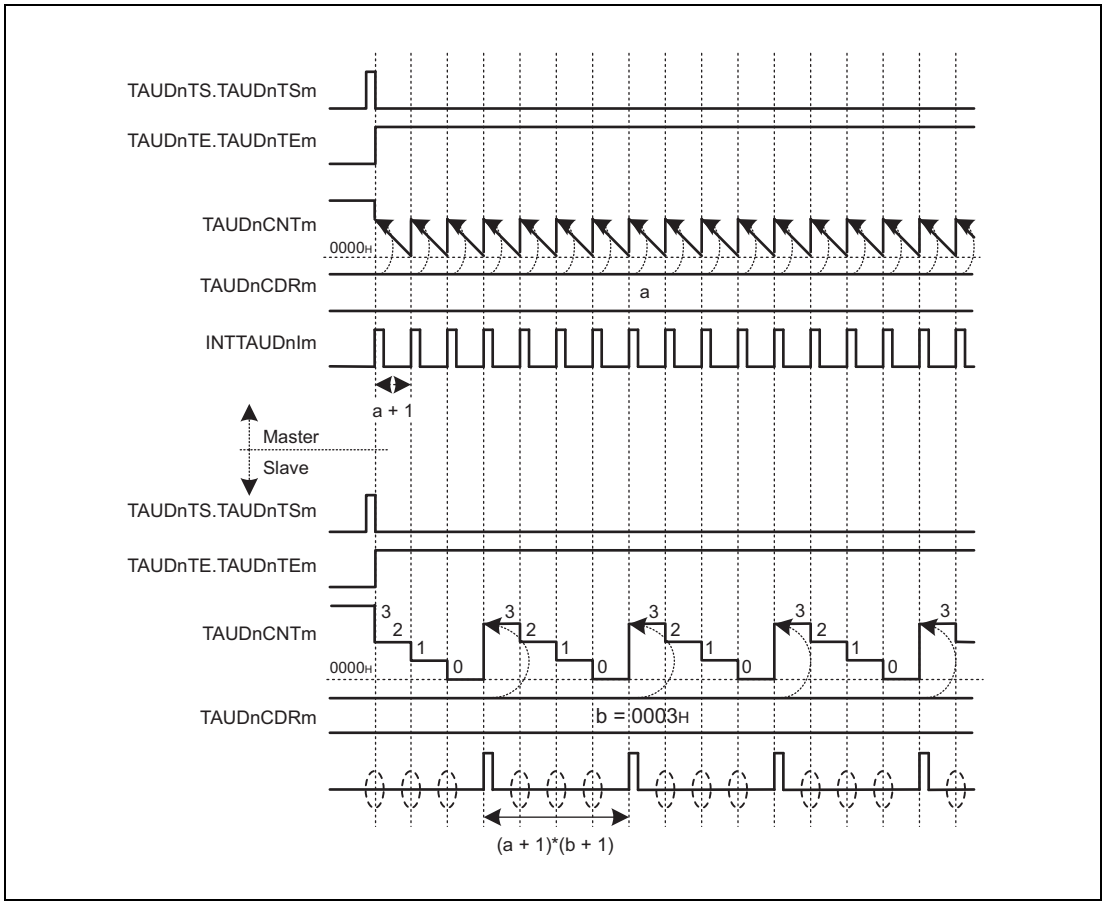


Figure 17.113 General Timing Diagram of Interrupt Request Signals Culling Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.186 Contents of TAUDnCMORm Register for Master Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation. 1: INTTAUDnIm generated at the beginning of operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.187 Contents of TAUDnCMURm Register for Master Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.188 Simultaneous Rewrite Settings for Master channels of Interrupt Request Signals Culling Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count. 1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(5) Register Settings for Slave Channels

(a) TAUDnCMORm for slave channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.189 Contents of TAUDnCMORm Register for Slave Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm not generated at the beginning of operation.

(b) TAUDnCMURm for slave channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.190 Contents of TAUDnCMURm Register for Slave Channels of Interrupt Request Signals Culling Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for the slave channel

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used with this function.

(d) Simultaneous rewrite for slave channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.191 Simultaneous Rewrite Settings for Slave Channels of Interrupt Request Signals Culling Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Selects master channel for simultaneous rewrite triggers. 1: Selects upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count. 1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

(6) Operating Procedure for Interrupt Request Signals Culling Function

Table 17.192 Operating Procedure for Interrupt Request Signals Culling Function

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channels**. Set the value of TAUDnCDRm register of every channel.	Channel operation is stopped.
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm (master and slave channels) is set to 1 and the counters of master and slave channels start. INTTAUDnIm is generated on the master channel.
During Operation	TAUDnCDRm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCNTm of master channel loads TAUDnCDRm value and counts down. When the counter reaches 0000_H: • INTTAUDnIm (master) occurs. • TAUDnCNTm (master) loads TAUDnCDRm value and continues count operation. • TAUDnCNTm of slave channels counts down each time INTTAUDnIm of master channel is detected. When TAUDnCNTm of the slave = 0000_H: • INTTAUDnIm (slave) occurs.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(7) Specific Timing Diagrams

(a) Interrupt count (master) = interrupt count (slave)

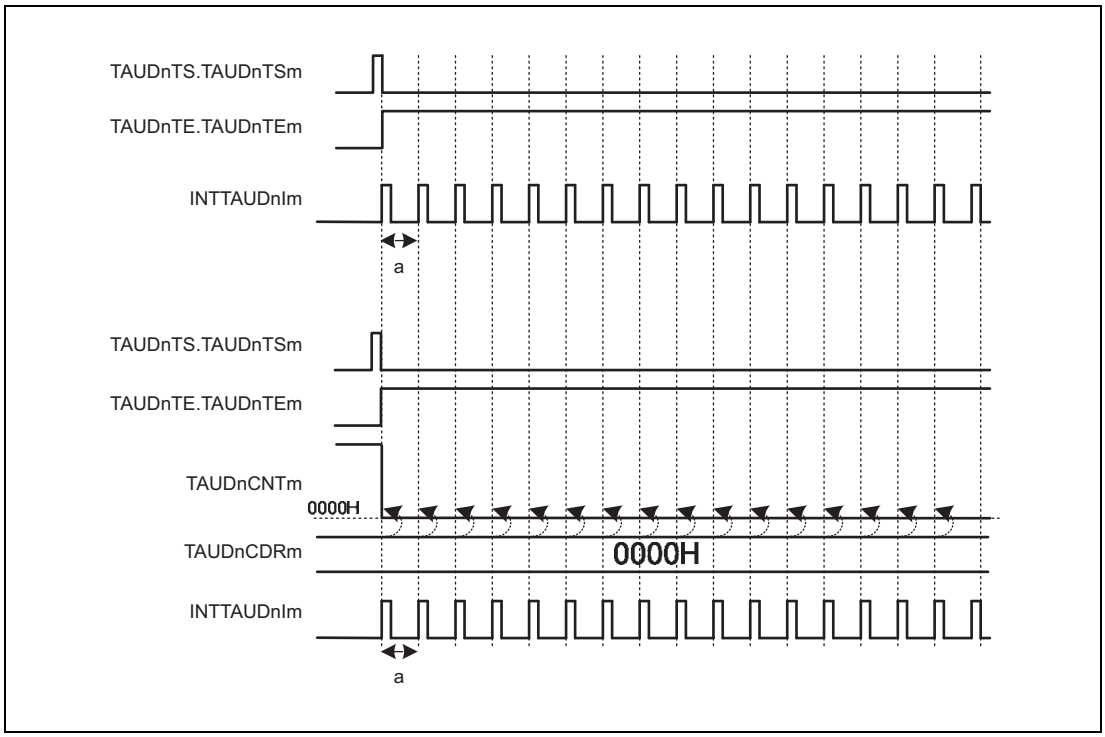


Figure 17.114 TAUDnCDRm (Slave) = 0000_H

- If TAUDnCDRm = 0000_H, TAUDnCDRm value of slave channels is loaded into TAUDnCNTm each time INTTAUDnIm of master channel is detected. In other words, TAUDnCNTm is always 0000_H.
- Therefore, an interrupt occurs on the master channel and simultaneously an interrupt occurs on slave channels.

17.4.12.11 One-Phase PWM Output Function

(1) Overview

Summary

This function adds dead time to a TAUDTTINm input signal. The resulting PWM signal is output via TAUDTTOUTm of the channel and TAUDTTOUTm of upper channels.

Prerequisites

- Each of two (or more) channels is enabled for dead time control (TAUDnTDE.TAUDnTDEm = 1).
- The operating mode for the lower channel should be set to one-count mode (see **Table 17.194, Contents of TAUDnCMORm Register for One-Phase PWM Output Function**).
- Any operating mode can be set to upper channels.
- Channel output mode for upper and lower channels should be set to synchronous channel output mode 2 with one-phase PWN output (see **Section 17.4.4, Channel Output Modes**).

Functional description

The counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation.

The counter starts when an effective TAUDTTINm input start edge is detected. The value of TAUDnCDRm is loaded into TAUDnCNTm and the counter starts to count down from the TAUDnCDRm value.

When the counter reaches 0000_H, an interrupt occurs. The counter is reset to $FFFF_H$ and waits for the next effective TAUDTTINm input start edge.

Table 17.193 TAUDTTOUTm to which Dead Time is Added and State of TAUDTTINm

TAUDnCMURm. TAUDnTIS[1:0]	TAUDnTOL. TAUDnTOLm	TAUDTTOUTm to which Dead Time is Added	TAUDnTDL. TAUDnTDLm	TAUDTTINm_lower State when Added
10	0	TAUDTTOUTm low	0	High
			1	Low
	1	TAUDTTOUTm high	0	High
			1	Low
11	0	TAUDTTOUTm low	0	Low
			1	High
	1	TAUDTTOUTm high	0	Low
			1	High

Conditions

- TAUDnCMURm.TAUDnTIS[1:0] bits specify the type of width measurement:
  - TAUDnCMURm.TAUDnTIS[1:0] = 10_B: Uses both edges as effective edges for detection (Low width measurement).
  - TAUDnCMURm.TAUDnTIS[1:0] = 11_B: Uses both edges as effective edges for detection (High width measurement).
- The TAUDnTDL.TAUDnTDLm bit specifies the operation of TAUDTTOUTm for each channel when an interrupt or effective TAUDTTINm edge is detected on the lower channel:
  - If TAUDnTDL.TAUDnTDLm = 0, an interrupt is used as a TAUDTTOUTm set trigger and an effective TAUDTTINm edge as a TAUDTTOUTm reset trigger.
  - If TAUDnTDL.TAUDnTDLm = 1, an effective TAUDTTINm edge is used as a TAUDTTOUTm set trigger and an interrupt as a TAUDTTOUTm reset trigger.
- This function cannot make a forced restart.

(2) Block Diagram and General Timing Diagram

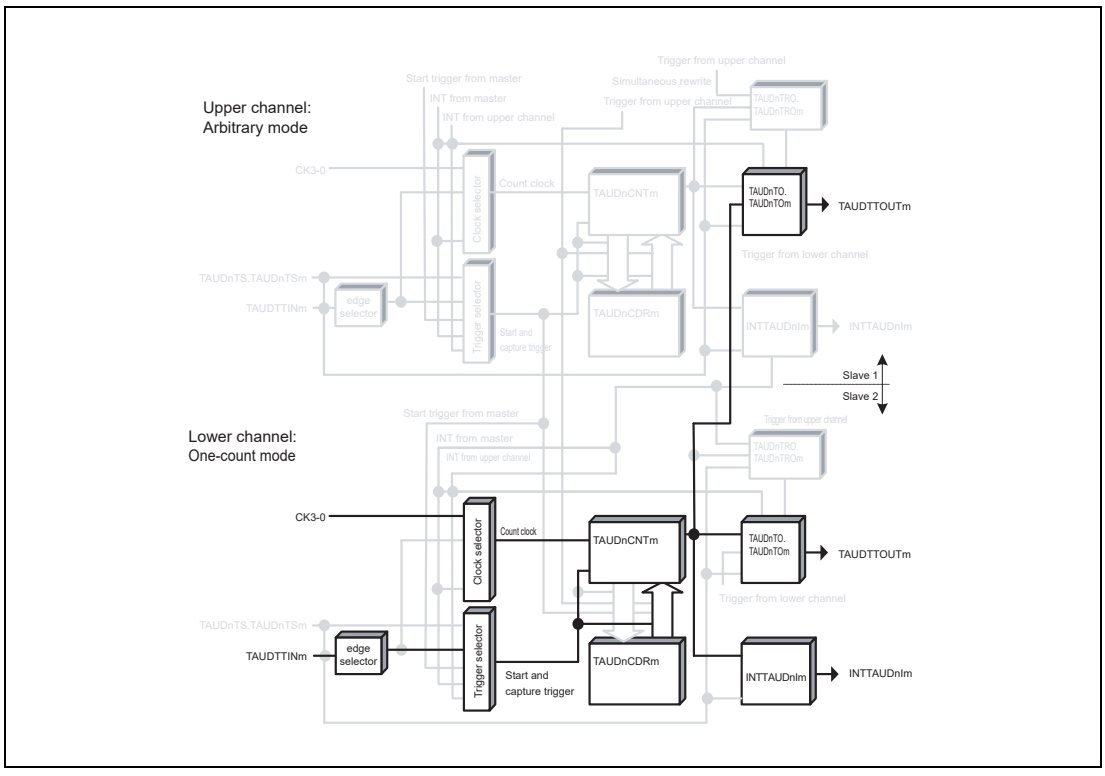


Figure 17.115 Block Diagram of One-Phase PWM Output Function

The following settings apply to the general timing diagram.

- Detection of rising and falling edges = high width measurement (TAUDnCMURm.TAUDnTIS[1:0] = 11_B)

This setting considers a duty as a high active.

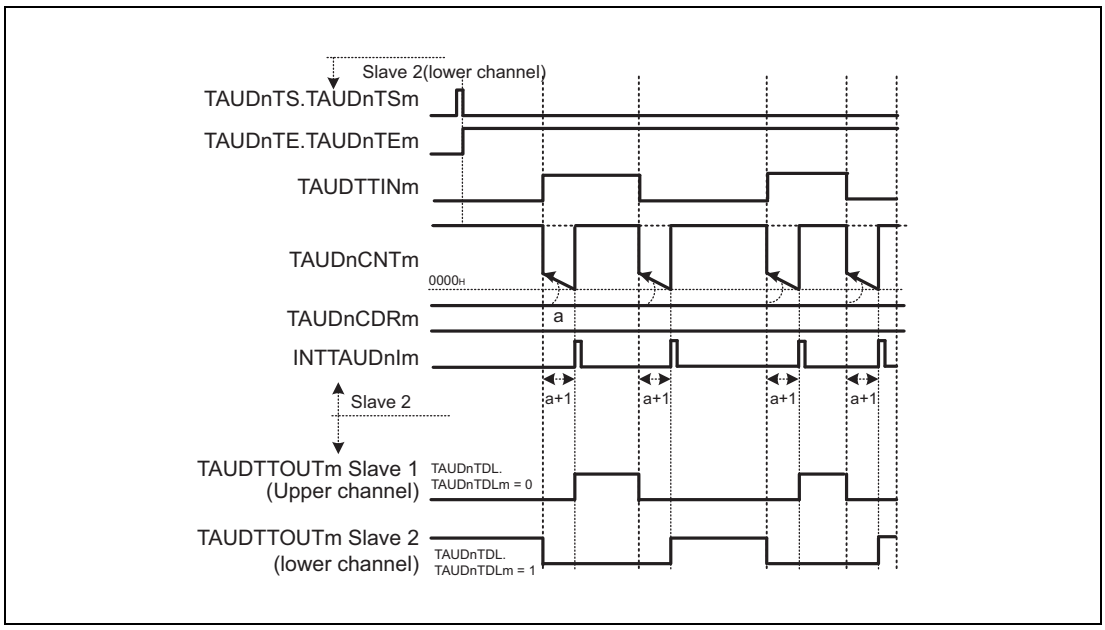


Figure 17.116 General Timing Diagram of One-Phase PWM Output Function

(3) Register Settings for Lower Channels

(a) TAUDnCMORm for lower channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.194 Contents of TAUDnCMORm Register for One-Phase PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Independent channel operation. Set to 0.
10 to 8	TAUDnSTS[2:0]	001: Effective edge of the TAUDTTINm input signal is used as an external start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for lower channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.195 Contents of TAUDnCMURm Register for One-Phase PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	10: Detection of rising and falling edges (low width measurement) 11: Detection of rising and falling edges (high width measurement)

(c) Channel output mode for lower channels

Table 17.196 Control Bit Settings in Synchronous Channel Output Mode 2 with One-Phase PWM Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode/
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	1: Adds dead time upon detection of a TAUDTTINm input edge on a lower odd-numbered channel.
TAUDnTDL.TAUDnTDLm	0: Adds dead time of the positive-phase width. 1: Adds dead time of the negative-phase width.
TAUDnTRE.TAUDnTREm	0: Disables real-time output.
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0.
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation/

CAUTION

Set TAUDnTDL.TAUDnTDLm to upper channels exclusively.

(d) Simultaneous rewrite for lower channels

Simultaneous rewrite registers (TAUDnRDE, TAUDnRDS, TAUDnRDM, and TAUDnRDC) cannot be used with this function. Therefore, these registers should be set to 0.

Table 17.197 Simultaneous Rewrite Settings for One-Phase PWM Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	0: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: When simultaneous rewrite is disabled (TAUDnRDE.TAUDnRDEm = 0), set these bits to 0
TAUDnRDM.TAUDnRDMm	
TAUDnRDC.TAUDnRDCm	

(4) Register Settings for Upper Channels

(a) TAUDnCMORm for upper channels

TAUDnCMORm register for upper channels can be set arbitrarily.

(b) TAUDnCMURm for upper channels

TAUDnCMURm register for upper channels can be set arbitrarily.

(c) Channel output mode for upper channels

Table 17.198 1Control Bit Settings for Upper Channels in Synchronous Channel Output Mode 2 with One-Phase PWM Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	1: Adds dead time upon detection of a TAUDTTINm input edge on a lower odd-numbered channel.
TAUDnTDL.TAUDnTDLm	0: Uses a lower channel interrupt as a TAUDTTOUTm set trigger and an effective lower channel TAUDTTINm edge as a TAUDTTOUTm reset trigger. 1: Uses an effective lower channel TAUDTTINm edge as a TAUDTTOUTm set trigger and a lower channel interrupt as a TAUDTTOUTm reset trigger.
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm to lower channels exclusively.

(d) Simultaneous rewrite for upper channels

Simultaneous rewrite register for upper channels can be set arbitrarily.

(5) Operating Procedure for One-phase PWM Output Function

Table 17.199 Operating Procedure for One-phase PWM Output Function

	Operation	TAUDn Status
Initial Channel Setting	Set TAUDnCMORm and TAUDnCMURm registers for the lower channel as described in **Table 17.194, Contents of TAUDnCMORm Register for One-Phase PWM Output Function**, and **Table 17.195, Contents of TAUDnCMURm Register for One-Phase PWM Output Function**. Set TAUDnCMORm and TAUDnCMURm registers for the upper channel as described in **Section (4), Register Settings for Upper Channels**. Set the value of TAUDnCDRm register. Set channel output mode by setting the control bits as described in **Table 17.196, Control Bit Settings in Synchronous Channel Output Mode 2 with One-Phase PWM Output**.	Channel operation is stopped.
Start Operation	Set TAUDnTOE.TAUDnTOEm (slave channels 1 and 2) to 1 (at restart time only). Set TAUDnTS.TAUDnTSm = 1 for slave channel 2. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0. Detection of TAUDTTINm start edge	TAUDnTE.TAUDnTEm is set to 1 (slave channel 2) and TAUDnCNTm waits for detection of TAUDTTINm start edge. TAUDnCNTm loads TAUDnCDRm value.
During Operation	The TAUDnCDRm register value can be changed at any time. The TAUDnCNTm register can be read at any time.	TAUDnCNTm of slave channel 2 counts down. When the counter reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCNTm stops counting. TAUDTTOUTm is changed by a TAUDTTINm edge detection signal and slave channel 2 INTTAUDnIm signal to output one-phase PWM waveform with dead time. Afterwards, this operation is repeated.
Stop Operation	Set TAUDnTT.TAUDnTTm = 1 for slave channel 2. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm stops. TAUDnCNTm and TAUDTTOUTm retain their current values.

Restart

17.4.13 Synchronous Non-Complementary and Complementary Modulation Output Functions

This section describes functions that generate 6-phase triangle PWM output using a master channel and 7 slave channels.

- **Section 17.4.13.1, Non-Complementary Modulation Output Function Type 1**
- **Section 17.4.13.2, Non-Complementary Modulation Output Function Type 2**
- **Section 17.4.13.3, Complementary Modulation Output Function**

17.4.13.1 Non-Complementary Modulation Output Function Type 1

(1) Overview

Summary

This function outputs a PWM signal, a high-level signal, or a low-level signal from TAUDTTOUTm depending on the values of the real-time output bits (TAUDnTRO.TAUDnTROm) and the modulation output enable bits (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- One master channel and seven slave channels
- The operation mode of the master channel must be set to interval timer mode (see **Table 17.201, Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 1**).
- The operating mode for slave channels 1 to 7 should be set to one-count mode (see **Table 17.204, Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1**).
- TAUDTTOUTm is not used with the master channel of this function.
- TAUDTTOUTm of slave channel 1 is not used with this function, but TAUDnTRC.TRCm should be set to 1 (see **Section 17.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 1 with non-complementary modulation output (see **Section 17.4.4, Channel Output Modes**).
- TAUDnCDRm of slave channel 1 should be set to 0000_H.

Functional description

The master/slave channel counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) is loaded into the counter (TAUDnCNTm) and the counter starts to count down. When the counter reaches 0000_H, INTTAUDnlm occurs.

- Slave channel 1:
  Slave channel 1 is set as a channel that triggers real-time output (TAUDnTRC.TAUDnTRCm = 1). If an interrupt occurs on slave channel 1 (TAUDnCDRm is fixed to 0000_H), the value of real-time output bit (TAUDnTRO.TAUDnTROm) of the channel that has detected the interrupt on slave channel 1 changes. After that, the counter returns to $FFFF_H$ and waits for the next interrupt of master channel.

- Slave channel 2:
  Slave channel 2 generates a PWM output. The master channel specifies a PWM output cycle and slave channel 2 specifies a duty cycle. After generating an interrupt, the counter returns to $FFFF_H$ and awaits the next interrupt from the master channel.

Slave channels 3 to 7 operate like slave channel 2.

As described in **Table 17.200, TAUDTTOUTm Output from One Pair of Slave Channels of Non-Complementary Modulation Output Function Type 1**, a signal output from TAUDTTOUTm depends on the value of the real-time output bit (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of slave channel.

This function cannot use a forced restart. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TMEm = 0 on slave channels 2 to 7:
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs a high-level signal.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TMEm = 1 on slave channels 2 to 7:
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs PWM corresponding to the channel.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. The PWM signal becomes inverted logic. Only the initial setting of TAUDnTOL.TAUDnTOLm is permitted (cannot be changed during operation).

Table 17.200 TAUDTTOUTm Output from One Pair of Slave Channels of Non-Complementary Modulation Output Function Type 1

TAUDnTME.TAUDnTMEm	TAUDnTRO.TAUDnTROm	TAUDTTOUTm Output
0	0	Low level
	1	High level
1	0	Low level
	1	PWM (positive logic)

- This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.
- TAUDnCDRm value of slave channel 1 should be set to 0000_H so that a real-time output is triggered at the same time with PWM generation on slave channels 2 to 7.
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set low before TAUDnTE.TAUDnTEm is set to 0.
- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set high before TAUDnTE.TAUDnTEm is set to 0.

(2) Equations

Slave channels 2 to 7:

PWM output cycle = [TAUDnCDRm (master) + 1] × count clock

PWM output duty time = [TAUDnCDRm (slave)] × count clock

(3) Block Diagram and General Timing Diagram

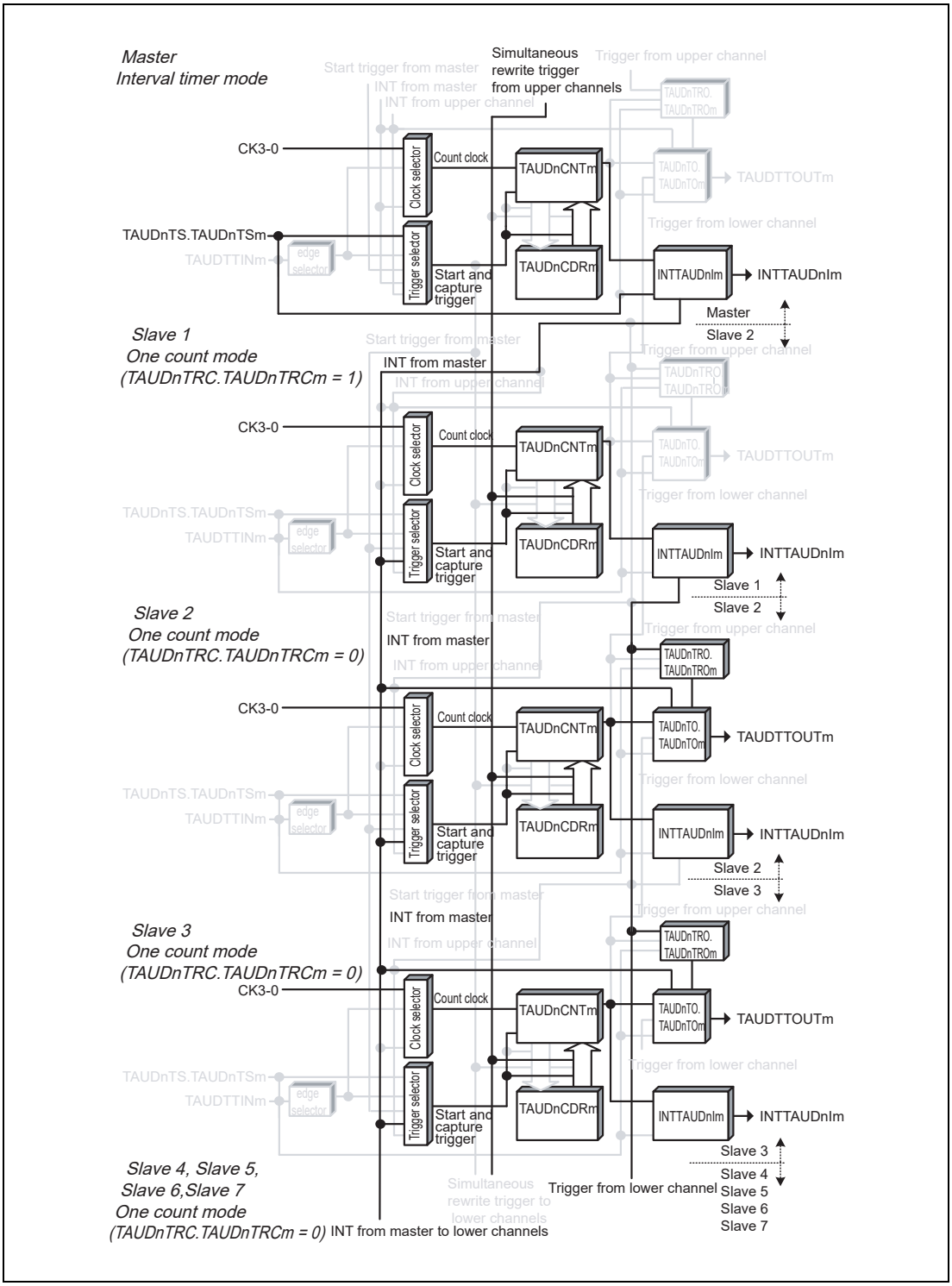


Figure 17.117 Block Diagram of Non-Complementary Modulation Output Function Type 1

The following settings apply to the general timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

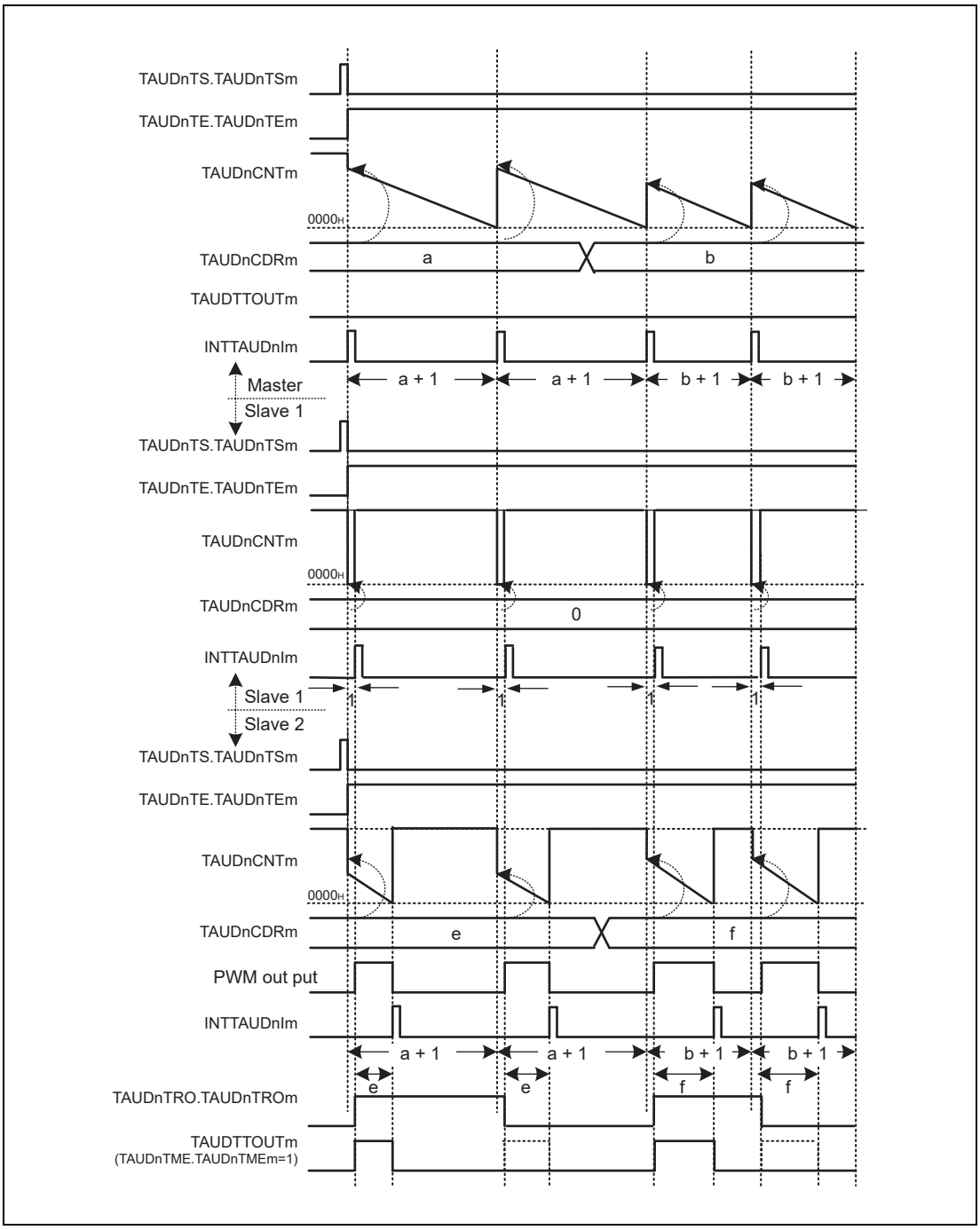


Figure 17.118 General Timing Diagram of Non-Complementary Modulation Output Function Type 1

NOTE

TAUDTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUDnIm of the master channel.

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.201 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	1: INTTAUDnIm is generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.202 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

TAUDnTOE.TAUDnTOEm is set to 0 because channel output mode is not used with this function.

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.203 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

Conduct operation settings under the following conditions.

- Simultaneous rewrite trigger output function type 1 setting channel:
  TAUDnRDCm = 1, TAUDnRDSm = 1
  In addition, TAUDnCDRm settings for this channel are as follows.
  = ((TAUDnCDRm setting for the master channel targeted for simultaneous rewrite + 1) × Interrupt count) – 1
- Master channel: TAUDnRDCm = 0, TAUDnRDSm = 1
- Slave channel: TAUDnRDCm = 0, TAUDnRDSm = 1

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.204 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is enabled.

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.205 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function. However, this mode can be used in independent channel output mode controlled by software.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.206 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register Settings for Slave Channels 2 to 7

(a) TAUDnCMORm for slave channels 2 to 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.207 Contents of TAUDnCMORm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	100: INTTAUDnIm of master channel is a start trigger.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Start trigger during operation is enabled.

(b) TAUDnCMURm for slave channels 2 to 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.208 Contents of TAUDnCMURm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for slave channels 2 to 7

Table 17.209 Control Bit Settings for Slave Channels 2 to 7 in Synchronous Channel Output Mode 1 with Non-Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

(d) Simultaneous rewrite of slave channels 2 to 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.210 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	0: Generates a simultaneous rewrite trigger signal when the master channel starts to count.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Operating Procedure for Non-Complementary Modulation Output Function Type 1

Table 17.211 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2 to 7: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (6), Register Settings for Slave Channels 2 to 7**. Set the value of TAUDnCDRm register of every channel. Set a pulse cycle with TAUDnCDRm of master channel, 0000_H in TAUDnCDRm of slave channel 1, and duty width with TAUDnCDRm of slave channels 2 to 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 17.211 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
During Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, and TAUDnTMEm.TAUDnTMEm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel, slave channel 1 and slave channels 2 to 7 is loaded into TAUDnCNTm to perform counting down. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • PWM output signals of slave channels 2 to 7 are set/reset. • TAUDnCDRm value of slave channel 1 is reloaded into TAUDnCNTm to perform counting down. • TAUDnCDRm value of slave channels 2 to 7 is reloaded into TAUDnCNTm to perform counting down. • When the counter of slave channel 1 or slave channels 2 to 7 reaches 0000_H: – INTTAUDnIm is generated. • When the counter of slave channels 2 to 7 reaches 0000_H: – INTTAUDnIm is generated. – PWM output signals of slave channels 2 to 7 are reset. TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bits (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

The following settings apply to the specific timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

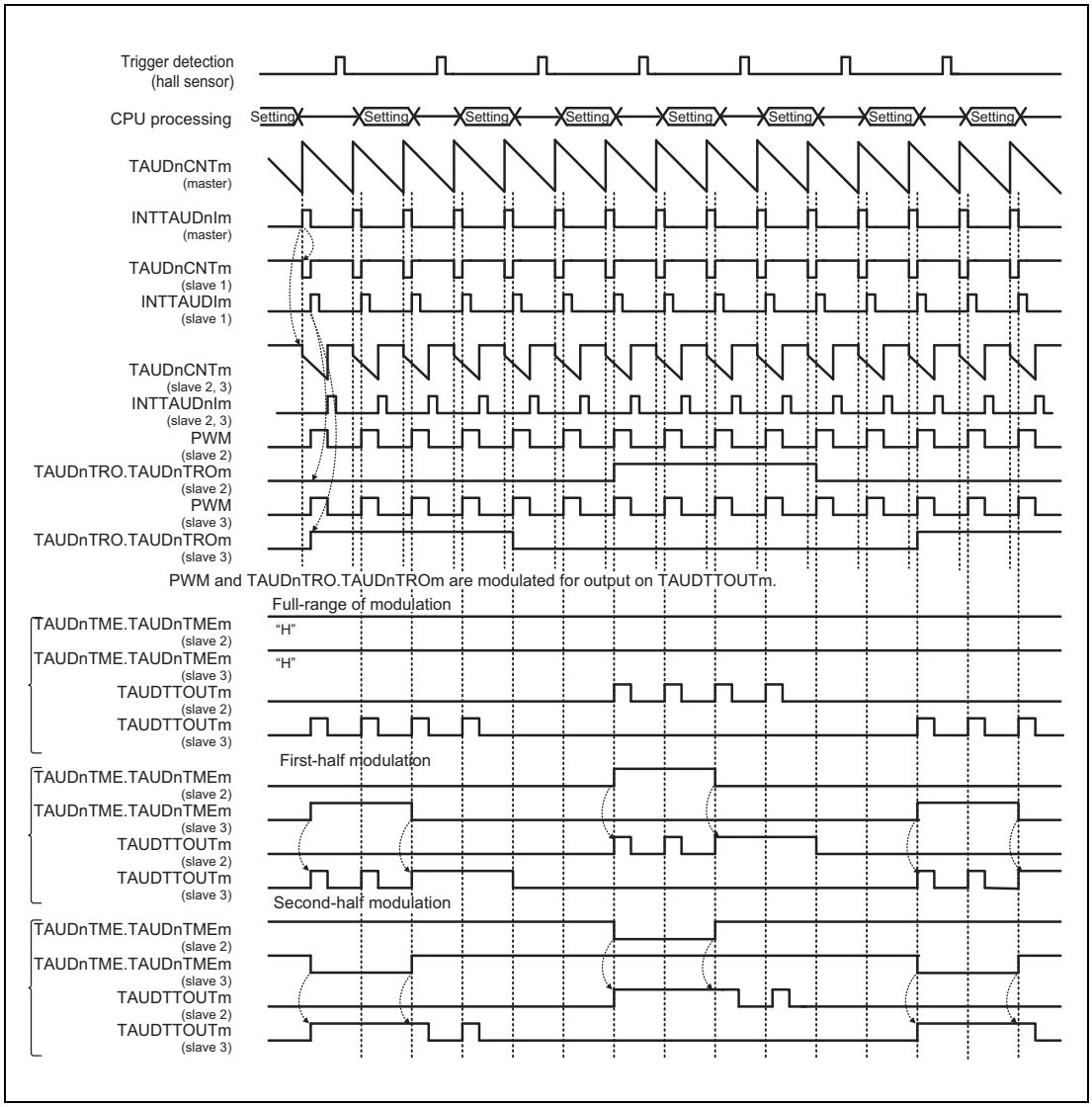


Figure 17.119 Specific Timing Diagram of Non-Complementary Modulation Output Function Type 1

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

The "Setting" symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, and TAUDnTRO.TAUDnTROm can be changed.

TAUDnTME.TAUDnTMEm setting is reflected by detecting the count start timing and master channel cycle. According to the modified setting, modulation waveforms are output from TAUDTTOUTm.

TAUDnTRO.TAUDnTROm bit value is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

17.4.13.2 Non-Complementary Modulation Output Function Type 2

(1) Overview

Summary

This function outputs a PWM signal, a high-level signal, or low-level signal from TAUDTTOUTm depending on the real-time output bit value (TAUDnTRO.TAUDnTROm) and the modulation output enable bit value (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- One master channel and seven slave channels
- The operation mode of the master channel must be set to interval timer mode (see **Table 17.213, Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 2**).
- The operating mode for slave channel 1 should be set to event count mode (see **Table 17.217, Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2**).
- The operating mode for slave channels 2 to 7 should be set to up/down count mode (see **Table 17.220, Contents of TAUDnCMORm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2**).
- The output mode of the master channel must be set to independent channel output mode 1 (see **Section 17.4.4, Channel Output Modes**).
- This function does not use TAUDTTOUTm of slave channel 1 but TAUDnTRC.TAUDnTRCm should be set to 1 (see **Section 17.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 2 with non-complementary modulation output (see **Section 17.4.4, Channel Output Modes**).

Functional description

The master/slave channel counter is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) is loaded into the counter (TAUDnCNTm).

- Master channel:
  The counter of master channel starts to count down. When the counter reaches 0000_H, INTTAUDnIm occurs.
- Slave channel 1:
  When slave channel 1 detects an interrupt from the master channel, the TAUDnCNTm value is decremented. When an interrupt from the master channel is detected for the (TAUDnCDRm + 1) times, INTTAUDnIm is generated. Then, the TAUDnCDRm value is loaded into TAUDnCNTm to continue operation subsequently.
  Since slave channel 1 is set as a real-time output trigger channel (TAUDnTRC.TAUDnTRCm = 1), the real-time output bit (TAUDnTRO.TAUDnTROm) of the channel which detects an interrupt on the corresponding channel is reflected to the respective TAUDTTOUTm outputs when an interrupt occurs on slave 1 channel.

- Slave channel 2:
  Once detecting an interrupt from the master channel, TAUDnCNTm counts in the reverse direction. When an interrupt is detected during count-up operation, TAUDnCDRm value is reloaded and then the counter starts to count down.
  If TAUDnCNTm = 0001_H, an interrupt occurs and a PWM output signal is set/reset.

The combined use of the master channel and slave channel 2 generates a PWM output signal. The master channel generates a PWM output cycle and slave channel 2 generate a duty cycle.

Slave channels 3 to 7 operates like slave channel 2.

A signal that is output from TAUDTTOUTm depends on a real-time output bit value (TAUDnTRO.TAUDnTROm) and a modulation output bit value (TAUDnTME.TAUDnTMEm) of the slave channel, as described in **Table 17.212, TAUDTTOUTm Output of a Pair of Slave Channels in Non-Complementary Modulation Output Function Type 2 (TAUDnTOL.TAUDnTOLm = 0)**.

This function cannot make a forced restart. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TAUDnTMEm = 0 on slave channels (TAUDnTOL.TAUDnTOLm = 0):
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs a high-level signal.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TAUDnTMEm = 1 on slave channels (TAUDnTOL.TAUDnTOLm = 0):
  - If the channel's TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm outputs PWM corresponding to the channel.
  - If the channel's TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. The PWM signal becomes inverted logic. Only the initial setting of TAUDnTOL.TAUDnTOLm is permitted (cannot be changed during operation).

Table 17.212 TAUDTTOUTm Output of a Pair of Slave Channels in Non-Complementary Modulation Output Function Type 2 (TAUDnTOL.TAUDnTOLm = 0)

TAUDnTME.TAUDnTMEm	TAUDnTRO.TAUDnTROm	TAUDTTOUTm Output
0	0	Low level
	1	High level
1	0	Low level
	1	PWM (positive logic)

- This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set low before TAUDnTE.TAUDnTEm is set to 0.
- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7, TAUDnTO.TAUDnTOm is set high before TAUDnTE.TAUDnTEm is set to 0.

(2) Equations

Slave channels 2 to 7:

Carrier cycle (down/up) = [TAUDnCDRm (master) + 1] × 2 × count clock cycle

Duty time = [TAUDnCDRm (master) + 1 – TAUDnCDRm (slave)] × 2 × count clock cycle

(3) Block Diagram and General Timing Diagram

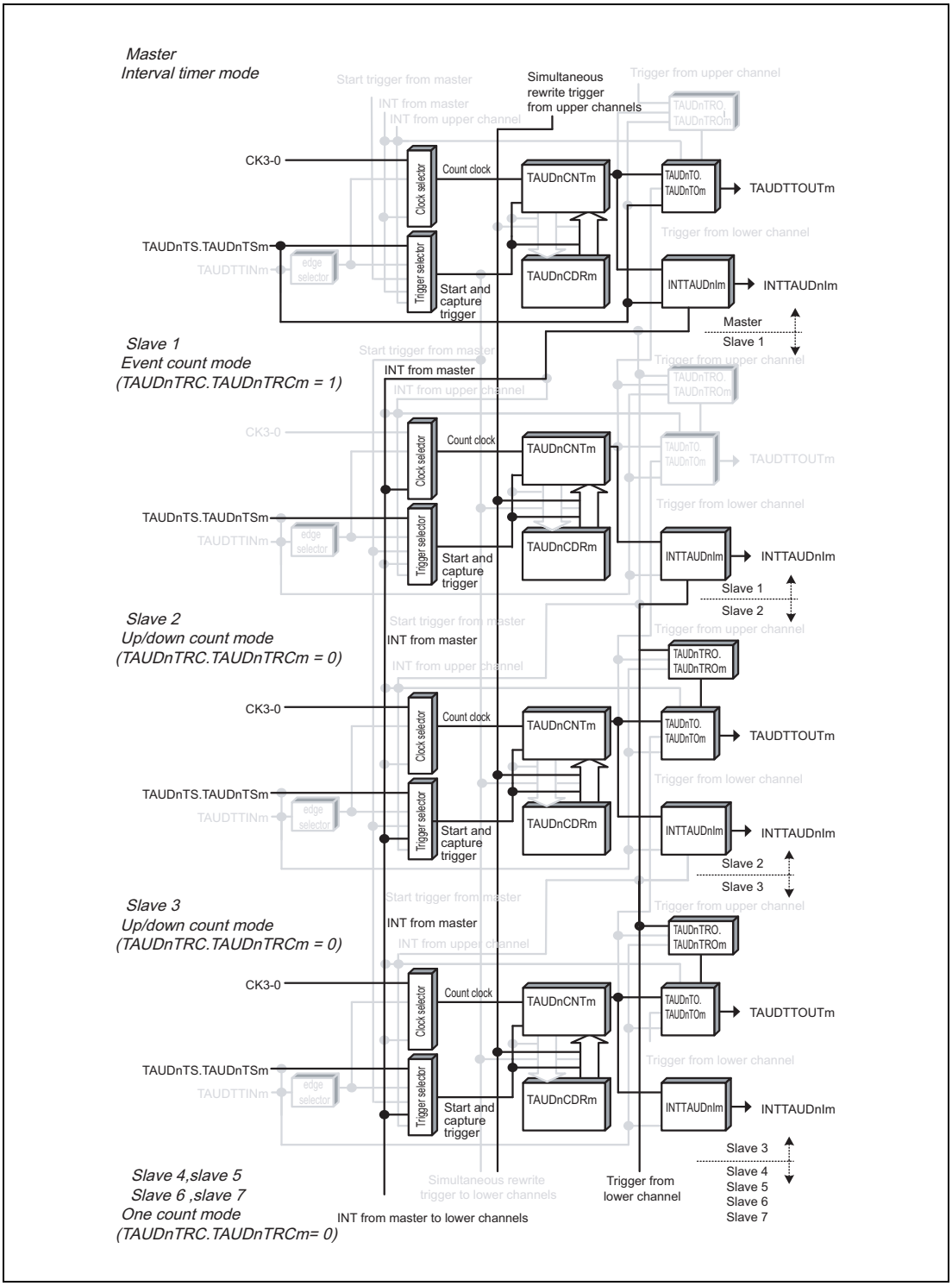


Figure 17.120 Block Diagram of Non-Complementary Modulation Output Function Type 2

The following settings apply to the general timing diagram.

- Master channel: INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

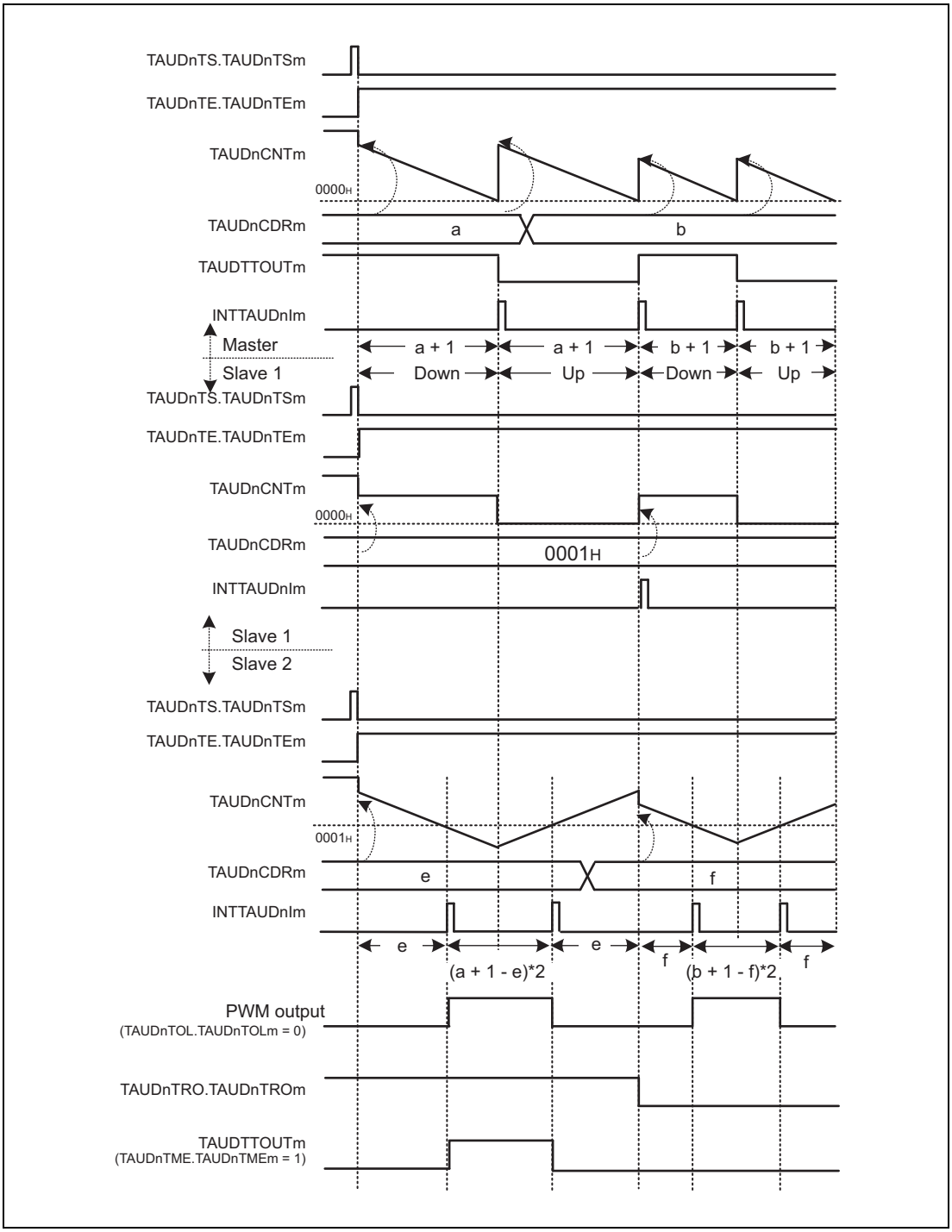


Figure 17.121 General Timing Diagram of Non-Complementary Modulation Output Function Type 2

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.213 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time. 1: INTTAUDnIm is generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 17.214 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode for master channels

Table 17.215 Control Bit Settings for Master Channels in Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (toggle mode with TAUDnTOM.TAUDnTOMm = 0)
TAUDnTOL.TAUDnTOLm	0:The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TAUDnTREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.216 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Disables simultaneous rewrite
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.217 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software. 011: Triggers simultaneous rewrite.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.218 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function. However, this mode can be used in independent channel output mode controlled by software.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.219 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register settings for slave channels 2 to 7

(a) TAUDnCMORm for slave channels 2 to 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.220 Contents of TAUDnCMORm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: The up/down output trigger signal of the master channel
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channels 2 to 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.221 Contents of TAUDnCMURm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 2 to 7

Table 17.222 Control Bit Settings for Slave Channels 2 to 7 in Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: The upper channel generates the real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

(d) Simultaneous rewrite for slave channels 2 to 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.223 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 2

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Operating Procedure for Non-Complementary Modulation Output Function Type 2

Table 17.224 Operating Procedure for Non-Complementary Modulation Output Function Type 2 (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2 to 7: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (6), Register settings for slave channels 2 to 7**. Set the value of TAUDnCDRm register of every channel. Set pulse cycle in TAUDnCDRm of master channel, and in TAUDnCDRm of slave channel 1, set the number of interrupts from master channel to be ignored before slave channel 1 generates an input signal. Set duty width in TAUDnCDRm of slave channels 2 to 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 17.224 Operating Procedure for Non-Complementary Modulation Output Function Type 2 (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
During Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, and TAUDnTMEm.TAUDnTMEm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channels 2 to 7 is loaded into TAUDnCNTm to perform count down. The TAUDnCDRm value of slave channel 1 is loaded and the counter waits for an interrupt from the master channel. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • The TAUDnCNTm value of slave channel 1 decrements by 1 and the counter waits for a next interrupt from the master channel. • TAUDnCNTm of slave channels 2 to 7 performs counting in opposite direction. • When the counter of slave channel 1 reaches 0000_H, it waits for a next interrupt from the master channel. When an interrupt is detected: – INTTAUDnIm is generated. • When the counter of slave channels 2 to 7 reaches 0001_H: – INTTAUDnIm is generated. – PWM output signals of slave channels 2 to 7 are set/reset. TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bits (TAUDnTRO.TAUDnTROm) and modulation output bit (TAUDnTME.TAUDnTMEm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(8) Specific Timing Diagrams

The following settings apply to the general timing diagram.

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

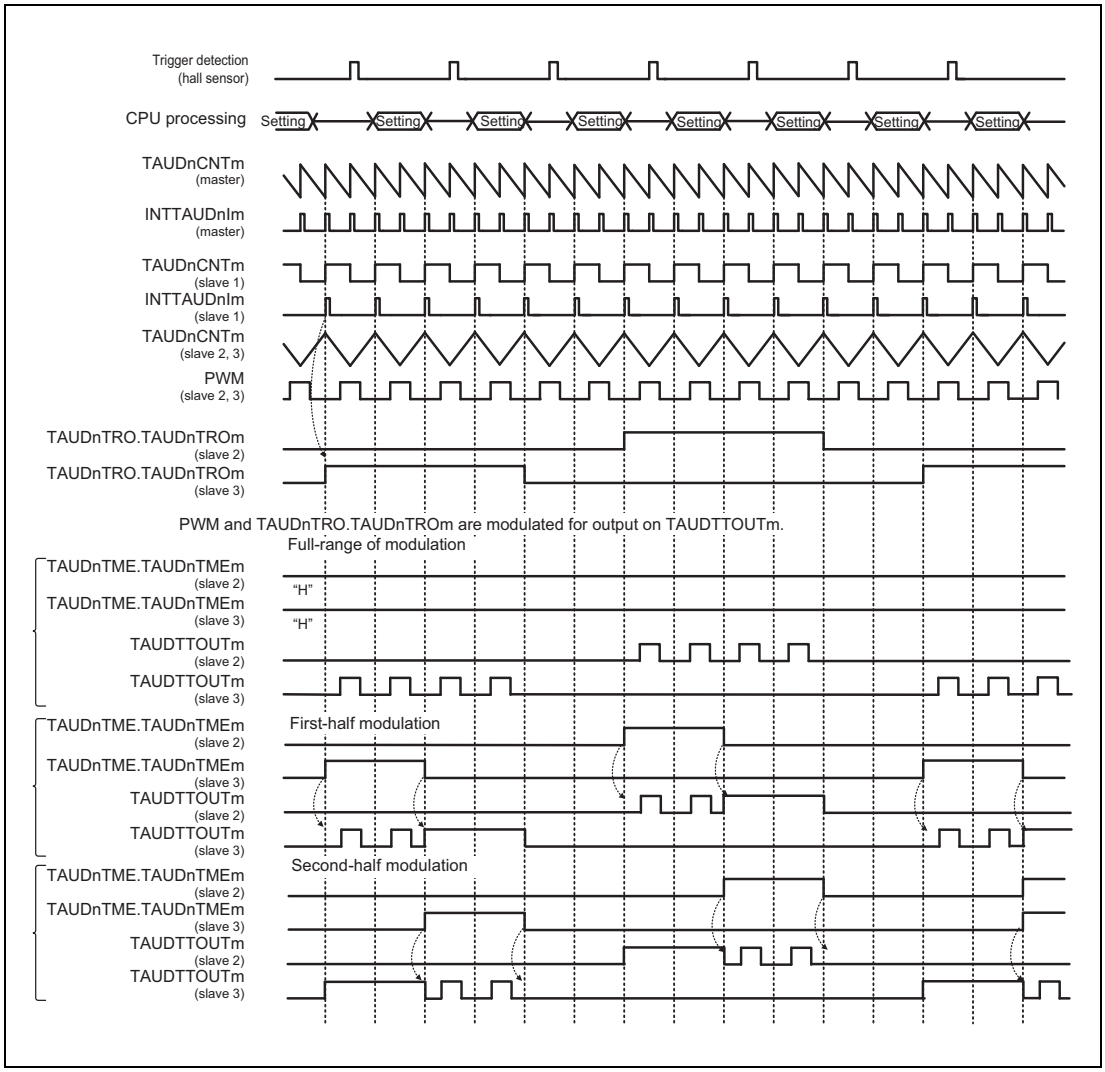


Figure 17.122 Specific Timing Diagram of Non-Complementary Modulation Output Function Type 2

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

The "Setting" symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, and TAUDnTRO.TAUDnTROm can be changed.

TAUDnTME.TAUDnTMEm setting is reflected by detecting the count start timing and triangle PWM carrier cycle (peak interrupt timing).

TAUDnTRO.TAUDnTROm bit value is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

17.4.13.3 Complementary Modulation Output Function

(1) Overview

Summary

This function outputs a triangle PWM output signal, a high-level signal, or low-level signal from TAUDTTOUTm depending on the real-time output bit value (TAUDnTRO.TAUDnTROm) and the modulation output bit value (TAUDnTME.TAUDnTMEm) of a pair of slave channels. Three pairs of channels are typically used.

Prerequisites

- One master channel and seven slave channels
- The operation mode of the master channel must be set to interval timer mode (see **Table 17.227, Contents of TAUDnCMORm Register for Master Channels of Complementary Modulation Output Function**).
- The operating mode for slave channel 1 should be set to event count mode (see **Table 17.231, Contents of TAUDnCMORm Register for Slave Channel 1 of Complementary Modulation Output Function**).
- The operating mode for slave channels 2, 4 and 6 should be set to up/down count mode (see **Table 17.234, Contents of TAUDnCMORm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function**).
- The operating mode for slave channels 3, 5 and 7 should be set to one-count mode (see **Table 17.238, Contents of TAUDnCMORm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function**).
- The output mode for master channels should be set to independent channel output mode 1 (see **17.4.4, Channel Output Modes**).
- This function does not use TAUDTTOUTm of slave channel 1 but TAUDnTRC.TAUDnTRCm should be set to 1 (see **Section 17.4.4, Channel Output Modes**).
- The channel output mode for slave channels 2 to 7 should be set to synchronous channel output mode 2 with complementary modulation output (see **Section 17.4.4, Channel Output Modes**).

Functional description

- Master channel:
  The counter of master channel is enabled by setting the channel trigger bit (TAUDnTS.TAUDnTSm) to 1. This sets TAUDnTE.TAUDnTEm = 1, enabling count operation. The value of data register (TAUDnCDRm) of master channel is loaded into the counter (TAUDnCNTm) and the counter starts to count down from this value.
  When the counter of master channel reaches 0000_H, INTTAUDnIm occurs. This decrements the counter value of slave channel 1 by 1 and the counter of slave channel 2 starts to count in the opposite direction.
- Slave channel 1:
  When the counter reaches 0000_H, slave channel 1 waits for the next interrupt from the master channel. And the TAUDnCDRm value is reloaded into TAUDnCNTm (slave 1) and INTTAUDnIm is generated.

Slave channel 1 is set as a real-time output trigger channel (TAUDnTRC.TAUDnTRCm = 1). The value of real-time output bit (TAUDnTRO.TAUDnTROm) of each channel is applied to the channel that detects the occurrence of an interrupt on slave channel 1. The real-time output bit value can be changed in any timing by application software but a new value is not applied until an interrupt occurs on slave channel 1.

- Slave channel 2:
  When the slave channel 2 counter reaches 0001_H, the slave channel 3 counter starts counting down. When the slave channel 3 counter reaches 0000_H, an interrupt occurs.
- Slave channels 2 and 3:
  The combined use of the master channel and slave channels 2 and 3 generates a PWM output signal. The master channel generates a PWM output cycle, slave channel 2 generates a duty cycle, and slave channel 3 generates dead time.
- Slave channels 4 to 7:
  Slave channels 4 and 6 operate like slave channel 2. Slave channels 5 and 7 operate like slave channel 3.

A signal that is output from TAUDTTOUTm depends on a real-time output bit value (TAUDnTRO.TAUDnTROm), a modulation output bit value (TAUDnTME.TAUDnTMEm), and an output level bit value (TAUDnTDL.TAUDnTDLm) of the slave channel, as described in **Table 17.225, TAUDTTOUTm Output (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function**.

It is, however, prohibited that a high-level signal is output from both channel 2 and channel 3 (in order to prevent a motor driver short circuit).

Forced restart is not possible for this function. The counter can be stopped by setting TAUDnTT.TAUDnTTm of master and slave channels to 1. This sets TAUDnTE.TAUDnTEm to 0. TAUDnCNTm and TAUDTTOUTm of master and slave channels stop but retain their values. The counters can be restarted by setting TAUDnTS.TAUDnTSm to 1.

Conditions

- If TAUDnTME.TAUDnTMEm of a pair of channels is set to 1 (TAUDnTOL.TAUDnTOLm = 0):
  - If TAUDnTRO.TAUDnTROm of one channel is set to 1, TAUDTTOUTm outputs the corresponding PWM of the channel.
  - If TAUDnTRO.TAUDnTROm of both channels is set to 0, a pair of TAUDTTOUTm outputs a low-level signal.
- If TAUDnTME.TAUDnTMEm of a pair of channels is set to 0 (TAUDnTOL.TAUDnTOLm = 0):
  - If TAUDnTRO.TAUDnTROm is set to 1, TAUDTTOUTm of the channel outputs a high-level signal.
  - If TAUDnTRO.TAUDnTROm is set to 0, TAUDTTOUTm of the channel outputs a low-level signal.
- If TAUDnTOL.TAUDnTOLm is set to 1, high-level and low-level signals output from TAUDTTOUTm are inverted. No PWM signals are changed depending on the setting of TAUDnTOL.TAUDnTOLm. The PWM signal is negative logic.

Table 17.225 TAUDTTOUTm Output (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function

TAUDnTME. TAUDnTME2	TAUDnTME. TAUDnTME3	TAUDnTRO. TAUDnTRO2	TAUDnTRO. TAUDnTRO3	TAUDnTDL. TAUDnTDL2	TAUDnTDL. TAUDnTDL3	TAUDTTOUT2 Output	TAUDTTOUT3 Output
0	0	0	0	x	x	Low level	Low level
		0	1	1	0	Low level	High level
		1	0	0	1	High level	Low level
		1	1	x	x	Setting prohibited	Setting prohibited
1	1	0	0	x	x	Low level	Low level
		0	1	1	0	~PWMm	PWMm
		1	0	0	1	PWMm	~PWMm
		1	1	x	x	Setting prohibited	Setting prohibited

NOTES

1. In the above table, PWM indicates a positive PWM signal and ~PWM indicates an inverted PWM signal (positive logic). PWM and ~PWM are set by TAUDnTDL.TAUDnTDLm.
2. Any settings not listed above are prohibited.

- If TAUDnTME.TAUDnTMEm is continuously set to 1 while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, full modulation is applied.
- If TAUDnTME.TAUDnTMEm is set to 1 at the first half of the period while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, first-half modulation is applied.
- If TAUDnTME.TAUDnTMEm is set to 1 at the second half of the period while TAUDnTRO.TAUDnTROm of one of paired channels is set to 1, second-half modulation is applied.
- Whether dead time is added to a normal or reverse phase PWM signal when two channels become high-level signal outputs simultaneously depends on a TAUDnTDL.TAUDnTDLm bit value.
  - If TAUDnTDL.TAUDnTDLm = 0, dead time is added to a normal phase PWM signal.
  - If TAUDnTDL.TAUDnTDLm = 1, dead time is added to a reverse phase PWM signal.
  - The operation defined by a TAUDnTDL.TAUDnTDLm bit value should be conducted by application software during operation. To modify TAUDnTDL.TAUDnTDLm, rewrite it during the period when TAUDnTRO.TAUDnTROm is 00_B.
- The TAUDnCDRm value of slave channel 1 should be set to the value to generate INTTAUDnIm of slave channel 1 at a carrier cycle (peak interrupt timing).
- If TAUDnTOL.TAUDnTOLm is set to 0 on slave channels 2 to 7:
  - If TAUDnTDL.TAUDnTDLm is set to 0, TAUDnTO.TAUDnTOm is set to 0 (low) before TAUDnTE.TAUDnTEm is set to 0.
  - If TAUDnTDL.TAUDnTDLm is set to 1, TAUDnTO.TAUDnTOm is set to 1 (high) before TAUDnTE.TAUDnTEm is set to 0.
- If TAUDnTOL.TAUDnTOLm is set to 1 on slave channels 2 to 7:

– If TAUDnTDL.TAUDnTDLm is set to 0, TAUDnTO.TAUDnTOm is set to 1 (high) before TAUDnTE.TAUDnTEm is set to 0.

– If TAUDnTDL.TAUDnTDLm is set to 1, TAUDnTO.TAUDnTOm is set to 0 (low) before TAUDnTE.TAUDnTEm is set to 0.

- This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.

Table 17.226 TAUDnTDL.TAUDnTDLm Settings (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function

TAUDnTME. TAUDnTME2	TAUDnTME. TAUDnTME3	TAUDnTRO. TAUDnTRO2	TAUDnTRO. TAUDnTRO3	TAUDnTDL. TAUDnTDL2	TAUDnTDL. TAUDnTDL3
0	0	0	0	1	1
		0	1	1	0
		1	0	0	1
1	1	0	0	1	1
		0	1	1	0
		1	0	0	1

- The value of TAUDnCDRm of slave channel 1 should be set to 1 so that INTTAUDnIm is generated on slave channel 1 at the peak of a carrier cycle.
- Set TAUDnCMORm.TAUDnMD0 of master channel to 0.
- This function enables simultaneous rewrite. See **Section 17.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse period = (TAUDnCDRm (master) + 1) × count clock cycle

$0000_H \leq$ TAUDnCDRm (master) $< FFFF_H$

Carrier cycle (down/up) = (TAUDnCDRm (master) + 1) × 2 × count clock cycle

For slave channels 2 and 3:

PWM signal width (positive phase) = [(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave 2) × 2) – (TAUDnCDRm (slave 3) + 1)] × count clock cycle

PWM signal width (negative phase) = [(TAUDnCDRm (master) + 1 – TAUDnCDRm (slave 2) × 2) + (TAUDnCDRm (slave 3) + 1)] × count clock cycle

For slave channels 4 to 7:

Slave channels 4 and 6 are calculated in the same way as slave channel 2, whereas slave channels 5 and 7 are calculated as slave channel 3.

(3) Block Diagram and General Timing Diagram

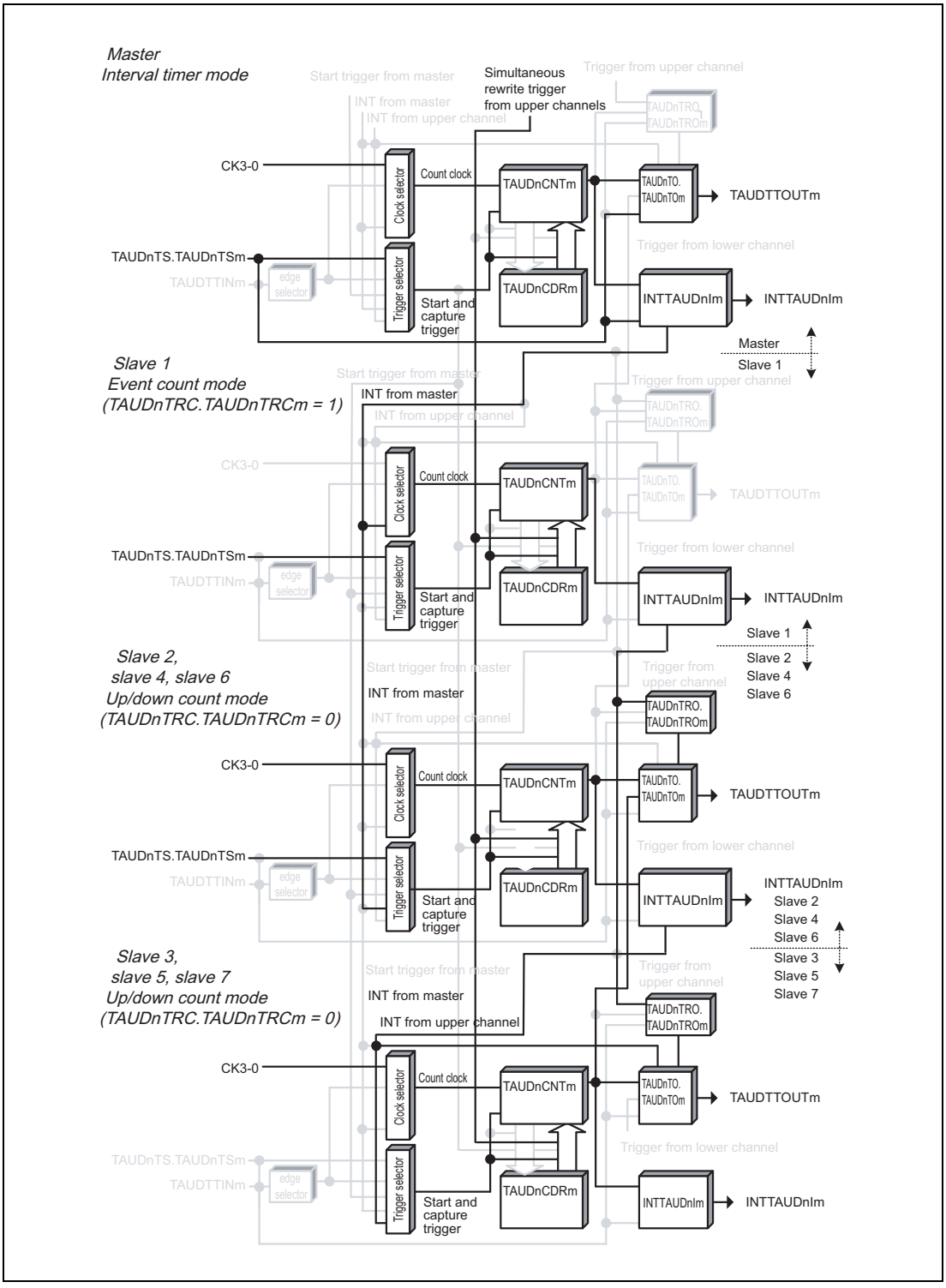


Figure 17.123 Block Diagram of Complementary Modulation Output Function

The following settings apply to the general timing diagram.

- Master channel: INTTAUDnIm not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Slave channels 1: TAUDnCDRm = 0001_H

- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

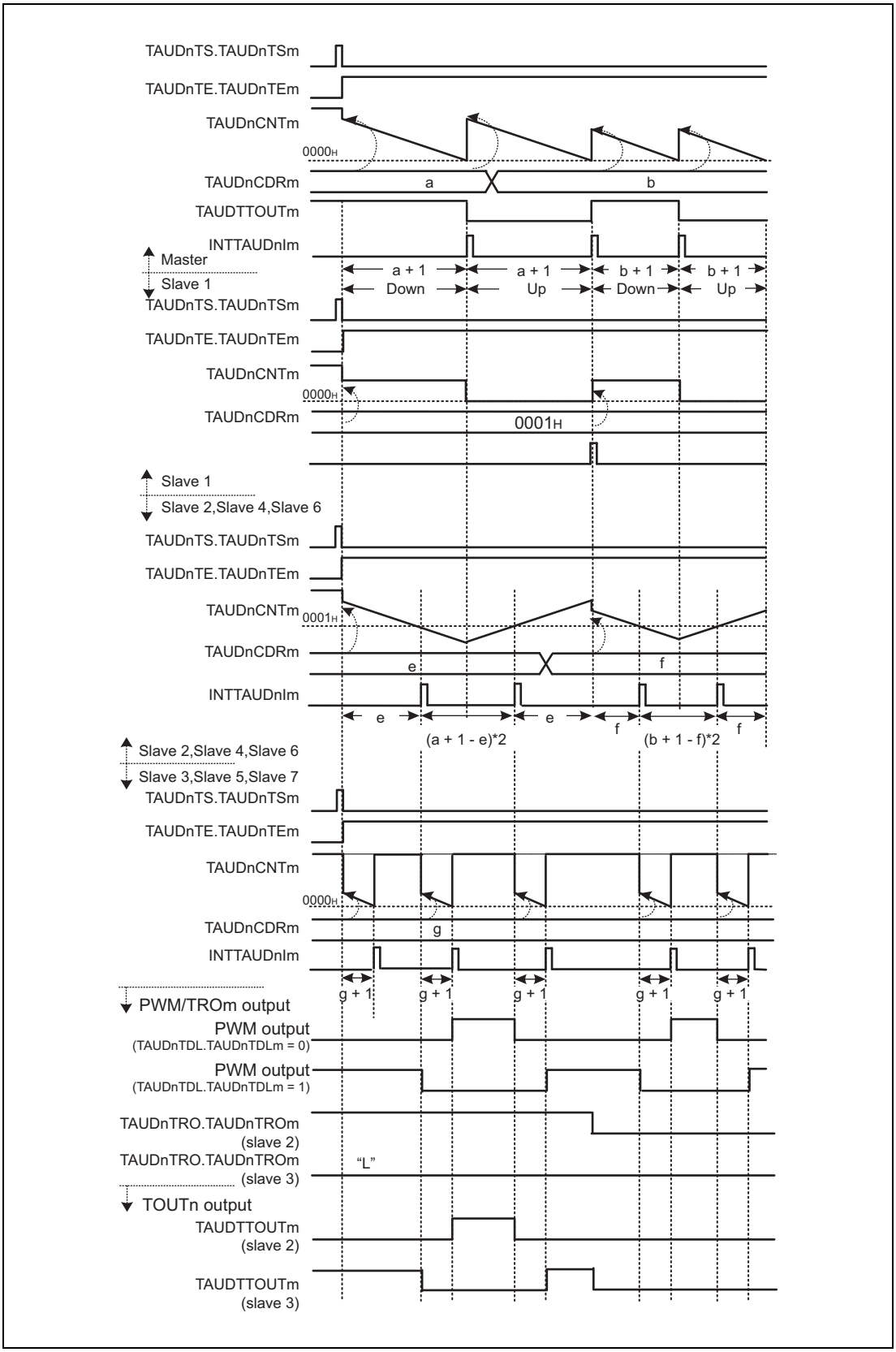


Figure 17.124 General Timing Diagram of Complementary Modulation Output Function

(4) Register Settings for Master Channels

(a) TAUDnCMORm for master channels

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.227 Contents of TAUDnCMORm Register for Master Channels of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	1: Channel is master channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0000: Interval timer mode
0	TAUDnMD0	0: INTTAUDnIm is not generated and TAUDTTOUTm is not toggled at the beginning of operation or at a restart time. 1: INTTAUDnIm is generated to toggle TAUDTTOUTm at the beginning of an operation.

(b) TAUDnCMURm for master channels

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.228 Contents of TAUDnCMURm Register for Master Channels of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

Table 17.229 Control Bit Settings in Independent Channel Output Mode 1

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	0: Independent channel output
TAUDnTOC.TAUDnTOCm	0: Operating mode 1 (Toggle mode if TAUDnTOM.TOMm = 0)
TAUDnTOL.TAUDnTOLm	0: The setting is disabled in toggle mode (the value after reset).
TAUDnTDE.TAUDnTDEm	0: Disables dead time operation
TAUDnTDM.TAUDnTDMm	0: When dead time operation is disabled (TAUDnTDE.TDEm = 0), set these bits to 0
TAUDnTDL.TAUDnTDLm	
TAUDnTRE.TAUDnTREm	0: Disables real-time output
TAUDnTRO.TAUDnTROm	0: When real-time output is disabled (TAUDnTRE.TREm = 0), set these bits to 0
TAUDnTRC.TAUDnTRCm	
TAUDnTME.TAUDnTMEm	0: Disables modulation

(d) Simultaneous rewrite for master channels

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.230 Simultaneous Rewrite Settings for Master Channels of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

NOTE

If TAUDnRDS.TAUDnRDSm = 1, it is necessary for an upper channel higher than a master channel to generate a simultaneous rewrite trigger signal.

(5) Register Settings for Slave Channel 1

(a) TAUDnCMORm for slave channel 1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.231 Contents of TAUDnCMORm Register for Slave Channel 1 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	11: INTTAUDnIm of the master channel is used as the count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	000: Trigger the counter using software. 011: Triggers simultaneous rewrite.
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0011: Event count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channel 1

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.232 Contents of TAUDnCMURm Register for Slave Channel 1 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Channel output mode

TAUDnTOE.TAUDnTOEm is set to 0 because the channel output mode is not used on slave channel 1 with this function. However, this mode can be used in independent channel output mode controlled by software.

CAUTION

TAUDnTRC.TAUDnTRCm should be set to 1 because slave channel 1 is used as a real-time output trigger channel.

(d) Simultaneous rewrite for slave channel 1

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.233 Simultaneous Rewrite Settings for Slave Channel 1 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(6) Register settings for slave channels 2, 4, and 6

(a) TAUDnCMORm for slave channels 2, 4, and 6

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.234 Contents of TAUDnCMORm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of the TAUDnCKS[1:0] bit of the master and slave channels must be identical.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	111: Up/down output trigger signal of master channel
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	1001: Up/down count mode
0	TAUDnMD0	0: INTTAUDnIm is not generated at the beginning of operation or at a restart time.

(b) TAUDnCMURm for slave channels 2, 4, and 6

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.235 Contents of TAUDnCMURm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 2, 4, and 6

Table 17.236 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from odd-numbered channels.

(d) Simultaneous rewrite for slave channels 2, 4, and 6

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.237 Simultaneous Rewrite Settings for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: A simultaneous rewrite trigger signal is generated when master channel starts to count and the corresponding slave channel is at the peak of a triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(7) Register settings for slave channels 3, 5, and 7

(a) TAUDnCMORm for slave channels 3, 5, and 7

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUDnCKS [1:0]		TAUDnCCS [1:0]		TAUDn MAS	TAUDnSTS[2:0]			TAUDnCOS [1:0]		—	TAUDnMD[4:1]				TAUDn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 17.238 Contents of TAUDnCMORm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
15, 14	TAUDnCKS [1:0]	These bits select the operation clock. 00: Operation clock = CK0 01: Operation clock = CK1 10: Operation clock = CK2 11: Operation clock = CK3 The value of TAUDnCKS[1:0] bits must be the same for the master channel and the slave channel.
13, 12	TAUDnCCS [1:0]	00: Uses the operation clock as a count clock
11	TAUDnMAS	0: Slave channel
10 to 8	TAUDnSTS[2:0]	110: Dead time trigger
7, 6	TAUDnCOS [1:0]	00: Unused. Set to 00.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUDnMD[4:1]	0100: One-count mode
0	TAUDnMD0	1: Enables start trigger detection while counting.

(b) TAUDnCMURm for slave channels 3, 5, and 7

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUDnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 17.239 Contents of TAUDnCMURm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUDnTIS[1:0]	00: Unused. Set to 00.

(c) Output mode for slave channels 3, 5, and 7

Table 17.240 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output

Bit Name	Setting
TAUDnTOE.TAUDnTOEm	1: Enables independent channel output mode
TAUDnTOM.TAUDnTOMm	1: Synchronous channel output
TAUDnTOC.TAUDnTOCm	1: Operating mode 2
TAUDnTOL.TAUDnTOLm	0: Positive logic 1: Negative logic
TAUDnTDE.TAUDnTDEm	1: Enables dead time operation.
TAUDnTDM.TAUDnTDMm	0: Adds dead time if an interrupt is detected on an even upper channel and the conditions set by TAUDnTDL.TAUDnTDLm are satisfied.
TAUDnTDL.TAUDnTDLm	0: Adds dead time to normal phase. 1: Adds dead time to reverse phase.
TAUDnTRE.TAUDnTREm	1: Enables real-time output.
TAUDnTRO.TAUDnTROm	0: Real-time output is low. 1: Real-time output is high.
TAUDnTRC.TAUDnTRCm	0: Upper channel generates a real-time output trigger for channel m.
TAUDnTME.TAUDnTMEm	0: Disables modulation 1: Enables modulation

CAUTION

Set TAUDnTDL.TAUDnTDLm exclusively from even-numbered channels.

(d) Simultaneous rewrite for slave channels 3, 5, and 7

Both master and slave channels should have the same simultaneous rewrite settings.

Table 17.241 Simultaneous Rewrite Settings for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function

Bit Name	Setting
TAUDnRDE.TAUDnRDEm	1: Enables simultaneous rewrite.
TAUDnRDS.TAUDnRDSm	0: Monitors master channel for simultaneous rewrite triggers. 1: Monitors upper channel other than the channel group for simultaneous rewrite triggers.
TAUDnRDM.TAUDnRDMm	1: Simultaneous rewrite trigger signal is generated when master channel counter is started and the corresponding slave channel is at the peak of triangular wave.
TAUDnRDC.TAUDnRDCm	0: Does not use the channel to generate the simultaneous rewrite trigger. When TAUDnRDS.TAUDnRDSm = 0, the master channel is monitored for simultaneous rewrite triggers regardless of the value of this bit.

(8) Operating Procedure for Complementary Modulation Output Function

Table 17.242 Operating Procedure for Complementary Modulation Output Function (1/2)

	Operation	TAUDn Status
Initial Channel Setting	Master channels: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (4), Register Settings for Master Channels**. Slave channel 1: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (5), Register Settings for Slave Channel 1**. Slave channels 2, 4, and 6: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (6), Register settings for slave channels 2, 4, and 6**. Slave channels 3, 5, and 7: Set TAUDnCMORm/ TAUDnCMURm register and the channel output mode as described in **Section (7), Register settings for slave channels 3, 5, and 7**. Set the value of TAUDnCDRm register of every channel. Set a pulse cycle using TAUDnCDRm of master channel, and an interrupt count of master channel ignored using TAUDnCDRm of slave channel 1. Also set a duty width in TAUDnCDRm of slave channels 2, 4, and 6, and a dead time delay on slave channels 3, 5, and 7. Set TAUDnTRC.TAUDnTRCm to 1 on slave channel 1.	Channel operation is stopped.

Table 17.242 Operating Procedure for Complementary Modulation Output Function (2/2)

	Operation	TAUDn Status
Start Operation	Set TAUDnTS.TAUDnTSm of master and slave channels to 1 simultaneously. TAUDnTS.TAUDnTSm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm of master and slave channels is set to 1 and the counter starts counting down.
During Operation	TAUDnCDRm, TAUDnTRO.TAUDnTROm, TAUDnTME.TAUDnTMEm, and TAUDnTDL.TAUDnTDLm can be changed at any time. TAUDnCNTm and TAUDnRSF.TAUDnRSFm can be read at any time. TAUDnRDT.TAUDnRDTm can be changed during operation.	TAUDnCDRm value of master channel and slave channels 2 to 7 is loaded into TAUDnCNTm to perform count down. TAUDnCDRm value of slave channel 1 is loaded and the counter waits for a master channel interrupt. When the counter of master channel reaches 0000_H: • INTTAUDnIm is generated. • TAUDnCDRm value is reloaded into TAUDnCNTm to continue counting down. • TAUDnCNTm value of slave channel 1 decreases by 1 and the counter waits for the next master channel interrupt. • TAUDnCNTm of slave channels 2, 4, and 6 performs counting in the reverse direction. • The counter of slave channel 1 waits for the next interrupt from the master channel when reaching 0000_H. When the interrupt is detected: – TAUDnCDRm value is reloaded into TAUDnCNTm and the counter waits for the next master channel interrupt. – INTTAUDnIm is generated. – TAUDnTRO.TAUDnTROm is changeable. • When the counter of slave channels 2, 4, and 6 reaches 0001_H: – INTTAUDnIm is generated. – PWM output of slave channel m is set/reset (when the specified condition of the channel output mode is matched). – TAUDnCDRm value of slave channels 3, 5, and 7 is loaded into TAUDnCNTm to perform count down. • When the counter of slave channels 3, 5, and 7 reaches 0000_H: – INTTAUDnIm is generated. TAUDTTOUTm of slave channels 2 to 7 outputs a PWM signal, a high-level signal or low-level signal depending on the values of real-time output bit (TAUDnTRO.TAUDnTROm), modulation output bit (TAUDnTME.TAUDnTMEm), and output level bit (TAUDnTDL.TAUDnTDLm) of a pair of slave channels.
Stop Operation	Set TAUDnTT.TAUDnTTm of master and slave channels to 1 simultaneously. TAUDnTT.TAUDnTTm is a trigger bit, which is automatically cleared to 0.	TAUDnTE.TAUDnTEm is cleared to 0 and the counter stops. TAUDnCNTm and TAUDTTOUTm stop and retain their current values.

Restart

(9) Specific Timing Diagrams

The following settings apply to the timing diagram.

- Master channel: INTTAUDnIm is not generated at the beginning of operation. (TAUDnCMORm.TAUDnMD0 = 0)
- Slave channel 1: TAUDnCDRm = 0001_H
- Slave channels 2 to 7: Positive logic (TAUDnTOL.TAUDnTOLm = 0)

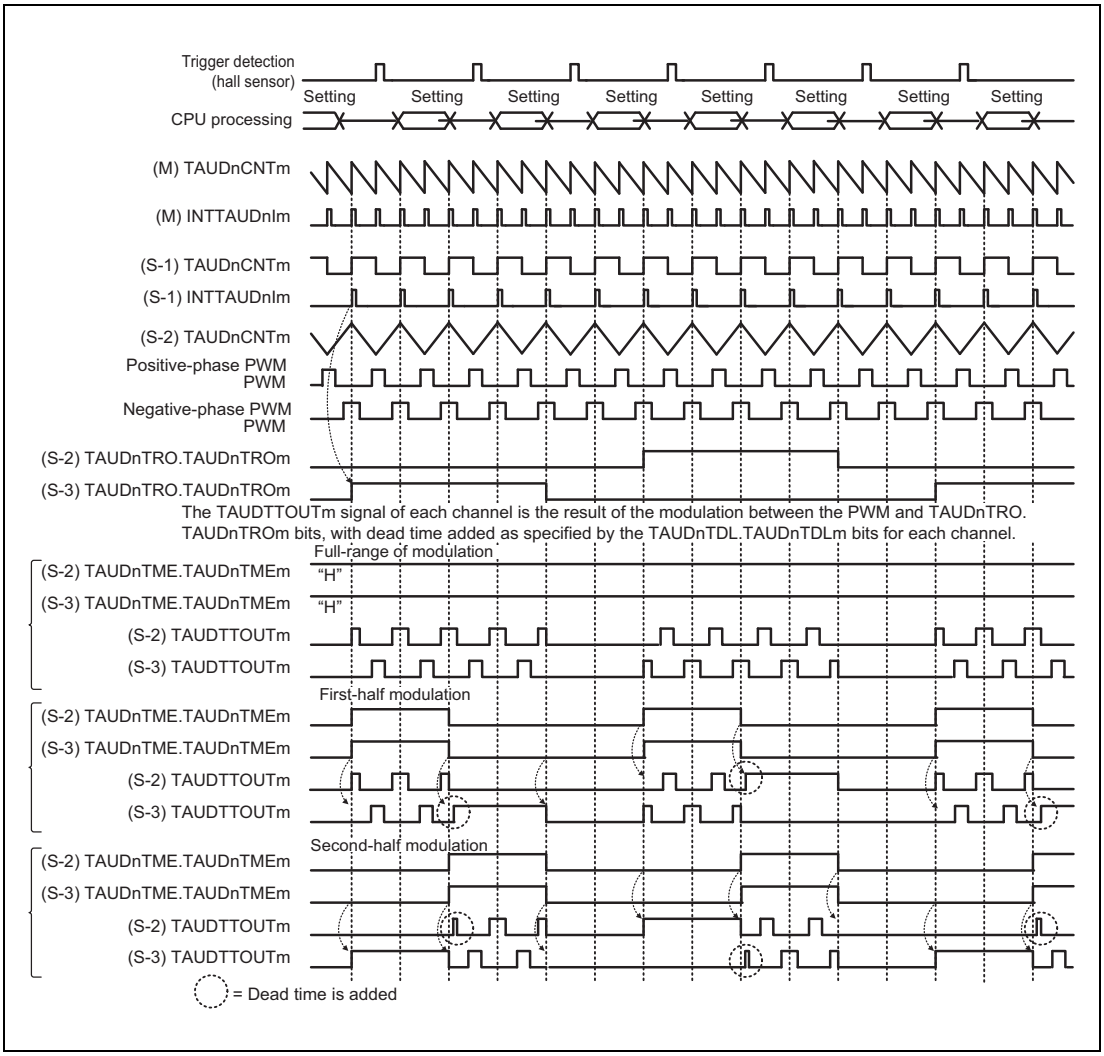


Figure 17.125 Specific Timing Diagram of Complementary Modulation Output Function

The above timing diagram shows how full modulation, first-half modulation, and second-half modulation can be achieved by modifying the TAUDnTME.TAUDnTMEm bits of lower slave channels during operation.

A modulated PWM output signal and TAUDnTRO.TAUDnTROm bit value are output from slave channels 2 and 3.

TAUDnTME.TAUDnTMEm and TAUDnTDL.TAUDnTDLm settings are reflected by detecting the count start timing and triangle PWM carrier cycle (peak interrupt timing).

The value of the TAUDnTRO.TAUDnTROm bit is set by software, but a new setting is applied only when an interrupt occurs on slave channel 1.

NOTE

Dead time is added to suppress simultaneous change of PWM edges of normal and reverse phases.

The “Setting” symbol indicates a time period when the values of TAUDnCDRm, TAUDnTME.TAUDnTMEm, TAUDnTRO.TAUDnTROm, and TAUDnTDL.TAUDnTDLm can be changed.

Section 18 Timer Array Unit J (TAUJ)

This section contains a generic description of the Timer Array Unit J (TAUJ).

The first part of this section describes all RH850/C1x specific properties, such as units, register base addresses, input/output signal names, etc.

The remainder of the section describes the features that apply to all implementations.

18.1 Overview of RH850/C1x TAUJ

18.1.1 Units

This microcontroller has following number of units of the Timer Array Unit J.

Table 18.1 Units of TAUJ

Product	RH850/C1x
Units	1
Name	TAUJ0

Table 18.2 Index

Index	Meaning
n	Throughout this section, the individual TAUJ units are identified by the index “n” (n = 0); for example, TAUJnTOM is the TAUJn channel output mode register.
m	The TAUJ has four channels. Throughout this section, the individual channels are identified by the index “m” (m = 0 to 3). A certain channel is denoted as CHm. The even numbered channels (m = 0, 2) are denoted as CHm_even. The odd numbered channels (m = 1, 3) are denoted as CHm_odd.

18.1.2 Register Base Addresses

TAUJ base addresses are listed in the following table.

TAUJ register addresses are given as offsets from the base addresses in general.

Table 18.3 Register base addresses

Base Address Name	Base Address
<TAUJ0_base>	$FFE5\ 0000_H$

18.1.3 Clock Supply

TAUJ clock supply is listed in the following table.

Table 18.4 TAUJn clock supply

Unit Name	Unit Clock Name	Supply Clock Name
TAUJ0	PCLK	CLKC_HSB (unmodulated high-speed peripheral clock)

18.1.4 Interrupts and DMA/DTS

TAUJ interrupt requests are listed in the following table.

Table 18.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMA Trigger Number	DTS Trigger Number
TAUJ0				
INTTAUJ0I0	Channel 0 interrupt	78	52	12
INTTAUJ0I1	Channel 1 interrupt	79	53	13
INTTAUJ0I2	Channel 2 interrupt	80	54	14
INTTAUJ0I3	Channel 3 interrupt	81	55	15

18.1.5 Reset Sources

TAUJ reset sources are listed in the following table. TAUJ is initialized by these reset sources.

Table 18.6 Reset Sources

Unit Name	Reset Source
TAUJ0	Reset by any reset source

18.1.6 External Input/Output Signals

External input/output signals of TAUJ are listed in the following table.

Table 18.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TAUJ0		
TAUJTTIN0-TAUJTTIN3	Channel 0 to 3 input	TAUJ0I0-TAUJ0I3
TAUJTTOUT0-TAUJTTOUT3	Channel 0 to 3 output	TAUJ0O0-TAUJ0O3

18.2 Overview

18.2.1 Functional Overview

The TAUJ has the following functions:

- Independent channel operation function (operated using a single channel)
- Synchronous channel operation function (operated using a master channel and multiple slave channels)

The timer array unit J is used to perform various count or timer operations and to output a signal which depends on the result of the operation. It contains one prescaler block for count clock generation and 4 channels, each equipped with a 32-bit counter TAUJnCNTm and a 32-bit data register TAUJnCDRm to hold the count start value or compare value.

It also contains several control and status registers.

Independent and synchronous operation

Every channel can operate in two operating modes, either independently or in combination with other channels (synchronously), i.e. multiple channels depend on each other with one master and one or more slave channels.

When a channel is operated independently, it can be operated independent of all other channels.

The synchronous operation function is implemented using a combination of channel groups (consisted of master and slave channels).

Several rules apply to the settings of channels.

18.2.1.1 Terms

In this section, the following terms are used.

Independent channel operation function / synchronous operation channel operation function

TAUJ has 4 channels, and provides an independent channel operation function that individual channels operate independently and a synchronous channel operation function that is implemented using a combination of channels.

- The independent channel operation function can use any channel independent of all other channels.
- The synchronous channel operation function is implemented using a combination of channel groups (comprised of master and slave channels).

Several rules apply to the settings of channels.

Channel group

In the synchronous channel operation function, all channels that depend on each other are referred to as a “channel group”.

A channel group has one master channel and one or more slave channels.

Operating mode

An operating mode can be selected for every channel m. The operating mode defines the basic operation and features of a channel.

In synchronous channel operation, every channel in the channel group can operate in a different operating mode.

Examples are capture mode and interval timer mode.

Channel output mode

The channel output mode defines the operation of TAUJTTOUTm

- of a single channel (independent output operation) or
- of all channels in a channel group (synchronous output operation).

The channel output mode includes independent channel output mode 1.

Channel operation function

The channel operation function defines the complete function and all features

- of a single channel (independent channel operation) or
- of all channels in a channel group (synchronous channel operation).

Upper/lower channel

Depending on the channel number m, a channel with a smaller channel number or higher channel number can be referred to as “upper” or “lower” channel:

- Upper channel: Channel with a smaller channel number
- Lower channel: Channel with a larger channel number

For instance, as to channel 2, channel 1 is an upper channel and channel 3 is a lower channel. Channel 0 is the highest channel and channel 3 is the lowest channel.

18.2.1.2 Operation Functions

TAUJ provides the following functions by operating individual channels independently or a combination of channels.

Table 18.8 TAUJ Operation Functions

Operation Function	Setting Example Reference Section
Independent Channel Operation Functions	**18.4.9**
Interval timer function	**18.4.9.1**
TAUJTTINm input interval timer function	**18.4.9.2**
TAUJTTINm input pulse interval measurement function	**18.4.9.3**
TAUJTTINm input signal width measurement function	**18.4.9.4**
TAUJTTINm input position detection function	**18.4.9.5**
TAUJTTINm input period count detection function	**18.4.9.6**
Synchronous Channel Operation Functions	**18.4.10**
PWM output function	**18.4.10.1**

18.2.1.3 Input/Output and Interrupt Request Signals

The following diagram illustrates TAUJ input/output and interrupt request signals.

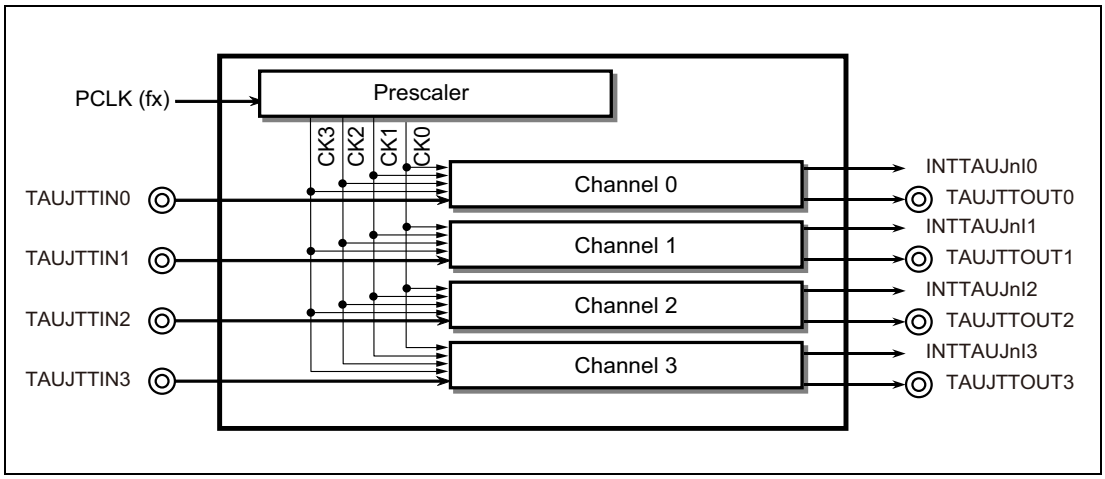


Figure 18.1 TAUJ Input/Output and Interrupt Request Signals

18.2.2 Block Diagram

The following figure shows the main components of the TAUJ:

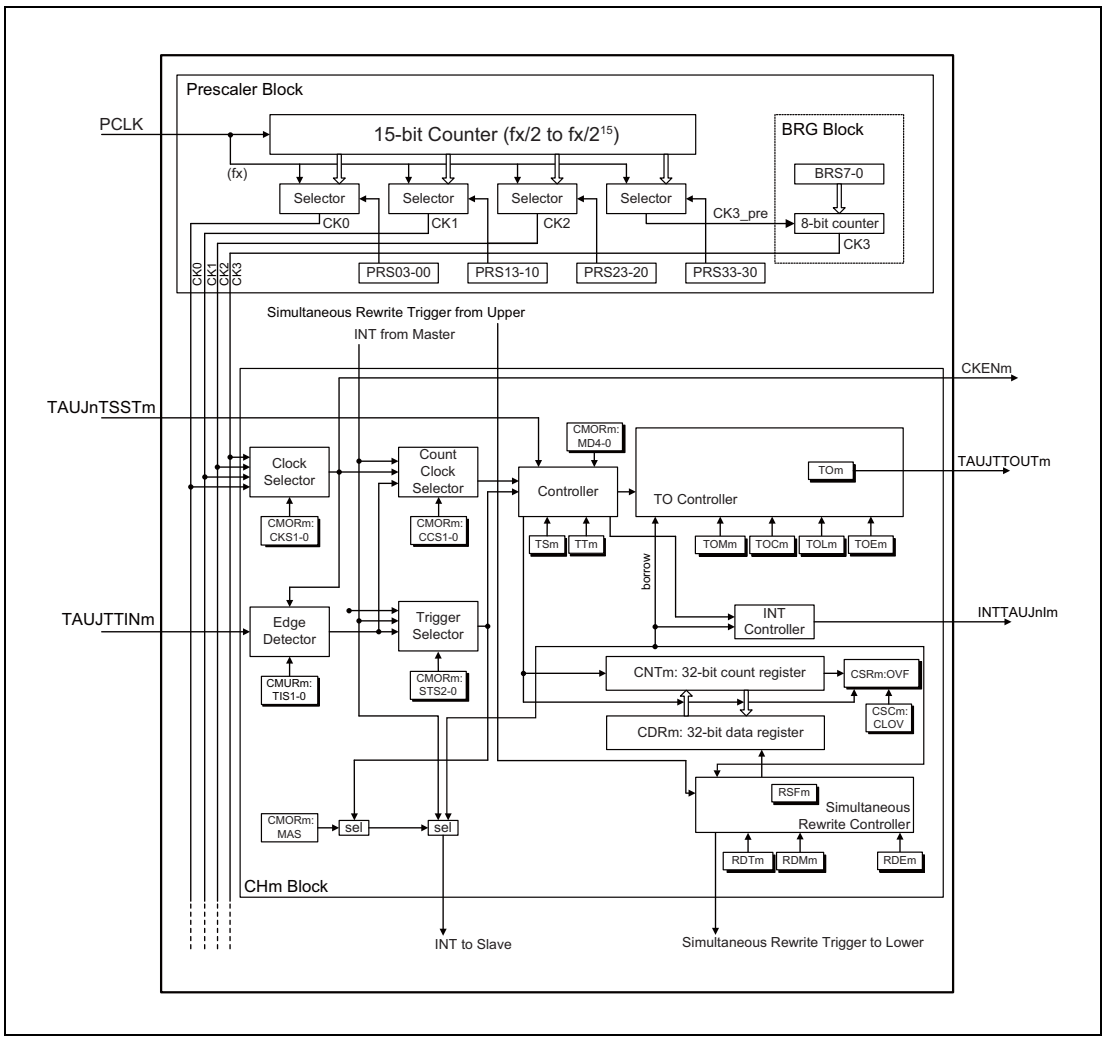


Figure 18.2 Block Diagram of the TAUJ

The prefix “TAUJn” has been omitted from the register names for the sake of clarity in the above figure.

- TAUJnTSSTm: Simultaneous start trigger (input from PIC1A)

18.2.2.1 Description of Blocks

The following describes the functional blocks:

Prescaler block

The prescaler block provides up to 4 clock signals (CK0 to CK3) that can be used as count clocks for all channels.

Count clocks CK0 to CK2 are derived from PCLK by a configurable prescaler division factor of 2^0 to 2^{15}. The fourth count clock CK3 can be adjusted more precisely by an additional division factor that is not a power of 2.

Clock and count clock selection

For every channel, the count clock selector selects which of the following is used as the clock source:

- One of the clocks CK0 to CK3 (selected by the clock selector)

Controller

The controller controls the main operations of the counter:

- Operating mode (selected with the TAUJnCMORm.TAUJnMD[4:0] bits)
- Counter start enable (TAUJnTS.TAUJnTSm) and counter stop (TAUJnTT.TAUJnTTm)

When counter start is enabled, status flag TAUJnTE.TAUJnTEm is set.

Trigger selector

Depending on the selected operation mode, the counter starts automatically when it is enabled (TAUJnTE.TAUJnTEm = 1), or it waits for an external start trigger signal. Any of the following signals can be used as the start trigger:

- Synchronous channel start trigger input TAUJnTSSTm
- Effective edge of TAUJTTINm input signal
- INTTAUJnIm from master channel

Simultaneous rewrite controller

Simultaneous rewrite control is a special function that can be used in synchronous operation functions. The data registers (TAUJnCDRm) of all channels in a channel group can be rewritten at any time. The simultaneous rewrite controller ensures that new data register values of all channels become effective at the same time.

TAUJnTO controller

The output control of every channel enables the generation of various output signals such as PWM signals.

18.3 Registers

18.3.1 List of Registers

TAUJ registers are listed in the following table.

For information on <TAUJn_base>, see **Section 18.1.2, Register Base Addresses**.

Table 18.9 TAUJn Registers Overview

Module name	Register name	Shortcut	Address
TAUJn prescaler registers			
TAUJn	TAUJn prescaler clock select register	TAUJnTPS	<TAUJn_base> + 90_H
TAUJn	TAUJn prescaler baud rate setting register	TAUJnBRS	<TAUJn_base> + 94_H
TAUJn control registers			
TAUJn	TAUJn channel data register m	TAUJnCDRm	<TAUJn_base> + m × 4_H
TAUJn	TAUJn channel counter register m	TAUJnCNTm	<TAUJn_base> + 10_H + m × 4_H
TAUJn	TAUJn channel mode OS register m	TAUJnCMORm	<TAUJn_base> + 80_H + m × 4_H
TAUJn	TAUJn channel mode user register m	TAUJnCMURm	<TAUJn_base> + 20_H + m × 4_H
TAUJn	TAUJn channel status register m	TAUJnCSRm	<TAUJn_base> + 30_H + m × 4_H
TAUJn	TAUJn channel status clear trigger register m	TAUJnCSCm	<TAUJn_base> + 40_H + m × 4_H
TAUJn	TAUJn channel start trigger register	TAUJnTS	<TAUJn_base> + 54_H
TAUJn	TAUJn channel enable status register	TAUJnTE	<TAUJn_base> + 50_H
TAUJn	TAUJn channel stop trigger register	TAUJnTT	<TAUJn_base> + 58_H
TAUJn output registers			
TAUJn	TAUJn channel output enable register	TAUJnTOE	<TAUJn_base> + 60_H
TAUJn	TAUJn channel output register	TAUJnTO	<TAUJn_base> + $5C_H$
TAUJn	TAUJn channel output mode register	TAUJnTOM	<TAUJn_base> + 98_H
TAUJn	TAUJn channel output configuration register	TAUJnTOC	<TAUJn_base> + $9C_H$
TAUJn	TAUJn channel output active level register	TAUJnTOL	<TAUJn_base> + 64_H
TAUJn reload data registers			
TAUJn	TAUJn channel reload data enable register	TAUJnRDE	<TAUJn_base> + $A0_H$
TAUJn	TAUJn channel reload data mode register	TAUJnRDM	<TAUJn_base> + $A4_H$
TAUJn	TAUJn channel reload data trigger register	TAUJnRDT	<TAUJn_base> + 68_H
TAUJn	TAUJn channel reload status register	TAUJnRSF	<TAUJn_base> + $6C_H$

18.3.2 TAUJnTPS — TAUJn Prescaler Clock Select Register

This register specifies clocks CK0, CK1, CK2, and CK3_PRE for all channels of the PCLK prescalers. CK3 is generated by dividing CK3_PRE by the factor specified in TAUJnBRS.

Access: Readable/writable in 16-bit units.

Address: <TAUJn_base> + 90_H

Value after reset: $FFFF_H$. Any reset source triggers initialization.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnPRS3[3:0]				TAUJnPRS2[3:0]				TAUJnPRS1[3:0]				TAUJnPRS0[3:0]			
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.10 TAUJnTPS Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	TAUJnPRS3 [3:0]	These bits specify a CK3_PRE clock. The CK3_PRE clock is an input clock of the BRG unit which supplies prescaler output CK3 to all channels.

TAUJnPRS3[3:0]	CK3_PRE Clock
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK3 are stopped (TAUJnTE.TAUJnTEm = 0).

Table 18.10 TAUJnTPS Register Contents (2/3)

Bit Position	Bit Name	Function
11 to 8	TAUJnPRS2 [3:0]	These bits specify a output CK2 clock.

TAUJnPRS2[3:0]	Prescaler Output CK2
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK2 are stopped (TAUJnTE.TAUJnTEm = 0).

7 to 4	TAUJnPRS1 [3:0]	These bits specify a output CK1 clock.

TAUJnPRS1[3:0]	Prescaler Output CK1
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK1 are stopped (TAUJnTE.TAUJnTEm = 0).

Table 18.10 TAUJnTPS Register Contents (3/3)

Bit Position	Bit Name	Function
3 to 0	TAUJnPRS0 [3:0]	These bits specify a CK0 clock.

TAUJnPRS0[3:0]	Prescaler Output CK0
0000_B	PCLK/2^0
0001_B	PCLK/2^1
0010_B	PCLK/2^2
0011_B	PCLK/2^3
0100_B	PCLK/2^4
0101_B	PCLK/2^5
0110_B	PCLK/2^6
0111_B	PCLK/2^7
1000_B	PCLK/2^8
1001_B	PCLK/2^9
1010_B	PCLK/2^{10}
1011_B	PCLK/2^{11}
1100_B	PCLK/2^{12}
1101_B	PCLK/2^{13}
1110_B	PCLK/2^{14}
1111_B	PCLK/2^{15}

The above bits are rewritable only when all the counters using CK0 are stopped (TAUJnTE.TAUJnTEm = 0).

NOTE

TAUJn clock input PCLK is defined in the first part of this section, **Section 18.1.3, Clock Supply**.

18.3.3 TAUJnBRS — TAUJn Prescaler Baud Rate Setting Register

This register specifies the division factor of prescaler clock CK3.

CK3 is generated by dividing CK3_PRE by the factor specified in this register plus one. The PCLK prescaler for CK3_PRE is specified in TAUJnTPS. TAUJnPRS3[3:0].

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 94_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	TAUJnBRS[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.11 TAUJnBRS Register Contents

Bit Position	Bit Name	Function
7 to 0	TAUJnBRS [7:0]	These bits specify a CK3_PRE clock division factor for generating CK3.

TAUJnBRS[7:0]	CK3 clock
$0000\ 0000_B$	CK3_PRE / 1
$0000\ 0001_B$	CK3_PRE / 2
$0000\ 0010_B$	CK3_PRE / 3
$0000\ 0011_B$	CK3_PRE / 4
...	...
$1111\ 1110_B$	CK3_PRE / 255
$1111\ 1111_B$	CK3_PRE / 256

18.3.4 TAUJnCDRm — TAUJn Channel Data Register

This register functions either as a compare register or as a capture register, depending on the operating mode specified in TAUJnCMORm.TAUJnMD[4:1].

Access: Readable/writable in 32-bit units.
- Readable in capture mode. Any write operation is ignored.
- Readable/writable in compare mode.

Address: <TAUJn_base> + 0_H + m × 4_H

Value after reset: 0000 0000_H. Any reset source triggers initialization.

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

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

Table 18.12 TAUJnCDRm Register Contents

Bit Position	Bit Name	Function
31 to 0	TAUJnCDR [31:0]	Data register for capture/compare values

18.3.5 TAUJnCNTm — TAUJn Channel Counter Register

This is a channel m counter register.

Access: Readable in 32-bit units.

Address: <TAUJn_base> + 10_H + m × 4_H

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TAUJnCNT[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 18.13 TAUJnCNTm Register Contents

Bit Position	Bit Name	Function
31 to 0	TAUJnCNT [31:0]	32-bit counter value

The read value depends on a counter, an operating mode change, or TAUJnTS.TAUJnTSm/ TAUJnTT.TAUJnTTm bit value.

The initial counter read value depends on the operating mode and how the counter is stopped.

- By a reset
- By a counter stop trigger (TAUJnTT.TAUJnTTm = 1)

The following table lists the initial counter read values after the counter is stopped (TAUJnTE.TAUJnTEm = 0) and re-enabled (TAUJnTS.TAUJnTSm = 1).

The table also contains the counter read value one count after the counter is enabled (TAUJnTS.TAUJnTSm = 1) with the counter waiting for a start trigger.

NOTE

If operating mode is changed while the counter is stopped, the initial counter value after a counter restart becomes undefined. Operating mode can be changed by the TAUJnCMORm.TAUJnMD[4:1] register.

Table 18.14 TAUJnCNTm Read Values after Re-Enabling Counter

Mode Name	Count Method (Up/Down)	TAUJnCNTm		
		Start Value*1	After Stop Trigger	After One Count
Interval timer mode	Count down	FFFF FFFF$_H$	Stop value	—
Capture mode	Count up	0000 0000$_H$	Stop value	—
One-count mode	Count down	FFFF FFFF$_H$	Stop value	Stop value
Capture and one-count mode	Count up	0000 0000$_H$	Stop value	Capture value + 1 (TAUJnCDRm)
Count capture mode	Count up	0000 0000$_H$	Stop value	—
Capture and gate count mode	Count up	0000 0000$_H$	Stop value	Stop value

Note 1. The value set for TAUJnCNTm when the operation mode is changed after reset release.

18.3.6 TAUJnCMORm — TAUJn Channel Mode OS Register

This register controls channel m operation.

Access: Readable/writable in 16-bit units. Writable only when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + 80_H + m × 4_H

Value after reset: 0000_H. Any reset source triggers initialization.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.15 TAUJnCMORm Register Contents (1/3)

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select an operation clock, which is used with the TAUJTTINm input edge detection circuit. Setting of TAUJnCMORm.TAUJnCCS[1:0] bit also allows the operation clock to serve as a count clock of TAUJnCNTm.

TAUJn CKS1	TAUJn CKS0	Selection of Operation Clock
0	0	CK0
0	1	CK1
1	0	CK2
1	1	CK3

Bit Position	Bit Name	Function
13, 12	TAUJnCCS[1:0]	These bits select a count clock for TAUJnCNTm counter.

TAUJn CKS1	TAUJn CKS0	Selection of Count Clock
0	0	Operation clock specified by TAUJnCMORm.TAUJnCKS[1:0].
0	1	Setting prohibited
1	0	
1	1	

Bit Position	Bit Name	Function
11	TAUJnMAS	This bit specifies whether the channel is a master or slave channel during synchronous channel operation. 0: Slave 1: Master This bit setting is valid only for even-numbered channels (CHm_even). Odd-numbered channels (CHm-odd) are fixed to 0.

Table 18.15 TAUJnCMORm Register Contents (2/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>10 to 8</td><td>TAUJnSTS[2:0]</td><td>These bits select a start trigger.
<table>
<tr><th>TAUJn STS2</th><th>TAUJn CKS1</th><th>TAUJn CKS0</th><th>Functional Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Software trigger</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>Effective edge of TAUJTTINm input signal, which is specified by TAUJnCMURm.TAUJnTIS[1:0].</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>Effective edge of TAUJTTINm input signal is used as a start trigger and the reverse edge as a stop trigger.</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>Simultaneous rewrite trigger</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>INT of master channel</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td rowspan="3">Setting prohibited</td></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table></td></tr>
<tr><td>7, 6</td><td>TAUJnCOS[1:0]</td><td>These bits specify the timing for updating capture register TAUJnCDRm and overflow flag TAUJnCSRm.
TAUJnOVF of channel m. These bits are only valid if channel m is for capture function (capture mode and capture & one-count mode).
<table>
<tr><th>TAUJn COS1</th><th>TAUJn COS0</th><th>TAUJnCDRm</th><th>TAUJnCSRm.TAUJnOVF</th></tr>
<tr><td>0</td><td>0</td><td rowspan="2">Updated when effective edge of TAUJTTINm input is detected.</td><td>Updated (cleared or set) when effective edge of TAUJTTINm input is detected:
• Set TAUJnCSRm. TAUJnOVF if a counter overflow has occurred since the last effective edge was detected.
• Clear TAUJnCSRm. TAUJnOVF if no counter overflow has occurred since the last effective edge was detected.</td></tr>
<tr><td>0</td><td>1</td><td>Set when a counter overflow occurs and cleared by setting TAUJnCSCm.TAUJnCLOV to 1.</td></tr>
<tr><td>1</td><td>0</td><td rowspan="2">Updated when effective edge of TAUJTTINm input is detected and when a counter overflow occurs.
• Detection of effective edge of TAUJTTINm input: The counter value is written into TAUJnCDRm.
• Occurrence of overflow: FFFF FFFF$_H$ is loaded into TAUJnCDRm. Detection of the next effective edge of TAUJTTINm is ignored.</td><td>No setting</td></tr>
<tr><td>1</td><td>1</td><td>Set when a counter overflow occurs and cleared by setting TAUJnCSCm.TAUJnCLOV to 1.</td></tr>
</table></td></tr>
<tr><td>5</td><td>Reserved</td><td>When writing, write the value after reset.</td></tr>
</table>

Table 18.15 TAUJnCMORm Register Contents (3/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4 to 0</td><td>TAUJnMD[4:0]</td><td>These bits specify an operating mode.
<table>
<tr><th>TAUJn MD4</th><th>TAUJn MD3</th><th>TAUJn MD2</th><th>TAUJn MD1</th><th>TAUJn MD0</th><th>Functional Description</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1/0</td><td>Interval timer mode</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>1/0</td><td>Capture mode</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1/0</td><td>One-count mode</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>Capture and one-count mode</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>1</td><td>1/0</td><td>Count capture mode</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>Capture and gate count mode</td></tr>
</table>
Settings other than the above are prohibited.
<table>
<tr><th>Mode</th><th>Role of TAUJnMD0 Bit</th></tr>
<tr><td>Interval timer mode
Capture mode
Count capture mode</td><td>Specifies whether INTTAUJnIm is generated at the beginning of count operation (when a start trigger is entered) or not.
0: INTTAUJnIm is not generated.
1: INTTAUJnIm is generated.</td></tr>
<tr><td>One-count mode</td><td>Enables or disables start trigger detection during counting.
0: Disables detection.
1: Enables detection.
CAUTION
In one-count mode, INTTAUJnIm signal is not output at the beginning of count operation.</td></tr>
<tr><td>Capture and one-count mode
Capture and gate count mode</td><td>This bit should be set to 0.
CAUTION
INTTAUJnIm signal is not output at the beginning of count operation. In addition, start trigger detected during counting is disabled.</td></tr>
</table>
</td></tr>
</table>

18.3.7 TAUJnCMURm — TAUJn Channel Mode User Register

This register specifies a type of effective edge detection used for TAUJTTINm input.

Access: Readable/writable in 8-bit units.

Address: <TAUJn_base> + 20_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.16 TAUJnCMURm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	These bits specify an effective edge of TAUJTTINm input signal.

TAUJn TIS1	TAUJn TIS0	Functional Description
0	0	Falling edge
0	1	Rising edge
1	0	Detection of falling and rising edges (selection of low width measurement) Start trigger: Falling edge Stop trigger (capture): Rising edge
1	1	Detection of falling and rising edges (selection of high width measurement) Start trigger: Rising edge Stop trigger (capture): Falling edge

Edge detection of TAUJTTINm input signal is based on the operation clock selected by TAUJnCMORm.TAUJnCKS[1:0].

18.3.8 TAUJnCSRm — TAUJn Channel Status Register

This register indicates the overflow status of channel m.

Access: Readable in 8-bit units.

Address: <TAUJn_base> + 30_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUJnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 18.17 TAUJnCSRm Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	Reserved	When read, the value returned is undefined.
0	TAUJnOVF	This bit indicates the counter overflow status: 0: No overflow occurs 1: Overflow occurs This bit is used only in the following modes: • Capture mode • Capture and one-count mode The function of this bit depends on the setting of control bits TAUJnCMORm.TAUJnCOS[1:0].

18.3.9 TAUJnCSCm — TAUJn Channel Status Clear Register

This register is a trigger register for clearing the overflow flag TAUJnCSRm.TAUJnOVF of channel m.

Access: Writable in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 40_H + m × 4_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAUJnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 18.18 TAUJnCSCm Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TAUJnCLOV	0: No function 1: Clears the overflow flag TAUJnCSRm.TAUJnOVF

18.3.10 TAUJnTS — TAUJn Channel Start Trigger Register

This register enables the counter operation for each channel.

Access: Writable in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 54_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTS03	TAUJnTS02	TAUJnTS01	TAUJnTS00
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 18.19 TAUJnTS Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnTSm	These bits enable the counter operation for channel m: 0: No function 1: Enables the counter operation and sets TAUJnTE.TAUJnTEm = 1 Only the counter operation is enabled even if TAUJnTE.TAUJnTEm = 1 Whether the counter is started or not depends on the selected of operating mode.

18.3.11 TAUJnTE — TAUJn Channel Enable Status Register

This register enables/disables a counter operation.

Access: Readable in 8-bit units.

Address: <TAUJn_base> + 50_H

Value after reset: 00_H. Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTE03	TAUJnTE02	TAUJnTE01	TAUJnTE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 18.20 TAUJnTE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3 to 0	TAUJnTEm	These bits enable or disable channel m's counter operation. 0: Disables the counter operation 1: Enables the counter operation These bits are set to 1 when trigger input of TAUJnTSSTm (synchronous channel start trigger signal) is detected or when TAUJnTS.TAUJnTSm are set to 1. These bits are set to 0 when TAUJnTT.TAUJnTTm are set to 1.

18.3.12 TAUJnTT — TAUJn Channel Stop Trigger Register

This register stops the counter operation of each channel.

Access: Writable in 8-bit units. The read value is always 00_H.

Address: <TAUJn_base> + 58_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTT03	TAUJnTT02	TAUJnTT01	TAUJnTT00
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 18.21 TAUJnTT Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnTTm	These bits stop channel m's counter operation. 0: No function 1: Stops the counter operation and resets TAUJnTE.TAUJnTEm TAUJnCNTm, TAUJnTO.TAUJnTOm, and TAUJTTOUTm retain the values provided before the counter is stopped.

18.3.13 TAUJnTOE — TAUJn Channel Output Enable Register

This register enables and disables Independent Channel Output Mode Controlled by Software.

Access: his register can be read/written in 8-bit units.

Address: <TAUJn_base> + 60_H

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOE03	TAUJnTOE02	TAUJnTOE01	TAUJnTOE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.22 TAUJnTOE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOEm	These bits enable or disable Independent Channel Output Mode: 0: Disables Independent Timer Output Function (controlled by software) 1: Enables Independent Timer Output Function

18.3.14 TAUJnTO — TAUJn Channel Output Register

This register specifies and reads the level of TAUJTTOUTm.

Access: This register can be read/written in 8-bit units.

Address: <TAUJn_base> + $5C_H$

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTO03	TAUJnTO02	TAUJnTO01	TAUJnTO00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.23 TAUJnTO Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOm	These bits specify and read the level of TAUJTTOUTm: 0: Low 1: High Only TAUJnTOm bits for which Independent Channel Output function is disabled (TAUJnTOEm = 0) can be written.

18.3.15 TAUJnTOM — TAUJn Channel Output Mode Register

This register specifies the output mode of each channel.

Access: This register can be read/written in 8-bit units. It can only be written when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + 98_H

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOM03	TAUJnTOM02	TAUJnTOM01	TAUJnTOM00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.24 TAUJnTOM Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOMm	These bits specify the channel output mode: 0: Independent channel output mode 1: Synchronous channel output mode The output mode depends on the setting of the output control bits (TAUJnTOE.TAUJnTOEm) of each channel.

18.3.16 TAUJnTOC — TAUJn Channel Output Configuration Register

This register specifies the output mode of each channel in combination with TAUJnTOMm.

Access: This register can be read/written in 8-bit units. It can only be written when the counter is stopped (TAUJnTE.TAUJnTEm = 0).

Address: <TAUJn_base> + $9C_H$

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOC03	TAUJnTOC02	TAUJnTOC01	TAUJnTOC00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.25 TAUJnTOC Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOCm	These bits specify the output mode: 0: Operation mode 1 (= Toggle mode) 1: No function These bits must be set to 0 for all output modes except Independent Channel Output Mode Controlled by Software. The output mode also depends on TAUJnTOM.TAUJnTOMm, as can be seen in the following table.

TAUJn TOMm	TAUJn TOCm	Functional Description
0	0	Toggle mode: Toggle operation is conducted when INTTAUJnIm occurs.
0	1	No function
1	0	Synchronous Channel Operation Mode 1: Set when INT occurs on the master channel and reset when INT occurs on the slave channel.
1	1	No function

18.3.17 TAUJnTOL — TAUJn Channel Output Level Register

This register specifies the output logic of the channel output bit (TAUJnTO.TAUJnTOm).

Access: This register can be read/written in 8-bit units.

Address: <TAUJn_base> + 64_H

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnTOL03	TAUJnTOL02	TAUJnTOL01	TAUJnTOL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.26 TAUJnTOL Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnTOLm	These bits specify the output logic of the channel m output bit (TAUJnTO.TAUJnTOm): 0: Positive logic (active high) 1: Negative logic (active low) These bits apply in all channel output modes except Independent Channel Output Mode Controlled by Software and independent channel output mode 1.

18.3.18 TAUJnRDE — TAUJn Channel Reload Data Enable Register

This register enables and disables simultaneous rewrite of the data register TAUJnCDRm. It also enables and disables simultaneous rewrite of the data register TAUJnTOLm for the PWM output function.

Access: This register can be read/written in 8-bit units. It can only be written when TAUJnTE.TAUJnTEm = 0.

Address: <TAUJn_base> + $A0_H$

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDE03	TAUJnRDE02	TAUJnRDE01	TAUJnRDE00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.27 TAUJnRDE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnRDEm	These bits enable or disable simultaneous rewrite of the data register of channel m: 0: Disables simultaneous rewrite 1: Enabled simultaneous rewrite

18.3.19 TAUJnRDM — TAUJn Channel Reload Data Mode Register

This register selects when the signal that controls simultaneous rewrite is generated.

Access: This register can be read/written in 8-bit units. It can only be written when TAUJnTE.TAUJnTEm=0.

Address: <TAUJn_base> + $A4_H$

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDM03	TAUJnRDM02	TAUJnRDM01	TAUJnRDM00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.28 TAUJnRDM Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3 to 0	TAUJnRDMm	These bits specify when the signal that triggers simultaneous rewrite is generated: 0: When the master channel counter starts counting 1: No function These bits only apply when TAUJnRDE.TAUJnRDEm = 1.

18.3.20 TAUJnRDT — TAUJn Channel Reload Data Trigger Register

This register triggers the simultaneous rewrite enabling state.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <TAUJn_base> + 68_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRDT03	TAUJnRDT02	TAUJnRDT01	TAUJnRDT00
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 18.29 TAUJnRDT Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3 to 0	TAUJnRDTm	These bits trigger the simultaneous rewrite enabling state: 0: No function 1: Simultaneous rewrite enabling state is triggered. The simultaneous rewrite enabling flag (TAUJnRSFm) is set to 1. The system waits for the simultaneous rewrite trigger. The setting of these bits is applied only to the following case: • TAUJnRDE.TAUJnRDEm = 1

18.3.21 TAUJnRSF — TAUJn Channel Reload Status Register

This flag register indicates the simultaneous rewrite status.

Access: This register can be read in 8-bit units.

Address: <TAUJn_base> + $6C_H$

Value after reset: 00_H Any reset source triggers initialization.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAUJnRSF03	TAUJnRSF02	TAUJnRSF01	TAUJnRSF00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 18.30 TAUJnRSF Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3 to 0	TAUJnRSFm	These bits indicate the simultaneous rewrite status: 0: Indicates that simultaneous rewrite has been completed due to the generation of simultaneous rewrite trigger. 1: Indicates that the system waits for a simultaneous rewrite trigger in the simultaneous rewrite enabling state (TAUJnRDTm = 1).

18.4 Function

18.4.1 General Operating Procedure

The following lists the general operation procedure for the TAUJn:

After reset release, the operation of each channel is stopped. Clock supply is started and writing to each register is enabled. All circuits and registers of all channels are initialized. The control register of TAUJTTOUTm is also initialized and outputs a low level.

1. Set the TAUJnTPS and TAUJnBRS registers to specify the clock frequency of CK0 to CK3.
2. Configure the desired TAUJn function:
   - Set the operation mode
   - Set any other control bits
3. Enable the counter by setting the TAUJnTS.TAUJnTSm bit to 1.
   The counter starts to count immediately, or when an appropriate trigger is detected, depending on the bit settings.
4. Stop the counter or perform a forced restart operation during count operation. The counter can be stopped by setting the TAUJnTT.TAUJnTTm bit to 1. The counter can be forcibly restarted by setting the TAUJnTS.TAUJnTSm bit to 1.
5. Stop the function by setting the TAUJnTT.TAUJnTTm bit to 1.

NOTE

A detailed description of the required control bits and the operation of the individual functions is given in **Section 18.4.10, Synchronous Channel Operation Functions.**

18.4.2 Concepts of Synchronous Channel Operation Function

The synchronous channel operation function is implemented using a combination of channel groups (comprised of master and slave channels).

Several rules apply to the settings of channels.

These rules are detailed in **Section 18.4.2.1, Rules of Synchronous Channel Operation Function**.

Two special features for the synchronous channel operation function are detailed in the following section:

- **Section 18.4.10, Synchronous Channel Operation Functions**

18.4.2.1 Rules of Synchronous Channel Operation Function

Number of master and slave channels

- Only even-numbered channels (CH0, CH2) can be set as master channels. Any channel apart from CH0 can be set as a slave channel.
- Only channels lower than the master channel can be set as slave channels, and several slave channels can be set for one master channel.
  Example: If CH2 is a master channel, CH3 can be set as slave channel.
- If two master channels are used, slave channels cannot cross the master.
  Example: If CH0 and CH2 are master channels, CH1 can be set as slave channels for CH0, but CH3 cannot.

Count clock

- The same count clock should be set for the master channel and the slave channels synchronizing to the master channel. This is achieved by setting the same value in the TAUJnCMORm.TAUJnCKS[1:0] bits of the master and slave channels.

Control trigger signal for master/salve channels

- Master channels can output control trigger signals to slave channels.
- Slave channels can use control trigger signals from master channels but cannot output control trigger signals for their own to lower channels.
- Master channels cannot use control trigger signals from upper master channels.

The basic concepts of master/slave usage and count clocks are illustrated in **Figure 18.3**.

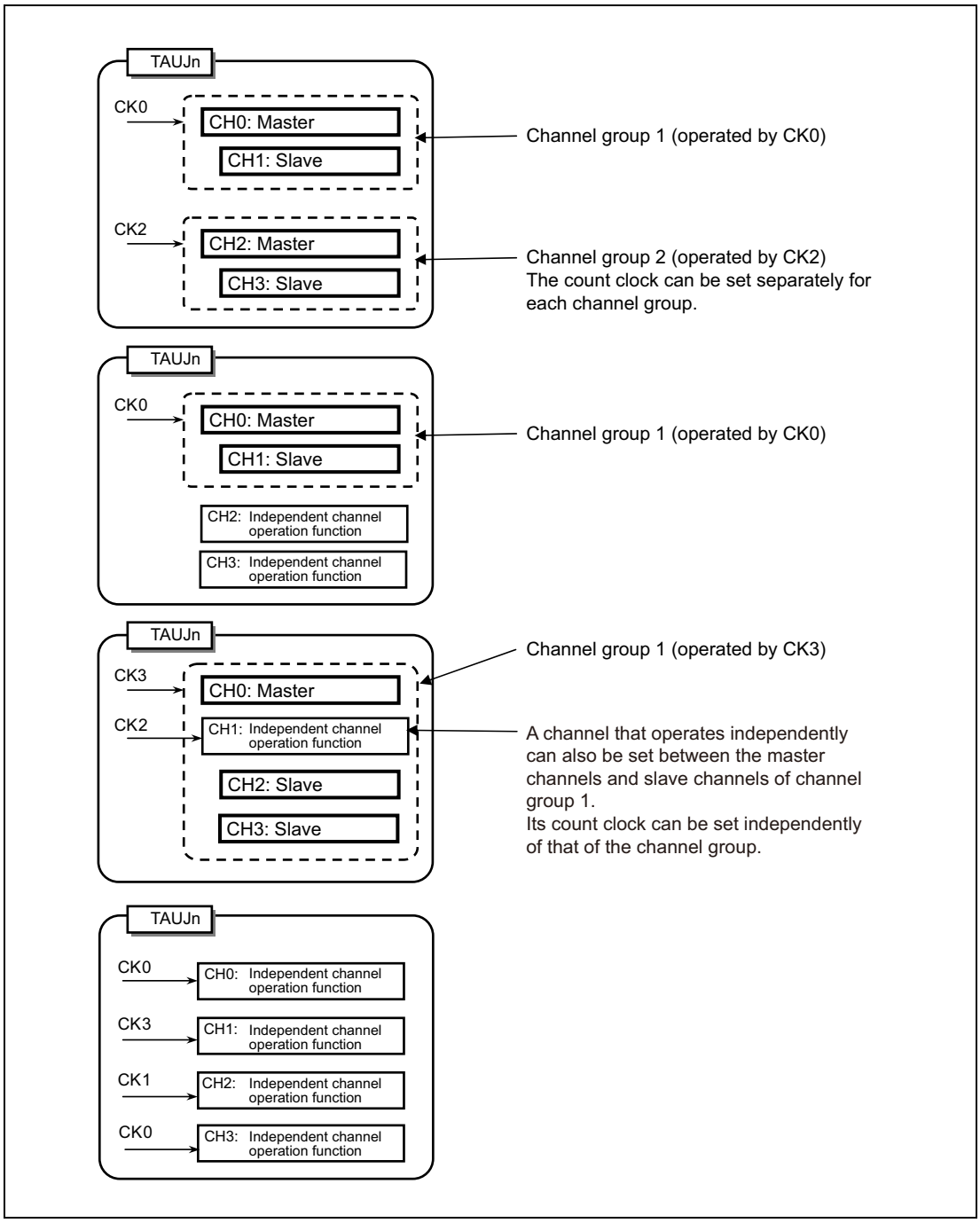


Figure 18.3 Grouping of Channels and Assignment of Count Clocks

18.4.2.2 Simultaneous start and stop of synchronous channel counters

Channels that are operated synchronously can be started and stopped simultaneously, both within a TAUJ unit and between TAUJ units.

(1) Simultaneous start and stop within a TAUJ unit

- To simultaneously start synchronized channels, the TAUJnTS.TAUJnTSm bits of the channels should be set at the same time.
- To simultaneously stop synchronized channels, the TAUJnTT.TAUJnTTm bits of the channels should be set at the same time.

Setting 1 in the TAUJnTS.TAUJnTSm bits sets the corresponding TAUJnTE.TAUJnTEm bits to 1, enabling counting. The count start timing of the counter depends on the operating mode.

(2) Simultaneous start between TAUJ units

Counters in different TAUJ units can also be started simultaneously if the corresponding counters are enabled before receiving the simultaneous trigger signal.

18.4.3 Simultaneous Rewrite

18.4.3.1 How to control simultaneous rewrite

The following figure shows the general procedure for simultaneous rewrite. The three main blocks (Initial settings, Start counter & count operation, and Simultaneous rewrite) are explained afterwards.

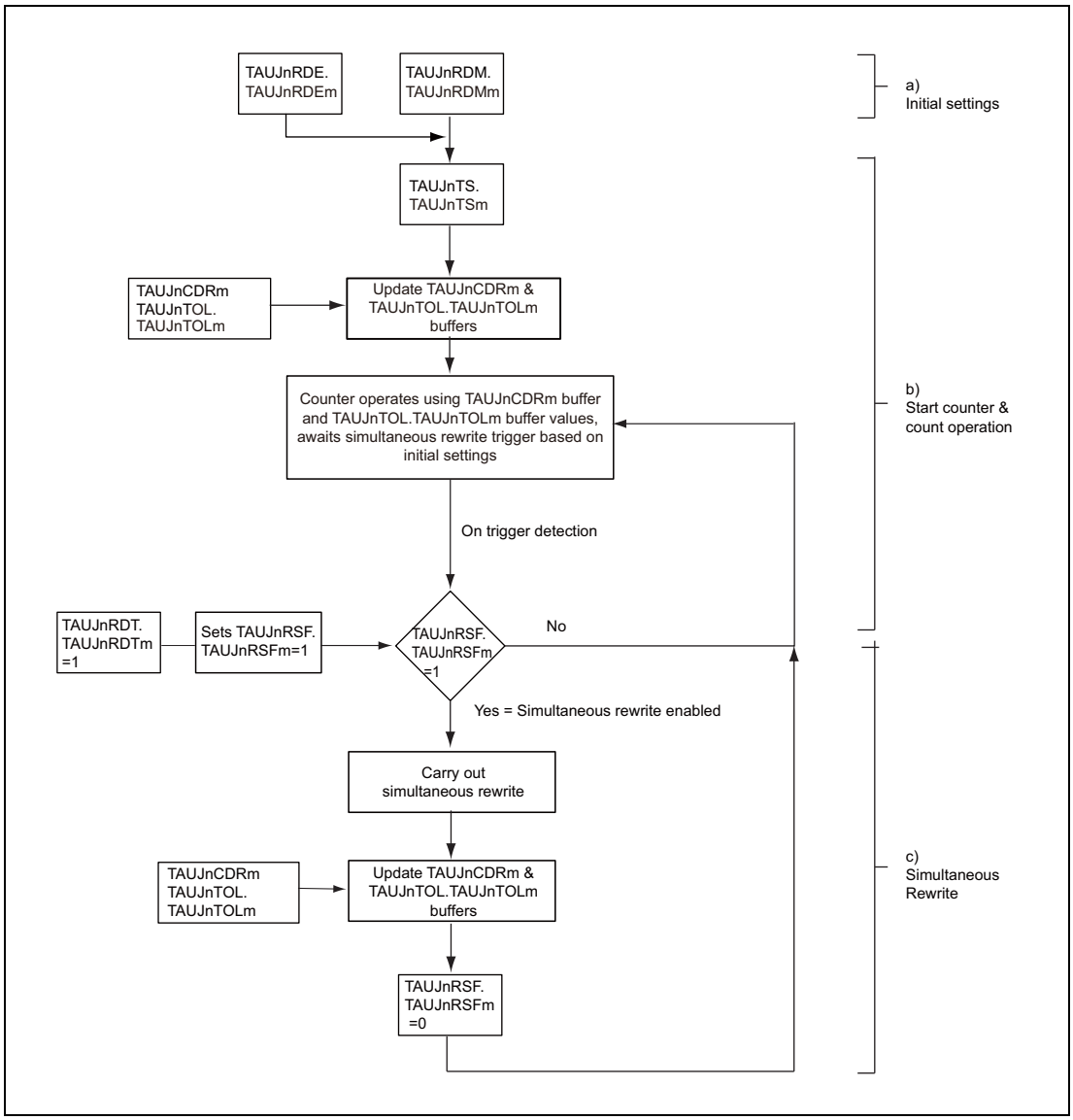


Figure 18.4 General procedure for simultaneous rewrite

(1) Initial settings

- To enable simultaneous rewrite in channel m, set TAUJnRDE.TAUJnRDEm = 1
- To select simultaneous rewrite when the master channel starts counting, set TAUJnRDM.TAUJnRDMm

(2) Start counter and count operation

- To start all the TAUJnCNTm counters in the channel group, set the corresponding TAUJnTS.TAUJnTSm bits to 1. TAUJnTOL.TAUJnTOLm and the values in the data registers (TAUJnCDRm) are loaded into the corresponding TAUJnTOL.TAUJnTOLm buffer (TAUJnTOL.TAUJnTOLm buf) and data buffer registers (TAUJnCDRm buf) and the counters start.
- Setting the reload data trigger bit (TAUJnRDT.TAUJnRDTm) to 1 sets the reload flag (TAUJnRSF.TAUJnRSFm) to 1, enabling simultaneous rewrite. TAUJnRSF.TAUJnRSFm remains set to 1 until simultaneous rewrite is completed.
- When a specified trigger for simultaneous rewrite is detected, the TAUJnRSF.TAUJnRSFm bit is checked to see if simultaneous rewrite is enabled (TAUJnRSF.TAUJnRSFm = 1). If enabled, simultaneous rewrite is carried out. Otherwise simultaneous rewrite is not carried out and the system waits for detection of the next simultaneous rewrite trigger.

(3) Simultaneous rewrite

- When the simultaneous rewrite trigger is detected and simultaneous rewrite is enabled (TAUJnRSF.TAUJnRSFm = 1), the current values of the data registers are copied to their buffers. These values are then loaded into the corresponding counters and the values are applied the next time the counter starts or restarts.
- When the simultaneous rewrite is completed, the TAUJnRSF.TAUJnRSFm bit is set to 0, and the system awaits the next simultaneous rewrite trigger.

18.4.3.2 Other general rules for simultaneous rewrite

The following rules also apply.

- TAUJnRDE.TAUJnRDEm and TAUJnRDM.TAUJnRDMm cannot be changed while the counter is in operation (TAUJnTE.TAUJnTEm = 1).
- TAUJnTOL.TAUJnTOLm can be rewritten only during operation using the PWM output function. For all other functions, TAUJnTOL.TAUJnTOLm should be written before the counter starts. If it is rewritten while any other function is used, TAUJTTOUTm outputs an invalid wave.

18.4.3.3 Simultaneous rewrite procedure

The simultaneous rewrite procedure in the PWM output function is described in the following figure.

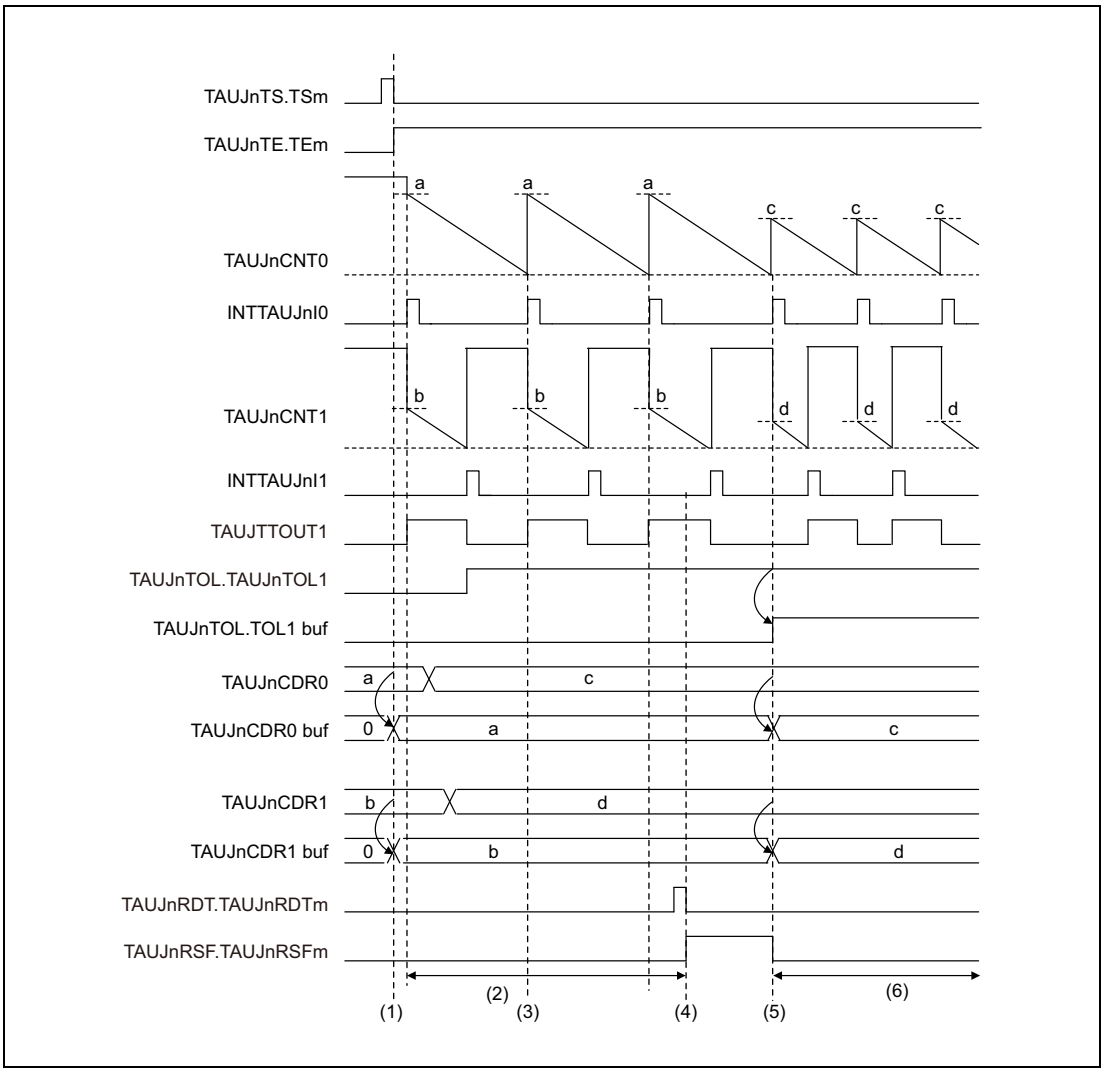


Figure 18.5 Simultaneous Rewrite when the Master Channel Starts/Restarts to Count

Setting

CH0 is a master channel of the PWM output function, and CH1 represents an slave channel of the PWM output function. Simultaneous rewrite is applied when the master channel starts counting.

Description:

(1) When TAUJnTS.TAUJnTSm = 1 is set, the value of TAUJnCDRm is copied to the TAUJnCDRm buffer and the value of TAUJnTOL.TAUJnTOLm is copied to the TAUJnTOL.TAUJnTOLm buffer.

(2) The TAUJnCDRm and TAUJnTOL.TAUJnTOLm registers can be written at any time.

(3) CH0 restarts counting, but simultaneous rewrite does not occur because it is disabled (TAUJnRSF.TAUJnRSFm = 0).

(4) The reload data trigger bit (TAUJnRDT.TAUJnRDTm) is set to 1 which sets the status flag (TAUJnRSF.TAUJnRSFm = 1), enabling simultaneous rewrite.

(5) Simultaneous rewrite is triggered when CH0 restarts counting, because simultaneous rewrite is enabled. The TAUJnCDRm value is loaded into the TAUJnCDRm buffer and the TAUJnTOL.TAUJnTOLm value is loaded into the TAUJnTOL.TAUJnTOLm buffer.

(6) The counters count down and await the next simultaneous rewrite trigger. The values of TAUJnCDRm and TAUJnTOL.TAUJnTOLm can be changed again.

18.4.4 Channel Output Modes

The output of the TAUJTTOUTm pin can be controlled in two ways, the latter of which can be further split into individual modes.

- By software (TAUJnTOE.TAUJnTOEm = 0)

  When controlled by software, the value written in the output register bit (TAUJnTO.TAUJnTOm) is sent out of the output pin (TAUJTTOUTm).

- By TAUJ signals (TAUJnTOE.TAUJnTOEm = 1)

  When controlled by TAUJ signals, the output level of TAUJTTOUTm is set or reset or toggled by internal signals. The value of TAUJnTO.TAUJnTOm is updated accordingly to reflect the value of TAUJTTOUTm.

  – Independently (TAUJnTOM.TAUJnTOMm = 0)

    In case of independent operation, the output of the TAUJTTOUTm pin is only affected by settings of channel m. Therefore, independent channel operation should be selected (TAUJnTOM.TAUJnTOMm = 0).

  – Synchronously (TAUJnTOM.TAUJnTOMm = 1)

    In case of synchronous operation, the output of the TAUJTTOUTm pin is affected by settings of channel m and those of other channels. Therefore, synchronous channel operation should be selected for all synchronized channels (TAUJnTOM.TAUJnTOMm = 1).

The TAUJnTO.TAUJnTOm bit can always be read to determine the current value of TAUJTTOUTm, regardless of whether the pin is controlled by software, operated independently, or operated synchronously.

Control bits

The settings of the control bits required to select a specific channel output mode are listed in **Table 18.31, Channel Output Modes**.

The channel output modes are described in details below.

- **Section 18.4.4.2, Channel Output Modes Controlled Independently by TAUJn Signals**
- **Section 18.4.4.3, Channel Output Modes Controlled Synchronously by TAUJn Signals**

Batch operation of TAUJnTOm bit

Whether a set value is reflected to the TAUJnTOm bit or not is controlled by the TAUJnTOE.TAUJnTOEm bit.

The TAUJnTOm setting is written only to the bit (channel) set with TAUJnTOE.TAUJnTOEm bit = 0 when a write to the TAUJnTO register is attempted. No TAUJnTOm setting is reflected to the bit (channel) set with TAUJnTOE.TAUJnTOEm bit = 1.

NOTE

TAUJnTO.TAUJnTOm bit is placed so that its bit number corresponds to a channel number.

Output logic

Positive or negative logic of the output is specified by control bit TAUJnTOL.TAUJnTOLm.

The value of TAUJnTOL.TAUJnTOLm bit should be set before the counter is started. It can only be changed during operation with PWM output function. Otherwise, changes to TAUJnTOL.TAUJnTOLm result in an invalid TAUJTTOUTm signal output.

See **Section 18.4.3, Simultaneous Rewrite**.

Table 18.31 lists the various channel output modes and the channel output control bits.

Table 18.31 Channel Output Modes

Channel Output Mode	TAUJnTOE. TAUJnTOEm	TAUJnTOM. TAUJnTOMm
By software		
Independent channel output mode controlled by software	0	x
By TAUJ signals, independently		
Independent channel output mode 1	1	0
By TAUJ signals, synchronously		
Synchronous channel output mode 1	1	1

CAUTIONS

1. **All combinations not listed in this table are forbidden.**
2. **Bits marked with an x can be set to any value.**
3. **The following bits cannot be changed during counter operation (TAUJnTE.TAUJnTEm = 1):**
   - **TAUJnTOM.TAUJnTOMm**
   - **TAUJnTOC.TAUJnTOCm**

18.4.4.1 General Procedures for Specifying a Channel Output Mode

This section describes the general procedures for specifying a TAUJTTOUTm channel output mode. The prerequisite is that timer output operation is disabled (TAUJnTOE.TAUJnTOEm = 0).

1. Set TAUJnTO.TAUJnTOm to specify the initial level of the TAUJTTOUTm output.
2. Set channel output mode according to **Table 18.31, Channel Output Modes**, and the output logic using the TAUJnTOL.TAUJnTOLm bit.
3. Start the counter (TAUJnTS.TAUJnTSm = 1).

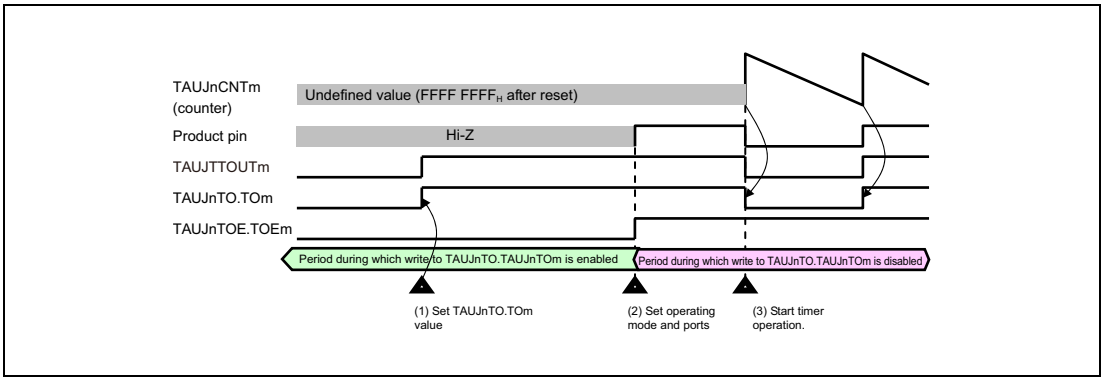


Figure 18.6 General Procedure for Specifying a TAUJTTOUTm Channel Output Mode

18.4.4.2 Channel Output Modes Controlled Independently by TAUJn Signals

This section lists the channel output modes that are controlled independently by TAUJn signals. The control bits used to specify a mode are listed in **Table 18.31, Channel Output Modes**.

(1) Independent Channel Output Mode 1

Set/reset conditions

In this output mode, TAUJTTOUTm toggles when INTTAUJnIm is detected. The value of TAUJnTOL.TAUJnTOLm is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 18.31, Channel Output Modes**.

18.4.4.3 Channel Output Modes Controlled Synchronously by TAUJn Signals

This section lists the channel output modes that are controlled synchronously by TAUJn signals. The control bits used to specify a mode are listed in **Table 18.31, Channel Output Modes**.

(1) Synchronous Channel Output Mode 1

Set/reset conditions

In this output mode, INTTAUJnIm of master channel serves as a set signal and INTTAUJnIm of the slave channel as a reset signal. If INTTAUJnIm of master channel and INTTAUJnIm of the slave channel are generated at the same time, INTTAUJnIm of the slave channel (reset signal) has priority over INTTAUJnIm (set signal) of master channel, i.e., the master channel is ignored.

Prerequisites

There are no prerequisites other than those shown in **Table 18.31, Channel Output Modes**.

18.4.5 Start Timing in Each Operating Mode

This section describes the timing at which the counter starts after TAUJnTS.TAUJnTSm is set to 1 in each operating mode.

In all modes, the value of data register (TAUJnCDRm) and whether or not an interrupt occurs depends on mode and register settings.

CAUTION

The count start timing described in this section is for your reference. Actually, the count start timing depends on the count clock timing.

18.4.5.1 Interval Timer Mode, Capture Mode, and Count Capture Mode

The counter starts operating with the next count clock after TAUJnTS.TAUJnTSm is set to 1. The value of data register is also loaded when the counter starts.

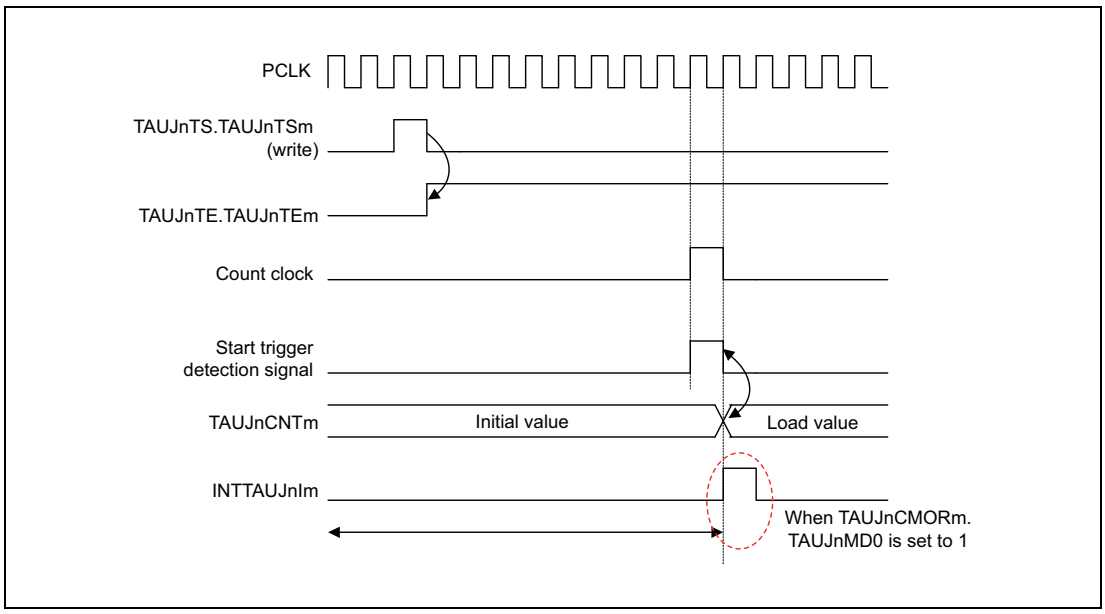


Figure 18.7 Start Timing in Interval Timer Mode, Capture Mode, and Count Capture Mode

18.4.5.2 Other Operating Modes

In other operating modes, the counter operation start timing is triggered only upon detection of an effective edge of TAUJTTINm. Once the counter starts, the value of data register is also loaded. The count clock cycles, which is irrelevant to start of counter operation, determine the frequency with which all operations take place.

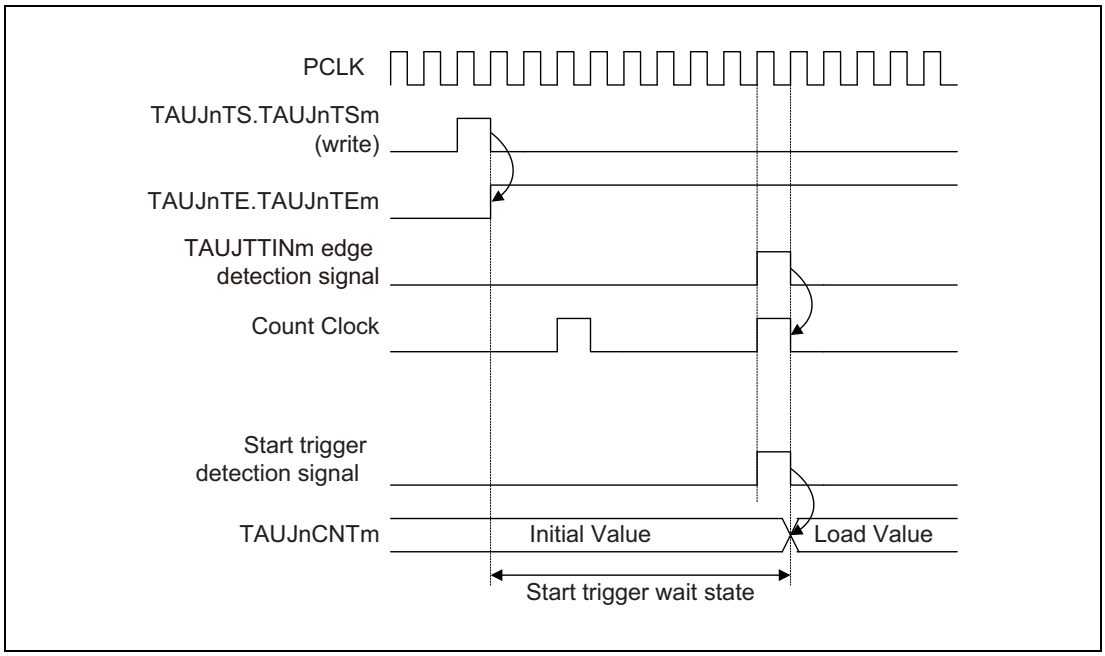


Figure 18.8 Start Timing in Other Operating Modes

18.4.6 TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts

When the counter starts, it is possible to specify whether an INTTAUJnIm is generated using the TAUJnCMORm.TAUJnMD0 bit. The effect of the bit depends on the selected mode, as shown in the table below. The effect of INTTAUJnIm on TAUJTTOUTm depends on the selected channel operation function.

Table 18.32 Effect of TAUJnCMORm.TAUJnMD0 Bit on Generation of INTTAUJnIm when Counter is Triggered

Mode	TAUJnCMORm. TAUJnMD0 Bit	INTTAUJnIm Generated when Counter Starts or Restarts or when TAUJTTINm Input Signal Trigger is Detected
Interval timer mode Capture mode Count capture mode	0	No
	1	Yes
Capture and one-count mode Capture and gate count mode	0	No
One-count mode	0/1	No, regardless of setting of TAUJnCMORm.TAUJnMD0 bit.

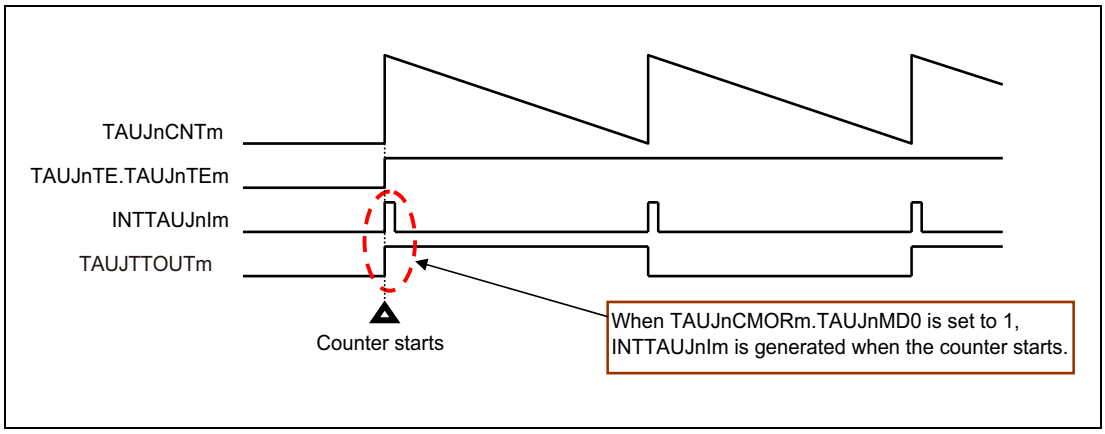


Figure 18.9 INTTAUJnIm Generated when Counter Starts

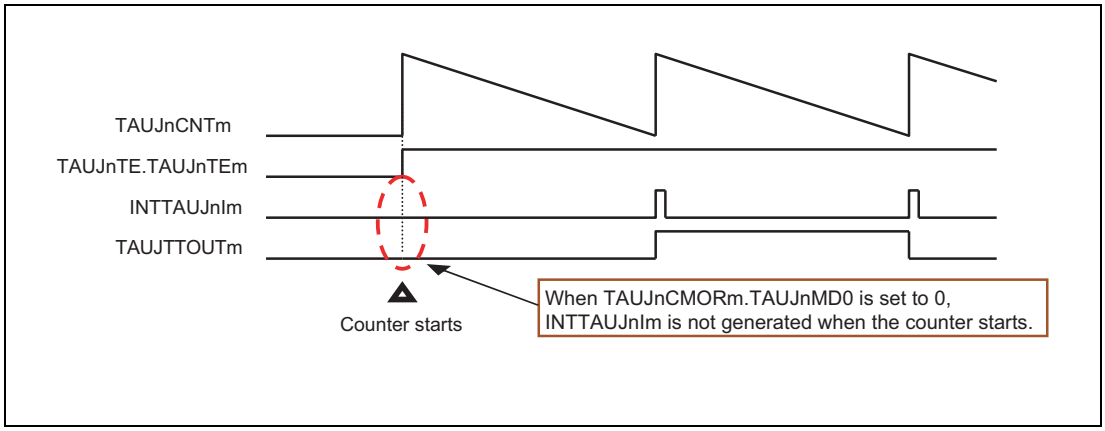


Figure 18.10 INTTAUJnIm Not Generated when Counter Starts

18.4.7 Interrupt Generation upon Overflow

Certain independent functions that count up, overflow without generating an interrupt when they reach FFFF $FFFF_H$. This section describes how it is possible to generate an interrupt, by combining a channel operating in one of these modes with a channel in a different operation mode which counts down.

The appropriate operation mode for the second channel depends on the operation mode of the first channel. Nevertheless, the principle is the same for all combinations:

- Find an operation mode for the second channel that counts down in such a manner, that it reaches 0000 0000_H at the same time as the first channel overflows (TAUJnCNTm = FFFF $FFFF_H$).
- Set TAUJnCDRm of the second channel to FFFF $FFFF_H$.
- The two channels must count at the same speed (i.e. they must have the same count clock).
- Both channels are triggered by the same TAUJTTINm input.
- The trigger detection settings (TAUJnCMORm.TAUJnSTS[2:0] and TAUJnCMURm.TAUJnTIS[1:0]) must be identical for both channels.

Result:

The down-counter of the second channel reaches 0000 0000_H at exactly the same time as the up-counter of the first channel overflows (TAUJnCNTm = FFFF $FFFF_H$). Thus the second channel generates the desired interrupt.

The following sections list the operating modes that count down that are required to match specific operating modes that count up, as well as example timing diagrams.

18.4.7.1 Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

When the capture trigger is input simultaneously to TAUJTTINm of both channels, INTTAUJnIm of the interval timer function can detect the overflow when TAUDJnCNTm of the TAUJTTINm input position detection function exceeds FFFF FFFF$_H$.

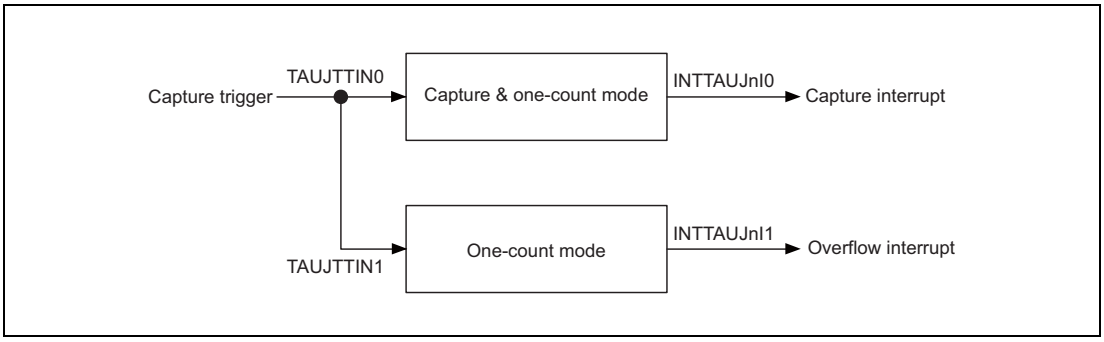


Figure 18.11 Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

Timing diagram

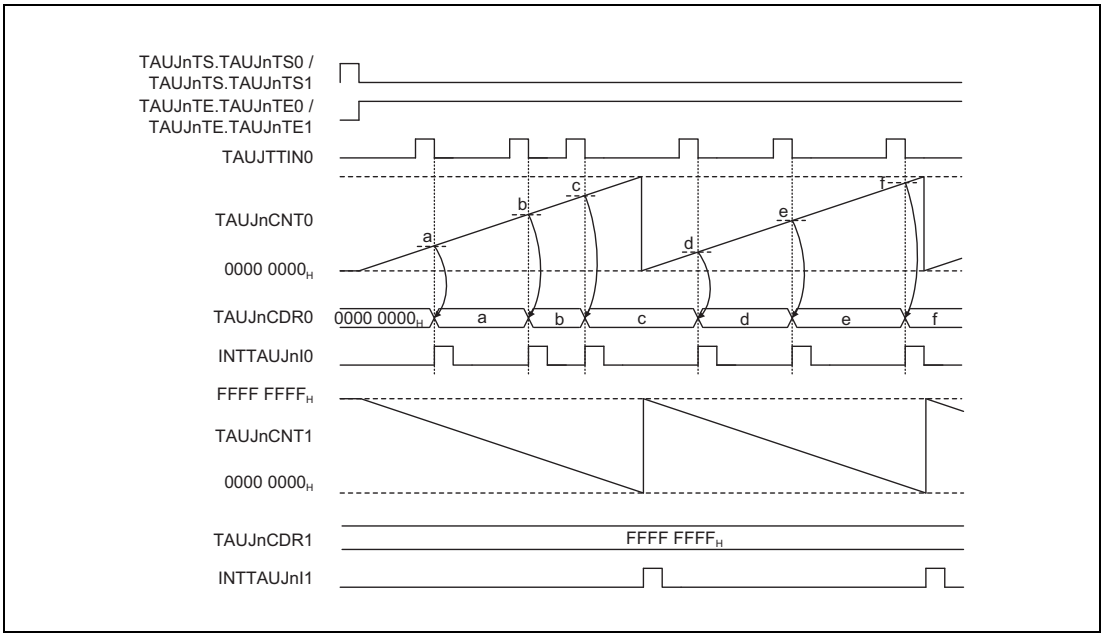


Figure 18.12 Interrupt Generation via Combination of the TAUJTTINm Input Position Detection Function and the Interval Timer Function

18.4.8 TAUJTTINm Edge Detection

Edge detection is based on the operation clock. This means that an edge can only be detected at the next rising edge of the operation clock. This can lead to a maximum delay of one operation clock cycle.

The following figure shows when edge detection takes place.

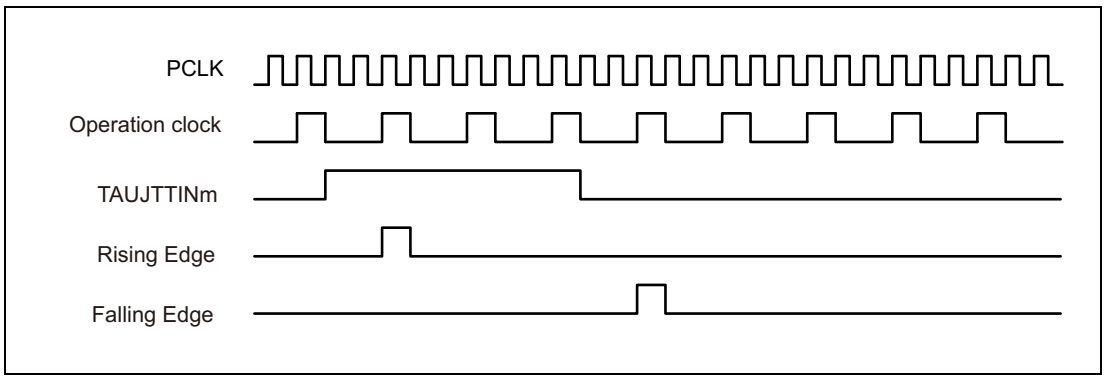


Figure 18.13 Basic Edge Detection Timing

CAUTION

Figure 18.13 shows an image of operation timing. In the actual operation, delay time occurs due to noise filter and synchronization circuit between the TAUJnIm pin and TAUJn.

18.4.9 Independent Channel Operation Functions

- **Section 18.4.9.1, Interval timer function**
- **Section 18.4.9.2, TAUJTTINm input interval timer function**
- **Section 18.4.9.3, TAUJTTINm input pulse interval measurement function**
- **Section 18.4.9.4, TAUJTTINm input signal width measurement function**
- **Section 18.4.9.5, TAUJTTINm input position detection function**
- **Section 18.4.9.6, TAUJTTINm input period count detection function**

18.4.9.1 Interval timer function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUJnIm) at regular intervals. When an interrupt is generated, the TAUJTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode must be set to interval timer mode (see **Table 18.33, Contents of TAUJnCMORm Register for Interval Timer Function**).
- The channel output mode must be set to independent channel output mode 1. See **Section 18.4.4, Channel Output Modes**.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The current value of TAUJnCDRm is written to TAUJnCNTm and the counter starts to count down from this value.

When the counter reaches 0000 0000_H, INTTAUJnIm is generated and the TAUJTTOUTm signal toggles. TAUJnCNTm then loads the TAUJnCDRm value and subsequently continues to operate.

The value of TAUJnCDRm can be rewritten at any time, and the changed value of TAUJnCDRm is applied the next time the counter starts to count down.

The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0. TAUJnCNTm and TAUJTTOUTm stop but retain their values. The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1. The counter can also be forcibly restarted (without stopping it first) by setting TAUJnTS.TAUJnTSm to 1 during operation.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated, and therefore TAUJTTOUTm does not toggle. This results in an inverted TAUJTTOUTm signal compared to when TAUJnCMORm.TAUJnMD0 is set to 1. For details refer to **Section 18.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

(2) Equations

INTTAUJnIm cycle = count clock cycle × (TAUJnCDRm + 1)

TAUJTTOUTm square wave cycle = count clock cycle × (TAUJnCDRm + 1) × 2

(3) Block diagram and general timing diagram

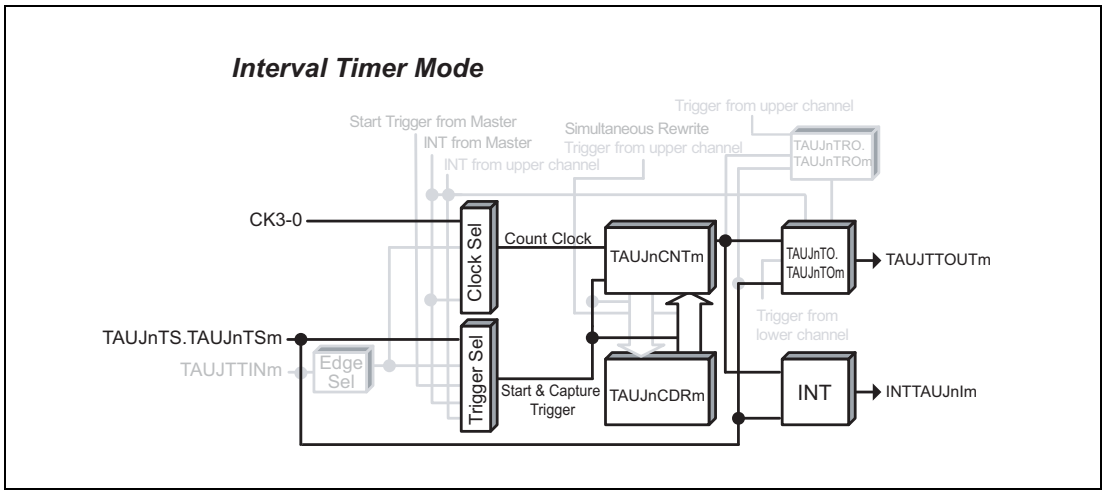


Figure 18.14 Block Diagram for Interval Timer Function

The following settings apply to the general timing diagram:

- INTTAUJnIm is generated at operation start (TAUJnCMORm.(TAUJnMD0 = 1)

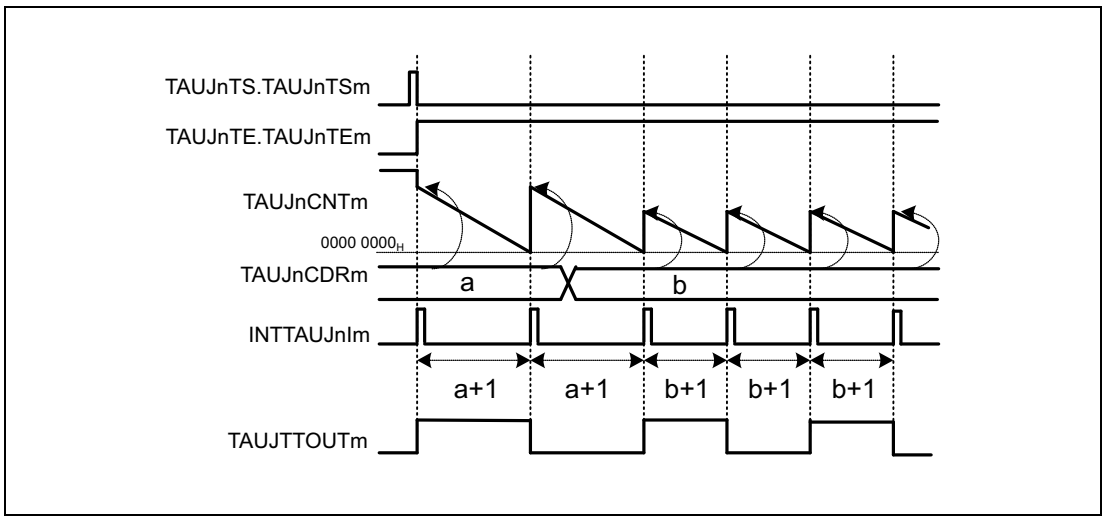


Figure 18.15 General Timing Diagram for Interval Timer Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.33 Contents of TAUJnCMORm Register for Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	000: A software trigger is used as an start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	0: INTTAUJnIm is not generated to toggle TAUJTTOUTm at the start of operation. 1: INTTAUJnIm is generated to toggle TAUJTTOUTm at the start or restart of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.34 Contents of TAUJnCMURm Register for Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Not used, so set to 00

(c) Channel output mode

Table 18.35 Control Bit Settings for Independent Channel Output Mode 1

Bit name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	0: Independent channel output
TAUJnTOC.TAUJnTOCm	0: Toggle mode
TAUJnTOL.TAUJnTOLm	0: Positive logic

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUJnTOE.TAUJnTOEm = 0. TAUJTTOUTm can then be controlled independently of the interrupts. For details refer to **Section 18.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the Interval Timer Function. Therefore, these registers must be set to 0.

Table 18.36 Simultaneous Rewrite Settings for Interval Timer Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for interval timer function

Table 18.37 Operating Procedure for Interval Timer Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.33, Contents of TAUJnCMORm Register for Interval Timer Function** and **Table 18.34, Contents of TAUJnCMURm Register for Interval Timer Function**. Set the value of the TAUJnCDRm register. Set the channel output mode by setting the control bits as described in **Table 18.35, Control Bit Settings for Independent Channel Output Mode 1**.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. TAUJnCNTm loads the TAUJnCDRm value. When TAUJnCMORm.TAUJnMD0 = 1, INTTAUJnIm is generated and TAUJTTOUTm toggles.
During operation	The TAUJnCDRm register value can be changed at any time. The TAUJnCNTm register can be read at all times.	TAUJnCNTm counts down. When the counter reaches 0000 0000_H: • TAUJnCNTm reloads the TAUJnCDRm value and continues count operation • INTTAUJnIm is generated and TAUJTTOUTm toggles.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(6) Specific timing diagrams

(a) TAUJnCDRm = 0000 0000_H, count clock = PCLK/2

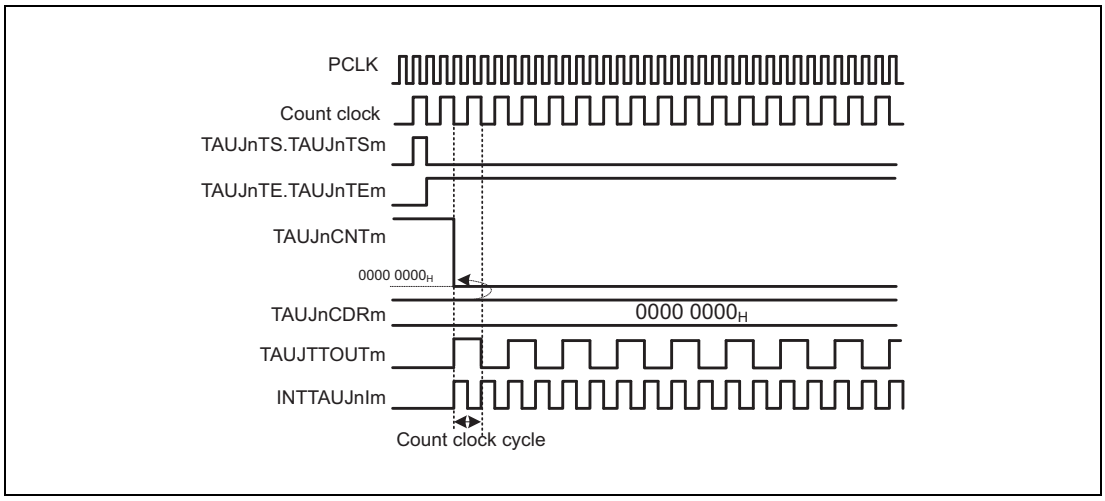


Figure 18.16 TAUJnCDRm = 0000 0000_H, Count Clock = PCLK/2

- If TAUJnCDRm = 0000 0000_H and the count clock = PCLK/2, the TAUJnCDRm value is written to TAUJnCNTm every count clock, meaning that TAUJnCNTm is always 0000 0000_H.
- INTTAUJnIm is generated every count clock, resulting in TAUJTTOUTm toggling every count clock.

(b) TAUJnCDRm = 0000 0000_H, count clock = PCLK

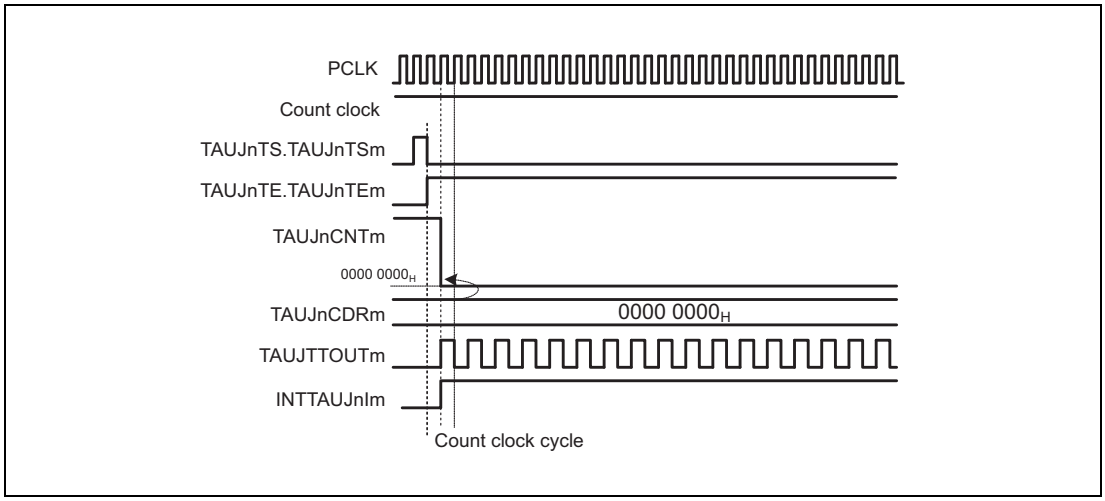


Figure 18.17 TAUJnCDRm = 0000 0000_H, Count Clock = PCLK

- If TAUJnCDRm = 0000 0000_H and the count clock = PCLK, the TAUJnCDRm value is written to TAUJnCNTm every PCLK clock, meaning that TAUJnCNTm is always 0000 0000_H.
- INTTAUJnIm is generated continuously, resulting in TAUJTTOUTm toggling every PCLK clock.

(c) Operation stop and restart

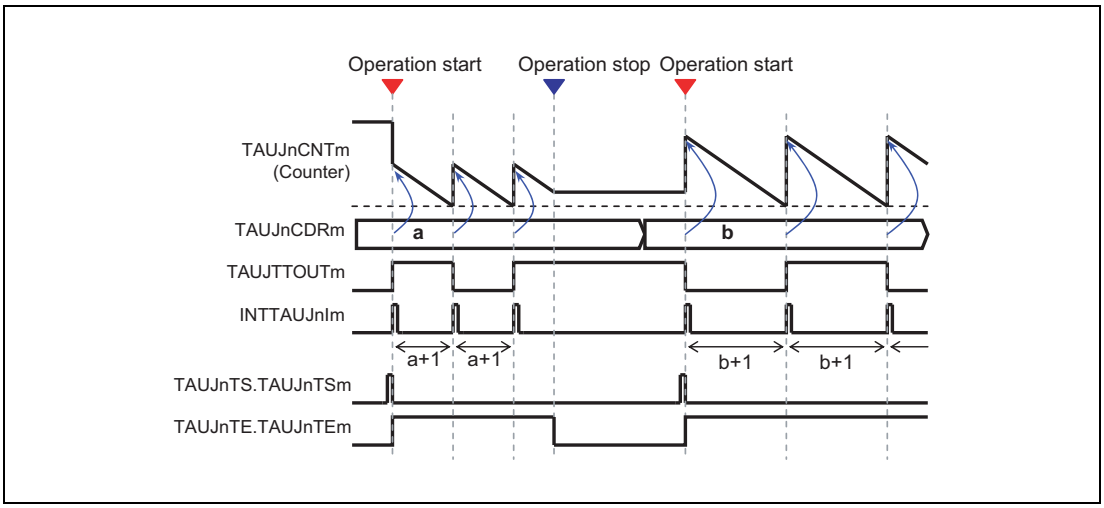


Figure 18.18 Operation Stop and Restart, TAUJnCMORm.TAUJnMD0 = 1

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm and TAUJTTOUTm stop but retain their values.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1.

(d) Forced restart

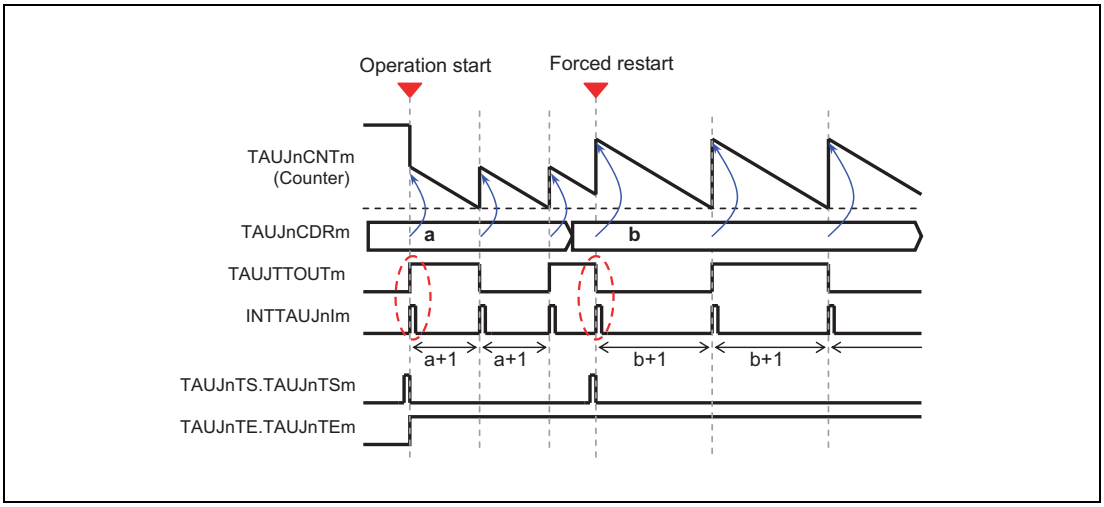


Figure 18.19 Forced Restart Operation, TAUJnCMORm.TAUJnMD0 = 1

- The counter can be forcibly restarted (without stopping it first) by setting TAUJnTS.TAUJnTSm to 1 during operation.
- If the TAUJnCMORm.TAUJnMD0 bit is set to 1, the first interrupt after a start or restart is generated.

18.4.9.2 TAUJTTINm input interval timer function

(1) Overview

Summary

This function is used as a reference timer for generating timer interrupts (INTTAUJnIm) at regular intervals or when an effective TAUJTTINm input edge is detected. When an interrupt is generated, the TAUJTTOUTm signal toggles, resulting in a square wave.

Prerequisites

- The operating mode must be set to interval timer mode (see **Table 18.15, TAUJnCMORm Register Contents**).
- The channel output mode must be set to independent channel output mode 1. See **Table 18.31, Channel Output Modes**.

Description

This function operates in an identical manner to the Interval Timer Function (see **Section 18.4.9.1, Interval timer function**), except that this function is restarted by an effective TAUJTTINm input edge. The type of edge used as the trigger is specified using the TAUJnCMURm.TAUJnTIS[1:0] bits. Either rising edge, falling edge, or rising and falling edge can be selected.

(2) Equations

INTTAUJnIm cycle = count clock cycle × (TAUJnCDRm + 1)

TAUJTTOUTm square wave cycle = count clock cycle × (TAUJnCDRm + 1) × 2

(3) Block diagram and general timing diagram

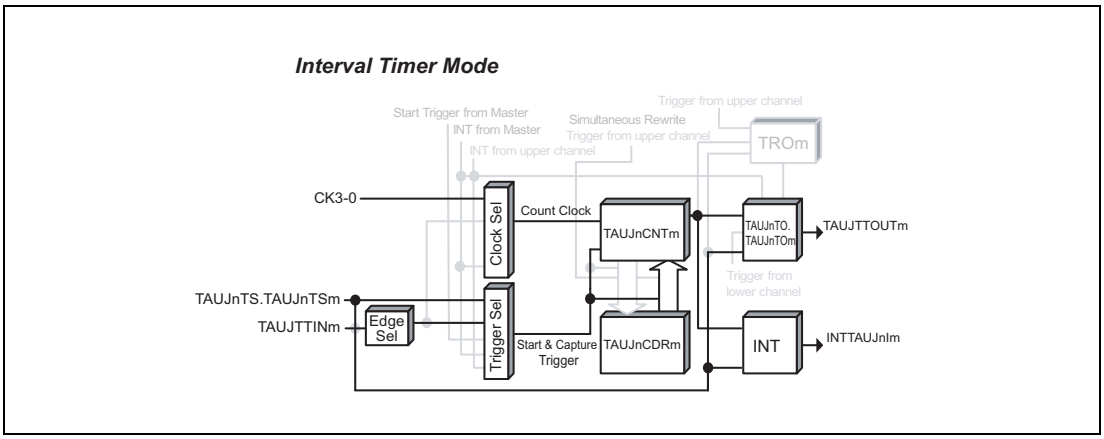


Figure 18.20 Block Diagram for TAUJTTINm Input Interval Timer Function

The following settings apply to the general timing diagram:

- INTTAUJnIm is generated at operation start (TAUJnCMORm.TAUJnMD0 = 0)
- Rising edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 01_B)

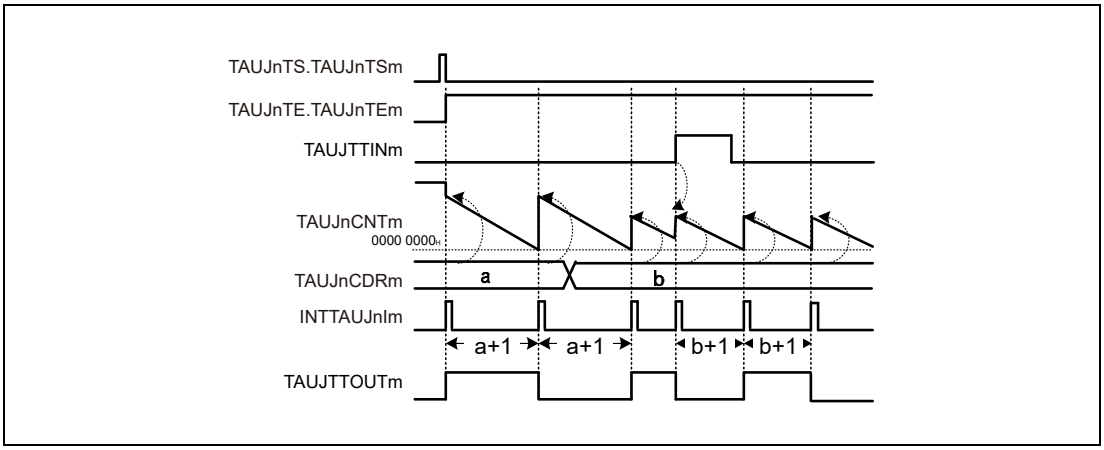


Figure 18.21 General Timing Diagram for TAUJTTINm Input Interval Timer Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.38 Contents of TAUJnCMORm Register for TAUJTTINm input Interval Timer Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	001: Effective TAUJTTINm input edge signal is used as an external start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	0: INTTAUJnIm is not generated to toggle TAUJTTOUTm at the start of operation. 1: INTTAUJnIm is generated to toggle TAUJTTOUTm at the start of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.39 Contents of TAUJnCMURm Register for TAUJTTINm input Interval Timer Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

Table 18.40 Control Bit Settings for Independent Channel Output Mode 1

Bit name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	0: Independent channel output
TAUJnTOC.TAUJnTOCm	0: Operating mode 1 (toggle mode if TAUJnTOM.TAUJnTOMm = 0)
TAUJnTOL.TAUJnTOLm	0: Positive logic

NOTE

The channel output mode can also be set to Channel Output Mode Controlled by Software by setting TAUJnTOE.TAUJnTOEm = 0. TAUJTTOUTm can then be controlled independently of the interrupts. For details refer to **Section 18.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Interval Timer Function. Therefore, these registers must be set to 0.

Table 18.41 Simultaneous Rewrite Settings for TAUJTTINm Input Interval Timer Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input interval timer function

Table 18.42 Operating Procedure for TAUJTTINm Input Interval Timer Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.38, Contents of TAUJnCMORm Register for TAUJTTINm input Interval Timer Function** and **Table 18.39, Contents of TAUJnCMURm Register for TAUJTTINm input Interval Timer Function**. Set the value of the TAUJnCDRm register Set the channel output mode by setting the control bits as described in **Table 18.40, Control Bit Settings for Independent Channel Output Mode 1**.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. TAUJnCNTm loads the TAUJnCDRm value. When TAUJnCMORm.TAUJnMD0 = 1, INTTAUJnIm is generated and TAUJTTOUTm toggles
During operation	The values of the TAUJnCMURm.TAUJnTIS[1:0] and TAUJnCDRm registers can be changed at any time. The TAUJnCNTm register can be read at all times. Detection of TAUJTTINm edge	TAUJnCNTm counts down. When the counter reaches 0000 0000_H: • TAUJnCNTm reloads the TAUJnCDRm value and continues count operation • INTTAUJnIm is generated and TAUJTTOUTm toggles When an effective TAUJTTINm input edge is detected during count operation, TAUJnCNTm reloads the TAUJnCDRm value and continues count operation. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(6) Specific timing diagrams

The timing diagrams in **Section 18.4.9.1, Interval timer function** apply, and in addition the counter can also be restarted by an effective TAUJTTINm input edge.

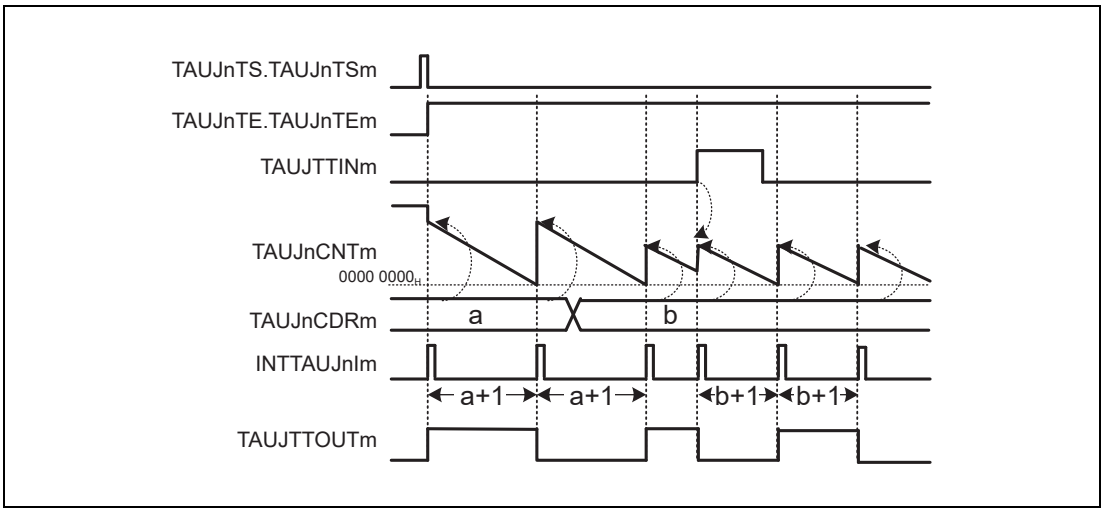


Figure 18.22 Counter Triggered by Rising TAUJTTINm Input Edge (TAUJnCMURm.TAUJnTIS[1:0] = 01_B), TAUJnCMORm.TAUJnMD0 = 1

If an effective TAUJTTINm input edge is detected, an interrupt is generated which causes TAUJTTOUTm to toggle. In this example, the effective edge is a rising edge (TAUJnCMURm.TAUJnTIS[1:0] = 01_B).

18.4.9.3 TAUJTTINm input pulse interval measurement function

(1) Overview

Summary

This function captures the count value and uses this value and the overflow bit TAUJnCSRm.TAUJnOVF to measure the interval of the TAUJTTINm input signal.

Prerequisites

- The operating mode must be set to capture mode (see **Table 18.44, Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function**).
- TAUJTTOUTm is not used with this function.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter TAUJnCNTm starts counting up from 0000 0000_H. When an effective TAUJTTINm edge is detected, the value of TAUJnCNTm is captured, transferred to TAUJnCDRm, and an interrupt INTTAUJnIm is generated. The counter resets to 0000 0000_H and subsequently continues to operate.

If the counter reaches FFFF $FFFF_H$ before an effective TAUJTTINm edge is detected, it overflows to 0000 0000_H. The counter is reset to 0000 0000_H and subsequently continues to operate. The values transferred to TAUJnCDRm and TAUJnCSRm.TAUJnOVF respectively depend on the values of bits TAUJnCMORm.TAUJnCOS[1:0]:

Table 18.43 Effects of an Overflow

TAUJnCMORm.TAUJnCOS[1:0]	When Overflow Occurs		When an Effective TAUJTTINm Input is then Detected	
	TAUJnCDRm	TAUJnCSRm.TAUJnOVF	TAUJnCDRm and TAUJnCNTm	TAUJnCSRm.TAUJnOVF
00	Unchanged	0	TAUJnCNTm written to TAUJnCDRm	1
01		1		
10	Set to FFFF $FFFF_H$	0	TAUJnCNTm set to 0, TAUJnCDRm unchanged	Unchanged
11		1		

If an overflow is set (TAUJnCSRm.TAUJnOVF = 1) when TAUJnCMORm.TAUJnCOS[0] = 1, it can only be cleared by setting TAUJnCSCm.TAUJnCLOV = 1.

The combination of the value of TAUJnCDRm and TAUJnCSRm.TAUJnOVF can be used to deduce the interval of the TAUJTTINm signal. However, if an overflow occurs multiple times before an effective TAUJTTINm input is detected, the overflow bit TAUJnCSRm.TAUJnOVF cannot indicate this.

The function can be stopped by setting TAUJnTT.TAUJnTTm = 1, which in turn sets TAUJnTE.TAUJnTEm = 0. TAUJnCNTm stops but retains its value. While the function is stopped, effective TAUJTTINm input edge detection and TAUJnCNTm capture are not performed.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details refer to **18.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

NOTE

When TAUJnCMORm.TAUJnCOS[1:0] = 1, the value of TAUJnCNTm is not written to TAUJnCDRm when the first effective TAUJTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUJTTINm input pulse interval = count clock cycle × [(TAUJnCSRm.TAUJnOVF × (FFFF FFFF$_H$ + 1)) + TAUJnCDRm capture value + 1]

(3) Block diagram and general timing diagram

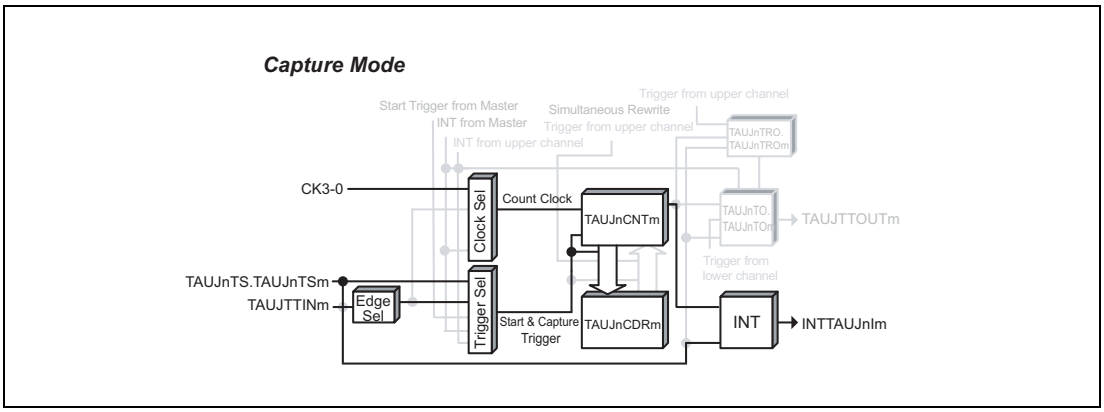


Figure 18.23 Block Diagram for TAUJTTINm Input Pulse Interval Measurement Function

The following settings apply to the general timing diagram:

- INTTAUJnIm not generated at operation start (TAUJnCMORm.TAUJnMD0 = 0)
- Falling edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 00_B)
- When an effective TAUJTTINm input is detected after an overflow, TAUJnCDRm is changed and TAUJnCSRm.TAUJnOVF is set to 1 (TAUJnCMORm.TAUJnCOS[1:0] = 00_B)

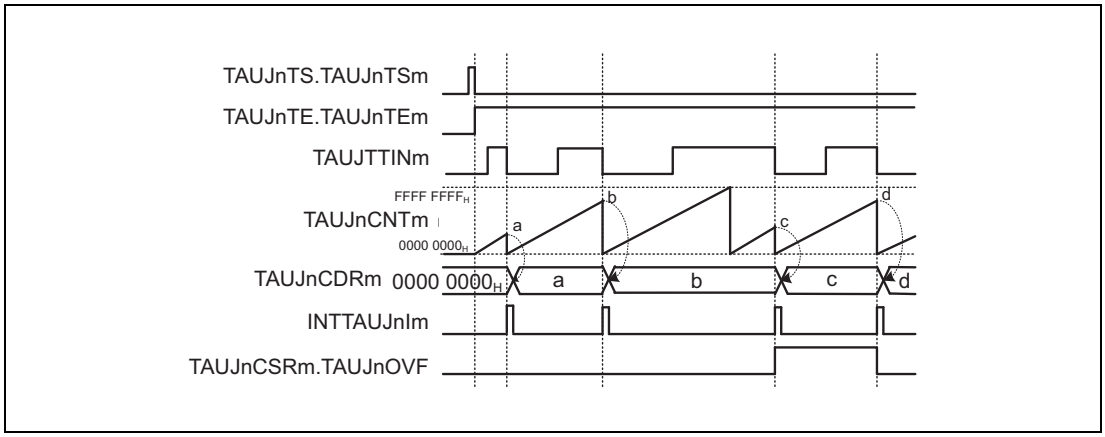


Figure 18.24 General Timing Diagram for TAUJTTINm Input Pulse Interval Measurement Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.44 Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	001: Effective edge of the TAUJTTINm input signal is used as an external capture trigger.
7, 6	TAUJnCOS[1:0]	See **Table 18.43, Effects of an Overflow**.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0010: Capture mode
0	TAUJnMD0	0: INTTAUJnIm is not generated at the start operation. 1: INTTAUJnIm is generated at the start operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.45 Contents of TAUJnCMURm Register for TAUJTTINm Input Pulse Interval Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used by this function. However, it can be used in Independent Channel Output Mode Controlled by Software.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Pulse Interval Measurement Function. Therefore, these registers must be set to 0.

Table 18.46 Simultaneous Rewrite Settings for TAUJTTINm Input Pulse Interval Measurement Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input pulse interval measurement function

Table 18.47 Operating Procedure for TAUJTTINm Input Pulse Interval Measurement Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.44, Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function** and **Table 18.45, Contents of TAUJnCMURm Register for TAUJTTINm Input Pulse Interval Measurement Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. TAUJnCNTm is cleared to 0000 0000_H. INTTAUJnIm is generated when TAUJnCMORm.TAUJnMD0 is set to 1.
During operation	Detection of TAUJTTINm edges. The TAUJnCMURm. TAUJnTIS[1:0] bits can be changed at any time. The TAUJnCDRm and TAUJnCSRm registers can be read at any time. TAUJnCSCm.TAUJnCLOV bit can be written to 1. (TAUJnCSRm.TAUJnOVF bit is cleared to 0.)	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • TAUJnCNTm transfers (captures) its value to TAUJnCDRm, and returns to 0000 0000_H • INTTAUJnIm is then generated. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm stops and both it and TAUJnCSRm.TAUJnOVF retain their current values.

Restart

(6) Specific timing diagrams: overflow behavior

(a) TAUJnCMORm.TAUJnCOS[1:0] = 00_B

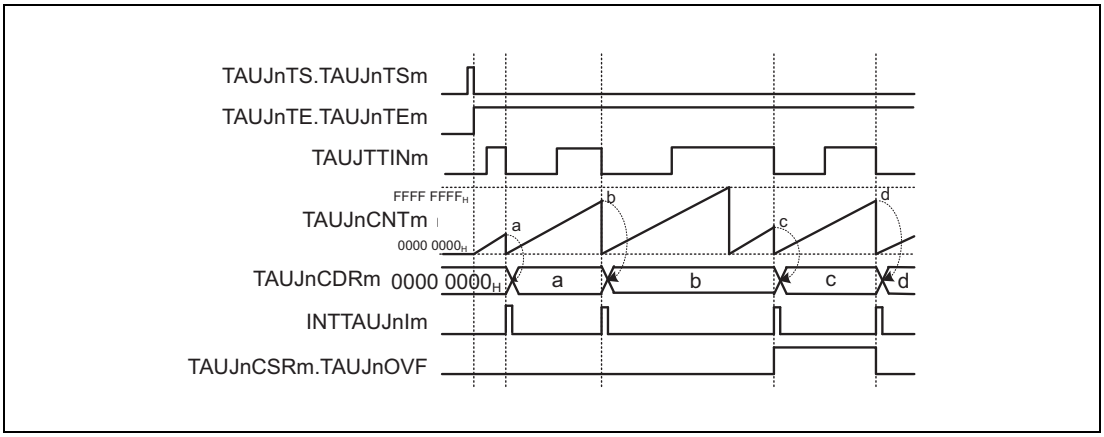


Figure 18.25 TAUJnCMORm.TAUJnCOS[1:0] = 00_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm and TAUJnCSRm.TAUJnOVF is set to1.
- If the next effective TAUJTTINm input edge is detected when no overflow occurs, TAUJnCSRm.TAUJnOVF is cleared to 0.

(b) TAUJnCMORm.TAUJnCOS[1:0] = 01_B

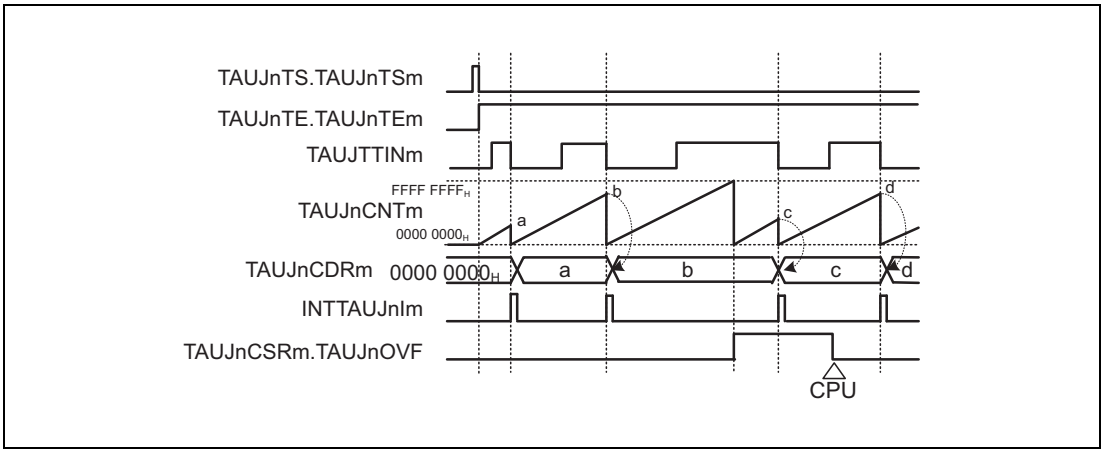


Figure 18.26 TAUJnCMORm.TAUJnCOS[1:0] = 01_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF is set to1.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm.
- TAUJnCSRm.TAUJnOVF is only cleared by a CPU command (by setting the TAUJnCSCm.TAUJnCLOV bit to 1).

(c) TAUJnCMORm.TAUJnCOS[1:0] = 10_B

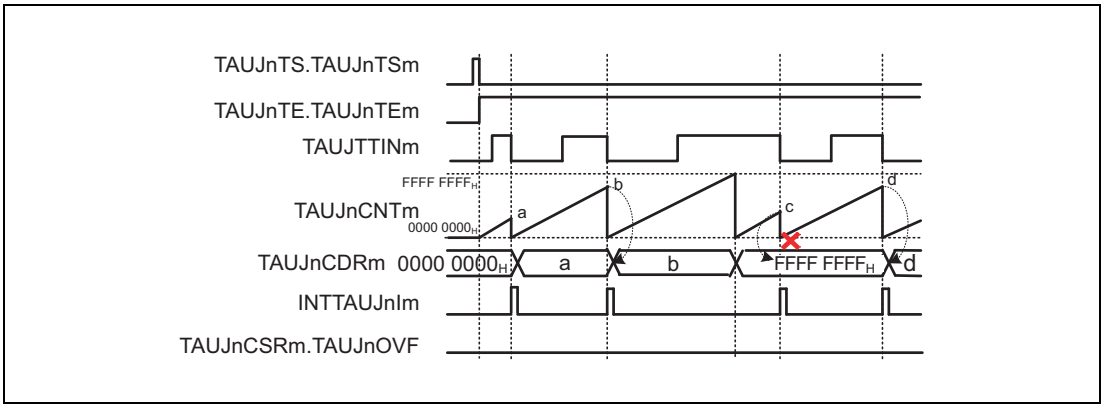


Figure 18.27 TAUJnCMORm.TAUJnCOS[1:0] = 10_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, TAUJnCDRm is set to FFFF $FFFF_H$ and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, TAUJnCNTm is reset to0, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.

(d) TAUJnCMORm.TAUJnCOS[1:0] = 11_B

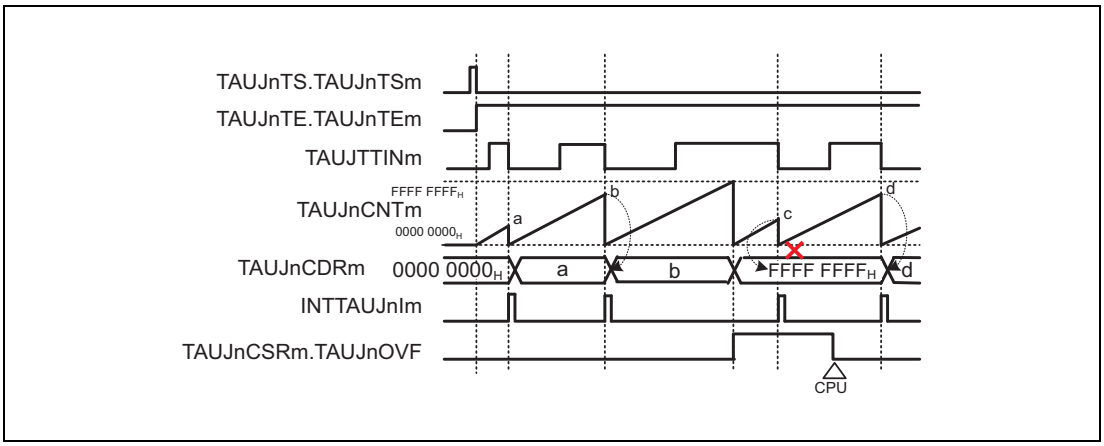


Figure 18.28 TAUJnCMORm.TAUJnCOS[1:0] = 11_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- When an overflow occurs, TAUJnCDRm is set to FFFF FFFF$_H$, and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, TAUJnCNTm is reset to 0, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.
- TAUJnCSRm.TAUJnOVF is cleared by setting the TAUJnCSCm.TAUJnCLOV bit to 1.

18.4.9.4 TAUJTTINm input signal width measurement function

(1) Overview

Summary

This function measures the width of a TAUJTTINm signal by starting counting on one edge of the TAUJTTINm signal and capturing the counter value on the opposite edge.

Prerequisites

- The operating mode should be set to capture and one-count mode (see **Table 18.49, Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function**).
- TAUJTTOUTm is not used with this function.
- TAUJnCMORm.TAUJnMD0 should be set to 0.

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. When an effective TAUJTTINm start edge is detected, the counter TAUJnCNTm starts counting up from 0000 0000_H. When an effective TAUJTTINm stop edge is detected, the value of TAUJnCNTm is captured, transferred to TAUJnCDRm, and an interrupt INTTAUJnIm is generated. The counter retains its value and awaits the next effective TAUJTTINm input start edge.

If the counter reaches FFFF $FFFF_H$ before an effective TAUJTTINm stop edge is detected, it overflows. The counter is reset to 0000 0000_H and subsequently continues to operate. The values transferred to TAUJnCDRm and TAUJnCSRm.TAUJnOVF respectively depend on the values of bits TAUJnCMORm.TAUJnCOS[1:0]:

Table 18.48 Effects of an overflow

TAUJnCMORm.TAUJnCOS[1:0]	When Overflow Occurs		When an Effective TAUJTTINm Input Stop Edge is Detected	
	TAUJnCDRm	TAUJnCSRm.TAUJnOVF	TAUJnCDRm and TAUJnCNTm	TAUJnCSRm.TAUJnOVF
00	Unchanged	0	TAUJnCNTm written to TAUJnCDRm	1
01		1		
10	Set to FFFF $FFFF_H$	0	TAUJnCNTm stops counting TAUJnCDRm unchanged	Unchanged
11		1		

If an overflow is set (TAUJnCSRm.OVF = 1) when TAUJnCMORm.TAUJnCOS[0] = 1, it can only be cleared by setting TAUJnCSCm.TAUJnCLOV = 1.

The combination of the value of TAUJnCDRm and TAUJnCSRm.TAUJnOVF can be used to deduce the width of the TAUJTTINm signal. However, if an overflow occurs multiple times before an effective TAUJTTINm input is detected, the overflow bit TAUJnCSRm.TAUJnOVF cannot indicate this.

This function cannot be forcibly restarted.

NOTE

When TAUJnCMORm.TAUJnCOS[1] = 1, the value of TAUJnCNTm is not written to TAUJnCDRm when the first effective TAUJTTINm input edge occurs after an overflow. However, an interrupt is generated.

(2) Equations

TAUJTTINm input signal width = count clock cycle × [(TAUJnCSRm.TAUJnOVF × (FFFF $FFFF_H$ + 1)) + TAUJnCDRm capture value + 1]

(3) Block diagram and general timing diagram

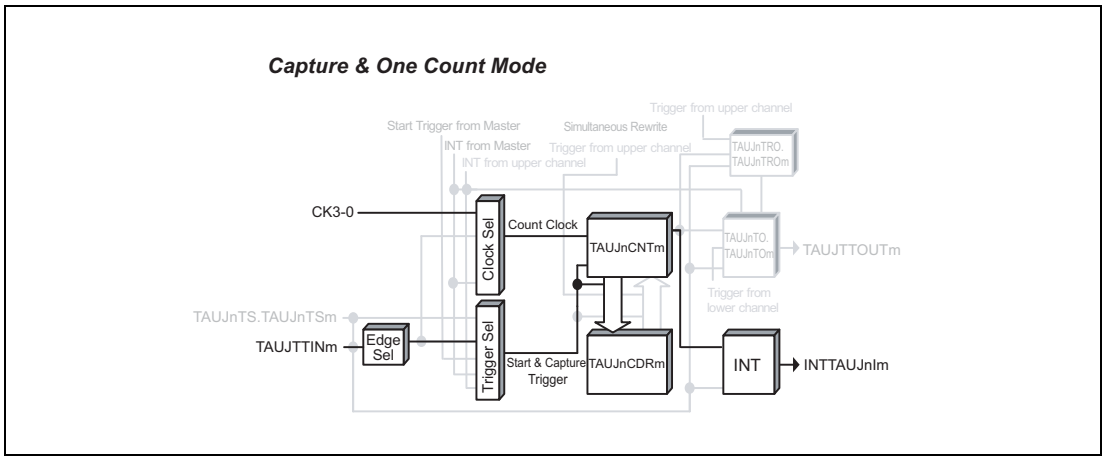


Figure 18.29 Block diagram for TAUJTTINm input signal width measurement function

The following settings apply to the general timing diagram:

- Rising and falling edge detection = high width measurement (TAUJnCMURm.TAUJnTIS[1:0] = 11_B)
- When an effective TAUJTTINm input is detected after an overflow, TAUJnCDRm is changed and TAUJnCSRm.TAUJnOVF is set to 1 (TAUJnCMORm.TAUJnCOS[1:0] = 00_B)

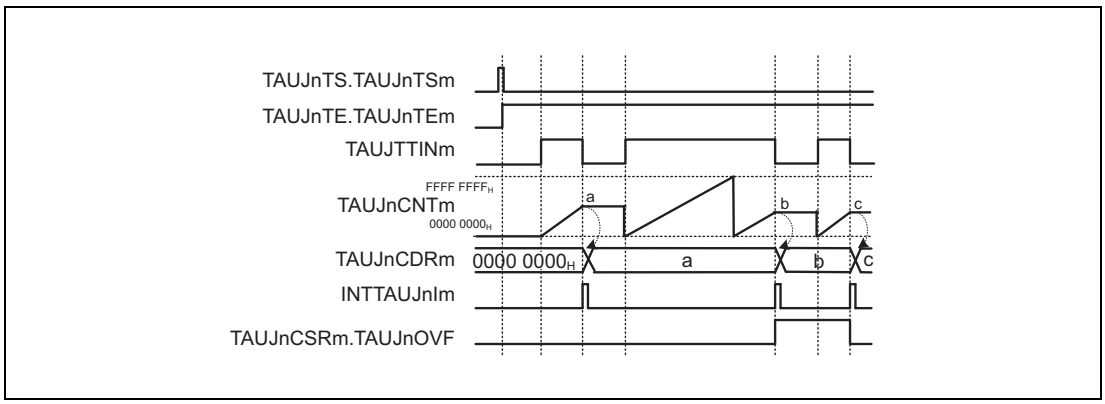


Figure 18.30 General timing diagram for TAUJTTINm input signal width measurement function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.49 Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	010: Effective edge of the TAUJTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUJnCOS[1:0]	See **Table 18.48, Effects of an overflow**.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0110: Capture and one-count mode
0	TAUJnMD0	0: Disables the start trigger during operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.50 Contents of TAUJnCMURm Register for TAUJTTINm Input Signal Width Measurement Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	10: Rising and falling edge detection (low width measurement) 11: Rising and falling edge detection (high width measurement)

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Signal Width Measurement Function. Therefore, these registers must be set to 0.

Table 18.51 Simultaneous rewrite settings for TAUJTTINm input signal width measurement function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm = 0), set these bits to 0

(5) Operating procedure for TAUJTTINm input signal width measurement function

Table 18.52 Operating procedure for TAUJTTINm input signal width measurement function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.49, Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function** and **Table 18.50, Contents of TAUJnCMURm Register for TAUJTTINm Input Signal Width Measurement Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. When a TAUJTTINm start edge is detected, TAUJnCNTm starts to count up.
During operation	Detection of TAUJTTINm edges. The TAUJnCDRm, TAUJnCNTm, and TAUJnCSRm registers can be read at any time. The TAUJnCSCm.TAUJnCLOV bit can be set to 1.	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • AUJnCNTm transfers (captures) its value to TAUJnCDRm, and retains its value • INTTAUJnIm is then generated. • Counting stops at the value that transferred to TAUJnCDRm + 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TAUJnTTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm stops and both it and TAUJnCSRm.TAUJnOVF retain their current values.

Restart

(6) Specific timing diagrams: overflow behavior

(a) TAUJnCMORm.TAUJnCOS[1:0] = 00_B

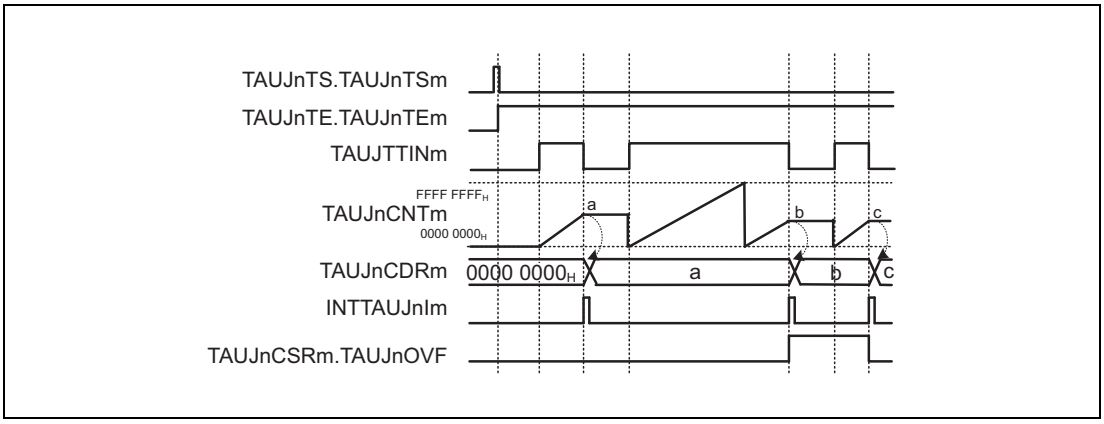


Figure 18.31 TAUJnCMORm.TAUJnCOS[1:0] = 00_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge with no overflow occurring, TAUJnCSRm.TAUJnOVF is cleared to 0.

(b) TAUJnCMORm.TAUJnCOS[1:0] = 01_B

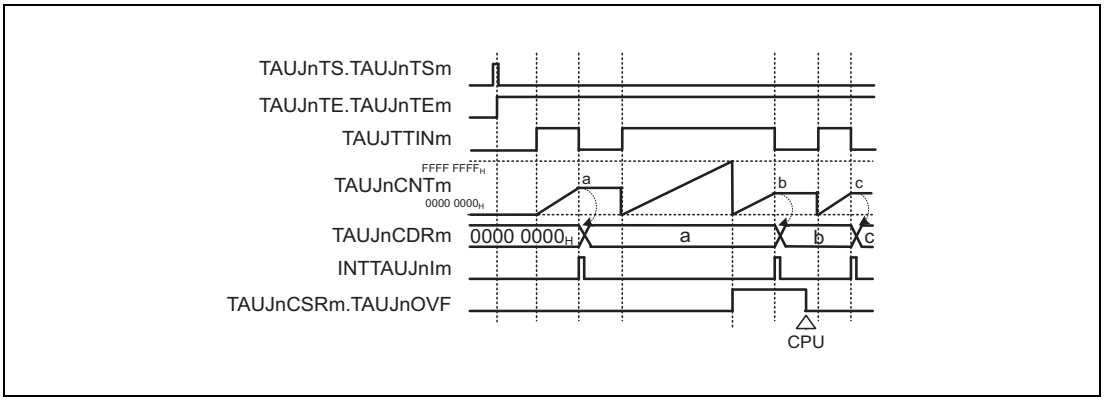


Figure 18.32 TAUJnCMORm.TAUJnCOS[1:0] = 01_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, the value of TAUJnCDRm remains unchanged and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next effective TAUJTTINm input edge, the value of TAUJnCNTm is written to TAUJnCDRm.
- TAUJnCSRm.TAUJnOVF is only cleared by a CPU command (by setting TAUJnCSCm.TAUJnCLOV bit = 1).

(c) TAUJnCMORm.TAUJnCOS[1:0] = 10_B

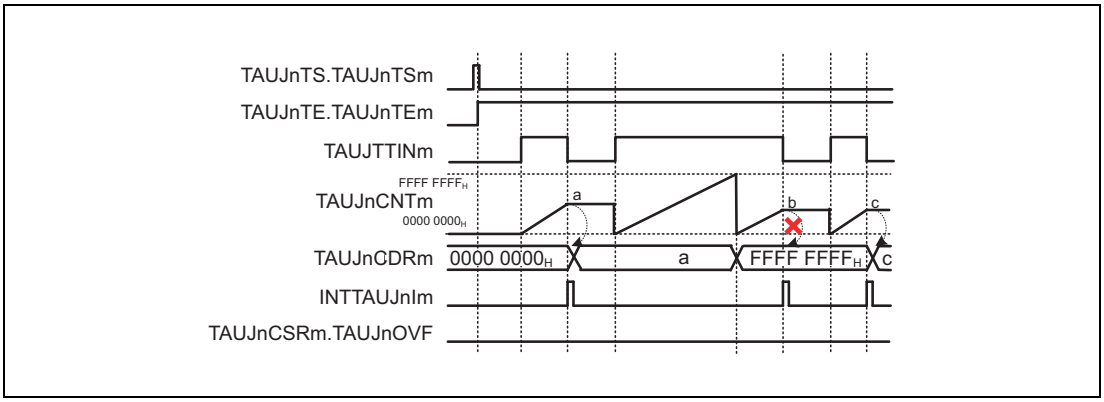


Figure 18.33 TAUJnCMORm.TAUJnCOS[1:0] = 10_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, TAUJnCDRm is set to FFFF FFFF$_H$ and TAUJnCSRm.TAUJnOVF remains = 0.
- Upon detection of the next valid TAUJTTINm input edge, TAUJnCNTm stops counting, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.

(d) TAUJnCMORm.TAUJnCOS[1:0] = 11_B

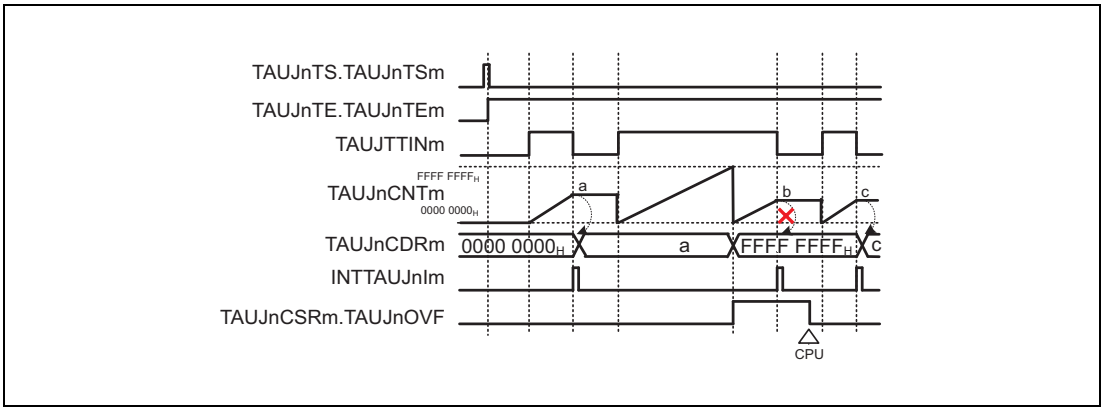


Figure 18.34 TAUJnCMORm.TAUJnCOS[1:0] = 11_B, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- When an overflow occurs, TAUJnCDRm is set to FFFF FFFF$_H$, and TAUJnCSRm.TAUJnOVF is set to 1.
- Upon detection of the next valid TAUJTTINm input edge, TAUJnCNTm stops counting, but TAUJnCDRm and TAUJnCSRm.TAUJnOVF remain unchanged.
- Thus, the next effective TAUJTTINm input edge after the overflow is ignored.
- TAUJnCSRm.TAUJnOVF is cleared by setting the TAUJnCSCm.TAUJnCLOV bit to 1.

18.4.9.5 TAUJTTINm input position detection function

(1) Overview

Summary

This function measures the interval of an input signal by capturing the counter value on an effective edge of the TAUJTTINm signal.

NOTE

The input TAUJTTINm is sampled at the frequency of the operation clock, specified by the TAUJnCMORm.TAUJnCKS[1:0] bits. As a result, the output cycle of TAUJTTOUTm has an error of ± 1 operation clock cycle.

Prerequisites

- The operating mode should be set to count capture mode (see **Table 18.53, Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function**).
- TAUJTTOUTm is not used for this function

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter starts to count from 0000 0000_H. When an effective TAUJTTINm input stop edge is detected, the current TAUJnCNTm value is written to TAUJnCDRm and an interrupt (INTTAUJnIm) is generated. The counter continues to count.

When the counter reaches FFFF $FFFF_H$, the counter restarts from 0000 0000_H.

Conditions

If the TAUJnCMORm.TAUJnMD0 bit is set to 0, the first interrupt after a start or restart is not generated. For details, see **Section 18.4.6, TAUJTTOUTm Output and INTTAUJnIm Generation when Counter Starts or Restarts**.

(2) Equations

Function duration at a TAUJTTINm input pulse =
count clock cycle × (TAUJnCDRm capture value + 1)

(3) Block diagram and general timing diagram

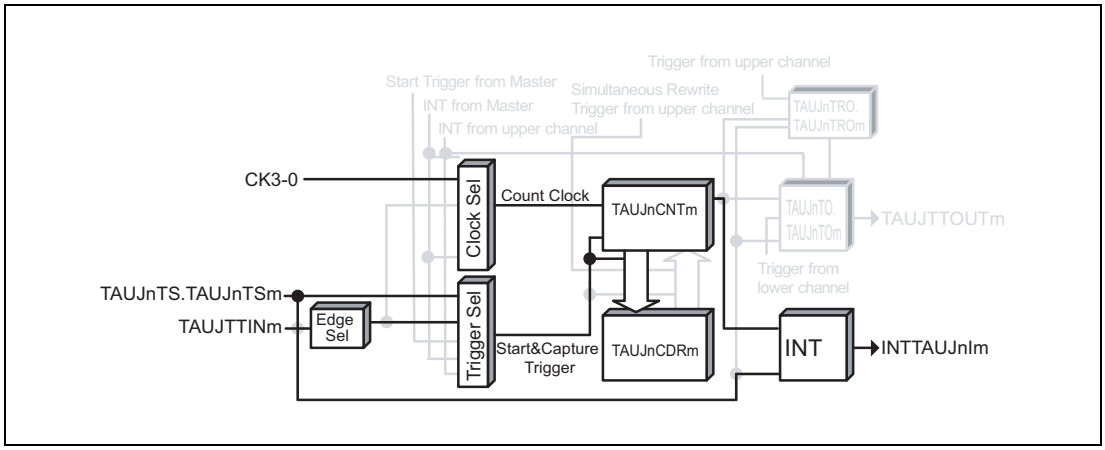


Figure 18.35 Block Diagram of TAUJTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram:

- INTTAUJnIm not generated at operation start (TAUJnCMORm.TAUJnMD0 = 0)
- Falling edge detection (TAUJnCMURm.TAUJnTIS[1:0] = 00_B)

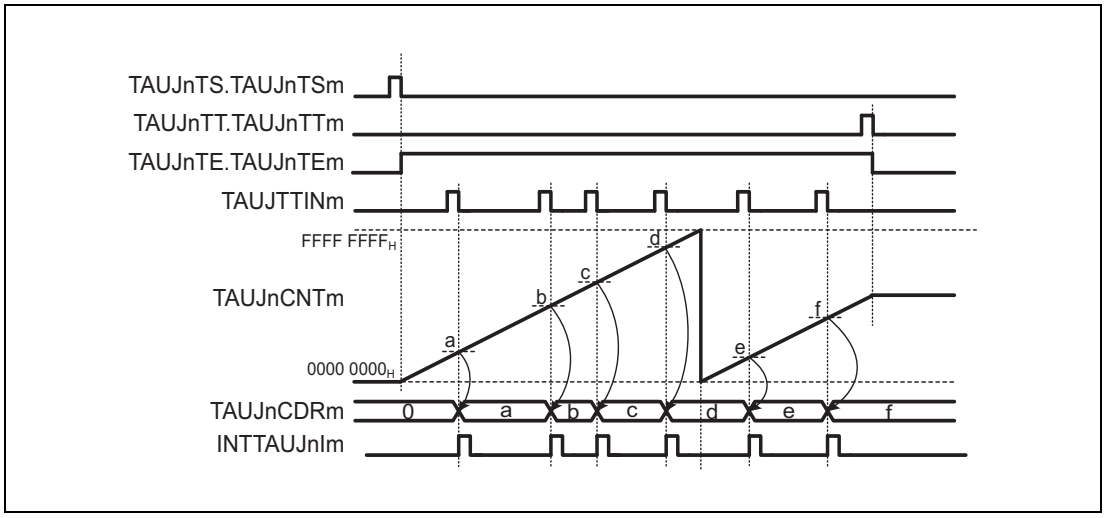


Figure 18.36 General Timing Diagram for TAUJTTINm Input Position Detection Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.53 Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	001: Effective TAUJTTINm input edge signal is used as an external capture trigger.
7, 6	TAUJnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	1011: Count capture mode
0	TAUJnMD0	0: INTTAUJnIm is not generated at the start of operation. 1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.54 Contents of TAUJnCMURm Register for TAUJTTINm Input Position Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Falling edge detection 01: Rising edge detection 10: Rising and falling edge detection

(c) Channel output mode

The channel output mode is not used by this function. However, it can be used in Independent Channel Output Mode Controlled by Software. For details, see **Section 18.4.4, Channel Output Modes**.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Position Detection Function. Therefore, these registers must be set to 0.

Table 18.55 Simultaneous Rewrite Settings for TAUJTTINm Input Position Detection Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm=0), set these bits to 0

(5) Operating procedure for TAUJTTINm input position detection function

Table 18.56 Operating Procedure for TAUJTTINm Input Position Detection Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.53, Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function** and **Table 18.54, Contents of TAUJnCMURm Register for TAUJTTINm Input Position Detection Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is set to 1 and the counter starts. INTTAUJnIm is generated when TAUJnCMORm.TAUJnMD0 is set to 1.
During operation	The TAUJnCMURm.TAUJnTIS[1:0] bits can be changed at any time. The TAUJnCDRm and TAUJnCSRm registers can be read at any time.	TAUJnCNTm starts to count up from 0000 0000_H. When an effective TAUJTTINm edge is detected: • TAUJnCNTm transfers (captures) its value to TAUJnCDRm • INTTAUJnIm is output. • The counter value is not cleared to 0000 0000_H and TAUJnCNTm continues count operation. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm stops. TAUJnCNTm retains its current value.

Restart

(6) Specific timing diagrams

(a) Operation stop and restart

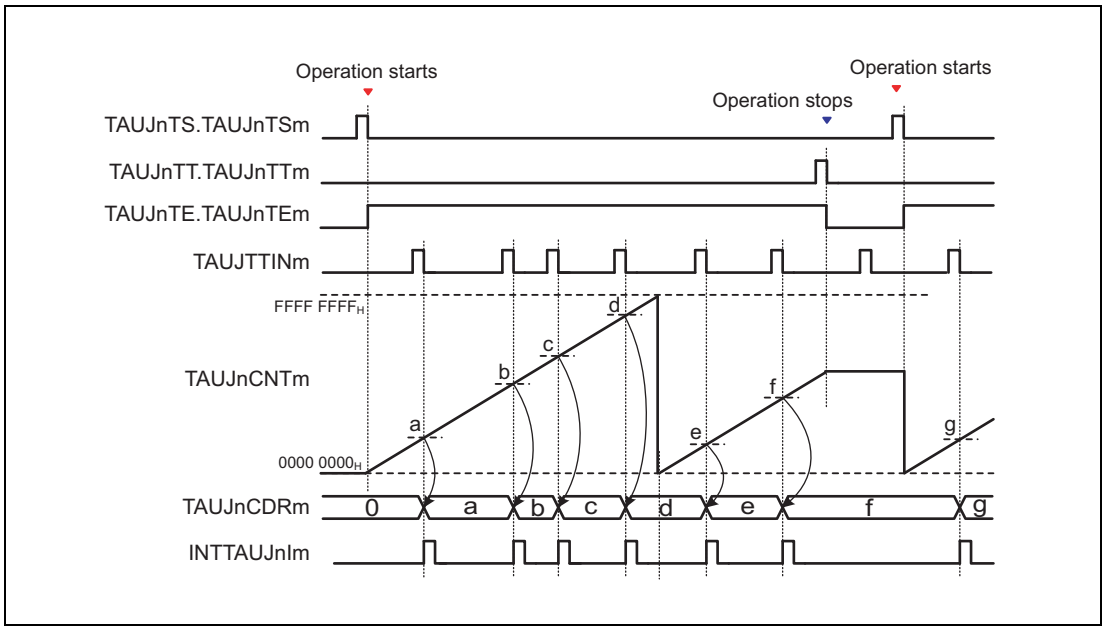


Figure 18.37 Operation Stop and Restart, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm stops and the current value is retained.
- If the counter is stopped, effective TAUJTTINm input edges are ignored.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm to 1. TAUJnCNTm restarts to count from 0000 0000_H.

18.4.9.6 TAUJTTINm input period count detection function

(1) Overview

Summary

This function measures the cumulative width of a TAUJTTINm input signal.

Prerequisites

- The operating mode should be set to capture and gate count mode (see **Table 18.57, Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function**).
- TAUJTTOUTm is not used for this function

Description

The counter is enabled by setting the channel trigger bit (TAUJnTS.TAUJnTSm) to 1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The counter awaits an effective TAUJTTINm input edge.

When an effective TAUJTTINm input start edge is detected, the counter starts to count from $0000\ 0000_H$.

When an effective TAUJTTINm input stop edge is detected, the current TAUJnCNTm value is written to TAUJnCDRm and an interrupt (INTTAUJnIm) is generated. The counter stops and retains its value until the next effective TAUJTTINm input start edge is detected.

When the next effective TAUJTTINm input start edge is detected, the counter restarts counting from the stop value.

If the counter reaches $FFFF\ FFFF_{H,}$ the counter restarts from $0000\ 0000_H$.

NOTE

The input TAUJTTINm signal is sampled at the frequency of the operation clock, specified by the TAUJnCMORm.TAUJnCKS[1:0] bits.

Conditions

The effective start and stop edges are specified by the TAUJnCMURm.TAUJnTIS[1:0] bits:

- If TAUJnCMURm.TAUJnTIS[1:0] = 10_B, the TAUJTTINm input low period is counted. The start trigger is a falling edge and the stop trigger is a rising edge.
- If TAUJnCMURm.TAUJnTIS[1:0] = 11_B, the TAUJTTINm input high period is counted. The start trigger is a rising edge and the stop trigger is a falling edge.

(2) Equations

Cumulative TAUJTTINm input width =
count clock cycle × (TAUJnCDRm capture value + 1)

(3) Block diagram and general timing diagram

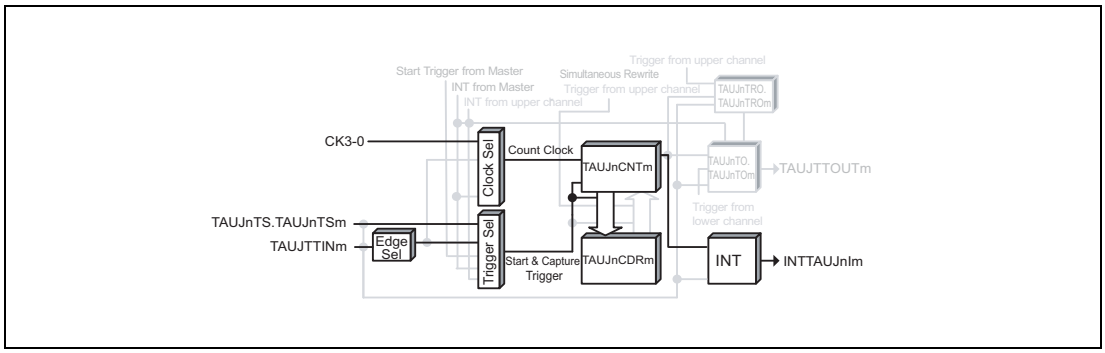


Figure 18.38 Block Diagram for TAUJTTINm Input Period Count Detection Function

The following settings apply to the general timing diagram:

- Rising and falling edge detection = high width measurement (TAUJnCMURm.TAUJnTIS[1:0] = 11_B)

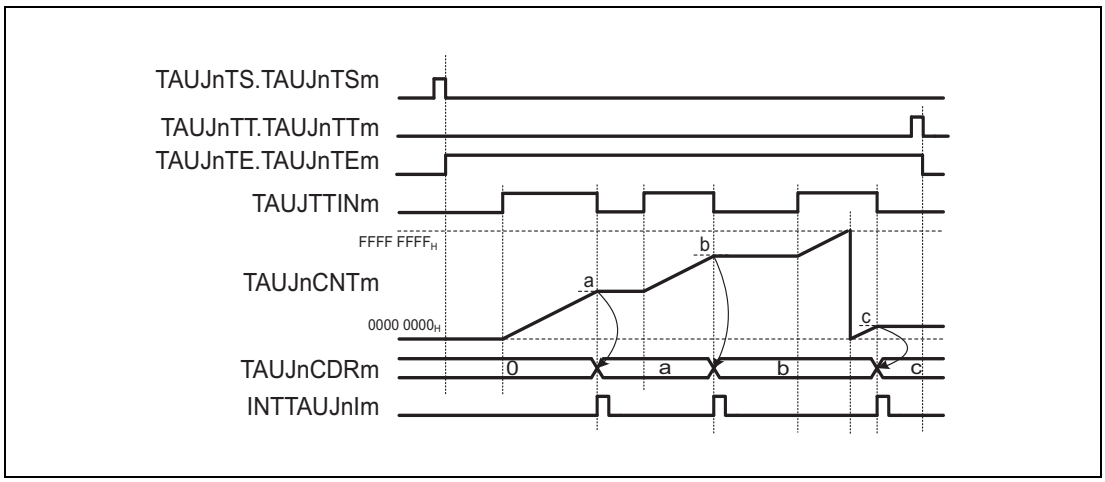


Figure 18.39 General Timing Diagram for TAUJTTINm Input Period Count Detection Function

(4) Register settings

(a) TAUJnCMORm

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.57 Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	00: Not used, so set to 00
10 to 8	TAUJnSTS[2:0]	010: Effective edge of the TAUJTTINm input signal is used as an external start trigger and the reverse edge as a stop trigger.
7, 6	TAUJnCOS[1:0]	01: This value must be set.
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	1101: Capture and gate count mode
0	TAUJnMD0	0: Disables the start trigger during operation.

(b) TAUJnCMURm

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.58 Contents of TAUJnCMURm Register for TAUJTTINm Input Period Count Detection Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	10: Rising and falling edge detection (Low width measurement) 11: Rising and falling edge detection (High width measurement)

(c) Channel output mode

TAUJnTOE.TAUJnTOEm is set to 0 because the channel output mode is not used with this function. However, this mode can be used in independent channel output mode controlled by software.

(d) Simultaneous rewrite

The simultaneous rewrite registers (TAUJnRDE and TAUJnRDM) cannot be used with the TAUJTTINm Input Period Count Detection Function. Therefore, these registers must be set to 0.

Table 18.59 Simultaneous Rewrite Settings for TAUJTTINm Input Period Count Detection Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	0: Disables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: When simultaneous rewrite is disabled (TAUJnRDE.TAUJnRDEm=0), set these bits to 0

(5) Operating procedure for TAUJTTINm input period count detection function

Table 18.60 Operating Procedure for TAUJTTINm Input Period Count Detection Function

	Operation	Status of TAUJn
Initial channel setting	Set the TAUJnCMORm register and TAUJnCMURm registers as described in **Table 18.57, Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function** and **Table 18.58, Contents of TAUJnCMURm Register for TAUJTTINm Input Period Count Detection Function**. The TAUJnCDRm register functions as a capture register.	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm to 1. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0. Detection of TAUJTTINm start edge.	TAUJnTE.TAUJnTEm is set to 1 and TAUJnCNTm waits for detection of the TAUJTTINm start edge. When a start edge is detected, TAUJnCNTm is cleared to 0000 0000_H and TAUJnCNTm starts to count up.
During operation	Detection of TAUJTTINm edges. The TAUJnCDRm, TAUJnCNTm, and TAUJnCSRm registers can be read at any time.	When a TAUJTTINm start edge (rising edge for high level width measurement, falling edge for low level width measurement) is detected, TAUJnCNTm starts to count up from the stop value. When TAUJnCNTm detects a start edge (falling edge for high level width measurement, rising edge for low level width measurement), it transfers the value to TAUJnCDRm and INTTAUJnIm is generated. Counting stops at the “value transferred to TAUJnCDRm + 1” and TAUJnCNTm waits for detection of the TAUJTTINm start edge. If the TAUJnCNTm reaches FFFF $FFFF_H$, the counter restarts count operation from 0000 0000_H. Afterwards, this procedure is repeated.
Stop operation	Set TAUJnTT.TTm to 1. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm stops. TAUJnCNTm retains its current value.

Restart

(6) Specific timing diagrams

(a) Operation stop and restart

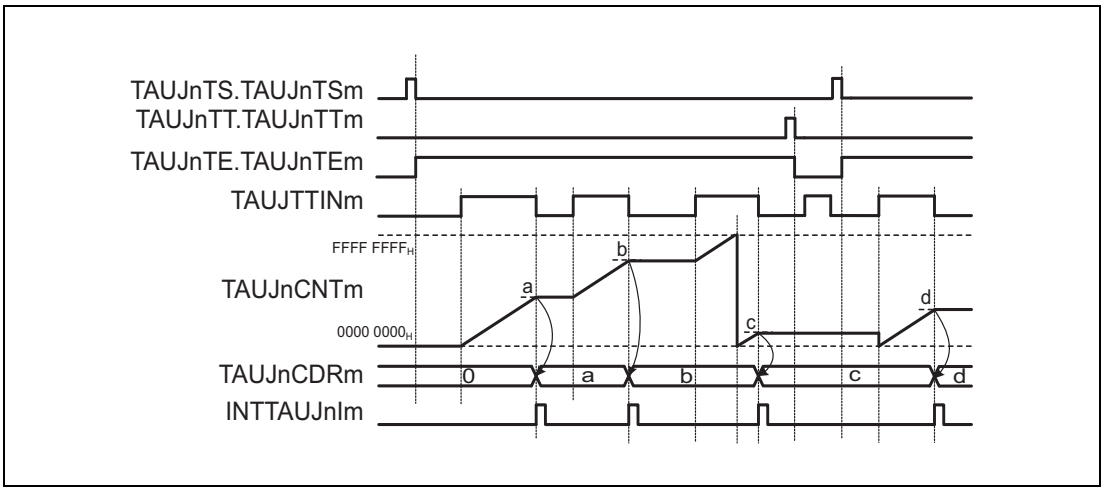


Figure 18.40 Operation Stop and Restart, TAUJnCMURm.TAUJnTIS[1:0] = 11_B

- The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1, which in turn sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm stops and the current value is retained.
- If the counter is stopped, effective TAUJTTINm input edges are ignored.
- The counter can be restarted by setting TAUJnTS. TAUJnTSm to 1. TAUJnCNTm restarts to count from 0000 0000_H.

18.4.10 Synchronous Channel Operation Functions

- **Section 18.4.10.1, PWM output function**

18.4.10.1 PWM output function

(1) Overview

Summary

This function generates multiple PWM outputs by using a master and multiple slave channels. It enables the pulse cycle (frequency) and the duty of the TAUJTTOUTm to be set. The pulse cycle is set in the master channel. The duty is set in the slave channel.

Prerequisites

- Two channels
- The operating mode for the master channel should be set to interval timer mode (see **Table 18.61, Contents of TAUJnCMORm Register of Master Channel for PWM Output Function**).
- The operating mode for the slave channels should be set to one-count mode (see **Table 18.64, Contents of TAUJnCMORm Register of Slave Channel for PWM Output Function**).
- TAUJTTOUTm is not used with the master channel of this function.
- The channel output mode for the slave channels should be set to synchronous channel output mode 1 (see **Section 18.4.4, Channel Output Modes**).

Description

The counters are enabled by setting the channel trigger bits (TAUJnTS.TAUJnTSm) to1. This in turn sets TAUJnTE.TAUJnTEm = 1, enabling count operation. The current value of TAUJnCDRm is written to TAUJnCNTm and the counters start to count down from these values. INTTAUJnIm is generated on the master channel and TAUJTTOUTm (slave) is set or reset to realize the PWM output.

- Master channel:

  When the counter of the master channel reaches 0000 0000_H, pulse cycle time has elapsed and INTTAUJnIm is generated. The TAUJnCDRm value is realized to TAUJnCNTm, and the counter counts down.

- Slave channel(s):

  The INTTAUJnIm of the master channel triggers the counter of the slave channel(s). The current value of TAUJnCDRm (slave) is written to TAUJnCNTm (slave) and the counter starts to count down from this value. The TAUJTTOUTm signal is set, to the active level.
  When the counter reaches 0000 0000_H, i.e. duty time has elapsed, INTTAUJnIm is generated and the TAUJTTOUTm signal is set to the inactive level. The counter returns to FFFF $FFFF_H$ and awaits the next INTTAUJnIm of the master channel, and thus the start of the next pulse cycle.

The counter can be stopped by setting TAUJnTT.TAUJnTTm to 1 for the master and slave channel(s), which in turn sets TAUJnTE.TAUJnTEm to 0. TAUJnCNTm and TAUJTTOUTm of master and slave channel(s) stop but retain their values. The counters can be restarted by setting TAUJnTS.TAUJnTSm to 1.

Conditions

Simultaneous rewrite can be used with this function. Please refer to **Section 18.4.3, Simultaneous Rewrite**.

(2) Equations

Pulse cycle = (TAUJnCDRm (master) + 1) × count clock cycle

Duty cycle [%] = (TAUJnCDRm (slave)/(TAUJnCDRm (master) +1)) × 100

– Duty cycle = 0%
  TAUJnCDRm (slave) = 0000 0000_H
– Duty cycle = 100%
  TAUJnCDRm (slave) ≥ TAUJnCDRm (master) + 1

(3) Block diagram and general timing diagram

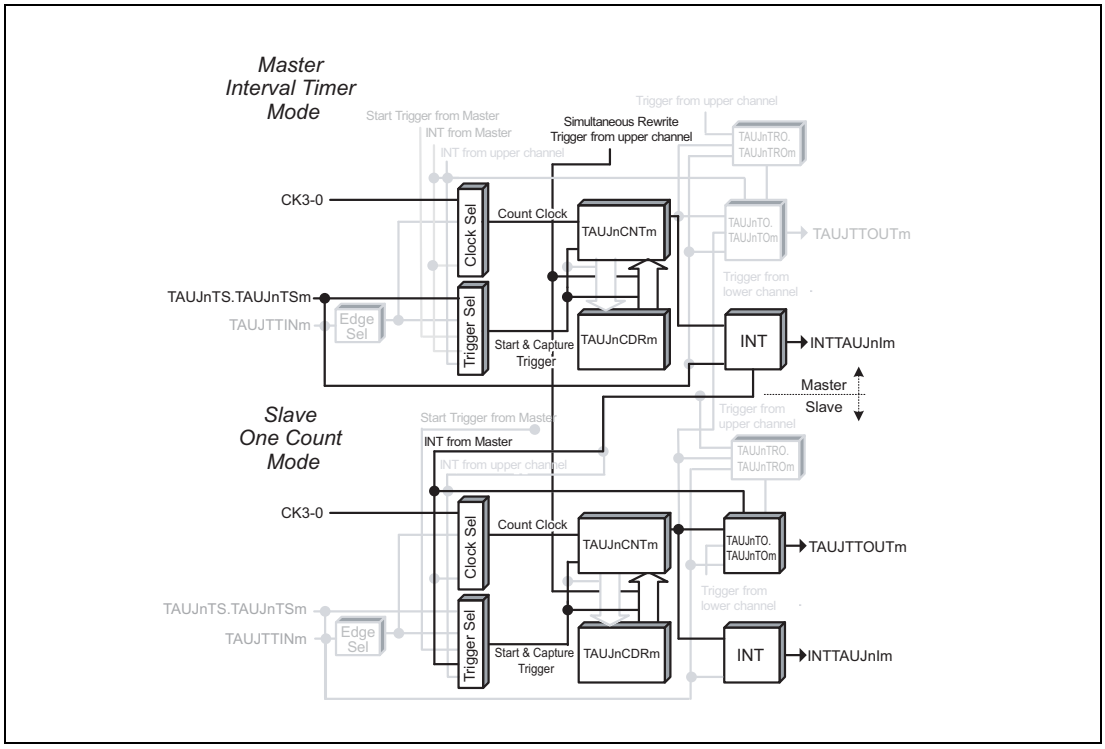


Figure 18.41 Block Diagram for PWM Output Function

The following settings apply to the general timing diagram:

- Slave channel: Positive logic (TAUJnTOL.TAUJnTOLm = 0)

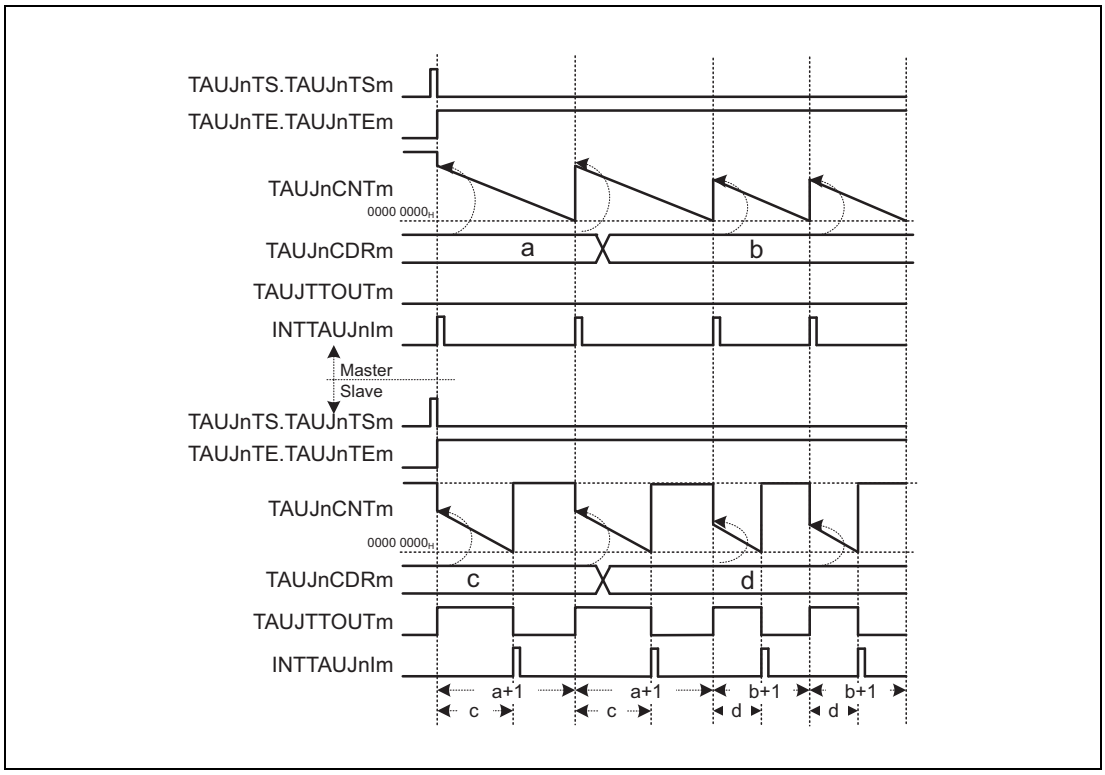


Figure 18.42 General Timing Diagram for PWM Output Function

NOTES

- The interval between the slave channel starting to count and an interrupt being generated is the value of corresponding TAUJnCDRm, whereas for the master channel the interval is the corresponding TAUJnCDRm + 1.
- TAUJTTOUTm of the slave channel rises with a delay of one clock count after the rise of INTTAUJnIm of the master channel.

(4) Register settings for the master channel

(a) TAUJnCMORm for the master channel

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.61 Contents of TAUJnCMORm Register of Master Channel for PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUJnCKS[1:0] bit of the master and slave channel(s) must be identical.
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	1: Channel is master channel
10 to 8	TAUJnSTS[2:0]	000: Trigger the counter using software.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0000: Interval timer mode
0	TAUJnMD0	1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm for the master channel

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.62 Contents of TAUJnCMURm Register of Master Channel for PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Not used so set to 00

(c) Channel output mode for the master channel

The channel output mode is not used by this function. However, it can be used by other functions or in Independent Channel Output Mode Controlled by Software.

(d) Simultaneous rewrite for the master channel

The simultaneous rewrite settings of the master and slave channel must be identical.

Table 18.63 Simultaneous Rewrite Settings for the Master Channel of the PWM Output Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	1: Enables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: The simultaneous rewrite trigger signal is generated when the master channel starts counting

(5) Register settings for the slave channel(s)

(a) TAUJnCMORm for the slave channel(s)

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TAUJnCKS[1:0]		TAUJnCCS[1:0]		TAUJn MAS	TAUJnSTS[2:0]			TAUJnCOS[1:0]		—	TAUJnMD[4:1]				TAUJn MD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 18.64 Contents of TAUJnCMORm Register of Slave Channel for PWM Output Function

Bit Position	Bit Name	Function
15, 14	TAUJnCKS[1:0]	These bits select the operation clock. 00: Prescaler output CK0 01: Prescaler output CK1 10: Prescaler output CK2 11: Prescaler output CK3 The value of the TAUJnCKS[1:0] bit of the master and slave channel(s) must be identical.
13, 12	TAUJnCCS[1:0]	00: Uses the operation clock as a counter clock.
11	TAUJnMAS	0: Channel is slave channel
10 to 8	TAUJnSTS[2:0]	100: INTTAUJnIm of master channel is a start trigger.
7, 6	TAUJnCOS[1:0]	00: Not used, so set to 00
5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4 to 1	TAUJnMD[4:1]	0100: One-count mode
0	TAUJnMD0	1: INTTAUJnIm is generated at the start of operation.

(b) TAUJnCMURm for the slave channel(s)

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TAUJnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 18.65 Contents of TAUJnCMURm Register of Slave Channel for PWM Output Function

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TAUJnTIS[1:0]	00: Not used so set to 00

(c) Channel output mode for the slave channel(s)

Table 18.66 Control Bit Settings for Synchronous Channel Output Mode 1

Bit name	Setting
TAUJnTOE.TAUJnTOEm	1: Enables independent channel output mode
TAUJnTO.TAUJnTOm	0: Low level 1: High level
TAUJnTOM.TAUJnTOMm	1: Synchronous channel operation
TAUJnTOC.TAUJnTOCm	0: Operating mode 1
TAUJnTOL.TAUJnTOLm	0: Positive logic 1: Negative logic

(d) Simultaneous rewrite for the slave channel(s)

The simultaneous rewrite settings of the master and slave channel must be identical.

Table 18.67 Simultaneous Rewrite Settings for the Slave Channel of the PWM Output Function

Bit name	Setting
TAUJnRDE.TAUJnRDEm	1: Enables simultaneous rewrite
TAUJnRDM.TAUJnRDMm	0: The simultaneous rewrite trigger signal is generated when the master channel starts counting

(6) Operating procedure for PWM output function

Table 18.68 Operating Procedure for PWM Output Function

	Operation	Status of TAUJn
Initial channel setting	Master channel: set the TAUJnCMORm and TAUJnCMURm registers and the channel output mode as described in **Section (4), Register settings for the master channel**. Slave channel: set the TAUJnCMORm and TAUJnCMURm registers and the channel output mode as described in **Section (5), Register settings for the slave channel(s)**. Set the values of the TAUJnCDRm registers of all channels	Channel operation is stopped.
Start operation	Set TAUJnTS.TAUJnTSm of the master and slave channels to 1 simultaneously. TAUJnTS.TAUJnTSm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm (master and slave channels) is set to 1 and the counters of the master and slave channels start. INTTAUJnIm is generated on the master channel and TAUJTTOUTm (slave) is set.
During operation	TAUJnCDRm can be changed at any time. TAUJnCNTm and TAUJnRSF.TAUJnRSFm can be read at any time. TAUJnRDT.TAUJnRDTm can be changed during operation.	TAUJnCNTm of the master channel loads TAUJnCDRm and counts down. When the counter reaches 0000 0000_H: • INTTAUJnIm (master) is generated • TAUJnCNTm (master) loads the TAUJnCDRm value and continues count operation • TAUJnCNTm (slave) loads the TAUJnCDRm value and counts down • TAUJTTOUTm (slave) is set to the active level. When TAUJnCNTm (slave) reaches 0000 0000_H: • INTTAUJnIm (slave) is generated • TAUJTTOUTm (slave) is set to the inactive level.
Stop operation	Set TAUJnTT.TAUJnTTm of the master and slave channels to 1 simultaneously. TAUJnTT.TAUJnTTm is a trigger bit, so it is automatically cleared to 0.	TAUJnTE.TAUJnTEm is cleared to 0 and the counter stops. TAUJnCNTm and TAUJTTOUTm stop and retain their current values.

Restart

(7) Specific timing diagrams

(a) Duty cycle = 0%

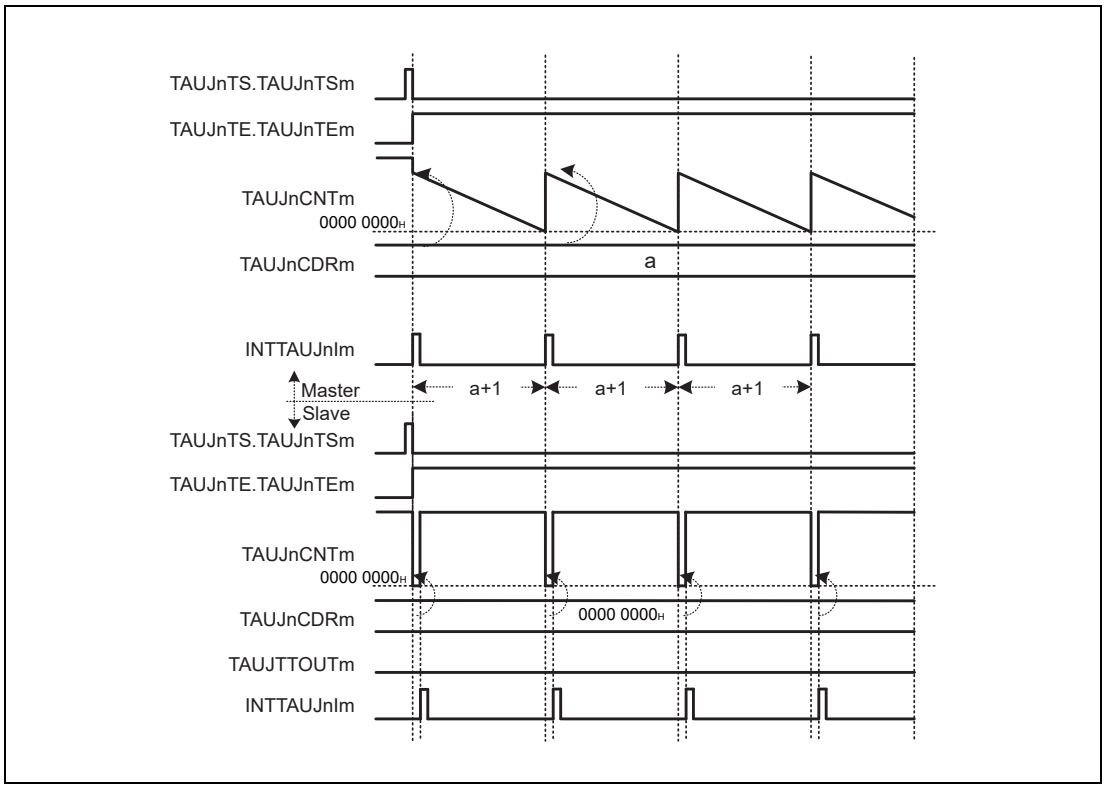


Figure 18.43 TAUJnCDRm (slave) = 0000 0000$_H$, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

- Every time the master channel generates an interrupt (INTTAUJnIm), 0000 0000$_H$ is written to TAUJnCNTm (slave). Therefore, TAUJnCNTm (slave) cannot start to count and TAUJTTOUTm remains at not active state.
- The value of TAUJnCDRm is loaded into TAUJnCNTm (slave) to generate an interrupt.

(b) Duty cycle = 100%

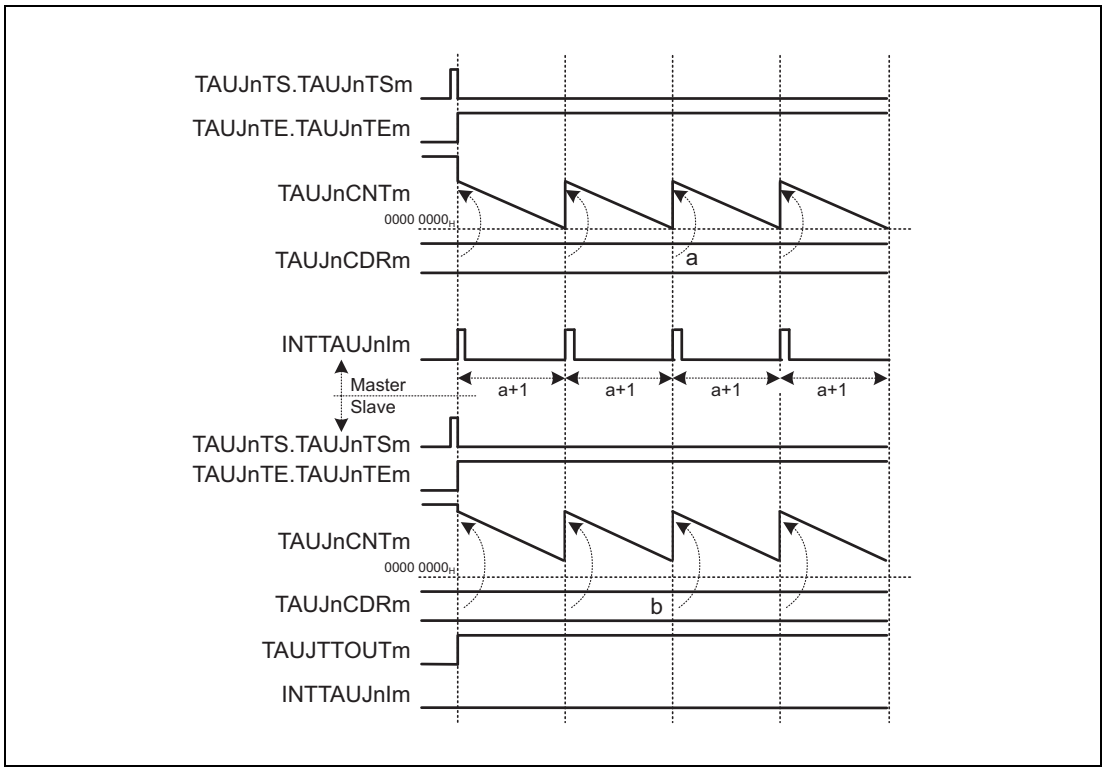


Figure 18.44 TAUJnCDRm (Slave) ≥ TAUJnCDRm (Master) + 1, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

If the TAUJnCDRm (slave) value is greater than the TAUJnCDRm (master) value, no interrupt occurs because the counter of the slave channel does not reach 0000 0000_H. TAUJTTOUTm remains active.

(c) Operation stop and restart

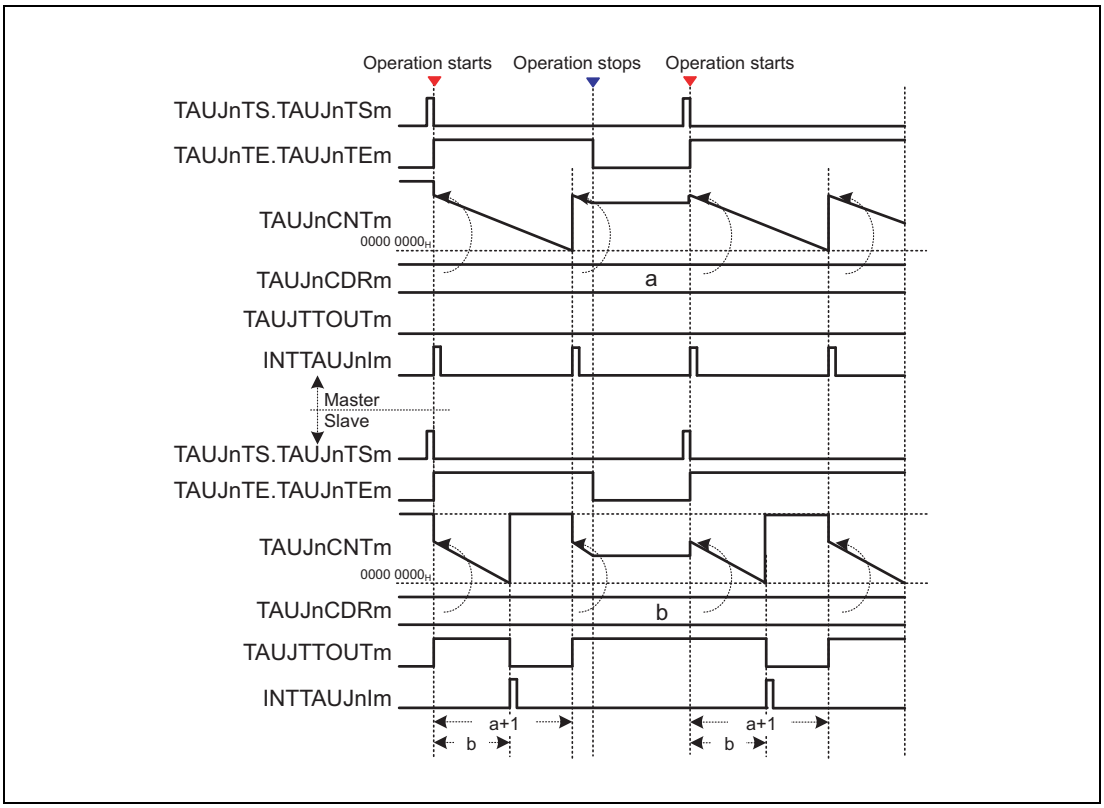


Figure 18.45 Stop and Restart Operation, Positive Logic (TAUJnTOL.TAUJnTOLm (slave) = 0)

- The counter can be stopped by setting TAUJnTT.TAUJnTTm of master and slave channels to 1. This sets TAUJnTE.TAUJnTEm to 0.
- TAUJnCNTm and TAUJTTOUTm of every channel stop and retain their current values. No interrupt occurs.
- The counter can be restarted by setting TAUJnTS.TAUJnTSm of master and slave channels to 1. TAUJnCDRm value of master and slave channels is loaded into TAUJnCNTm. The counter starts to count down from this value.

Section 19 Motor Control Timer (TSG3)

This section contains a generic description of the motor control timer (TSG3).

The first part of the section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the TSG3.

19.1 Features of RH850/C1x TSG3

19.1.1 Number of Units

This LSI has the following number of units of TSG3.

Table 19.1 Units

Product	RH850/C1H	RH850/C1M
Number of Units	2	2
Name	TSG3n (n = 0, 1)	TSG3n (n = 0, 1)

Table 19.2 Index

Index	Meaning
n	Throughout this section, the individual TSG3 units are identified by the index “n” (n = 0, 1), for example, TSG3nCTL0 for the TSG3n control register 0.
m, k	Throughout this section, the variables used for description are indicated by the letter “m” or “k”, for example, TSG3nCMPmE is a non-specified compare register.

19.1.2 Register Base Address

TSG3 base addresses are listed in the following table.

TSG3 register addresses are given as offsets from the base addresses in general.

Table 19.3 Register Base Address

Base Address Name	Base Address
<TSG30_base>	FFE7 0000_H
<TSG31_base>	FFE7 1000_H

19.1.3 Clock Supply

Clock supply by and to TSG3 is listed in the following table.

Table 19.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
TSG3n	PCLK	CLKC_HSB (Unmodulated high-speed peripheral clock)

19.1.4 Interrupt Requests

TSG3 interrupt requests are listed in the following table.

Table 19.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number	DTS Trigger Number
TSG30				
INTTSG30I0	TSG30 compare match interrupt 0	164	—	—
INTTSG30I1	TSG30 compare match interrupt 1	136	—	—
INTTSG30I2	TSG30 compare match interrupt 2	137	—	—
INTTSG30I3	TSG30 compare match interrupt 3	138	—	—
INTTSG30I4	TSG30 compare match interrupt 4	139	—	—
INTTSG30I5	TSG30 compare match interrupt 5	140	—	—
INTTSG30I6	TSG30 compare match interrupt 6	141	—	—
INTTSG30I7	TSG30 compare match interrupt 7	142	—	—
INTTSG30I8	TSG30 compare match interrupt 8	143	—	—
INTTSG30I9	TSG30 compare match interrupt 9	144	—	—
INTTSG30I10	TSG30 compare match interrupt 10	145	—	—
INTTSG30I11	TSG30 compare match interrupt 11	146	56	92
INTTSG30I12	TSG30 compare match interrupt 12	147	57	93
INTTSG30IPEK	TSG30 peak interrupt	165	58	94
INTTSG30IVLY	TSG30 trough interrupt	166	59	95
INTTSG30IER	TSG30 error interrupt	148	—	—
INTTSG30IWN	TSG30 warning interrupt	149	—	—
TSG31				
INTTSG31I0	TSG31 compare match interrupt 0	167	—	—
INTTSG31I1	TSG31 compare match interrupt 1	150	—	—
INTTSG31I2	TSG31 compare match interrupt 2	151	—	—
INTTSG31I3	TSG31 compare match interrupt 3	152	—	—
INTTSG31I4	TSG31 compare match interrupt 4	153	—	—
INTTSG31I5	TSG31 compare match interrupt 5	154	—	—
INTTSG31I6	TSG31 compare match interrupt 6	155	—	—
INTTSG31I7	TSG31 compare match interrupt 7	156	—	—
INTTSG31I8	TSG31 compare match interrupt 8	157	—	—
INTTSG31I9	TSG31 compare match interrupt 9	158	—	—
INTTSG31I10	TSG31 compare match interrupt 10	159	—	—
INTTSG31I11	TSG31 compare match interrupt 11	160	60	96
INTTSG31I12	TSG31 compare match interrupt 12	161	61	97
INTTSG31IPEK	TSG31 peak interrupt	168	62	98
INTTSG31IVLY	TSG31 trough interrupt	169	63	99
INTTSG31IER	TSG31 error interrupt	162	—	—
INTTSG31IWN	TSG31 warning interrupt	163	—	—

19.1.5 Reset Sources

TSG3 reset sources are listed in the following table.

Table 19.6 Reset Source

Unit Name	Reset Source
TSG3n	All reset sources

19.1.6 External Input/Output Signals

External input/output signals of TSG3 are listed in the following table.

Table 19.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
TSG30		
TSG30PTSI0-TSG30PTSI2	External pattern input	ENCA0E0, ENCA0E1, ENCA0EC*1
TSG30O0-TSG30O7	Timer output	TSG30O0-TSG30O7
TSG31		
TSG31PTSI0-TSG31PTSI2	External pattern input	ENCA1E0, ENCA1E1, ENCA1EC*1
TSG31O0-TSG31O7	Timer output	TSG31O0-TSG31O7

Note 1. For the C1x products, external pattern input is shared with ENCAn input. For the detail specification, see **Section 23.2.2.13, PIC1ATSGHALLSEL — Hall Sensor Input Select Register**.

19.2 Overview

19.2.1 Functional Overview

The TSG3n is an 18-bit timer counter with various motor control functions.

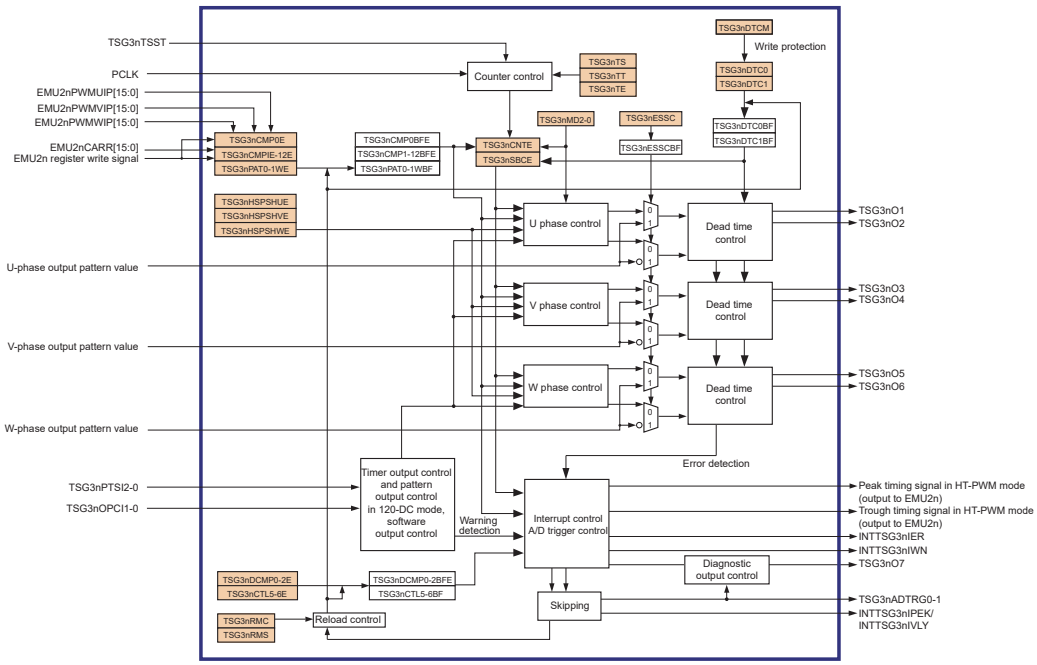
- Count clock resolution: Minimum 12.5 ns (count clock = 80 MHz)
- Operating mode corresponding to various motor control methods
- Compare registers with reload buffer
- 10-bit dead time counter
  - Dead time counter with reload buffer
  - Independent dead time can be set for positive to inverse phase change and inverse to positive phase change.
- A/D conversion trigger signal generation
  - Three A/D conversion trigger signals can be generated by the compare registers TSG3nDCMP0E, TSG3nDCMP1E, and TSG3nDCMP2E.
  - Skipping function of A/D conversion trigger signals TSG3nADTRG0 and TSG3nADTRG1 can be set independently. The skipping ratio can be selected among 1/1, 1/2, 1/4, and 1/8.
  - The dedicated pin (TSG3nO7) can be used to output the toggle or diagnostic signal set by the TSG3nADTRG0 signal and reset by the TSG3nADTRG1 signal.
- Interrupt skipping
  - Skipping rate: 1/1 to 1/32
- Forced output stop function
  - Using the timer option (TAPA) function allows the high impedance control of the TSG3nO1 to TSG3nO6 pin outputs.
- Compare value setting
  - Reload (simultaneous rewrite) or anytime rewrite can be selected.
- Reload mode
  - Writing to TSG3nCMP1E enables reload (the reload request flag (TSG3nRSF) is set), and allows simultaneous transfer of the values of multiple registers.
  - Data can be transferred at peak/trough/peak or trough reload timing
  - Reload request flag (TSG3nRSF)
  - Register address assignment allowing DMA transfer
  - Reload skipping
- HT-PWM mode
  - 0 to 100% PWM duty cycle output is possible (with possible dead time reduction).
  - The LSB in the compare register can be used to append an additional pulse to the PWM output during count up, thus improving the output resolution without extra load on software.

- 120-DC control
  - Semi-automatic cruise function (trigger signal can be generated by an offset in conjunction with two-phase encoder, three-phase encoder, or ENCA).
- Three-phase encoder function (hall sensor signals can be input).
- Active level of the output pins TSG3nO1 to TSG3nO6 can be set individually.
- Fail-safe function (warning interrupt or error interrupt can be generated)
  - Simultaneous active output detect function for positive and inverse phases.
  - Abnormal input detection function of the three-phase encoder
- Direct transfer of carrier-cycle setting and PWM duty setting from EMU2.
- Output selectable between rectangular waveform output from EMU2 and PWM generate by TSG3.

NOTE

In this section, active level is indicated as high level.

19.2.2 Block Diagram



Note: EMU2nPWMUIP[15:0]: EMU2n U-phase compare value register (input from EMU2n)
EMU2nPWMVIP[15:0]: EMU2n V-phase compare value register (input from EMU2n)
EMU2nPWMWIP[15:0]: EMU2n W-phase compare value register (input from EMU2n)
EMU2nCARR[15:0]: EMU2n carrier cycle register (input from EMU2n)
EMU2n register write signal: Write timing signal of the above register value (input from EMU2n)
U-phase output pattern value: Input from EMU2n rectangular waveform generator
V-phase output pattern value: Input from EMU2n rectangular waveform generator
W-phase output pattern value: Input from EMU2n rectangular waveform generator

Figure 19.1 TSG3n Block Diagram

- TSG3nTSST: Simultaneous start trigger (input from PIC1A)

19.3 List of Registers

19.3.1 List of Registers

TSG3n registers are listed in the following table.

For <TSG3n_base>, see **Section 19.1.2, Register Base Address.**

Table 19.8 List of Registers (1/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n control register0	TSG3nCTL0	<TSG3n_base> + 208_H	Disabled
TSG3n	TSG3n control register1	TSG3nCTL1	<TSG3n_base> + $20C_H$	Disabled
TSG3n	TSG3n control register 3	TSG3nCTL3	<TSG3n_base> + 004_H	Disabled
TSG3n	TSG3n control register 4	TSG3nCTL4	<TSG3n_base> + $07C_H$	Enabled
TSG3n	TSG3n control register 5	TSG3nCTL5	<TSG3n_base> + 008_H	Disabled
TSG3n	TSG3n control register 6	TSG3nCTL6	<TSG3n_base> + $00C_H$	Disabled
TSG3n	TSG3n control register 7	TSG3nCTL7	<TSG3n_base> + 218_H	Disabled
TSG3n	TSG3n control register 8	TSG3nCTL8	<TSG3n_base> + $21C_H$	Disabled
TSG3n	TSG3n I/O control register 0	TSG3nIOC0	<TSG3n_base> + 200_H	Disabled
TSG3n	TSG3n I/O control register 1	TSG3nIOC1	<TSG3n_base> + 204_H	Disabled
TSG3n	TSG3n I/O control register 2	TSG3nIOC2	<TSG3n_base> + 000_H	Disabled
TSG3n	TSG3n I/O control register 3	TSG3nIOC3	<TSG3n_base> + 074_H	Enabled
TSG3n	TSG3n status register 0	TSG3nSTR0	<TSG3n_base> + 010_H	Disabled
TSG3n	TSG3n status register 1	TSG3nSTR1	<TSG3n_base> + 014_H	Disabled
TSG3n	TSG3n status register 2	TSG3nSTR2	<TSG3n_base> + 018_H	Disabled
TSG3n	TSG3n status clear trigger register	TSG3nSTC	<TSG3n_base> + $01C_H$	Disabled
TSG3n	TSG3n option register 0	TSG3nOPT0	<TSG3n_base> + 020_H	Disabled
TSG3n	TSG3n option register 1	TSG3nOPT1	<TSG3n_base> + 024_H	Disabled
TSG3n	TSG3n option register 2	TSG3nOPT2	<TSG3n_base> + $03C_H$	Disabled
TSG3n	TSG3n option 2 buffer register	TSG3nOPT2BF	<TSG3n_base> + $0CC_H$	Disabled
TSG3n	TSG3n trigger register 0	TSG3nTRG0	<TSG3n_base> + 030_H	Disabled
TSG3n	TSG3n trigger register 1	TSG3nTRG1	<TSG3n_base> + 034_H	Disabled
TSG3n	TSG3n trigger register 2	TSG3nTRG2	<TSG3n_base> + 038_H	Disabled
TSG3n	TSG3n counter read buffer register	TSG3nCNT	<TSG3n_base> + 028_H	Disabled
TSG3n	TSG3n bit extended counter read buffer register	TSG3nCNTE	<TSG3n_base> + $1A0_H$	Disabled
TSG3n	TSG3n sub-counter read buffer register	TSG3nSBC	<TSG3n_base> + $02C_H$	Disabled
TSG3n	TSG3n bit extended sub-counter read buffer register	TSG3nSBCE	<TSG3n_base> + $1A4_H$	Disabled
TSG3n	TSG3n compare register 0	TSG3nCMP0	<TSG3n_base> + 058_H	Enabled
TSG3n	TSG3n bit extended compare register 0	TSG3nCMP0E	<TSG3n_base> +$14C_H$	Enabled
TSG3n	TSG3n compare register 1, 2	TSG3nCMP1W	<TSG3n_base> + 040_H	Enabled
TSG3n	TSG3n compare register 5, 6	TSG3nCMP5W	<TSG3n_base> + 044_H	Enabled
TSG3n	TSG3n compare register 9, 10	TSG3nCMP9W	<TSG3n_base> + 048_H	Enabled
TSG3n	TSG3n compare register 3, 4	TSG3nCMP3W	<TSG3n_base> + $04C_H$	Enabled
TSG3n	TSG3n compare register 7, 8	TSG3nCMP7W	<TSG3n_base> + 050_H	Enabled
TSG3n	TSG3n compare register 11, 12	TSG3nCMP11W	<TSG3n_base> + 054_H	Enabled
TSG3n	TSG3n compare register 1	TSG3nCMP1	<TSG3n_base> + 080_H	Enabled
TSG3n	TSG3n compare register 2	TSG3nCMP2	<TSG3n_base> + 084_H	Enabled
TSG3n	TSG3n compare register 3	TSG3nCMP3	<TSG3n_base> + 098_H	Enabled

Table 19.8 List of Registers (2/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n compare register 4	TSG3nCMP4	<TSG3n_base> + $09C_H$	Enabled
TSG3n	TSG3n compare register 5	TSG3nCMP5	<TSG3n_base> + 088_H	Enabled
TSG3n	TSG3n compare register 6	TSG3nCMP6	<TSG3n_base> + $08C_H$	Enabled
TSG3n	TSG3n compare register 7	TSG3nCMP7	<TSG3n_base> + $0A0_H$	Enabled
TSG3n	TSG3n compare register 8	TSG3nCMP8	<TSG3n_base> + $0A4_H$	Enabled
TSG3n	TSG3n compare register 9	TSG3nCMP9	<TSG3n_base> + 090_H	Enabled
TSG3n	TSG3n compare register 10	TSG3nCMP10	<TSG3n_base> + 094_H	Enabled
TSG3n	TSG3n compare register 11	TSG3nCMP11	<TSG3n_base> + $0A8_H$	Enabled
TSG3n	TSG3n compare register 12	TSG3nCMP12	<TSG3n_base> + $0AC_H$	Enabled
TSG3n	TSG3n bit extended compare register 1	TSG3nCMP1E	<TSG3n_base> + $17C_H$	Enabled
TSG3n	TSG3n bit extended compare register 2	TSG3nCMP2E	<TSG3n_base> + 178_H	Enabled
TSG3n	TSG3n bit extended compare register 3	TSG3nCMP3E	<TSG3n_base> + 164_H	Enabled
TSG3n	TSG3n bit extended compare register 4	TSG3nCMP4E	<TSG3n_base> + 160_H	Enabled
TSG3n	TSG3n bit extended compare register 5	TSG3nCMP5E	<TSG3n_base> + 174_H	Enabled
TSG3n	TSG3n bit extended compare register 6	TSG3nCMP6E	<TSG3n_base> +170_H	Enabled
TSG3n	TSG3n bit extended compare register 7	TSG3nCMP7E	<TSG3n_base> + $15C_H$	Enabled
TSG3n	TSG3n bit extended compare register 8	TSG3nCMP8E	<TSG3n_base> + 158_H	Enabled
TSG3n	TSG3n bit extended compare register 9	TSG3nCMP9E	<TSG3n_base> + $16C_H$	Enabled
TSG3n	TSG3n bit extended compare register 10	TSG3nCMP10E	<TSG3n_base> + 168_H	Enabled
TSG3n	TSG3n bit extended compare register 11	TSG3nCMP11E	<TSG3n_base> + 154_H	Enabled
TSG3n	TSG3n bit extended compare register 12	TSG3nCMP12E	<TSG3n_base> + 150_H	Enabled
TSG3n	TSG3n diagnostic output compare register 0, 1	TSG3nDCMP0W	<TSG3n_base> + $05C_H$	Enabled
TSG3n	TSG3n diagnostic output compare register 2	TSG3nDCMP2	<TSG3n_base> + 060_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 0	TSG3nDCMP0E	<TSG3n_base> + 148_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 1	TSG3nDCMP1E	<TSG3n_base> + 144_H	Enabled
TSG3n	TSG3n bit extended diagnostic output compare register 2	TSG3nDCMP2E	<TSG3n_base> + 140_H	Enabled
TSG3n	TSG3n pattern register 0	TSG3nPAT0W	<TSG3n_base> + 064_H	Enabled
TSG3n	TSG3n pattern register 1	TSG3nPAT1W	<TSG3n_base> + 068_H	Enabled
TSG3n	TSG3n dead time control register 0	TSG3nDTC0W	<TSG3n_base> + $06C_H$	Enabled
TSG3n	TSG3n dead time control register 1	TSG3nDTC1W	<TSG3n_base> + 070_H	Enabled
TSG3n	TSG3n HT-PWM U phase compare register	TSG3nCMPU	<TSG3n_base> + $0B0_H$	Enabled
TSG3n	TSG3n HT-PWM V phase compare register	TSG3nCMPV	<TSG3n_base> + $0B4_H$	Enabled
TSG3n	TSG3n HT-PWM W phase compare register	TSG3nCMPW	<TSG3n_base> + $0B8_H$	Enabled
TSG3n	TSG3n bit extended HT-PWM U phase compare register	TSG3nCMPUE	<TSG3n_base> + 188_H	Enabled
TSG3n	TSG3n bit extended HT-PWM V phase compare register	TSG3nCMPVE	<TSG3n_base> + 184_H	Enabled
TSG3n	TSG3n bit extended HT-PWM W phase compare register	TSG3nCMPWE	<TSG3n_base> + 180_H	Enabled
TSG3n	TSG3n SP-PWM U phase active width register	TSG3nUPW	<TSG3n_base> + $0BC_H$	Enabled
TSG3n	TSG3n SP-PWM V phase active width register	TSG3nVPW	<TSG3n_base> + $0C0_H$	Enabled
TSG3n	TSG3n SP-PWM W phase active width register	TSG3nWPW	<TSG3n_base> + $0C4_H$	Enabled
TSG3n	TSG3n bit extended SP-PWM U phase active width register	TSG3nUPWE	<TSG3n_base> + 198_H	Enabled
TSG3n	TSG3n bit extended SP-PWM V phase active width register	TSG3nVPWE	<TSG3n_base> + 194_H	Enabled
TSG3n	TSG3n bit extended SP-PWM W phase active width register	TSG3nWPWE	<TSG3n_base> + 190_H	Enabled

Table 19.8 List of Registers (3/3)

Module Name	Register Name	Symbol	Address	Reload
TSG3n	TSG3n HSP-PWM W phase shift register	TSG3nHSPSHWE	<TSG3n_base> + 120_H	Enabled
TSG3n	TSG3n HSP-PWM V phase shift register	TSG3nHSPSHVE	<TSG3n_base> + 124_H	Enabled
TSG3n	TSG3n HSP-PWM U phase shift register	TSG3nHSPSHUE	<TSG3n_base> + 128_H	Enabled
TSG3n	TSG3n HSP-PWM W phase shift register	TSG3nHSPCMWE	<TSG3n_base> + $12C_H$	Enabled
TSG3n	TSG3n HSP-PWM V phase compare register	TSG3nHSPCMVE	<TSG3n_base> + 130_H	Enabled
TSG3n	TSG3n HSP-PWM U phase compare register	TSG3nHSPCMUE	<TSG3n_base> + 134_H	Enabled
TSG3n	TSG3n dead timer protection register	TSG3nDTPR	<TSG3n_base> + 210_H	Disabled

19.3.2 TSG3nCTL0 — TSG3n Control Register 0

This register specifies the pulse width for the diagnostic output and operating mode of the TSG3n.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 208_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TSG3nDWD	—	TSG3nMD[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R/W	R/W

Table 19.9 TSG3nCTL0 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TSG3nDWD	Selects the pulse width for the diagnostic output. 0: The output pulse width is set to 8 clocks. 1: The output pulse width is set to 16 clocks. The setting of this bit is valid when diagnostic output is enabled (TSG3nIOC1.TSG3nTGS = 1).
3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	TSG3nMD[2:0]	Selects timer mode

TSG3n MD2	TSG3n MD1	TSG3n MD0	Timer Mode
0	0	0	PWM mode
0	0	1	HT-PWM mode (HT-PWM)
0	1	0	Shift pulse PWM mode (SP-PWM)
0	1	1	120-DC mode
1	0	0	High-accuracy shift pulse PWM mode (HSP-PWM)
Other than above			Setting prohibited

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer

19.3.3 TSG3nCTL1 — TSG3n Control Register 1

This register controls the flags of TSG3n.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $20C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nTBA2	TSG3nTBA1	TSG3nTBA0	TSG3nPPC	TSG3nPEC	TSG3nTDC	TSG3nNDC	TSG3nPRC	TSG3nPTC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.10 TSG3nCTL1 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	TSG3nTBA2	Enables or disables detection of the simultaneous active states of the TSG3nO5 and TSG3nO6 pins. 0: Disables detection of simultaneous active states of the TSG3nO5 and TSG3nO6 pins. 1: Enables detection of simultaneous active states of the TSG3nO5 and TSG3nO6 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA2 bits are 1, the positive phase and inverse phase simultaneous active state detection flag 2 (TSG3nTBF2) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
8	TSG3nTBA1	Enables or disables detection of the simultaneous active states of the TSG3nO3 and TSG3nO4 pins. 0: Disables detection of simultaneous active states of the TSG3nO3 and TSG3nO4 pins. 1: Enables detection of simultaneous active states of the TSG3nO3 and TSG3nO4 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA1bits are 1, the positive phase and inverse phase simultaneous active state detection flag 1 (TSG3nTBF1) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
7	TSG3nTBA0	Enables or disables detection of the simultaneous active states of the TSG3nO1 and TSG3nO2 pins. 0: Disables detection of simultaneous active states of the TSG3nO1 and TSG3nO2 pins. 1: Enables detection of simultaneous active states of the TSG3nO1 and TSG3nO2 pins. If the simultaneous active state is detected when the TSG3nIOC1.TSG3nEOC and TSG3nTBA0 bits are 1, the positive phase and inverse phase simultaneous active state detection flag 0 (TSG3nTBF0) is set to 1, and an error interrupt (INTTSG3nIER) is generated.
6	TSG3nPPC	Enables or disables detection of the pattern phase difference (TSG3nSTR2.TSG3nPPF) between the TSG3nPTSI2-0 and TSG3nOPF2-0. 0: Disables detection of I/O pattern difference. 1: Enables detection of I/O pattern difference
5	TSG3nPEC	Enables or disables detection of the pattern error (TSG3nSTR2.TSG3nPEF) of the TSG3nPTSI2-0 pins. 0: Disables detection of the pattern error of the TSG3nPTSI2-0 pins. 1: Enables detection of the pattern error of the TSG3nPTSI2-0 pins.

Table 19.10 TSG3nCTL1 Register Contents (2/2)

Bit Position	Bit Name	Function
4	TSG3nTDC	Enables or disables detection of the simultaneous trigger (TSG3nSTR2.TSG3nTDF) of the TSG3nOPCI0 and TSG3nOPCI1. 0: Disables detection of the simultaneous trigger of the TSG3nOPCI0 and TSG3nOPCI1. 1: Enables detection of the simultaneous trigger of the TSG3nOPCI0 and TSG3nOPCI1.
3	TSG3nNDC	Enables or disables detection of the noise generation (two or more pins change simultaneously) (TSG3nSTR2.TSG3nNDF) on the TSG3nPTSI2-0 pins. 0: Disables detection of the noise generation on the TSG3nPTSI2-0 pins. 1: Enables detection of the noise generation on the TSG3nPTSI2-0 pins.
2	TSG3nPRC	Enables or disables detection of the reversal of the pattern (TSG3nSTR2.TSG3nPRF) of the TSG3nPTSI2-0 pins. 0: Disables detection of the reversal of the pattern of the TSG3nPTSI2-0 pins. 1: Enables detection of the reversal of the pattern of the TSG3nPTSI2-0 pins.
1, 0	TSG3nPTC[1:0]	Enables or disables detection of an abnormal toggle (TSG3nSTR2.TSG3nPTF) of the TSG3nPTSI2-0 pins between TSG3nOPCI1 and TSG3nOPCI10 triggers

TSG3nPTC1	TSG3nPTC0	Function
0	0	Disables detection of an abnormal toggle of the TSG3nPTSI2-0 pins.
0	1	
1	0	Enables detection of an abnormal toggle of the TSG3nPTSI2-0 pins.
1	1	Enables detection of an abnormal toggle of the TSG3nPTSI2-0 pins. When an abnormal toggle is detected, the pattern output switch trigger is automatically switched from trigger switch to pattern switch (TSG3nPOT is switched from 1 to 0.)

CAUTIONS

1. If TSG3nDTC0 or TSG3nDTC1 is set to 0000_H (without dead time), the TSG3nTBA2-0 bits should be set to 0.
2. This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer

19.3.4 TSG3nCTL3 — TSG3n Control Register 3

This register selects the rewrite method of the compare registers.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 004_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nRIA	TSG3nRMC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 19.11 TSG3nCTL3 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	TSG3nRIA	Selects the reload timing of the compare register values. 0: The reload timing is set to peak reload timing (set by TSG3nCTL4.TSG3nPRE) and trough reload timing (set by TSG3nCTL4.TSG3nVRE). 1: The reload timing is set to peak interrupt timing and trough interrupt timing. The setting of this bit is valid in reload mode (TSG3nRMC = 0).
0	TSG3nRMC	Selects the transfer timing of the compare register values. 0: Reload mode (simultaneous rewrite) Writing to registers to be reloaded enables reloading and the register values are rewritten simultaneously at the next reload timing. Writing to any register other than registers to be reloaded does not enable reloading. For the register to be reloaded, see **Section 19.3.1, List of Registers**. 1: Anytime rewrite mode The compare registers are rewritten independently. Whenever a value is written to the compare register, the written value is reflected immediately. TSG3nRSF is cleared. Do not set TSG3nRMC to 1 when operated in 120-DC mode or in HSP-PWM mode.

19.3.5 TSG3nCTL4 — TSG3n Control Register 4

This register enables or disables generation of a peak interrupt and a trough interrupt, and the reload timing.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $07C_H$

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3n PRE	TSG3n VRE	TSG3n PIE	TSG3n VIE	TSG3nRCC[04:00]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.12 TSG3nCTL4 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	TSG3nPRE	Enables or disables the peak reload timing. 0: Disables reload operation at the peak timing of the 18-bit counter. 1: Enables reload operation at the peak timing of the 18-bit counter. • The peak reload timing means the peak timing of the 18-bit counter in HT-PWM mode and the clear timing of the 18-bit counter by compare match in any mode other than HT-PWM mode. • When the reload operation at the peak timing of the 18-bit counter is disabled (TSG3nPRE = 0), reload is not executed in any mode other than HT-PWM mode.
7	TSG3nVRE	Enables or disables the trough reload timing. 0: Disables reload operation at the trough timing of the 18-bit counter. 1: Enables reload operation at the trough timing of the 18-bit counter. The setting of this bit is valid only in HT-PWM mode.
6	TSG3nPIE	Enables or disables generation of a peak interrupt (INTTSG3nIPEK). 0: Disables generation of a peak interrupt (INTTSG3nIPEK) at the peak timing of the 18-bit counter. Interrupts are not skipped. 1: Enables generation of a peak interrupt (INTTSG3nIPEK) at the peak timing of the 18-bit counter. Interrupts are skipped.
5	TSG3nVIE	Enables or disables generation of a trough interrupt (INTTSG3nIVLY). 0: Disables generation of a trough interrupt (INTTSG3nIVLY) at the trough timing of the 18-bit counter. Interrupts are not skipped. 1: Enables generation of a trough interrupt (INTTSG3nIVLY) at the trough timing of the 18-bit counter. Interrupts are skipped. The setting of this bit is valid only in HT-PWM mode.

Table 19.12 TSG3nCTL4 Register Contents (2/2)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4 to 0</td><td>TSG3nRCC [04:00]</td><td>Specifies the skipping rate of the interrupts (INTTSG3nIPEK and INTTSG3nIVLY) and reload.
<table>
<tr><th>TSG3n RCC04</th><th>TSG3n RCC03</th><th>TSG3n RCC02</th><th>TSG3n RCC01</th><th>TSG3n RCC00</th><th>Skipping Rate</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Skipping disabled</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1/2</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1/3</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1/4</td></tr>
<tr><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td><td>:</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1/30</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1/31</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1/32</td></tr>
</table>
When a write access is made (including a write of the same value to TSG3nRCC04-TSG3nRCC00) to TSG3nCTL4 during timer operation (TSG3nSTR0.TSG3nTE = 1), the interrupt skipping counter is cleared</td></tr>
</table>

19.3.6 TSG3nCTL5 — TSG3n Control Register5

This register controls A/D conversion trigger output (TSG3nADTRG0).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> +008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nACC [01:00]		TSG3nAT09	TSG3nAT08	TSG3nAT07	TSG3nAT06	TSG3nAT05	TSG3nAT04	TSG3nAT03	TSG3nAT02	TSG3nAT01	TSG3nAT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.13 TSG3nCTL5 Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG3nACC [01:00]	Specifies the skipping rate of the A/D conversion trigger (TSG3nADTRG0). **TSG3nACC01 / TSG3nACC00 / Skipping Rate** 0 / 0 / Skipping disabled 0 / 1 / 1/2 1 / 0 / 1/4 1 / 1 / 1/8 When a write access is made (including a write of the same value to TSG3nACC01 and TSG3nACC00) to TSG3nCTL5 during timer operation (TSG3nSTR0.TSG3nTE = 1), the interrupt skipping counter is cleared.
9	TSG3nAT09	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the (peak) timing when the 18-bit sub-counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. • The TSG3nAT09 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT09 bit should be set to 0. • Do not set the TSG3nAT09 bit to 1 when TSG3nDTC0W is not 0000_H and TSG3nDTC1W is 0000_H. A/D conversion trigger is not generated at the peak timing of the 18-bit sub-counter even if set so.
8	TSG3nAT08	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the (trough) timing when the 18-bit sub-counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. • The TSG3nAT08 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT08 bit should be set to 0. • Do not set the TSG3nAT08 bit to 1 when TSG3nDTC0W is 0000_H and TSG3nDTC1W is not 0000_H. A/D conversion trigger is not generated at the trough timing of the 18-bit sub-counter even if set so.

Table 19.13 TSG3nCTL5 Register Contents (2/3)

Bit Position	Bit Name	Function
7	TSG3nAT07	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
6	TSG3nAT06	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value
5	TSG3nAT05	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
4	TSG3nAT04	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1Evalue. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value.
3	TSG3nAT03	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
2	TSG3nAT02	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E.
1	TSG3nAT01	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the timing (peak interrupt) when the 18-bit counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped.

Table 19.13 TSG3nCTL5 Register Contents (3/3)

Bit Position	Bit Name	Function
0	TSG3nAT00	Specifies generation of A/D conversion trigger (TSG3nADTRG0) at the timing (trough interrupt) when the 18-bit counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a trough interrupt (INTTSG3nIVLY) after being skipped. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.

19.3.7 TSG3nCTL6 — TSG3n Control Register 6

This register controls the A/D conversion trigger output (TSG3nADTRG1).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $00C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nACC [11:10]		TSG3n AT19	TSG3n AT18	TSG3n AT17	TSG3n AT16	TSG3n AT15	TSG3n AT14	TSG3n AT13	TSG3n AT12	TSG3n AT11	TSG3n AT10
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.14 TSG3nCTL6 Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG3nACC [11:10]	Specifies the skipping rate of the A/D conversion trigger (TSG3nADTRG1). **TSG3nACC11 / TSG3nACC10 / Skipping Rate** 0 / 0 / Skipping disabled 0 / 1 / 1/2 1 / 0 / 1/4 1 / 1 / 1/8 When a write access is made (including a write of the same value to TSG3nACC11 and TSG3nACC10) to TSG3nCTL6 during timer operation (TSG3nSTR0.TSG3nTE = 1), the skipping counter is cleared.
9	TSG3nAT19	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the (peak) timing when the 18-bit sub-counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the peak timing of the 18-bit sub-counter. • The TSG3nAT19 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT19 bit should be set to 0. • Do not set the TSG3nAT19 bit to 1 when TSG3nDTC0W is not 0000_H and TSG3nDTC1W is 0000_H. A/D conversion trigger is not generated at the peak timing of the 18-bit sub-counter even if set so.
8	TSG3nAT18	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the (trough) timing when the 18-bit sub-counter switches from decrementing to incrementing. 0: Disables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. 1: Enables generation of the A/D conversion trigger at the trough timing of the 18-bit sub-counter. • The TSG3nAT18 bit can be set to 1 only in HT-PWM mode. In other modes, the TSG3nAT18 bit should be set to 0. • Do not set the TSG3nAT18 bit to 1 when TSG3nDTC0W is 0000_H and TSG3nDTC1W is not 0000_H. A/D conversion trigger is not generated at the trough timing of the 18-bit sub-counter even if set so.

Table 19.14 TSG3nCTL6 Register Contents (2/3)

Bit Position	Bit Name	Function
7	TSG3nAT17	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation and the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP2E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
6	TSG3nAT16	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP2E value.
5	TSG3nAT15	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP1E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
4	TSG3nAT14	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP1E value.
3	TSG3nAT13	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during decrementation with the TSG3nDCMP0E value. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit should be set to 0.
2	TSG3nAT12	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E. 0: Disables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value. 1: Enables generation of the A/D conversion trigger at the match timing of the 18-bit counter value during incrementation with the TSG3nDCMP0E value.
1	TSG3nAT11	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the timing (peak interrupt) when the 18-bit counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIPEK) after being skipped.

Table 19.14 TSG3nCTL6 Register Contents (3/3)

Bit Position	Bit Name	Function
0	TSG3nAT10	Specifies generation of A/D conversion trigger (TSG3nADTRG1) at the timing (peak interrupt) when the 18-bit counter switches from incrementing to decrementing. 0: Disables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIVLY) after being skipped. 1: Enables generation of the A/D conversion trigger at the timing of a peak interrupt (INTTSG3nIVLY) after being skipped. This bit can be set to 1 only in HT-PWM mode. In other modes, this bit must be set to 0.

19.3.8 TSG3nCTL7 — TSG3n Control Register 7

This register sets the level of PWM output from the TSG3O1 to TSG3O6 pins at operation start (TSG3nTE is changed from 0 to 1) and at operation restart in SP-PWM mode.

This register can be written only when the SP-PWM mode is selected (TSG3nMD2-0 = 010) and when the operation is stopped (TSG3nTE = 0).

Do not rewrite this register when other modes are selected (PWM mode, HT-PWM mode, 120-DC mode, and HSP-PWM mode), or during operation (TSG3nTE = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 218_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSG3n SPSTL2	TSG3n SPSTL1	TSG3n SPSTL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 19.15 TSG3nCTL7 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2	TSG3nSPSTL2	SP-PWM Mode Start Level Control Bit 2 0: TSG3nO5 (W phase) is cleared and TSG3nO6 (WB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO5 (W phase) is set and TSG3nO6 (WB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.
1	TSG3nSPSTL1	SP-PWM Mode Start Level Control Bit 1 0: TSG3nO3 (V phase) is cleared and TSG3nO4 (VB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO3 (V phase) is set and TSG3nO4 (VB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.
0	TSG3nSPSTL0	SP-PWM Mode Start Level Control Bit 0 0: TSG3nO1 (U phase) is cleared and TSG3nO2 (UB phase) is set at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode. 1: TSG3nO3 (U phase) is set and TSG3nO1 (UB phase) is cleared at operation start (TSG3nTE changed from 0 to 1) or restart in SP-PWM mode.

NOTE

The settings of bits TSG3nSPSTL2 to TSG3nSPSTL0 affect output on the TSG3nO1 to TSG3nO6 pins when operation starts or is restarted. The set dead time is always inserted at these times.

19.3.9 TSG3nCTL8 — TSG3n Control Register 8

This register specifies timer output timing when input patterns are changed in 120-DC mode.

This register can be written only when 120-DC mode is selected (TSG3nMD2-0 = 011) and the timer is stopped (TSG3nTE = 0).

Do not rewrite this register in other modes (PWM mode, SP-PMW mode, HT-PWM mode, HSP-PWM mode) or while the timer is operating (TSG3nTE = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> +$21C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3n S120DCO
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 19.16 TSG3nCTL8 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TSG3n S120DCO	120-DC Mode Control Bit 0 0: When the input patterns are changed while 120-DC mode is selected, the main counter (TSG3nCNTE) is cleared and the change of input patterns is immediately reflected to timer output. 1: When the input patterns are changed while 120-DC mode is selected, the change of input patterns is reflected to timer output after a match of the main counter (TSG3nCNTE) with TSG3nCMP0E (from the next timer period).

CAUTION

When TSG3nS120DCO is set to 1 in 120DC mode, set the TSG3nOPT0.TSG3nSOC and TSG3nOPT02.TSG3nESSC bits to 0.

The settings of the TSG3nOPT0.TSG3nSTE and TSG3nOPT0.TSG3nPOT bits must not be changed while the timer is operating (TSG3nSTR0.TSG3nTE = 1).

19.3.10 TSG3nIOC0 — TSG3n I/O Control Register0

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> +200_H

Value after reset: $7E_H$

Bit	7	6	5	4	3	2	1	0
	—	TSG3nTOE6	TSG3nTOE5	TSG3nTOE4	TSG3nTOE3	TSG3nTOE2	TSG3nTOE1	—
Value after reset	0	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 19.17 TSG3nIOC0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTOE6 to TSG3nTOE1	Enables or disables control of TSG3nO6 to TSG3nO1 by rewriting TSG3nIOC2. When TSG3nTOEm = 1 (m = 1 to 6), rewriting TSG3nIOC2 is ignored. 0: Disabled 1: Enabled
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

19.3.11 TSG3nIOC1 — TSG3n I/O Control Register1

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 204_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TSG3nPTS	TSG3nEOC	TSG3nWOC	TSG3nTGS	TSG3nTOS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 19.18 TSG3nIOC1 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TSG3nPTS	Enables or disables output of the edge detection signal (TSG3nPTE) of TSG3nPTSI0 to TSG3nPTSI2 and two-phase encoder count signal (TSG3nPEC). 0: Disables output of the toggle signal by edge detection of TSG3nPTSI0 to TSG3nPTSI2. 1: Enables output of the toggle signal by edge detection of TSG3nPTSI0 to TSG3nPTSI2.
3	TSG3nEOC	Enables or disables detection of the error condition at the motor control. 0: Disables generation of an error interrupt (INTTSG3nIER). 1: Enables generation of an error interrupt (INTTSG3nIER). For details on controlling the error interrupt, see **Section 19.4.6.1, Error Interrupt Function**.
2	TSG3nWOC	Enables or disables detection of the warning condition at the motor control. 0: Disables generation of a warning interrupt (INTTSG3nIWN). 1: Enables generation of a warning interrupt (INTTSG3nIWN). For details on the controlling generation of warning interrupt, see **Section 19.4.6.2, Warning Interrupt Function**.
1	TSG3nTGS	Selects the A/D conversion trigger diagnostic output (TSG3nO7) signal. 0: Selects A/D conversion trigger output. 1: Selects diagnostic output.
0	TSG3nTOS	Selects the timer counter increment/decrement status output (TSG3nO0) signal. 0: Outputs the up/down count flag of the 18-bit counter. 1: Outputs the up/down count flag of the 18-bit sub-counter. • When TSG3nTOS is 0, the status of TSG3nSTR0.TSG3nCUF is output to TSG3nO0. When TSG3nTOS is 1, the status of TSG3nSTR0.TSG3nSUF is output to TSG3nO0. • The setting of this bit is valid only in HT-PWM mode.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

19.3.12 TSG3nIOC2 — TSG3n I/O Control Register2

This register controls the timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + 000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TSG3nOL6	TSG3nOL5	TSG3nOL4	TSG3nOL3	TSG3nOL2	TSG3nOL1	—	—	TSG3nTO6	TSG3nTO5	TSG3nTO4	TSG3nTO3	TSG3nTO2	TSG3nTO1	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 19.19 TSG3nIOC2 Register Contents

Bit Position	Bit Name	Function
15	Reserved	When read, the value after reset is read. When writing, write the value after reset.
14 to 9	TSG3nOL6 to TSG3nOL1	Specifies the active level of TSG3nO6 to TSG3nO1 outputs. 0: Active level is high level 1: Active level is low level
8 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTO6 to TSG3nTO1	Specifies the latch level of the output buffer of the TSG3nO6 to TSG3nO1. 0: Latch level of output buffer is low level 1: Latch level of output buffer is high level
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

When the counting is stopped (TSG3nSTR0.TSG3nTE = 0), the TSG3nO1-6 pins maintain their previous output states. The output level should be changed by setting the TSG3nIOC0.TSG3nTOEm bit to 0, using the TSG3nTOm bit. This register can be rewritten when TSG3nIOC0.TSG3nTOEm = 0 (m = 1 to 6).

NOTE

While control of TSG3nOm is enabled (TSG3nIOC0.TSG3nTOEm = 0) by rewriting TSG3nIOC2, TSG3nOLm and TSG3nTOm of TSG3nIOC2 select the output level for the corresponding TSG3nOm output as listed in the table below.

TSG3nOLm	TSG3nTOm	Output level of TSG3nOm
0	0	Low level
0	1	High level
1	0	High level
1	1	Low level

19.3.13 TSG3nIOC3 — TSG3n I/O Control Register3

This register controls timer output pins (TSG3nO1 to TSG3nO6).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 074_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TSG3n TOL6	TSG3n TOL5	TSG3n TOL4	TSG3n TOL3	TSG3n TOL2	TSG3n TOL1	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 19.20 TSG3nIOC3 Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG3nTOL6 to TSG3nTOL1	Controls the set/clear level of output. 0: Outputs the normal level. 1: Outputs the reversed level. Setting of this bit is reflected at the start of output. The change of the output level is reflected at the next compare match timing after the change.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

CAUTION

TSG3nTOL6 to 1 should be set to 0 in HT-PWM mode and HSP-PWM mode.

19.3.14 TSG3nSTR0 — TSG3n Status Register 0

This register consists of various status flags.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + 010_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nCUF	TSG3nSUF	TSG3nRSF	TSG3nTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.21 TSG3nSTR0 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3	TSG3nCUF	Indicates the count direction of the 18-bit counter. 0: The 18-bit counter is incremented. 1: The 18-bit counter is decremented. TSG3nCUF is valid only in HT-PWM mode. In other modes, it is invalid (TSG3nCUF = 0).
2	TSG3nSUF	Indicates the count direction of the 18-bit sub-counter. 0: The 18-bit sub-counter is incremented. 1: The 18-bit sub-counter is decremented. • TSG3nSUF detects counting of the 18-bit sub-counter from 0000_H to (TSG3nCMP0E value - 0002_H) as up-counting, and counting from the TSG3nCMP0E value to 0002_H as down-counting. • This bit is valid only in HT-PWM mode.
1	TSG3nRSF	Indicates whether there is a reload request. 0: No reload request or reload has completed. 1: There is a reload request. • This bit is valid only in TSG3nRMC = 0. • This bit indicates that the data to be transferred next is held. • This bit is set to 1 by writing to registers to be reloaded, and cleared to 0 when reload has completed. • When TSG3nRMC is changed from 0 to 1 in HT-PWM mode, TSG3nRSF is cleared to 0. For registers to be reloaded, see **Section 19.3.1, List of Registers**.
0	TSG3nTE	Indicates the TSG3n operation status. 0: TSG3n is stopped. 1: TSG3n is operating. This bit is set when TSG3nTRG0.TSG3nTS = 1, and cleared when TSG3nTRG1.TSG3nTT = 1.

19.3.15 TSG3nSTR1 — TSG3n Status Register 1

This register consists of various status flags.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + 014_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG3nTSF	TSG3nOPF[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.22 TSG3nSTR1 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read.
3	TSG3nTSF	Indicates the pattern change order of TSG3nPTSI0 to TSG3nPTSI2. 0: Indicates that patterns are input to TSG3nPTSI0 to TSG3nPTSI2 in the normal rotation pattern order 1: Indicates that patterns are input to TSG3nPTSI0 to TSG3nPTSI2 in the reverse rotation pattern order. Normal Rotation → Reverse Rotation ← TSG3nPTSI2-TSG3nPTSI0: [1,0,1] [1,0,0] [1,1,0] [0,1,0] [0,1,1] [0,0,1] Normal or reverse rotation can be detected from the first change of TSG3nPTSI0 to TSG3nPTSI2 after TSG3nTRG0.TSG3nTS has been set to 1. For details, see **Section 19.4.3.5 (b), Detection of Input Pattern Order**.
2 to 0	TSG3nOPF [2:0]	Indicates the output pattern of the timer output pins (TSG3nO1 to TSG3nO6).

19.3.16 TSG3nSTR2 — TSG3n Status Register 2

This register consists of various status flags.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + 018_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3n TBF2	TSG3n TBF1	TSG3n TBF0	TSG3n PPF	TSG3n PEF	TSG3n TDF	TSG3n NDF	TSG3n PRF	TSG3n PTF	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 19.23 TSG3nSTR2 Register Contents (1/3)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read.
9	TSG3nTBF2	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA2 is 1. 0: Positive phase (TSG3nO5) and inverse phase (TSG3nO6) are not active simultaneously. 1: Positive phase (TSG3nO5) and inverse phase (TSG3nO6) are active simultaneously. • TSG3nTBF2 is set to 1 when the simultaneous active state of the positive phase (TSG3nO5) and inverse phase (TSG3nO6) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF2 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR2. • The simultaneous active state is not detected when TSG3nTBA2 = 0.
8	TSG3nTBF1	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA1 is 1. 0: Positive phase (TSG3nO3) and inverse phase (TSG3nO4) are not active simultaneously. 1: Positive phase (TSG3nO3) and inverse phase (TSG3nO4) are active simultaneously. • TSG3nTBF1 is set to 1 when the simultaneous active state of the positive phase (TSG3nO3) and inverse phase (TSG3nO4) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF1 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR1. • The simultaneous active state is not detected when TSG3nTBA1 = 0.
7	TSG3nTBF0	Indicates whether the simultaneous active state of the positive phase and inverse phase is detected when TSG3nCTL1.TSG3nTBA0 is 1. 0: Positive phase (TSG3nO1) and inverse phase (TSG3nO2) are not active simultaneously. 1: Positive phase (TSG3nO1) and inverse phase (TSG3nO2) are active simultaneously. • TSG3nTBF0 is set to 1 when the simultaneous active state of the positive phase (TSG3nO1) and inverse phase (TSG3nO2) is detected, and an error interrupt (INTTSG3nIER) is generated. TSG3nTBF0 can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start) or by writing 1 to TSG3nSTC.TSG3nTBR0. • The simultaneous active state is not detected when TSG3nTBA0 = 0.

Table 19.23 TSG3nSTR2 Register Contents (2/3)

Bit Position	Bit Name	Function
6	TSG3nPPF	Indicates detection of the difference between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6) after they are compared. 0: No phase difference detected between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6). 1: A phase difference detected between the input patterns (TSG3nPTSI0 to TSG3nPTSI2) and the output patterns (TSG3nO1 to TSG3nO6). • TSG3nPPF is set to 1 when a difference between input and output patterns is detected, and a warning interrupt (INTTSG3nIWN) is generated. This bit can be cleared either by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), setting TSG3nTRG0.TSG3nTS to 1 (starting the timer), input to TSG3nTSST (restarting the timer), or writing 1 to TSG3nSTC.TSG3nPPR.
5	TSG3nPEF	Indicates whether an abnormal input (000_B or 111_B is input to TSG3nPTSI0 to TSG3nPTSI2) is detected. 0: No abnormal input (000_B or 111_B) to TSG3nPTSI0 to TSG3nPTSI2 detected. 1: Abnormal input (000_B or 111_B) to TSG3nPTSI0 to TSG3nPTSI2 detected. TSG3nPEF is set to 1 when an input of 000_B or 111_B to TSG3nPTSI0 to TSG3nPTSI2 is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPEF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nPER. TSG3nPEF is valid when TSG3nCTL1.TSG3nPEC = 1.
4	TSG3nTDF	Indicates whether simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers is detected. 0: Simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers not detected. 1: Simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers detected. TSG3nTDF is set to 1 when simultaneous generation of the TSG3nOPCI0 and TSG3nOPCI1 triggers is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nTDF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nTDR. TSG3nTDF is valid when TSG3nCTL1.TSG3nTDC = 1.
3	TSG3nNDF	Indicates whether noise on TSG3nPTSI0 to TSG3nPTSI2 is detected. 0: Noise on TSG3nPTSI0 to TSG3nPTSI2 due to simultaneous change of two or more pins not detected. 1: Noise on TSG3nPTSI0 to TSG3nPTSI2 due to simultaneous change of two or more pins detected. TSG3nNDF is set to 1 when simultaneous change of two or more pins in TSG3nPTSI0 to TSG3nPTSI2 is detected, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nNDF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1(starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nNDR. TSG3nNDF is valid when TSG3nCTL1.TSG3nNDC = 1.

Table 19.23 TSG3nSTR2 Register Contents (3/3)

Bit Position	Bit Name	Function
2	TSG3nPRF	Indicates whether reversal of the TSG3nPTSI0 to TSG3nPTSI2 input order is detected. 0: The reversal of the TSG3nPTSI0 TSG3nPTSI2 input order not detected. 1: The reversal of the TSG3nPTSI0 to TSG3nPTSI2 input order detected. TSG3nPRF is set to 1 when TSG3nSTR1.TSG3nTSF changes, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPRF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (operation start), by setting TSG3nTRG0.TSG3nTS to 1 (timer operation start), by input to TSG3nTSST(timer restart), or by writing 1 to TSG3nSTC.TSG3nPRR. Detection is possible from the second TSG3nPTSI0 to TSG3nPTSI2 change timing after setting TSG3nTRG0.TSG3nTS = 1. TSG3nPRF is valid when TSG3nCTL1.TSG3nPRC = 1.
1	TSG3nPTF	Indicates whether an abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 is detected. 0: No abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 detected. 1: An abnormal toggle of TSG3nPTSI0 to TSG3nPTSI2 detected. TSG3nPTF is set to 1 when TSG3nPTSI0 to TSG3nPTSI2 (TSG3nPTE signal toggle) are changed three times or more during TSG3nOPCI0 trigger or TSG3nPTSI0 to TSG3nPTSI2 (TSG3nPTE signal toggle) are changed three times or more during TSG3nOPCI1 trigger, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPTF can be cleared by changing TSG3nSTR0.TSG3nTE from 0 to 1 (starting operation), by setting TSG3nTRG0.TSG3nTS to 1 (starting timer operation), by input to TSG3nTSST (restarting the timer), or by writing 1 to TSG3nSTC.TSG3nPTR. TSG3nPTF is valid when TSG3nCTL1.TSG3nPTC[1:0] = 10_B or 11_B.
0	Reserved	When read, the value after reset is read.

19.3.17 TSG3nSTC — TSG3n Status Clear Trigger Register

This register controls the flags.

Access: This register can be written only in 16-bit units.

Address: <TSG3n_base> + $01C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3n TBR2	TSG3n TBR1	TSG3n TBR0	TSG3n PPR	TSG3n PER	TSG3n TDR	TSG3n NDR	TSG3n PRR	TSG3n PTR	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W	W	W	W	W	W	W	W	R

Table 19.24 TSG3nSTC Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When writing, write the value after reset.
9	TSG3nTBR2	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF2. 0: Does not clear TSG3nTBF2. 1: Clears TSG3nTBF2. When TSG3nTBR2 writing and TSG3nSTR2.TSG3nTBF2 setting occur simultaneously, TSG3nSTR2.TSG3nTBF2 setting has a priority, and the flag is not cleared.
8	TSG3nTBR1	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF1. 0: Does not clear TSG3nTBF1. 1: Clears TSG3nTBF1. When TSG3nTBR1 writing and TSG3nSTR2.TSG3nTBF1 setting occur simultaneously, TSG3nSTR2.TSG3nTBF1 setting has a priority, and the flag is not cleared.
7	TSG3nTBR0	This is a trigger bit that clears TSG3nSTR2.TSG3nTBF0. 0: Does not clear TSG3nTBF0. 1: Clears TSG3nTBF0. When TSG3nTBR0 writing and TSG3nSTR2.TSG3nTBF0 setting occur simultaneously, TSG3nSTR2.TSG3nTBF0 setting has a priority, and the flag is not cleared.
6	TSG3nPPR	This is a trigger bit that clears TSG3nSTR2.TSG3nPPF. 0: Does not clear TSG3nPPF. 1: Clears TSG3nPPF. When TSG3nPPR writing and TSG3nSTR2.TSG3nPPF setting occur simultaneously, TSG3nSTR2.TSG3nPPF setting has a priority, and the flag is not cleared.
5	TSG3nPER	This is a trigger bit that clears TSG3nSTR2.TSG3nPEF. 0: Does not clear TSG3nPEF. 1: Clears TSG3nPEF. When TSG3nPER writing and TSG3nSTR2.TSG3nPEF setting occur simultaneously, TSG3nSTR2.TSG3nPEF setting has a priority, and the flag is not cleared.
4	TSG3nTDR	This is a trigger bit that clears TSG3nSTR2.TSG3nTDF. 0: Does not clear TSG3nTDF. 1: Clears TSG3nTDF. When TSG3nTDR writing and TSG3nSTR2.TSG3nTDF setting occur simultaneously, TSG3nSTR2.TSG3nTDF setting has a priority, and the flag is not cleared.

Table 19.24 TSG3nSTC Register Contents (2/2)

Bit Position	Bit Name	Function
3	TSG3nNDR	This is a trigger bit that clears TSG3nSTR2.TSG3nNDF. 0: Does not clear TSG3nNDF. 1: Clears TSG3nNDF. When TSG3nNDR writing and TSG3nSTR2.TSG3nNDF setting occur simultaneously, TSG3nSTR2.TSG3nNDF setting has a priority, and the flag is not cleared.
2	TSG3nPRR	This is a trigger bit that clears TSG3nSTR2.TSG3nPRF. 0: Does not clear TSG3nPRF. 1: Clears TSG3nPRF. When TSG3nPRR writing and TSG3nSTR2.TSG3nPRF setting occur simultaneously, TSG3nSTR2.TSG3nPRF setting has a priority, and the flag is not cleared.
1	TSG3nPTR	This is a trigger bit that clears TSG3nSTR2.TSG3nPTF. 0: Does not clear TSG3nPTF. 1: Clears TSG3nPTF. When TSG3nPTR writing and TSG3nSTR2.TSG3nPTF setting occur simultaneously, TSG3nSTR2.TSG3nPTF setting has a priority, and the flag is not cleared.
0	Reserved	When writing, write the value after reset.

19.3.18 TSG3nOPT0 — TSG3n Option Register 0

This register sets the optional functions.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> + 020_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TSG3nSOC	TSG3nSTE	TSG3nPOT	TSG3nPSS	TSG3nIDC	TSG3nPSC	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 19.25 TSG3nOPT0 Register Contents (1/2)

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6	TSG3nSOC	Enables or disables control of the timer output (TSG3nO1 to TSG3nO6 pins) by software. 0: Disables control by software. 1: Enables control by software. When TSG3nSOC is set to 1, timer output is switched to the software control/ trigger control output pattern specified by TSG3nSPC2 to TSG3nSPC0. The dead time is secured by the dead time counter.
5	TSG3nSTE	Enables or disables control by the pattern output trigger. 0: Disables the TSG3nPTSI0 to TSG3nPTSI2 and TSG3nOPCI0, and TSG3nOPCI1 inputs. 1: Enables TSG3nPTSI0 to TSG3nPTSI2 and TSG3nOPCI0, and TSG3nOPCI1 inputs. • The pattern output trigger is selected by TSG3nPOT. • TSG3nSTE is valid in 120-DC mode and when software output control function is enabled.
4	TSG3nPOT	Selects the pattern output trigger. 0: Switches the output pattern by the external pattern input pins (TSG3nPTSI0 to TSG3nPTSI2) (pattern switch method). 1: Switches the output pattern by the rising edge of the TSG3nOPCI0 and TSG3nOPCI1 (trigger switch method).
3	TSG3nPSS	Selects the pattern output order switch factor. 0: The pattern output order is not switched by TSG3nPSC. 1: The pattern output order is switched by TSG3nPSC.
2	TSG3nIDC	Determines the output pattern from the TSG3nO1 to TSG3nO6 pins in combination with the TSG3nIDC and TSG3nSTR1.TSG3nTSF and TSG3nPSC signals. For the timer output order and patterns to be output, see **Figure 19.87** to **Figure 19.90**, Example of Operation in 120-DC Mode, in **Section 19.4.7.6 (5), Operation in 120-DC Mode**.

Table 19.25 TSG3nOPT0 Register Contents (2/2)

Bit Position	Bit Name	Function
1	TSG3nPSC	Selects the pattern output order when the semi-automatic cruise function is enabled. 0: Switches the timer output (TSG3nO1 to TSG3nO6) in the normal rotation. 1: Switches the timer output (TSG3nO1 to TSG3nO6) in the reverse rotation. • TSG3nPSC specifies the timer output pattern order assuming the output pattern specified by TSG3nSPC2 to TSG3nSPC0 as the initial pattern. TSG3nPSC is valid when TSG3nPOT = 1 and TSG3nPSS = 1. • It is recommended to rewrite TSG3nPSC when TSG3nSTR0.TSG3nTE = 0 or TSG3nPOT = 0. If TSG3nPSC is rewritten when TSG3nPOT = 1, the unexpected timer output pattern might be caused. • If the signal input to TSG3nPTSI0 to TSG3nPTSI2 changes with TSG3n operation being stopped (TSG3nSTR0.TSG3nTE = 0), the TSG3nTRG0.TSG3nTS bit should be set to 1 after matching the input signal change logic with the TSG3nPSC order. • For output order in normal or reverse rotation, see **Section 19.4.7.6, 120-DC Mode**. Here, normal rotation and reverse rotation refer to the change of output, and they are different from normal rotation and reverse rotation of a motor.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

19.3.19 TSG3nOPT1 — TSG3n Option Register 1

This register sets the optional functions.

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> +024_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSG3nSPC[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 19.26 TSG3nOPT1 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	TSG3nSPC [2:0]	Specifies the timer output pattern when software output function is enabled and in 120-DC mode. For the output pattern, see **Section 19.4.7.10, Software Output Control Function**, and **Section 19.4.7.6, 120-DC Mode**.

19.3.20 TSG3nOPT2 — TSG3n Option Register 2

This register sets rectangular waveform output from EMU2 to be used as PWM output by TSG3. This register can be set to 1 only when HT-PWM mode is selected (TSG3nMD2-0 = 001) and the timer is stopped (TSG3nRMC = 0). Do not rewrite this register in other modes (PWM mode, SP-PMW mode, 120-DC mode, HSP-PWM mode) or while the timer is operating (TSG3nRMC = 1).

Access: This register can be read/written in 8-bit units.

Address: <TSG3n_base> +$03C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nESSC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 19.27 TSG3nOPT2 Register Contents

Bit Position	Bit Name	Function
0	TSG3nESSC	EMU2 Rectangular Waveform Select Control 0: Timer output of TSG3 is set to HT-PWM mode selected by TSG3nMD2-0. 1: Timer output of TSG3 is switched to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value, input from EMU2 (ESW function). When TSG3nESSC is changed from 0 to 1, the timer output is switched to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value immediately. When TSG3nESSC is changed from 1 to 0, the timer output is switched to HT-PWM mode at the next reload timing.

19.3.21 TSG3nOPT2BF — TSG3n Option 2 Buffer Register

TSG3nOPT2BF is the buffer register for TSG3nOPT2. The value set in TSG3nOPT2 is captured at the timing described below.

TSG3nOPT2BF is a read-only register that indicates whether the output from TSG3nO1-6 is the rectangular waveform from EMU2. Writing to this register is ignored.

Access: This register can be read only in 8-bit units.

Address: <TSG3n_base> + $0CC_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nESSCBF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 19.28 TSG3nOPT2BF Register Contents

Bit Position	Bit Name	Function
0	TSG3nESSCBF	EMU2 Rectangular Waveform Select Control Buffer 0: Timer output of TSG3 (TSG3nO1-6) is set to the PWM output mode selected by TSG3nMD2-0. 1: Timer output of TSG3 (TSG3nO6-1) is switched to the output using ESW function (dead time is added to U-phase output pattern value, V-phase output pattern value, or W-phase output pattern value, input from EMU2).

Capture Timing

- At operation start (1 is written to TSG3nTS when TSG3nTE = 0).
- At restart (1 is written to TSG3nTS when TSG3nTE = 1).
- At rising of the next PCLK if TSG3nESSC is changed from 0 to 1.
- At the next reload timing if TSG3nESSC is changed from 1 to 0

19.3.22 TSG3nTRG0 — TSG3n Trigger Register 0

This register controls the start of the timer.

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> + 030_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 19.29 TSG3nTRG0 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nTS	This bit is a trigger bit that controls the start of the timer. 0: The timer is not started. 1: The timer is started (restarted if TSG3nSTR0.TSG3nTE = 1). When restarted, the 18-bit counter is initialized. This bit is always read as 0.

19.3.23 TSG3nTRG1 — TSG3n Trigger Register 1

This register controls the stop of the timer.

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> +034_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 19.30 TSG3nTRG1 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nTT	This is a trigger bit that controls the stop of the timer. 0: The timer is not stopped. 1: The timer is stopped (TSG3nSTR0.TSG3nTE = 0). This bit is always read as 0.

19.3.24 TSG3nTRG2 — TSG3n Trigger Register 2

TSG3nTRG2 is a trigger bit to reflect the PWM duty setting to TSG3nO1-6 in anytime rewrite mode of HT-PWM mode.

This register can be set to 1 only in HT-PWM mode and when anytime rewrite mode is selected (TSG3nRMC = 1). Do not rewrite this register in other modes (PWM mode, SP-PMW mode, 120-DC mode, HSP-PWM mode) or when reload mode is selected (TSG3nRMC = 0).

Access: This register can be written only in 8-bit units.

Address: <TSG3n_base> + 038_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG3nIMT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 19.31 TSG3nTRG2 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TSG3nIMT	Anytime Rewrite Trigger 0: Disabled 1: Changes in duty settings for U, V, and W phases are reflected to timer output in HT-PWM mode and when anytime rewrite mode is selected.

19.3.25 TSG3nCNT — TSG3n Counter Read Buffer Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCNTE.

For the operation of this register, see **Section 19.3.26, TSG3nCNTE — TSG3n Bit-Extended Counter Read Buffer Register**.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + 028_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

19.3.26 TSG3nCNTE — TSG3n Bit-Extended Counter Read Buffer Register

The counter values can be read from this register. This register mirrors the contents of TSG3nCNT from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read only in 32-bit units.

Address: <TSG3n_base> + $1A0_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit counter	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

18-bit counter

This register is a timer read buffer register from which the 18-bit counter value can be read. In HT-PWM mode, the 18-bit counter provides the triangular waveform control in which the counter value is incremented and decremented by 2. Bit 0 is always read as 0.

In other modes, the 18-bit counter provides the sawtooth waveform control in which the counter value is incremented by 1.

Table 19.32 TSG3nCNTE Register Count Value

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nDTC0	TSG3nDTC0	TSG3nDTC0+TSG3nCMP0E[*1]
Other modes	00000_H	00000_H	TSG3nCMP0E

Note 1. Set the value as TSG3nDTC0+TSG3nCMP0E < $3FFFF_H$.

19.3.27 TSG3nSBC — TSG3n Sub-Counter Read Buffer Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nSBCE.

For the operation of this register, see **Section 19.3.28, TSG3nSBCE — TSG3n Bit Extended Sub-Counter Read Buffer Register**.

Access: This register can be read only in 16-bit units.

Address: <TSG3n_base> + $02C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit sub-counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

19.3.28 TSG3nSBCE — TSG3n Bit Extended Sub-Counter Read Buffer Register

The sub-counter values can be read from this register. This register mirrors the contents of TSG3nSBC from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read only in 32-bit units.

Address: <TSG3n_base> + $1A4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit sub-counter	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit sub-counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

18-bit counter

This register is a timer read buffer register from which the 18-bit sub-counter value can be read. In HT-PWM mode, the 18-bit counter provides the triangular waveform control in which the counter value is incremented and decremented by 2. Bit 0 is always read as 0. (Available only in HT-PWM mode.).

Table 19.33 TSG3nSBCE Register Count Value

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nDTC0	00000_H	TSG3nDTC0 + TSG3nDTC1 + TSG3nCMP0E*1
Other modes	00000_H	00000_H	00000_H

Note 1. Set the value as TSG3nDTC0 + TSG3nDTC1 + TSG3nCMP0E < $3FFFF_H$.

19.3.29 TSG3nCMP0 — TSG3n Compare Register 0

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP0E.

For the operation of this register, see **Section 19.3.30, TSG3nCMP0E — TSG3n Bit Extended Compare Register 0**.

Access: his register can be read/written in 32-bit units.

Address: <TSG3n_base> + 058_H

Value after reset: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.30 TSG3nCMP0E — TSG3n Bit Extended Compare Register 0

This register is an 18-bit compare register that specifies the PWM period in all modes. This register mirrors the contents of TSG3nCMP0 from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> +$14C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.34 TSG3nCMP0E Register Setting

Operating mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG3nCMP0E[*1]	00002_H	$3FFFE_H$
Other modes	TSG3nCMP0E + 1	1 (TSG3nCMP0E = 00000_H)	40000_H (TSG3nCMP0E = $3FFFF_H$)

Note 1. In HT-PWM mode, the least significant bit is ignored.

19.3.31 TSG3nCMP1W — TSG3n Compare Register 1, 2

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP1E and TSG3nCMP2E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 040_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP2 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP1 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.32 TSG3nCMP3W — TSG3n Compare Register 3, 4

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP3E and TSG3nCMP4E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $04C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP4 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP3 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.33 TSG3nCMP5W — TSG3n Compare Register 5, 6

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP5E and TSG3nCMP6E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 044_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP6 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP5 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.34 TSG3nCMP7W — TSG3n Compare Registers 7, 8

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP7E and TSG3nCMP8E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 050_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP8 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP7 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.35 TSG3nCMP9W — TSG3n Compare Registers 9, 10

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP9E and TSG3nCMP10E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 048_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP10 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP9 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.36 TSG3nCMP11W — TSG3n Compare Registers 11, 12

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMP11E and TSG3nCMP12E.

For the operation of this register, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 054_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nCMP12 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMP11 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.37 TSG3nCMP1 to TSG3nCMP12 — TSG3n Compare Registers 1 to 12

These registers can access the 16 lower-order bits of the 18-bit registers TSG3nCMP1E-12E.

For the operation of these registers, see **Section 19.3.38, TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12**.

Access: This register can be read/written in 16-bit units.

Address:
TSG3nCMP1 <TSG3n_base> + 080_H
TSG3nCMP2 <TSG3n_base> + 084_H
TSG3nCMP3 <TSG3n_base> + 098_H
TSG3nCMP4 <TSG3n_base> + $09C_H$
TSG3nCMP5 <TSG3n_base> + 088_H
TSG3nCMP6 <TSG3n_base> + $08C_H$
TSG3nCMP7 <TSG3n_base> + $0A0_H$
TSG3nCMP8 <TSG3n_base> + $0A4_H$
TSG3nCMP9 <TSG3n_base> + 090_H
TSG3nCMP10 <TSG3n_base> + 094_H
TSG3nCMP11 <TSG3n_base> + $0A8_H$
TSG3nCMP12 <TSG3n_base> + $0AC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.38 TSG3nCMP1E to TSG3nCMP12E — TSG3n Bit Extended Compare Registers 1 to 12

The compare value is set by these registers. These registers mirror the contents of TSG3nCMP1-12, TSG3nCMP1W, 3W, 5W, 7W, 9W, and 11W from which the 16 lower-order bits of these registers can be accessed.

Access: This register can be read/written in 32-bit units.

Address:
TSG3nCMP1E <TSG3n_base> + $17C_H$
TSG3nCMP2E <TSG3n_base> + 178_H
TSG3nCMP3E <TSG3n_base> + 164_H
TSG3nCMP4E <TSG3n_base> + 160_H
TSG3nCMP5E <TSG3n_base> + 174_H
TSG3nCMP6E <TSG3n_base> + 170_H
TSG3nCMP7E <TSG3n_base> + $15C_H$
TSG3nCMP8E <TSG3n_base> + 158_H
TSG3nCMP9E <TSG3n_base> + $16C_H$
TSG3nCMP10E <TSG3n_base> + 168_H
TSG3nCMP11E <TSG3n_base> + 154_H
TSG3nCMP12E <TSG3n_base> + 150_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.35 TSG3nCMP1E-TSG3nCMP12E Register Setting (1/2)

register	PWM Mode	HT-PWM Mode SP-PWM Mode	120-DC Mode	HSP-PWM Mode
TSG3nCMP1E	TSG3nO1 clear timing	TSG3nO1 clear timing /TSG3nO2 set timing	Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO1 clear timing
TSG3nCMP2E	TSG3nO1 set timing	TSG3nO1 set timing/TSG3nO2 clear timing		TSG3nO1 set timing
TSG3nCMP3E	TSG3nO2 clear timing	—	Duty whenTSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO2 clear timing
TSG3nCMP4E	TSG3nO2 set timing	—		TSG3nO2 set timing
TSG3nCMP5E	TSG3nO3 clear timing	TSG3nO3 clear timing/ TSG3nO4 set timing	Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO3 clear timing
TSG3nCMP6E	TSG3nO3 set timing	TSG3nO3 set timing/TSG3nO4 clear timing		TSG3nO3 set timing
TSG3nCMP7E	TSG3nO4 clear timing	—	Duty when TSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO4 clear timing
TSG3nCMP8E	TSG3nO4 set timing	—		TSG3nO4 set timing
TSG3nCMP9E	TSG3nO5 clear timing	TSG3nO5 clear timing/ TSG3nO6 set timing	Duty when TSG3nO1, 3, or 5 output pattern is selected by TSG3nPAT0.	TSG3nO5 clear timing
TSG3nCMP10E	TSG3nO5 set timing	TSG3nO5 set timing/TSG3nO6 clear timing		TSG3nO5 set timing

Table 19.35 TSG3nCMP1E-TSG3nCMP12E Register Setting (2/2)

register	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode	HSP-PWM Mode
TSG3nCMP11E	TSG3nO6 clear timing	—		Duty when TSG3nO2, 4, or 6 output pattern is selected by TSG3nPAT1.	TSG3nO6 clear timing
TSG3nCMP12E	TSG3nO6 set timing	—			TSG3nO6 set timing

NOTE

The dead time function is used in all operating modes.

In HT-PWM mode, a compare match occurs not only in TSG3nCNTE but also in TSG3nSBCE.

In 120-DC mode, the output from TSG3nO1-6 is controlled by the TSG3nCMPmE, TSG3nPAT0, and TSG3nPAT1 registers.

19.3.39 TSG3nDCMP0W — TSG3n Diagnostic Output Compare Register 0, 1

This register can access the 16 lower-order bits of the 18-bit register TSG3nDCMP0E and TSG3nDCMP1E.

For the operation of this register, see **Section 19.3.41, TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $05C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG3nDCMP1(16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nDCMP0(16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.40 TSG3nDCMP2 — TSG3n Diagnostic Output Compare Register 2

This register can access the 16 lower-order bits of the 18-bit register TSG3nDCMP2E.

For the operation of this register, see **Section 19.3.41, TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 060_H

Value after reset: 0000 0000_H

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nDCMP2 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.41 TSG3nDCMP0E to 2E — TSG3n Bit Extended Diagnostic Output Compare Register 0 to 2

The compare value is set by these registers. These registers mirror the contents of TSG3nDCMP0W and TSG3nDCMP2 from which the 16 lower-order bits of these registers can be accessed.

Access: This register can be read/written in 32-bit units.

Address: TSG3nDCMP0E <TSG3n_base> + 148_H
TSG3nDCMP1E <TSG3n_base> + 144_H
TSG3nDCMP2E <TSG3n_base> + 140_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

These registers control the diagnostic output timing or AD conversion trigger timing in all modes. A pulse is generated at the match timing of the 18-bit counter value with this register.

19.3.42 TSG3nPAT0W — TSG3n Pattern Register 0

This register specifies the output pattern.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 064_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5T	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5T	PAT4T			PAT3T			PAT2T			PAT1T			PAT0T		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output Pattern

This register controls UT/VT/WT output in 120-DC mode.

Table 19.36 TSG3nPAT0W Register Setting value and Output Control

PATmT Value	Output Control
000	Fixed Low level
001	PWM output set by TSG3nCMP1E
010	PWM output set by TSG3nCMP2E
011	PWM output set by TSG3nCMP5E
100	PWM output set by TSG3nCMP6E
101	PWM output set by TSG3nCMP9E
110	PWM output set by TSG3nCMP10E
111	Fixed High level

Note: m = 0, 1, 2, 3, 4, 5

19.3.43 TSG3nPAT1W — TSG3n Pattern Register 1

This register specifies the output pattern.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 068_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5B	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5B	PAT4B			PAT3B			PAT2B			PAT1B			PAT0B		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output Pattern

This register controls UB/VB/WB output in 120-DC mode.

Table 19.37 TSG3nPAT1W Register Setting value and Output Control

PATmB Value	Output Control
000	Fixed Low level
001	PWM output set by TSG3nCMP3E
010	PWM output set by TSG3nCMP4E
011	PWM output set by TSG3nCMP7E
100	PWM output set by TSG3nCMP8E
101	PWM output set by TSG3nCMP11E
110	PWM output set by TSG3nCMP12E
111	Fixed High level

Note: m = 0, 1, 2, 3, 4, 5

19.3.44 TSG3nDTC0W — TSG3n Dead Time Setting Register 0

This register sets the dead time value (the period from inverse phase inactivation to positive phase activation).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $06C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	Write Protection Code Check														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nDTC0(10-bit dead time compare)									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

To rewrite TSG3nDTC0W[0:9], set bit 14 to bit 0 and TSG3nDTCM to 0 in TSG3nDTPR, and rewrite the TSG3nDTC0W. At this time, when the rewritten value of TSG3nDTC0W[30:16] and the TSG3nDTPR value match, TSG3nDTC0W is rewritten.

During timer operation (TSG3nSTR0.TSG3nTE = 1), rewriting should be performed in reload mode (TSG3nCTL3.TSG3nRMC = 0).

19.3.45 TSG3nDTC1W — TSG3n Dead Time Setting Register 1

This register sets the dead time (the period from positive phase inactivation to inverse phase activation).

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 070_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	Write Protection Code Check														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG3nDTC1(10-bit dead time compare)									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

To rewrite TSG3nDTC1W[0:9], set bit 14 to bit 0 and TSG3nDTCM to 0 in TSG3nDTPR, and rewrite the TSG3nDTC1W. At this time, when the rewritten value of TSG3nDTC1W[30:16] and the TSG3nDTPR value match, TSG3nDTC1W is rewritten.

During timer operation (TSG3nSTR0.TSG3nTE = 1), rewriting should be performed in reload mode (TSG3nCTL3.TSG3nRMC = 0).

19.3.46 TSG3nCMPU — TSG3n HT-PWM U Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPUE.

For the operation of this register, see **Section 19.3.49, TSG3nCMPUE — TSG3n Bit Extended U Phase Compare Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPU(16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.47 TSG3nCMPV — TSG3n HT-PWM V Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPVE.

For the operation of this register, see **Section 19.3.50, TSG3nCMPVE — TSG3n Bit Extended V Phase Compare Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPV (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.48 TSG3nCMPW — TSG3n HT-PWM W Phase Compare Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nCMPWE.

For the operation of this register, see **Section 19.3.51, TSG3nCMPWE — TSG3n Bit Extended W Phase Compare Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0B8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nCMPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.49 TSG3nCMPUE — TSG3n Bit Extended U Phase Compare Register

This register sets the compare value for U phase in HT-PWM. In addition to the functions of TSG3nCMP1E and TSG3nCMP2E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP1E and TSG3nCMP2E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 19.2**). When this register is read, the same value as TSG3nCMP1E is returned. This register mirrors the contents of TSG3nCMPU from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> +188_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

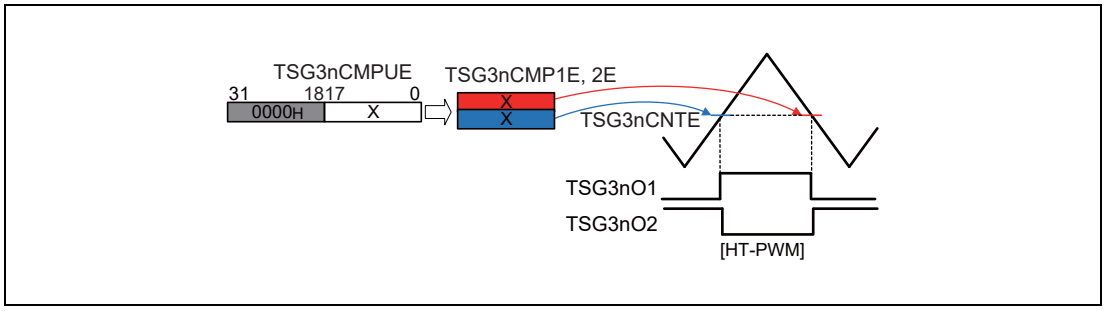


Figure 19.2 TSG3nCMPUE Register Accesses

19.3.50 TSG3nCMPVE — TSG3n Bit Extended V Phase Compare Register

This register sets the compare value for V phase in HT-PWM. In addition to the functions of TSG3nCMP5E and TSG3nCMP6E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP5E and TSG3nCMP6E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 19.3**). When this register is read, the same value as TSG3nCMP5E is returned. This register mirrors the contents of TSG3nCMPV from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> +184_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

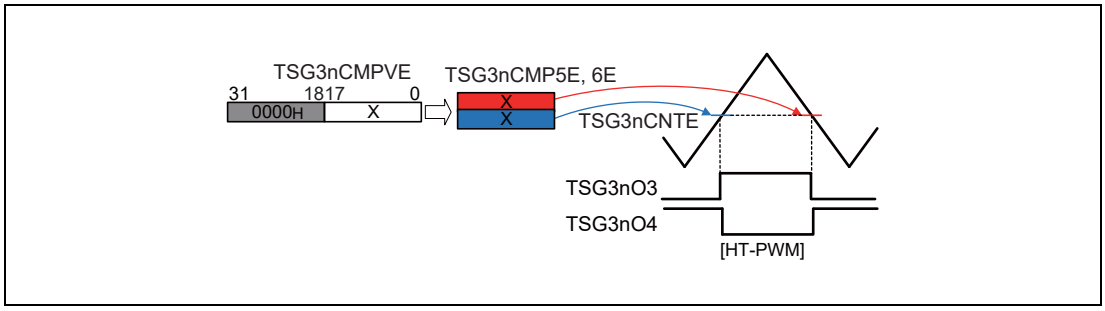


Figure 19.3 TSG3nCMPVE Register Accesses

19.3.51 TSG3nCMPWE — TSG3n Bit Extended W Phase Compare Register

This register sets the compare value for W phase in HT-PWM. In addition to the functions of TSG3nCMP9E and TSG3nCMP10E, this register can access specific registers.

The data written to this register is stored in the TSG3nCMP9E and TSG3nCMP10E registers which allows a generation of symmetry triangular waveform of PWM by one write access (see **Figure 19.4**). When this register is read, the same value as TSG3nCMP9E is returned. This register mirrors the contents of TSG3nCMPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> +180_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

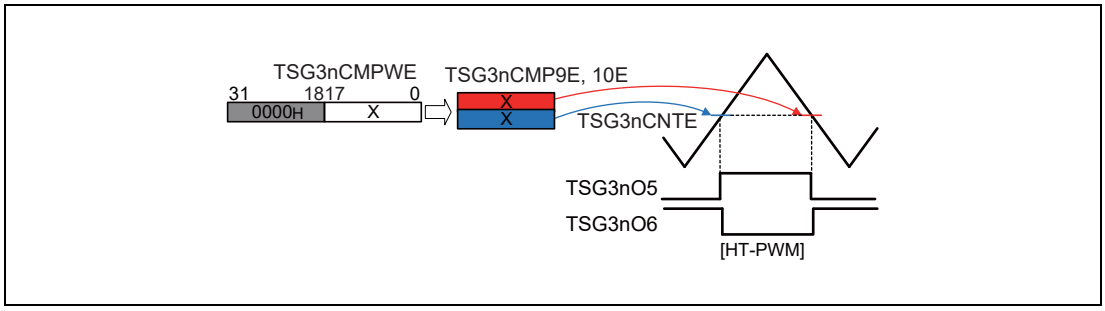


Figure 19.4 TSG3nCMPWE Register Accesses

19.3.52 TSG3nUPW — TSG3n SP-PWM U Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nUPWE.

For the operation of this register, see **Section 19.3.55, TSG3nUPWE — TSG3n Bit Extended U Phase Active Width Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0BC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nUPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.53 TSG3nVPW — TSG3n SP-PWM V Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nVPWE.

For the operation of this register, see **Section 19.3.56, TSG3nVPWE — TSG3n Bit Extended V Phase Active Width Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0C0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nVPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.54 TSG3nWPW — TSG3n SP-PWM W Phase Active Width Register

This register can access the 16 lower-order bits of the 18-bit register TSG3nWPWE.

For the operation of this register, see **Section 19.3.57, TSG3nWPWE — TSG3n Bit Extended W Phase Active Width Register**.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + $0C4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3nWPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.55 TSG3nUPWE — TSG3n Bit Extended U Phase Active Width Register

This register sets the active width for U phase in SP-PWM mode. The sum of the TSG3nUPWE value and the TSG3nCMP2E value is stored in TSG3nCMP1E (see **Figure 19.5**). When this register is read, the same value as TSG3nCMP1E is returned. This register mirrors the contents of TSG3nUPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 198_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

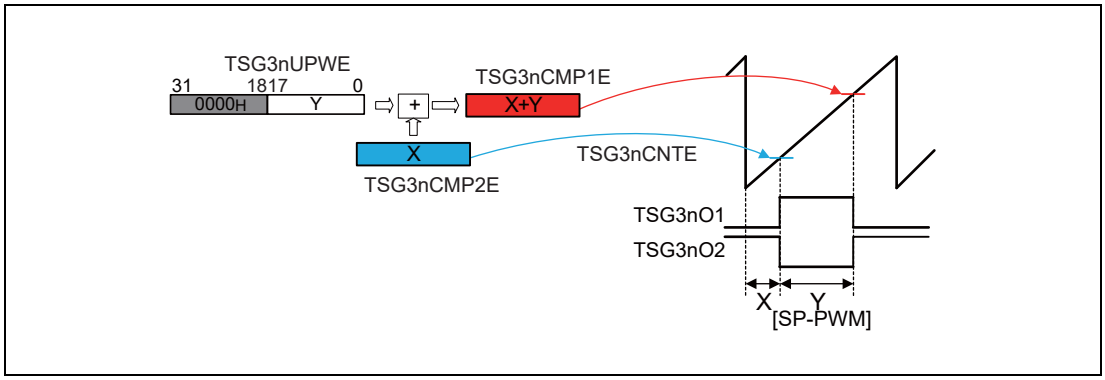


Figure 19.5 TSG3nUPWE Register Accesses

19.3.56 TSG3nVPWE — TSG3n Bit Extended V Phase Active Width Register

This register sets the active width for V phase in SP-PWM mode. The sum of the TSG3nVPWE value and the TSG3nCMP6E value is stored in TSG3nCMP5E (see **Figure 19.6**). When this register is read, the same value as TSG3nCMP5E is returned. This register mirrors the contents of TSG3nVPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 194_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

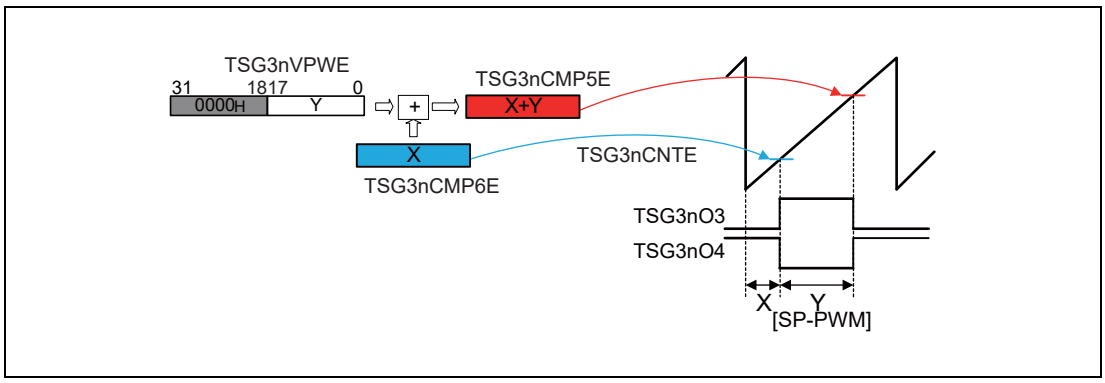


Figure 19.6 TSG3nVPWE Register Accesses

19.3.57 TSG3nWPWE — TSG3n Bit Extended W Phase Active Width Register

This register sets the active width for W phase in SP-PWM mode. The sum of the TSG3nWPWE value and the TSG3nCMP10E value is stored in TSG3nCMP9E (see **Figure 19.7**). When this register is read, the same value as TSG3nCMP9E is returned. This register mirrors the contents of TSG3nWPW from which the 16 lower-order bits of this register can be accessed.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 190_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

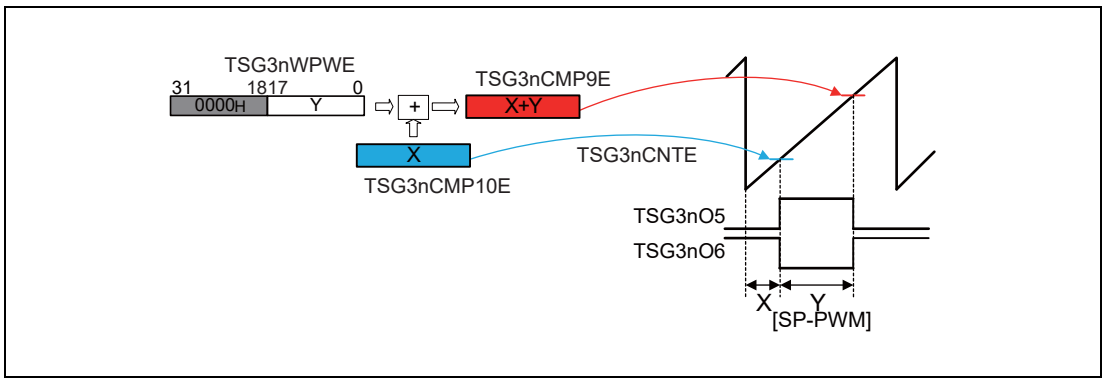


Figure 19.7 TSG3nWPWE Register Accesses

19.3.58 TSG3nHSPCMUE — TSG3n HSP-PWM Mode U Phase Compare Register

This register sets the PWM output width for U phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP1E to TSG3nCMP4E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

When this register is read, the same value as TSG3nCMP1E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 134_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.59 TSG3nHSPCMVE — TSG3n HSP-PWM Mode V Phase Compare Register

This register sets the PWM output width for V phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP5E to TSG3nCMP8E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

When this register is read, the same value as TSG3nCMP5E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 130_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.60 TSG3nHSPCMWE — TSG3n HSP-PWM Mode W Phase Compare Register

This register sets the PWM output width for W phase in HSP-PWM mode.

When a write access is made to this register, the values are set to TSG3nCMP9E to TSG3nCMP12E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

When this register is read, the same value as TSG3nCMP9E is returned.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + $12C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit compare register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.61 TSG3nHSPSHUE — TSG3n HSP-PWM Mode U Phase Shift Register

This register sets the PWM shift width for U phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMUE after setting this register, the values are set to TSG3nCMP1E to TSG3nCMP4E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 128_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.62 TSG3nHSPSHVE — TSG3n HSP-PWM Mode V Phase Shift Mode Register

This register sets the PWM shift width for V phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMVE after setting this register, the values are set to TSG3nCMP5E to TSG3nCMP8E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 124_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.63 TSG3nHSPSHWE — TSG3n HSP-PWM Mode W Phase Sift Register

This register sets the PWM shift width for U phase in HSP-PWM mode.

When a write access is made to TSG3nHSPCMWE after setting this register, the values are set to TSG3nCMP9E to TSG3nCMP12E registers according to the formulas described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode**.

Access: This register can be read/written in 32-bit units.

Address: <TSG3n_base> + 120_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	18-bit shift register	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	18-bit shift register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

19.3.64 TSG3nDTPR — TSG3n Dead Time Protection Register

This register controls protection of the write access to the dead time register.

Access: This register can be read/written in 16-bit units.

Address: <TSG3n_base> + 210_H

Value after reset: $0000\ 0000_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG3n DTCM	Write Protection Code														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.38 TSG3nDTPR Register Contents

Bit Position	Bit Name	Function
15	TSG3nDTCM	Enables or disables rewriting of TSG3nDTC0 and TSG3nDTC1. 0: Enables rewriting of TSG3nDTC0 and TSG3nDTC1. 1: Disables rewriting of TSG3nDTC0 and TSG3nDTC1.
14 to 0	TSG3nDTPR [14:0]	Sets the write protection code (any value from 0000 to 7FFF).

This register protects TSG3nDTC0 and TSG3nDTC1 from illegal rewriting.

Functions are described below.

- TSG3nDTCM enables or disables rewriting of TSG3nDTC0 and TSG3nDTC1.
- Rewriting of TSG3nDTC0 and TSG3nDTC1 is enabled or disabled by checking a match of the write protection code (bits 30 to 16) of TSG3nDTC0 with TSG3nDTC1 and the write protection code of TSG3nDTPR, and the TSG3nDTCM setting.

CAUTION

This register should be set when the timer is stopped (TSG3nSTR0.TSG3nTE = 0). Only the same value can be written during timer operation (TSG3nSTR0.TSG3nTE = 1). If the different value is written to this register when TSG3nSTR0.TSG3nTE = 1, timer operation cannot be guaranteed. If this register is erroneously rewritten, set this register again after stopping the timer.

19.4 Function

Table 19.39 List of Modes

TSG3nCTL0 Register			Timer Mode
TSG3nMD2	TSG3nMD1	TSG3nMD0	
0	0	0	PWM mode
0	0	1	HT-PWM mode (HT-PWM)
0	1	0	Shift pulse PWM mode (SP-PWM)
0	1	1	120-DC mode
1	0	0	High-accuracy shift pulse PWM mode (HSP-PWM)
Other than above			Setting prohibited

19.4.1 Basic Operation

19.4.1.1 Basic Operation of 18-Bit Counter

The basic operation of the 18-bit counter is described. For details, see **Section 19.4.7, Operating Modes**.

Counting start

The 18-bit counter of TSG3n starts counting in HT-PWM mode when the initial value is 00000_H and after the TSG3nDTC0 value is loaded. The counter starts counting from the initial value 00000_H in all modes except for HT-PWM mode.

In HT-PWM mode, the counter value is incremented by 2 from the value of TSG3nDTC0 and decremented by 2 up to the value of TSG3nDTC0 after the counter value matches with the value of TSG3nCMP0E + TSG3nDTC0. The counter increments from 00000_H, 00001_H, 00002_H, 00003_H, … in all modes except for HT-PWM mode.

Counter clear

The 18-bit counter is cleared by the match of the counter value and the value of TSG3nCMP0E in all modes except for HT-PWM mode. (Clearing operation is not available in HT-PWM mode.)

Counter read during counting

In the TSG3n, the 18-bit counter value during counting can be read through TSG3nCNTE.

Count stop operation

When count operation is stopped (when TS0TE is changed from 1 to 0), TSG3nCNTE and TSG3nSBCE retain the counter value when stopped.

Interrupt operation

In the TSG3n, the following interrupts are generated.

- INTTSG3nI0: A period interrupt by a match of the 18-bit counter value with the TSG3nDTC0 value in HT-PWM mode. A compare match interrupt of the 18-bit counter value with the TSG3nCMP0E buffer register in any mode other than HT-PWM mode.
- INTTSG3nI1: A compare match interrupt of the 18-bit counter value with the TSG3nCMP1E buffer register.
- INTTSG3nI2: A compare match interrupt of the 18-bit counter value with the TSG3nCMP2E buffer register.
- INTTSG3nI3: A compare match interrupt of the 18-bit counter value with the TSG3nCMP3E buffer register.
- INTTSG3nI4: A compare match interrupt of the 18-bit counter value with the TSG3nCMP4E buffer register.
- INTTSG3nI5: A compare match interrupt of the 18-bit counter value with the TSG3nCMP5E buffer register.
- INTTSG3nI6: A compare match interrupt of the 18-bit counter value with the TSG3nCMP6E buffer register.
- INTTSG3nI7: A compare match interrupt of the 18-bit counter value with the TSG3nCMP7E buffer register.
- INTTSG3nI8: A compare match interrupt of the 18-bit counter value with the TSG3nCMP8E buffer register.
- INTTSG3nI9: A compare match interrupt of the 18-bit counter value with the TSG3nCMP9E buffer register.
- INTTSG3nI10: A compare match interrupt of the 18-bit counter value with the TSG3nCMP10E buffer register.
- INTTSG3nI11: A compare match interrupt of the 18-bit counter value with the TSG3nCMP11E buffer register.
- INTTSG3nI12: A compare match interrupt of the 18-bit counter value with the TSG3nCMP12E buffer register.
- INTTSG3nIPEK: A peak interrupt when the 18-bit counter switches from incrementing to decrementing.
- INTTSG3nIVLY: A trough interrupt when the 18-bit counter switches from decrementing to incrementing.
- INTTSG3nIER: A simultaneous active state detection interrupt of the positive phase and inverse phase
- INTTSG3nIWN: A warning detection interrupt

19.4.1.2 Function of Compare Registers

The functions of the compare registers in each operating mode are shown in the following tales.

Table 19.40 Compare Register Functions in Each Mode (1/7)

Operating Mode	TSG3nCMP0E	TSG3nCMP1E	TSG3nCMP2E
PWM mode	PWM period	TSG3nO1 clear timing	TSG3nO1 set timing
HT-PWM mode	PWM period	TSG3nO1 clear timing TSG3nO2 set timing	TSG3nO1 set timing TSG3nO2 clear timing
SP-PWM mode	PWM period	TSG3nO1 clear timing TSG3nO2 set timing	TSG3nO1 set timing TSG3nO2 clear timing
120-DC mode	PWM period	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0
HSP-PWM mode	PWM period	TSG3nO1 clear timing	TSG3nO1 set timing

Table 19.40 Compare Register Functions in Each Mode (2/7)

Operating Mode	TSG3nCMP3E	TSG3nCMP4E	TSG3nCMP5E	TSG3nCMP6E
PWM mode	TSG3nO2 clear timing	TSG3nO2 set timing	TSG3nO3 clear timing	TSG3nO3 set timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	TSG3nO3 clear timing TSG3nO4 set timing	TSG3nO3 set timing TSG3nO4 clear timing
SP-PWM mode	—	—	TSG3nO3 clear timing TSG3nO4 set timing	TSG3nO3 set timing TSG3nO4 clear timing
120-DC mode	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W
HSP-PWM mode	TSG3nO2 clear timing	TSG3nO2 set timing	TSG3nO3 clear timing	TSG3nO3 set timing

Table 19.40 Compare Register Functions in Each Mode (3/7)

Operating Mode	TSG3nCMP7E	TSG3nCMP8E	TSG3nCMP9E	TSG3nCMP10E
PWM mode	TSG3nO4 clear timing	TSG3nO4 set timing	TSG3nO5 clear timing	TSG3nO5 set timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	TSG3nO5 clear timing TSG3nO6 set timing	TSG3nO5 set timing TSG3nO6 clear timing
SP-PWM mode	—	—	TSG3nO5v TSG3nO6 set timing	TSG3nO5 set timing TSG3nO6 clear timing
120-DC mode	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, or TSG3nO6 output by TSG3nPAT1W	Select TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W	TSG3nO1, TSG3nO3, or TSG3nO5 output by TSG3nPAT0W
HSP-PWM mode	TSG3nO4 clear timing	TSG3nO4 set timing	TSG3nO5 clear timing	TSG3nO5 set timing

Table 19.40 Compare Register Functions in Each Mode (4/7)

Operating Mode	TSG3nCMP11E	TSG3nCMP12E	TSG3nDCMP0E	TSG3nDCMP1E
PWM mode	TSG3nO6 clear timing	TSG3nO6 set timing	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
HT-PWM mode	Compare match interrupt	Compare match interrupt	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
SP-PWM mode	—	—	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
120-DC mode	Select TSG3nO2, TSG3nO4, TSG3nO6 by TSG3nPAT1W	Select TSG3nO2, TSG3nO4, TSG3nO6 by TSG3nPAT1W	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing
HSP-PWM mode	TSG3nO6 clear timing	TSG3nO6 set timing	Diagnostic output or A/D conversion trigger timing	Diagnostic output or A/D conversion trigger timing

Table 19.40 Compare Register Functions in Each Mode (5/7)

Operating Mode	TSG3nDCMP2E	TSG3nCMPUE	TSG3nCMPVE	TSG3nCMPWE
PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—
HT-PWM mode	Diagnostic output or A/D conversion trigger timing	The TSG3nCMPUE set value is used as the set value of TSG3nCMP1E and TSG3nCMP2E.	The TSG3nCMPVE set value is used as the set value of TSG3nCMP5E and TSG3nCMP6E.	The TSG3nCMPWE set value is used as the set value of TSG3nCMP9E and TSG3nCMP10E.
SP-PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—
120-DC mode	Diagnostic output or A/D conversion trigger timing	—	—	—
HSP-PWM mode	Diagnostic output or A/D conversion trigger timing	—	—	—

Table 19.40 Compare Register Functions in Each Mode (6/7)

Operating Mode	TSG3nUPWE	TSG3nVPWE	TSG3nWPWE
PWM mode	—	—	—
HT-PWM mode	—	—	—
SP-PWM mode	The sum of the TSG3nUPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.	The sum of the TSG3nVPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.	The sum of the TSG3nWPWE set value and the TSG3nCMP2E set value is used as the TSG3nCMP1E set value.
120-DC mode	—	—	—
HSP-PWM mode	—	—	—

Table 19.40 Compare Register Functions in Each Mode (7/7)

Operating Mode	TSG3nHSPCMUE, TSG3nHSPSHUE	TSG3nHSPCMVE, TSG3nHSPSHVE	TSG3nHSPCMWE, TSG3nHSPSHWE
PWM mode	—	—	—
HT-PWM mode	—	—	—
SP-PWM mode	—	—	—
120-DC mode	—	—	—
HSP-PWM mode	TSG3nCMP1E-4E is set based on the TSG3nHSPCMUE set value and the set values in TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHUE.	TSG3nCMP5E-8E is set based on the TSG3nHSPCMVE set value and the set values of TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHVE.	TSG3nCMP9E-12E is set based on the TSG3nHSPCMWE set value and the set values of TSG3nCMP0E, TSG3nDTC0, TSG3nDTC1, and TSG3nHSPSHWE.

19.4.1.3 Compare Register Rewrite Operation

TSG3 can be set to reload mode or anytime rewrite mode using theTSG3nRMC bit.

Reload mode is enabled when TSG3nRMC = 0. The registers listed as “enabled” on the “Reload” column in **Section 19.3.1, List of Registers** are simultaneously updated at the reload timing.

Anytime rewrite mode is enabled when TSG3nRMC = 1. The registers are updated independently every time the value is written to the relevant register.

The update timing of the registers to be reloaded in reload mode and anytime rewrite mode in each mode are listed in the following table.

Table 19.41 Updating Timing of Compare Registers by Mode

Mode	Anytime Rewrite TSG3nRMC = 1	Reload TSG3nRMC = 0
PWM mode	TSG3nCMP0E: At the next counter clear timing of the 18-bit counter	At reload timing
	Registers other than TSG3nCMP0E: At a write access to the register	
HT-PWM mode	TSG3nCMP0E: At the next peak or trough timing of TSG3nCNTE	At reload timing
	TSG3nCMP1E, 2E, 5E, 6E, 9E, 10E: At writing 1 to the TSG3nIMT bit	
	Registers other than TSG3nCMP0E,1E, 2E, 5E, 6E, 9E, 10E: At a write access to the register	
SP-PWM mode	TSG3nCMP0E: At the next counter clear timing of the 18-bit counter	At reload timing
	Registers other than TSG3nCMP0E: At a write access to the register	
120-DC mode	Setting prohibited	At reload timing
HSP-PWM mode	Setting prohibited	At reload timing

Anytime Rewrite Mode

In this mode, the compare registers are rewritten independently. Whenever a value is written to the compare register, the written value is reflected at the timing of **Table 19.41**.

Reload mode (Simultaneous Rewrite Function)

Writing to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) enables reload (sets the reload request flag (TSG3nSTR0.TSG3nRSF)), and the values of all the pertinent registers are updated simultaneously at the next reload timing (reload).

The reload timing is the peak or trough timing of the 18-bit counter when the TSG3nTRG0.TSG3nTS bit is changed from 0 to 1. Reloading is controlled by TSG3nCTL4.TSG3nPRE, and TSG3nVRE.

Writing to any register other than TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) does not enable reloading.

Do not write to the registers to be reloaded until the next reload timing after reloading is enabled by writing to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE).

The pertinent registers should be rewritten when the reload request flag (TSG3nSTR0.TSG3nRSF) is 0.

Rewriting registers to be reloaded by DMA transfer

Some of the registers to be reloaded can be rewritten by DMA transfer. DMA transfer is performed as follows.

Table 19.42 Example of DMA Transfer Order of Registers to be Reloaded

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 040_H	TSG3nCMP1W	
<TSG3n_base> + 044_H	TSG3nCMP5W	
<TSG3n_base> + 048_H	TSG3nCMP9W	
<TSG3n_base> + $04C_H$	TSG3nCMP3W	
<TSG3n_base> + 050_H	TSG3nCMP7W	
<TSG3n_base> + 054_H	TSG3nCMP11W	
<TSG3n_base> + 058_H	TSG3nCMP0	
<TSG3n_base> + $05C_H$	TSG3nDCMP0W	
<TSG3n_base> + 060_H	TSG3nDCMP2	
<TSG3n_base> + 064_H	TSG3nPAT0W	
<TSG3n_base> + 068_H	TSG3nPAT1W	
<TSG3n_base> + $06C_H$	TSG3nDTC0W	
<TSG3n_base> + 070_H	TSG3nDTC1W	

Table 19.43 Example of DMA Transfer Order of Registers to be Reloaded

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 140_H	TSG3nDCMP2E	
<TSG3n_base> + 144_H	TSG3nDCMP1E	
<TSG3n_base> + 148_H	TSG3nDCMP0E	
<TSG3n_base> + $14C_H$	TSG3nCMP0E	
<TSG3n_base> + 150_H	TSG3nCMP12E	
<TSG3n_base> + 154_H	TSG3nCMP11E	
<TSG3n_base> + 158_H	TSG3nCMP8E	
<TSG3n_base> + $15C_H$	TSG3nCMP7E	
<TSG3n_base> + 160_H	TSG3nCMP4E	
<TSG3n_base> + 164_H	TSG3nCMP3E	
<TSG3n_base> + 168_H	TSG3nCMP10E	
<TSG3n_base> + $16C_H$	TSG3nCMP9E	
<TSG3n_base> + 170_H	TSG3nCMP6E	
<TSG3n_base> + 174_H	TSG3nCMP5E	
<TSG3n_base> + 178_H	TSG3nCMP2E	
<TSG3n_base> + $17C_H$	TSG3nCMP1E	

Table 19.44 Duty Setting in HT-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 180_H	TSG3nCMPWE	
<TSG3n_base> + 184_H	TSG3nCMPVE	
<TSG3n_base> + 188_H	TSG3nCMPUE	

Table 19.45 Active Width Setting in SP-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 190_H	TSG3nWPWE	↓
<TSG3n_base> + 194_H	TSG3nVPWE	
<TSG3n_base> + 198_H	TSG3nUPWE	

Table 19.46 Shift Width and Duty Setting in HSP-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG3n_base> + 120_H	TSG3nHSPSHWE	↓
<TSG3n_base> + 124_H	TSG3nHSPSHVE	
<TSG3n_base> + 128_H	TSG3nHSPSHUE	
<TSG3n_base> + $12C_H$	TSG3nHSPCMWE	
<TSG3n_base> + 130_H	TSG3nHSPCMVE	
<TSG3n_base> + 134_H	TSG3nHSPCMUE	

NOTES

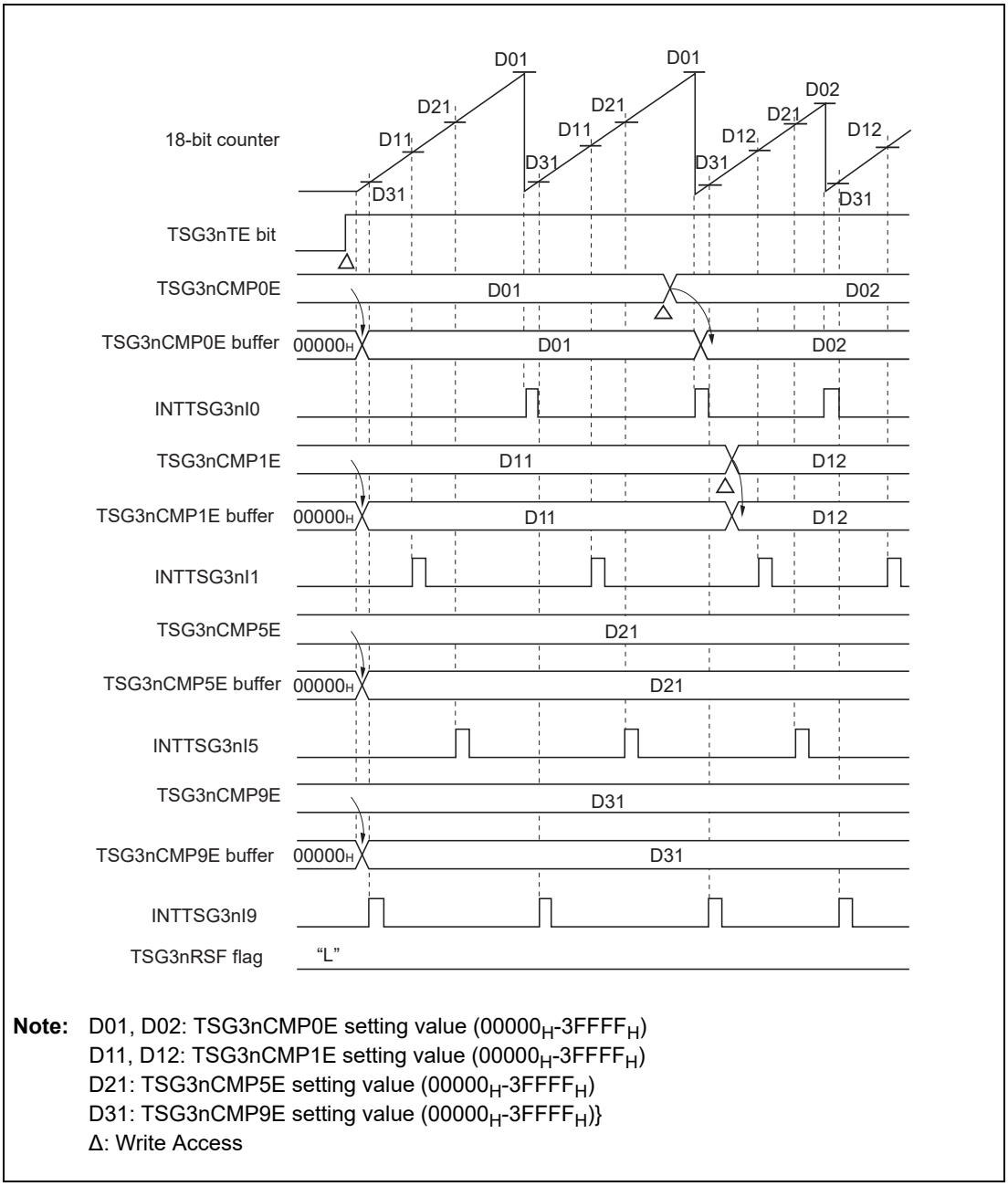
1. TSG3nCTL4 and TSG3nIOC3 should be rewritten individually.
2. Since writing to TSG3nCMP1E (including TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE) enables reloading, it should be rewritten after all the other registers to be reloaded have been rewritten (ready to be reloaded)

(1) Example of Operation in Anytime Rewrite Mode

In this mode, the values written to the compare registers (TSG3nCMP1E to TSG3nCMP12E) are transferred to the internal buffer registers immediately, and are compared with the counter value.

The values are transferred to the internal compare buffer registers one clock cycle (PCLK) after being written to the compare registers (TSG3nCMP1E to TSG3nCMP12E).

The transfer timing of the TSG3nCMP0E is the peak or trough timing (only in HT-PWM mode) of the 18-bit counter after being written to the compare registers, or at the match timing of the TSG3nCMP0E value with the 18-bit counter value (in any mode other than HT-PWM mode).



Note: D01, D02: TSG3nCMP0E setting value (00000_H-$3FFFF_H$)
D11, D12: TSG3nCMP1E setting value (00000_H-$3FFFF_H$)
D21: TSG3nCMP5E setting value (00000_H-$3FFFF_H$)
D31: TSG3nCMP9E setting value (00000_H-$3FFFF_H$)}
Δ: Write Access

Figure 19.8 Anytime Rewrite Timing (Example in PWM Mode)

(a) Data reflection on PWM an Anytime Rewrite in HT-PWM

In anytime rewrite operation in HT-PWM mode, the values are transferred to the buffer at the timing to write 1 to the TSG3nIMT bit after the settings of TSG3nCMP1E, 2E, 5E, 6E, 9E, and 10E registers are modified and PWM output is forcibly set/cleared depending on the modified set value.

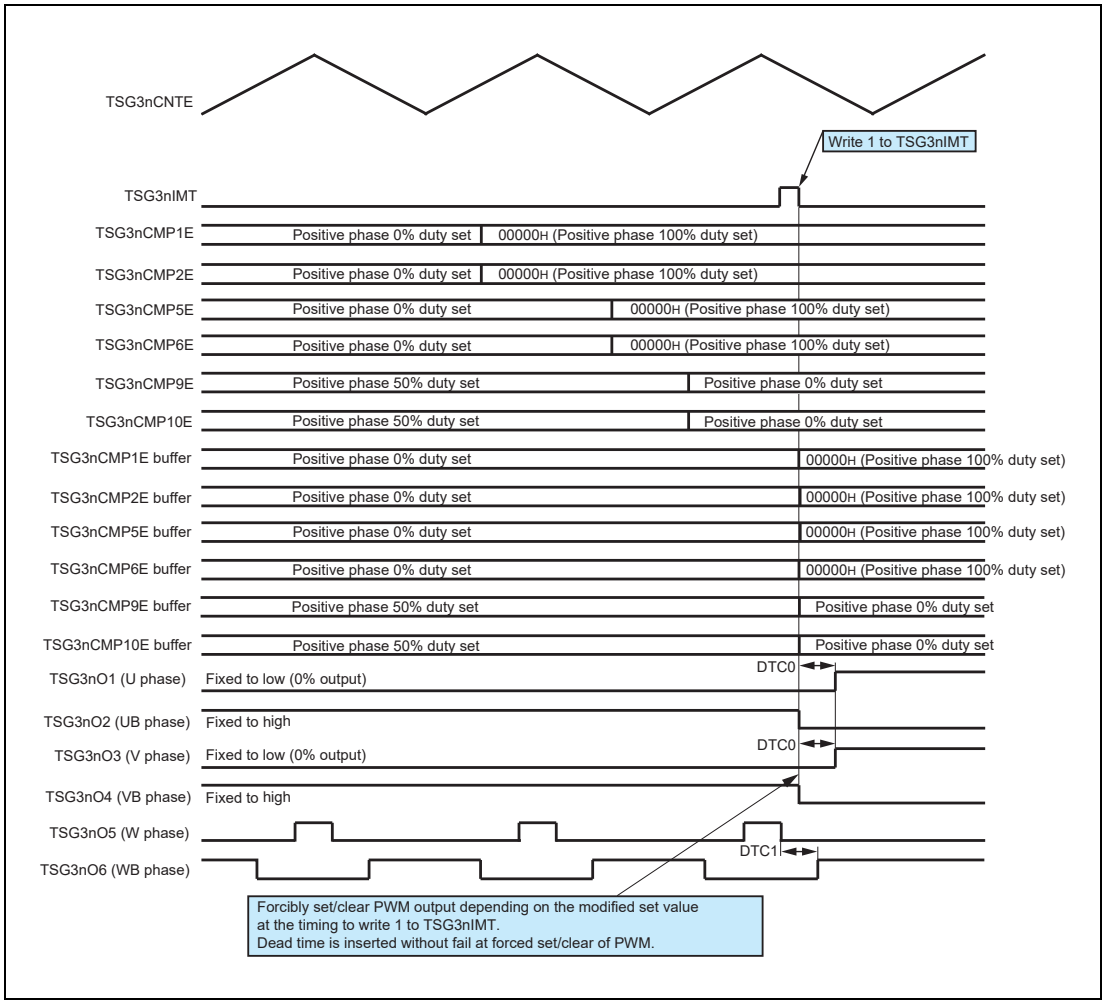


Figure 19.9 Update Timing of TSG3nCMP1E, 2E, 5E, 6E, 9E,and 10E at Anytime Rewrite Operation in HT-PWM Mode

(2) Example of Operation in Reload Mode (Simultaneous Rewrite Function)

The rewritten values of the registers to be reloaded (the registers listed in the **Section 19.3.1, List of Registers** with "Enabled" in the column of "Reload") can be transferred to the corresponding buffer registers simultaneously at the reload timing.

The registers should be rewritten when the pertinent reload request flag (TSG3nSTR0.TSG3nRSF) is 0.

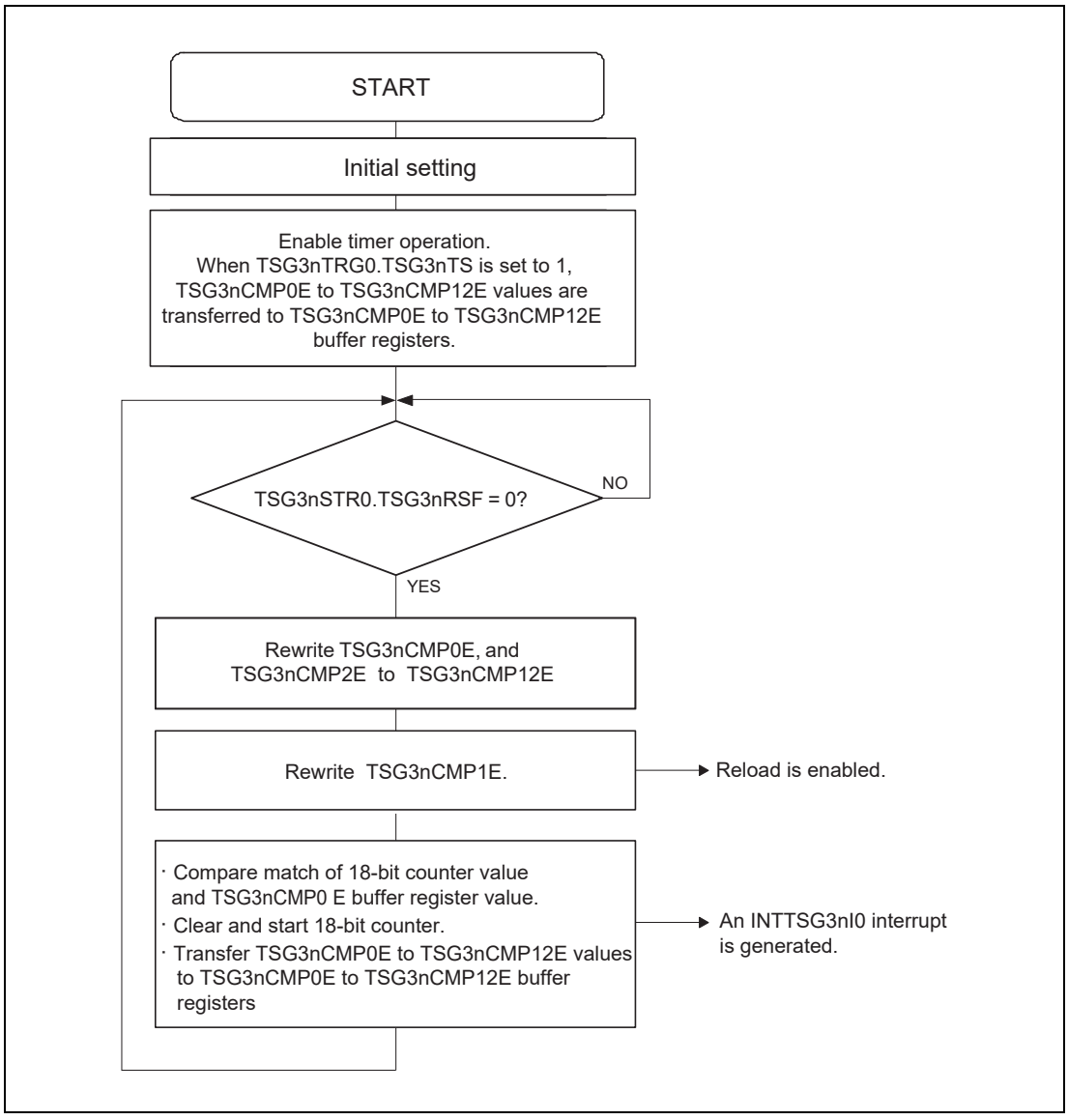


Figure 19.10 Basic Operation Flow in Reload Mode (Simultaneous Rewrite Function) (Example of PWM Mode)

CAUTION

Writing to TSG3nCMP1E also enables reloading. Therefore, TSG3nCMP1E should be rewritten after TSG3nCMP0E and TSG3nCMP2E to TSG3nCMP12E registers have been rewritten.

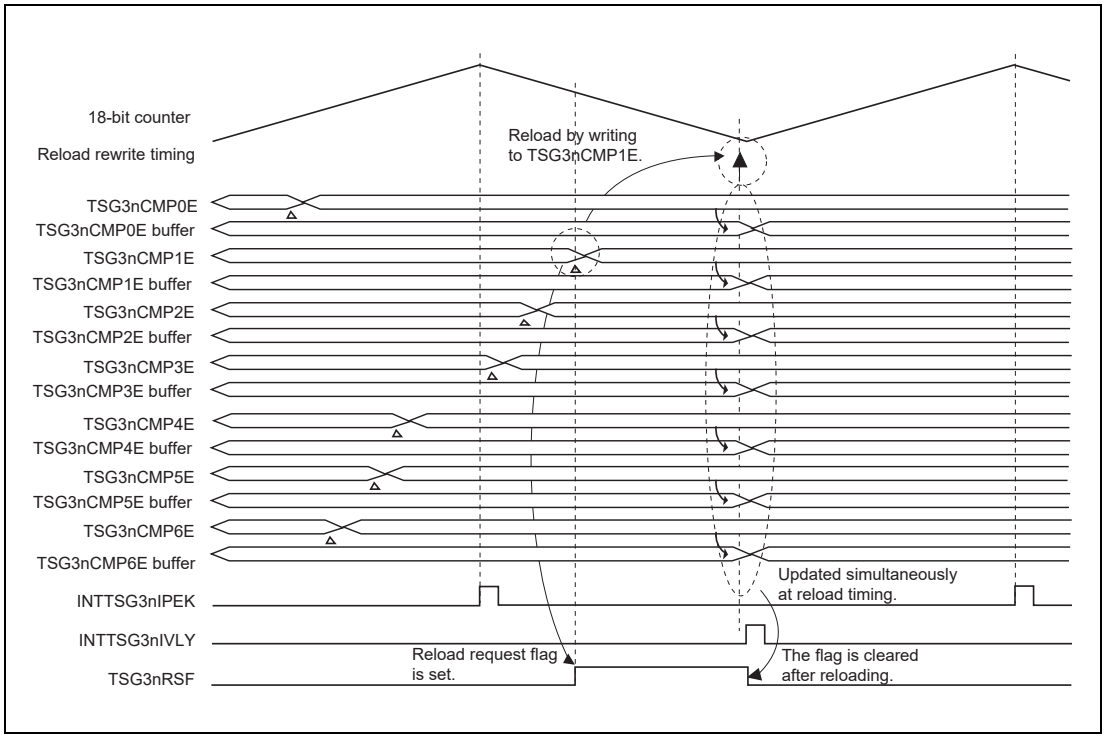


Figure 19.11 Simultaneous Rewrite Timing (Example of HT-PWM Mode) (1/2)

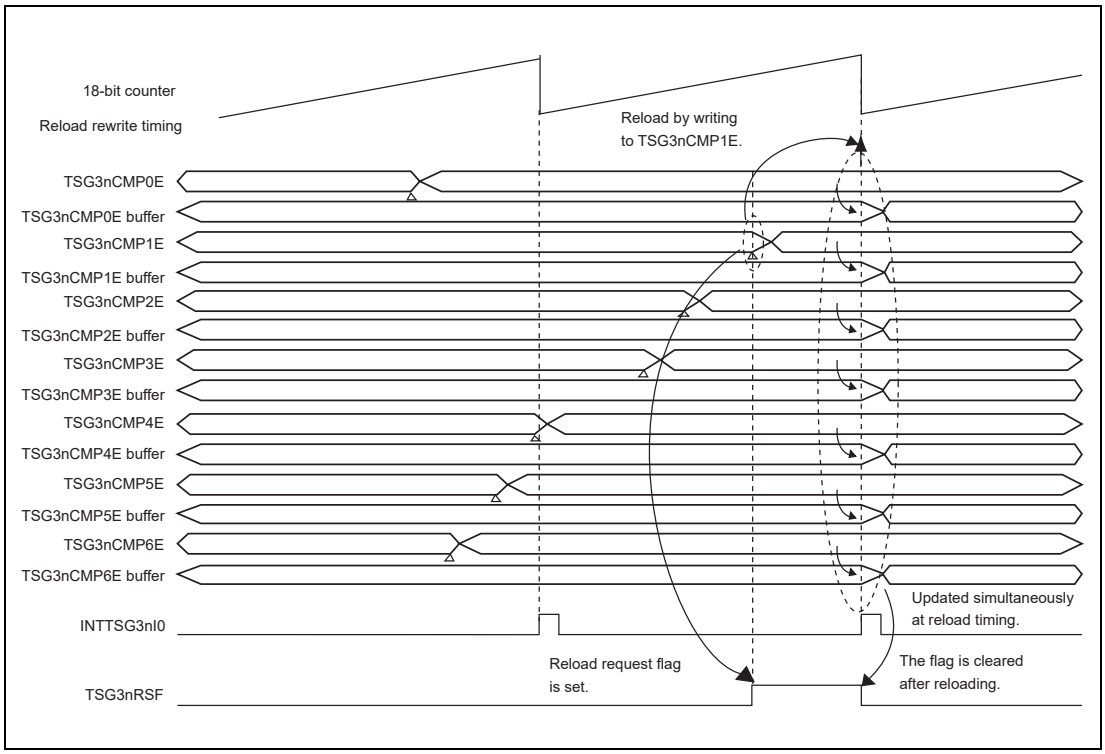


Figure 19.11 Simultaneous Rewrite Timing (Example of PWM Mode) (2/2)

(a) Reload Rewrite Setting Example in Each Mode

Reloading conditions and setting examples are shown in the following tables.

Table 19.47 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 0)

Mode	TSG3nCTL4. TSG3nPRE	TSG3nCTL4. TSG3nVRE	TSG3nCTL4. TSG3nPIE	TSG3nCTL4. TSG3nVIE	TSG3nCTL4. TSG3nRCC04- TSG3nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode HSP-PWM mode	0	0/1	0/1	0/1	Any value	Setting prohibited
	1	0	0/1	0/1	Any value	When INTTSG3nI0 is generated.
	1	1	0/1	0/1	Any value	When INTTSG3nI0 is generated.
HT-PWM mode	0	0	0/1	0/1	Any value	Setting prohibited
	0	1	0/1	0/1	Any value	When INTTSG3nIVLY is generated.
	1	0	0/1	0/1	Any value	When INTTSG3nIPEK is generated
	1	1	0/1	0/1	Any value	When INTTSG3nIPEK or INTTSG3nIVLY is generated.

Table 19.48 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 1)

Mode	TSG3nCTL4. TSG3nPRE	TSG3nCTL4. TSG3nVRE	TSG3nCTL4. TSG3nPIE	TSG3nCTL4. TSG3nVIE	TSG3nCTL4. TSG3nRCC04- TSG3nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode HSP-PWM mode	0	0/1	0/1	0/1	Any value	Setting prohibited
	1	0	0	0/1	Any value	Setting prohibited
	1	0	1	0/1	Any value	When INTTSG3nI0 is generated
	1	1	0	0/1	Any value	Setting prohibited
	1	1	1	0/1	Any value	When INTTSG3nI0 is generated
HT-PWM mode	0	0	0/1	0/1	Any value	Setting prohibited
	0	1	0	0	Any value	Setting prohibited
	0	1	0	1	Any value	When INTTSG3nIVLY is generated
	0	1	1	0	Any value	Setting prohibited
	0	1	1	1	Any value	When INTTSG3nIVLY is generated
	1	0	0	0/1	Any value	Setting prohibited
	1	0	1	0/1	Any value	When INTTSG3nIPEK is generated
	1	1	0	0	Any value	Setting prohibited
	1	1	0	1	Any value	When INTTSG3nIVLY is generated
	1	1	1	0	Any value	When INTTSG3nIPEK is generated
	1	1	1	1	Any value	When INTTSG3nIPEK or INTTSG3nIVLY is generated

19.4.1.4 List or Outputs in Each Mode

The list of timer outputs (TSG3nO0-7 pins) in each mode is shown in the following tables.

Table 19.49 List of Timer Outputs in Each mode (1/3)

Operating Mode	TSG3nO0 Pin	TSG3nO1 Pin	TSG3nO2 Pin
PWM mode	— (Fixed to low)	Outputs a PWM signal by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs a PWM signal by compare match of TSG3nCMP3E and TSG3nCMP4E.
HT-PWM mode	Outputs the status indicating whether the 18-bit counter or 18-bit sub-counter is incremented or decremented.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO1 pin.
SP-PWM mode	— (Fixed to low)	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO1.
120-DC mode	— (Fixed to low)	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.
HSP-PWM mode	— (Fixed to low)	Outputs a PWM signal by compare match of TSG3nCMP1E and TSG3nCMP2E.	Outputs a PWM signal by compare match of TSG3nCMP3E and TSG3nCMP4E.

Table 19.49 List of Timer Outputs in Each mode (2/3)

Operating Mode	TSG3nO3 Pin	TSG3nO4 Pin	TSG3nO5 Pin
PWM mode	Outputs a PWM signal by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs a PWM signal by compare match of TSG3nCMP7E and TSG3nCMP8E.	Outputs a PWM signal by compare match of TSG3nCMP9E and TSG3nCMP10E.
HT-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO3.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP9E and TSG3nCMP10E.
SP-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs an inverse phase PWM signal (with dead time) to TSG3nO3.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG3nCMP9E and TSG3nCMP10E.
120-DC mode	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.	Outputs a PWM signal by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E.
HSP-PWM mode	Outputs a PWM signal by compare match of TSG3nCMP5E and TSG3nCMP6E.	Outputs a PWM signal by compare match of TSG3nCMP7E and TSG3nCMP8E.	Outputs a PWM signal by compare match of TSG3nCMP9E and TSG3nCMP10E.

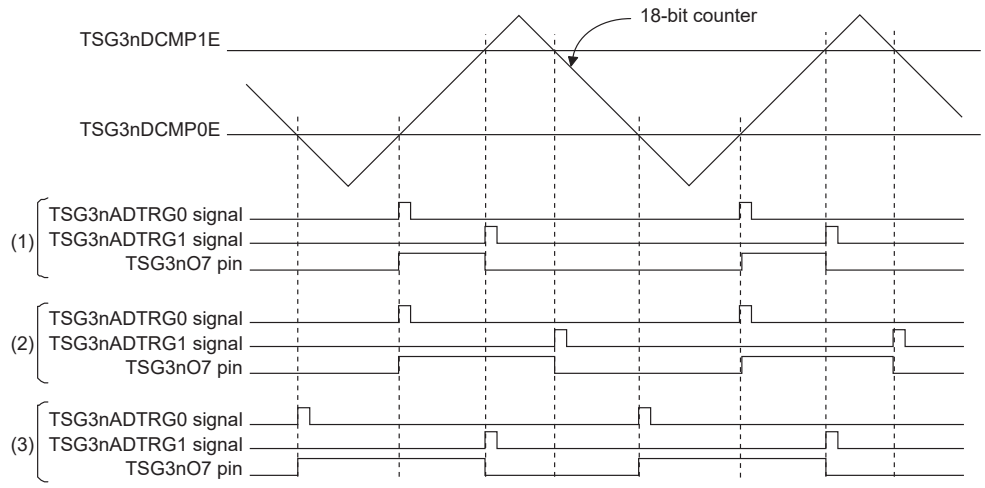
Table 19.49 List of Timer Outputs in Each mode (3/3)

Operating Mode	TSG3nO6 Pin	TSG3nO7 Pin
PWM mode	Outputs a PWM signal by compare match of TSG3nCMP11E and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1
HT-PWM mode	Outputs an inverse phase PWM signal (with dead time) to TSG3nO5.	Outputs a diagnostic signal or A/D conversion trigger.*1
SP-PWM mode	Outputs an inverse phase PWM signal (with dead time) to TSG3nO5.	Outputs a diagnostic signal or A/D conversion trigger.*1
120-DC mode	Outputs a PWM signal by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1
HSP-PWM mode	Outputs a PWM signal by compare match of TSG3nCMP11E and TSG3nCMP12E.	Outputs a diagnostic signal or A/D conversion trigger.*1

Note 1. For TSG3nO7, See **Section 19.4.1.4 (a), TSG3nO7 Pin Output Control**

(a) TSG3nO7 Pin Output Control

The TSG3nO7 pin can output a pulse of A/D conversion trigger (TSG3nIOC1.TSG3nTGS = 0) or diagnostic output (TSG3nIOC1.TSG3nTGS = 1). When outputting a pulse of A/D conversion trigger, the TSG3nO7 pin is activated at the rising edge of the TSG3nADTRG0 signal, and inactivated at the rising edge of the TSG3nADTRG1 signal. When the TSG3nADTRG0 signal is detected while the TSG3nO7 pin is active, the TSG3nO7 pin remains active. When the TSG3nADTRG1 signal is detected while the TSG3nO7 pin is inactive, the TSG3nO7 pin remains inactive. If TSG3nADTRG0 and TSG3nADTRG1 signal triggers occur simultaneously, the TSG3nO7 pin is inactivated.



Note: (1) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0004_H, and TSG3nCTL6 register = 0010_H

(2) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0004_H, and TSG3nCTL6 register = 0020_H

(3) When TSG3nDCMP0E register<TSG3nDCMP1E register, TSG3nCTL5 register = 0008_H, and TSG3nCTL6 register = 0010_H

Figure 19.12 Example of A/D Trigger Output Timing of TSG3nO7 Pin (TSG3nIOC1.TSG3nTGS = 0)

During diagnostic output, the active level is output on the TSG3nO7 pin with the width specified by the TSG3nCTL0.TSG3nDWD bit when the values of the TSG3nDCMP0E to TSG3nDCMP2E bits match that of the 18-bit counter. If a TSG3nDCMP0E to TSG3nDCMP2E value again matches the value of the 18-bit counter value while the diagnostic output on the TSG3nO7 pin is already at the active level, pulse output on the TSG3nO7 pin continues for the number of cycles of PCLK (8 or 16) set by the TSG3nDWD bit from the point of the later match.

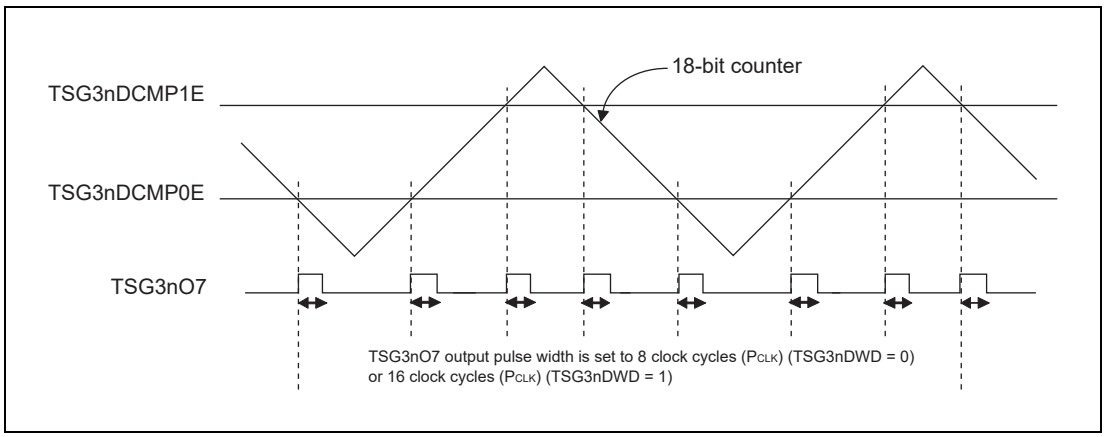


Figure 19.13 Example of TSG3nO7 Pin Diagnostic Pulse Output Timing (1) (TSG3nIOC1.TSG3nTGS = 1)

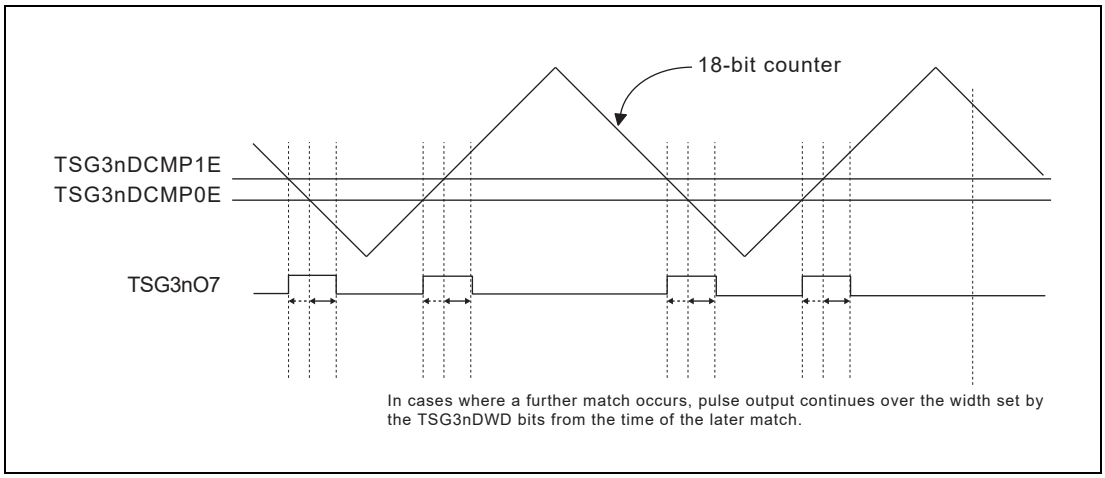


Figure 19.14 Example of TSG3nO7 Pin Diagnostic Pulse Output Timing (2) (with Pulse Output Width Overlapped)

19.4.2 Match Interrupt

The TSG3n can generate interrupts such as a compare match interrupt (INTTSG3nIm), a peak interrupt (INTTSG3nIPEK), and a trough interrupt (INTTSG3nIVLY). For an error interrupt and warning interrupt (INTTSG3nIER and INTTSG3nIWN), see **Section 19.4.6, Error/Warning Interrupt**.

A period interrupt (INTTSG3nI0) is generated for each timer period. In HT-PWM mode, it is generated when the TSG3nDTC0 buffer register value matches with the 18-bit counter value. When the 18-bit counter performs sawtooth waveform operation (PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode), it is generated after the 18-bit counter value has matched with the TSG3nCMP0E buffer register value.

A compare-match interrupt (INTTSG3nIm) is generated by a match of the TSG3nCMPmE buffer register value with the 18-bit counter value depending on the compare register to be used in each operating mode (m = 1 to 12).

A peak interrupt (INTTSG3nIPEK) is generated in all the modes. In HT-PWM mode, it is generated when the 18-bit counter switches from incrementing to decrementing. When the 18-bit counter performs sawtooth waveform operation (PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode), it is generated after the 18-bit counter value has matched with the TSG3nCMP0E buffer register value (the same timing as an INTTSG3nI0 interrupt).

A trough interrupt (INTTSG3nIVLY) is generated when the 18-bit counter switches from decrementing to incrementing in HT-PWM mode.

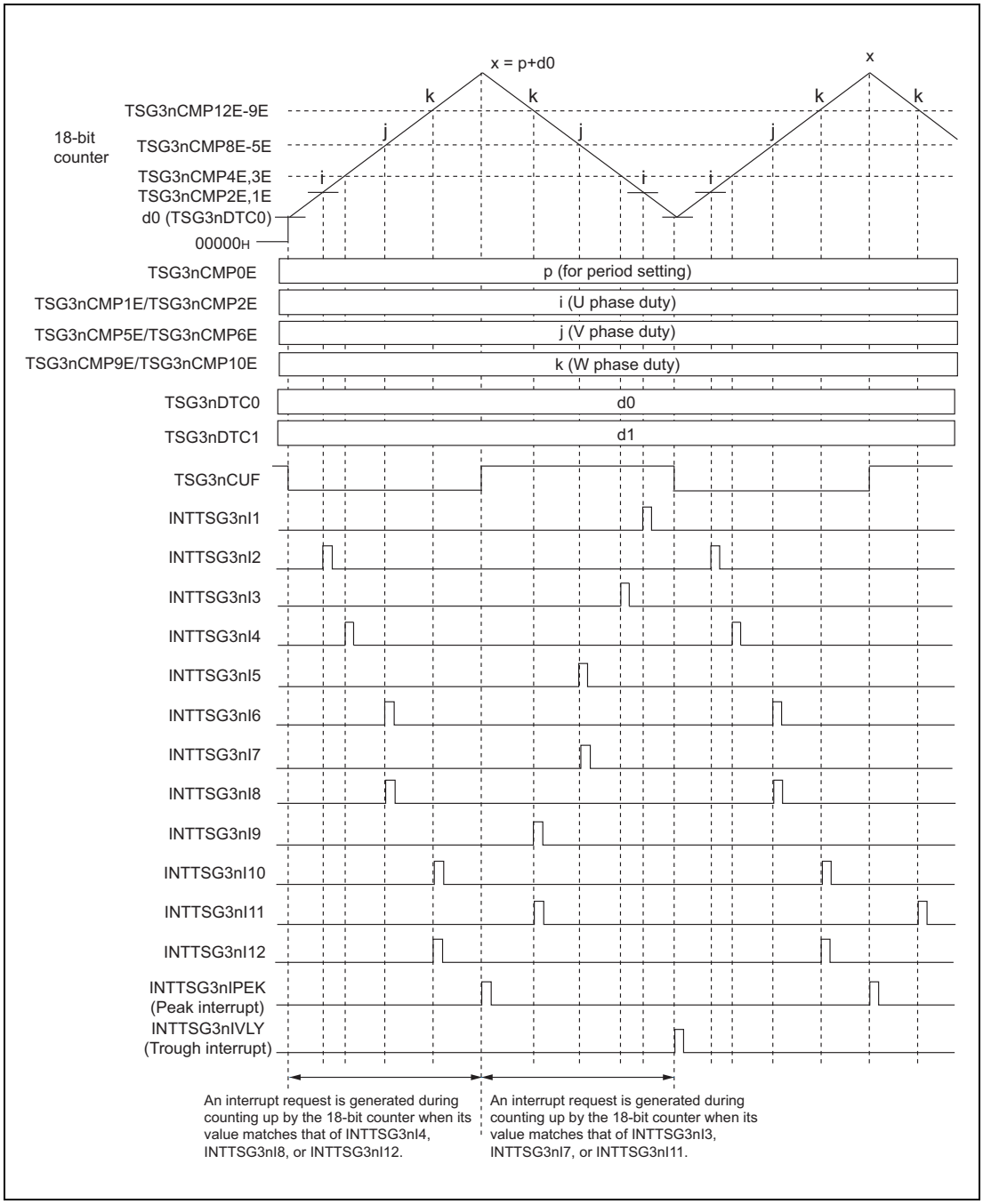


Figure 19.15 Interrupt Generation Example (Example of HT-PWM Mode)

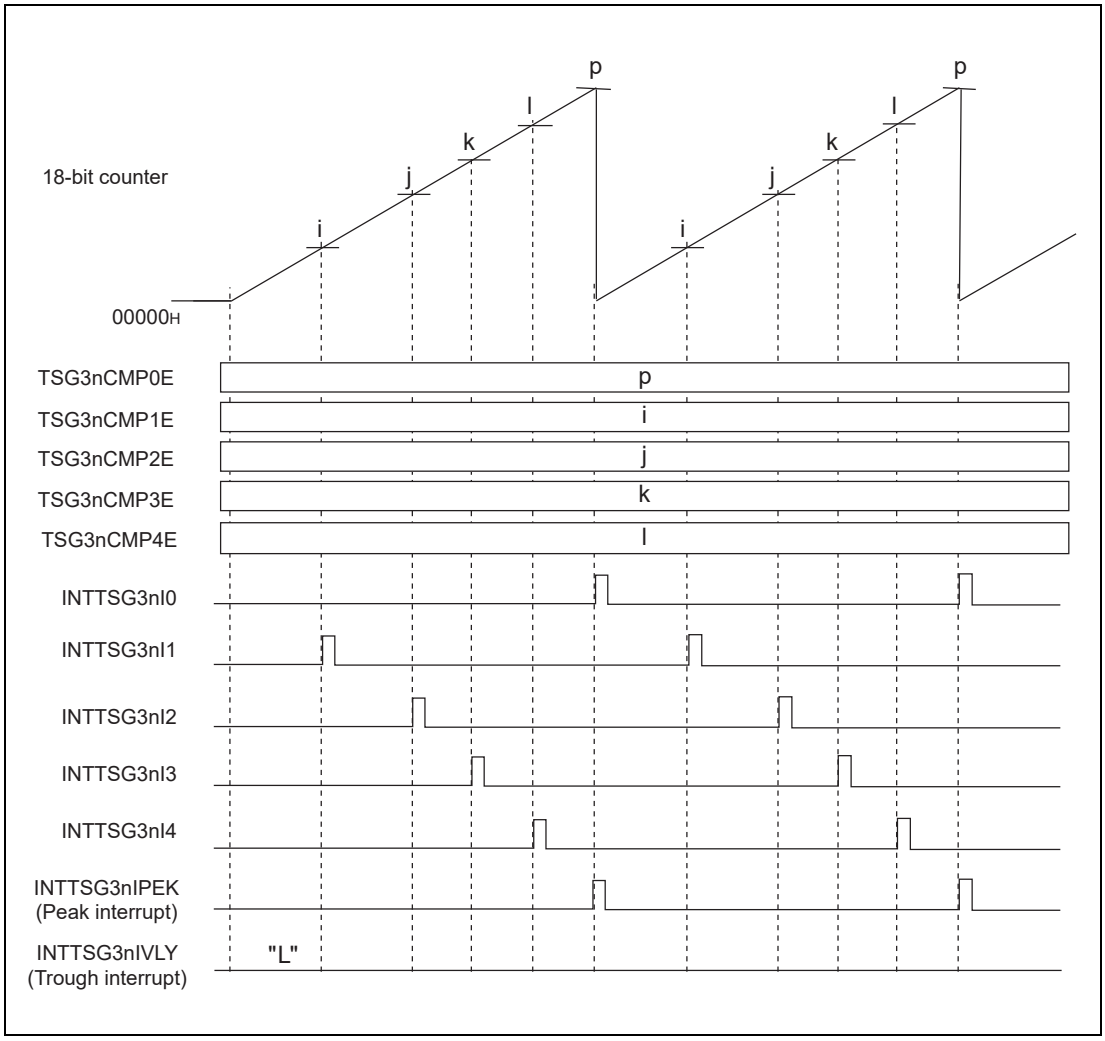


Figure 19.16 Interrupt Generation Example (Example of PWM Mode)

Interrupt in each mode (INTTSG3nI0-INTTSG3nI12, INTTSG3nIPEK, INTTSG3nIVLY, INTTSG3nIER, and INTTSG3nIWN) are listed in **Table 19.50.**

Table 19.50 List of Interrupts in Each Mode (1/5)

Operating Mode	INTTSG3nI0	INTTSG3nI1	INTTSG3nI2	INTTSG3nI3
PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1
HT-PWM mode	Period interrupt	TSG3nCMP1E compare match interrupt*2	TSG3nCMP1E compare match interrupt*2	TSG3nCMP3E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	—
120-DC mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1
HSP-PWM mode	TSG3nCMP0E compare match interrupt	TSG3nCMP1E compare match interrupt*1	TSG3nCMP2E compare match interrupt*1	TSG3nCMP3E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 19.50 List of Interrupts in Each Mode (2/5)

Operating Mode	INTTSG3nI4	INTTSG3nI5	INTTSG3nI6	INTTSG3nI7
PWM mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1
HT-PWM mode	TSG3nCMP4E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP5E compare match interrupt*2	TSG3nCMP6E compare match interrupt*2	TSG3nCMP7E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	—	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	—
120-DC mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1
HSP-PWM mode	TSG3nCMP4E compare match interrupt*1	TSG3nCMP5E compare match interrupt*1	TSG3nCMP6E compare match interrupt*1	TSG3nCMP7E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 19.50 List of Interrupts in Each Mode (3/5)

Operating Mode	INTTSG3nI8	INTTSG3nI9	INTTSG3nI10	INTTSG3nI11
PWM mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1
HT-PWM mode	TSG3nCMP8E compare match interrupt*2 when incrementing (TSG3nCUF = 0)	TSG3nCMP9E compare match interrupt*2	TSG3nCMP10E compare match interrupt*2	TSG3nCMP11E compare match interrupt*2 when decrementing (TSG3nCUF = 1)
SP-PWM mode	—	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	—
120-DC mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1
HSP-PWM mode	TSG3nCMP8E compare match interrupt*1	TSG3nCMP9E compare match interrupt*1	TSG3nCMP10E compare match interrupt*1	TSG3nCMP11E compare match interrupt*1

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 19.50 List of Interrupts in Each Mode (4/5)

Operating Mode	INTTSG3nI12	INTTSG3nIPEK	INTTSG3nIVLY
PWM mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—
HT-PWM mode	TSG3nCMP12E compare match interrupt*2 when incrementing (TSG3nCUF=0)	Peak interrupt	Trough interrupt
SP-PWM mode	—	Peak interrupt at the same timing with INTTSG3nI0	—
120-DC mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—
HSP-PWM mode	TSG3nCMP12E compare match interrupt*1	Peak interrupt at the same timing with INTTSG3nI0	—

Note 1. A compare match interrupt is not generated when TSG3nCMP0E < TSG3nCMPmE (m = 1 to 12).

Note 2. A compare match interrupt is not generated when $00000_H \leq$ TSG3nCMPmE < TSG3nDTC0, (TSG3nCMP0E + TSG3nDTC0) < TSG3nCMPmE.

Table 19.50 List of Interrupts in Each Mode (5/5)

Operating Mode	INTTSG3nIER	INTTSG3nIWN
PWM mode	Error interrupt	Warning interrupt
HT-PWM mode	Error interrupt	Warning interrupt
SP-PWM mode	Error interrupt	Warning interrupt
120-DC mode	Error interrupt	Warning interrupt
HSP-PWM mode	Error interrupt	Warning interrupt

19.4.3 Flags

Table 19.51 List of Flags

Number	Flag Name	Symbol	Registers	Operating Mode
(1)	Up count flag	TSG3nCUF	TSG3nSTR0	HT-PWM mode
		TSG3nSUF	TSG3nSTR0	
(2)	Positive phase and inverse phase simultaneous active state detection flag	TSG3nTBF0-TSG3nTBF2	TSG3nSTR2	All operating modes
(3)	Reload request flag	TSG3nRSF	TSG3nSTR0	All operating modes
(4)	Noise Detection Flag	TSG3nNDF	TSG3nSTR2	All operating modes
(5)	Pattern order detection flag	TSG3nTSF	TSG3nSTR1	All operating modes
(6)	Pattern error detection flag	TSG3nPEF	TSG3nSTR2	All operating modes
(7)	Pattern reversal detection flag	TSG3nPRF	TSG3nSTR2	All operating modes
(8)	TSG3nPTSI2 to TSG3nPTSI0 pin abnormal toggle detection flag	TSG3nPTF	TSG3nSTR2	All operating modes
(9)	TSG3nOPCI0 and TSG3nOPCI1 signal simultaneous trigger detection flag	TSG3nTDF	TSG3nSTR2	All operating modes
(10)	Pattern phase difference detection flag	TSG3nPPF	TSG3nSTR2	All operating modes
(11)	Timer output pattern flag	TSG3nOPF0 to TSG3nOPF2	TSG3nSTR1	All operating modes
(12)	Pattern switch detection signal (internal signal)	TSG3nPTE	—	All operating modes

19.4.3.1 Up Count Flag (TSG3nCUF and TSG3nSUF)

Name

Up count flag (TSG3nSTR0.TSG3nCUF and TSG3nSUF)

Description

There are following two up count flags.

TSG3nCUF is an up/down count flag of the 18-bit counter.

TSG3nSUF is an up/down count flag of the 18-bit sub-counter.

For both TSG3nCUF and TSG3nSUF, 0 means increment, and 1 means decrement.

TSG3nCUF and TSG3nSUF can be used only in HT-PWM mode.

Example of operation

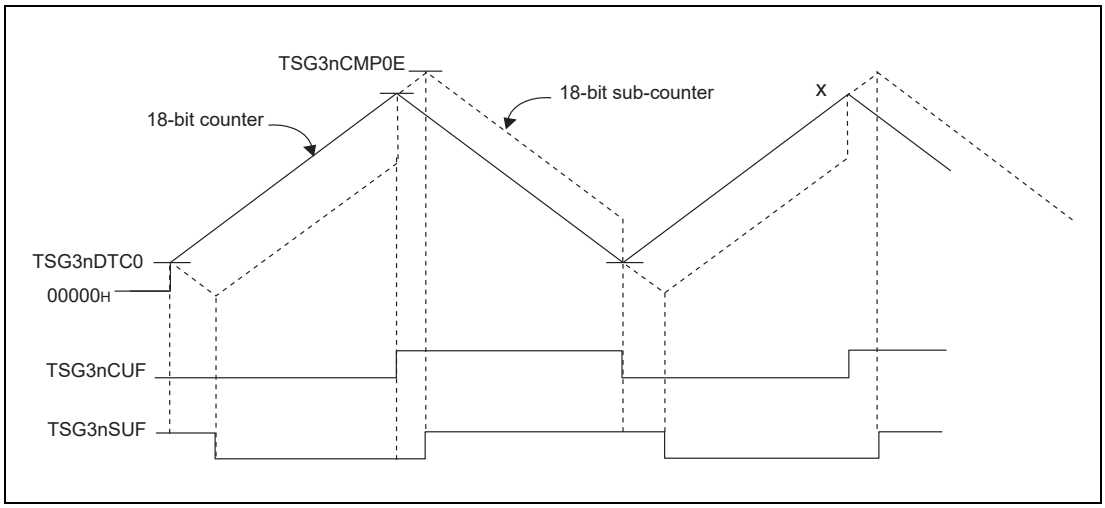


Figure 19.17 Example of Up Count Flag Operation

NOTES

1. TSG3nCUF value is:
   0 (up count) when TSG3nDTC0 ≤ 18-bit counter ≤ (TSG3nCMP0E + TSG3nDTC0-2)
   1 (down count) when (TSG3nCMP0E + TSG3nDTC0) ≥ 18-bit counter ≥ TSG3nDTC0 + 2
2. TSG3nSUF value is:
   0 (up count) when 0 ≤ 18-bit sub-counter ≤ (TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1-2)
   1 (down count) when (TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1) ≥ 18-bit sub-counter ≥ 2

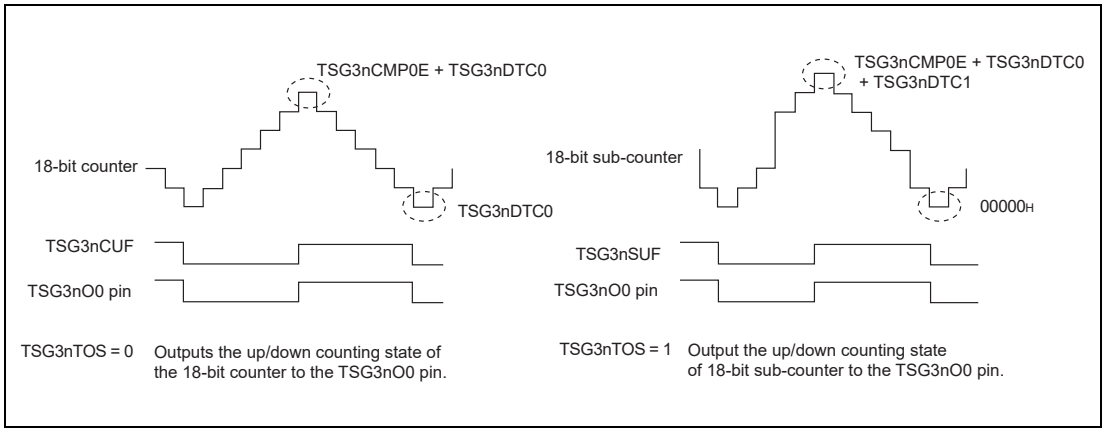


Figure 19.18 TSG3nO0 Pin Output depending on TSG3nIOC1.TSG3nTOS Setting

Operating Mode

TSG3nCUF and TSG3nSUF can be used only in HT-PWM mode.

19.4.3.2 Positive Phase and Inverse Phase Simultaneous Active State Detection Flag (TSG3nTBF0 to TSG3nTBF2)

Name

Positive phase and inverse phase simultaneous active state detection flag (TSG3nSTR2.TSG3nTBF0 to TSG3nTBF2 flags)

Description

When any of TSG3nCTL1.TSG3nTBA2 to TSG3nTBA0 is 1, TSG3nTBF0 to TSG3nTBF2 can detect the simultaneous active state of the positive phase and inverse phase of TSG3n.

When the simultaneous active state of the positive phase and inverse phase of the TSG3n is detected, the corresponding TSG3nTBF0 to TSG3nTBF2 flags are set to 1, and an error interrupt (INTTSG3nIER) is generated. The flags are cleared when 1 is written to TSG3nSTC.TSG3nTBR0 to TSG3nTBR2, respectively.

Example of operation

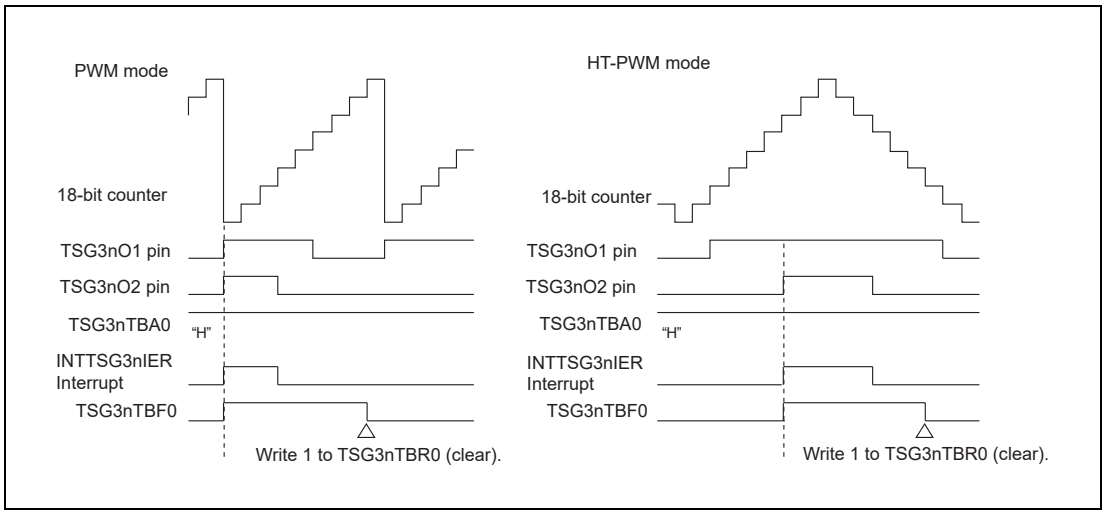


Figure 19.19 Example of Positive Phase and Inverse Phase Simultaneous Active State Detection Flag Operation

Operating Mode

Available in all operating modes.

CAUTION

TSG3nTBF0 to TSG3nTBF2 are valid only when TSG3nCTL1.TSG3nTBA0 to TSG3nTBA2 = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.3.3 Reload Request Flag (TSG3nRSF)

Name

Reload request flag (TSG3nSTR0.TSG3nRSF)

Description

TSG3nRSF is set to 1 when a reload request is generated (when a value is written to TSG3nCMP1E (TSG3nCMP1, TSG3nCMP1W, TSG3nCMPUE, TSG3nCMPU, TSG3nUPWE, TSG3nUPW, and TSG3nHSPCMUE)), and cleared to 0 when the value is transferred to all the buffer registers.

Example of operation

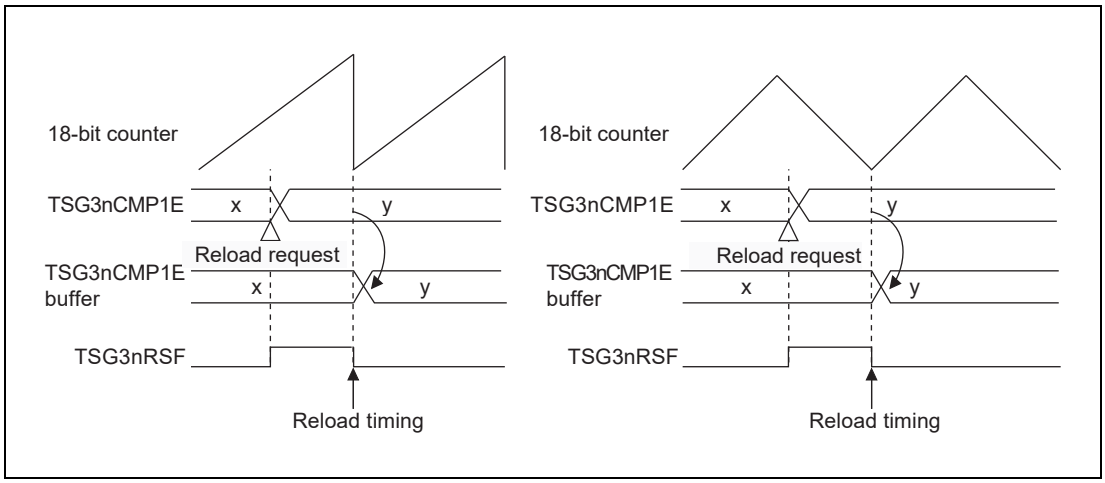


Figure 19.20 Example of Reload Request Flag Operation

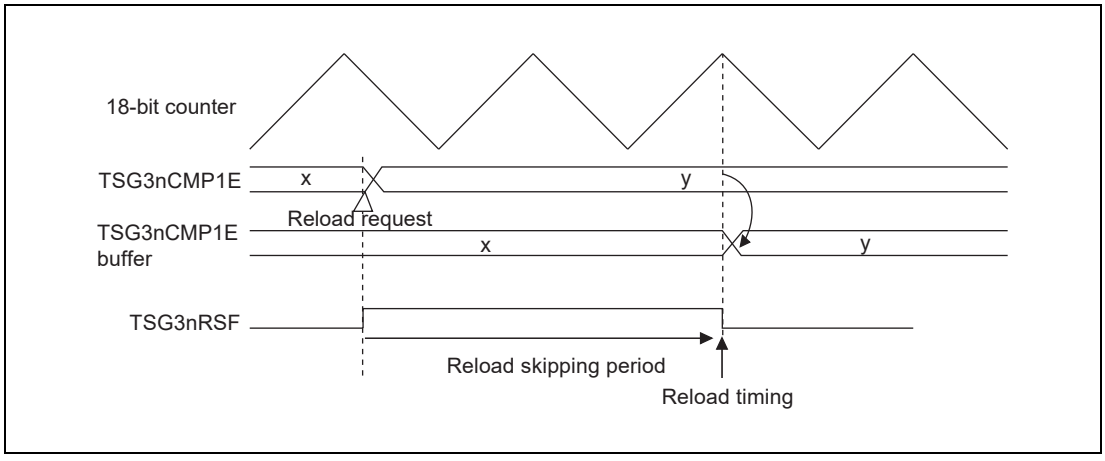


Figure 19.21 Reload Request Flag and Reload Skipping Period

Operating Mode

Available in all operating modes.

19.4.3.4 Noise Detection Flag (TSG3nNDF)

Name

Noise detection flag (TSG3nSTR2.TSG3nNDF)

Description

TSG3nNDF can detect that two or more pins of TSG3nPTSI2 to TSG3nPTSI0 have changed simultaneously (a noise is generated).

TSG3nNDF is set to 1 when two or more pins of TSG3nPTSI2 to TSG3nPTSI0 have changed simultaneously (a noise is generated), and a warning interrupt (INTTSG3nIWN) is generated. The TSG3nNDF flag is cleared to 0 when 1 is written to the TSG3nSTC.TSG3nNDR bit.

Example of operation

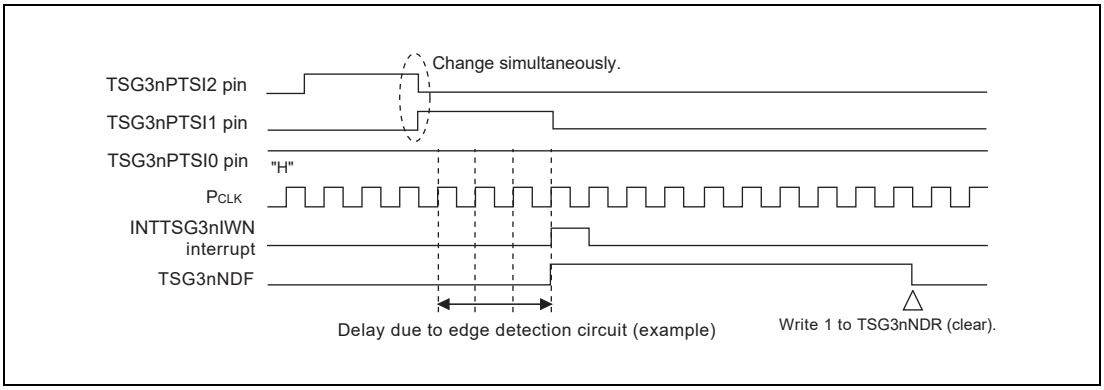


Figure 19.22 Example of Noise Detection Flag Operation

Operation mode

Available in all operating modes.

CAUTION

TSG3nNDF is valid only when TSG3nCTL1.TSG3nNDC = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.3.5 Pattern Order Detection Flag (TSG3nTSF)

Name

Pattern order detection flag (TSG3nSTR1.TSG3nTSF)

Description

TSG3nTSF can detect the order of patterns input to the TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nTSF is set depending on the values input to the TSG3nPTSI2 to TSG3nPTSI0 pins as shown in the table below

Table 19.52 Pattern Order Detection Flag and Pattern Input Order

TSG3nTSF	Values input to TSG3nPTSI2 to TSG3nPTSI0 Pins
0	[1,0,1] → [1,0,0] → [1,1,0] → [0,1,0] → [0,1,1] → [0,0,1]
1	[1,0,1] ← [1,0,0] ← [1,1,0] ← [0,1,0] ← [0,1,1] ← [0,0,1]

Example of operation

(a) When Normal Input to TSG3nPTSI2 to TSG3nPTSI0 Pins is Detected

As shown in **Figure 19.23**, if the TSG3nPTSI2 to TSG3nPTSI0 pins change in the normal order, 0 or 1 is set according to the change order at the change timing.

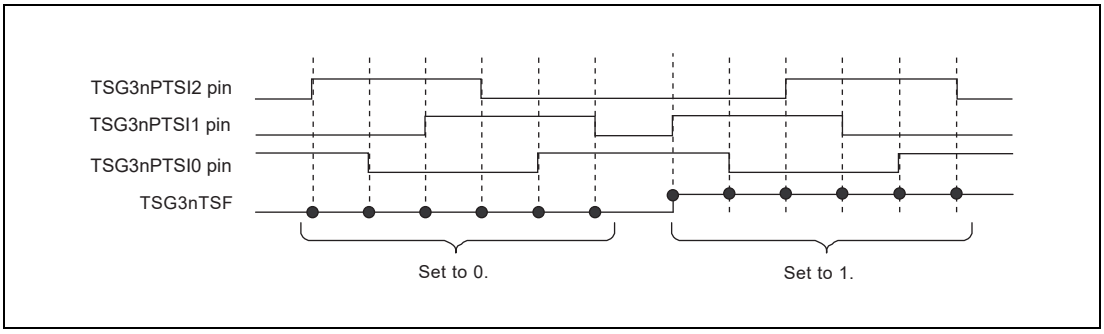


Figure 19.23 Example of Pattern Order Detection Flag Operation (Normal Operation)

(b) Detection of Input Pattern Order

Immediately after TSG3n starts operation, the rotation direction cannot be determined. Therefore, TSG3nTSF cannot detect the change (normal or reverse rotation) in the patterns input to the TSG3nPTSI2 to TSG3nPTSI0 pins. To enable detection of change immediately after the beginning of operation, TSG3nPSC should be set before operation starts (when TSG3nTE = 0, the TSG3nPSC value is reflected).

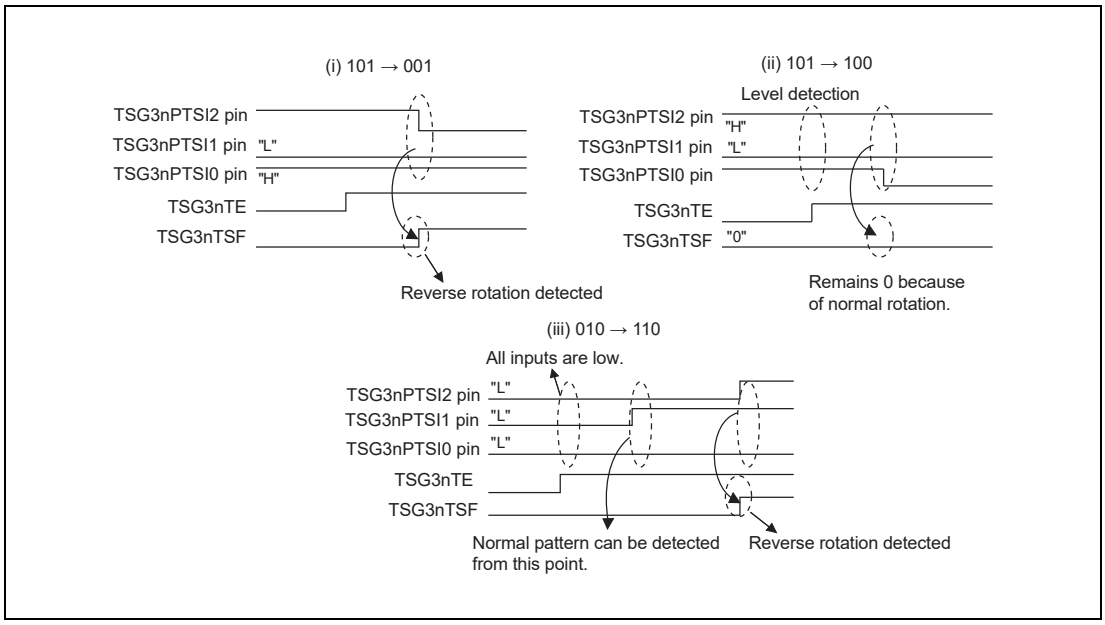


Figure 19.24 Example of Detecting Change (Normal/Reverse Rotation) in Pattern Input to TSG3nPTSI2 to TSG3nPTSI0 Pins

(c) When Abnormal Input to TSG3nPTSI2 to TSG3nPTSI0 Pins is Detected

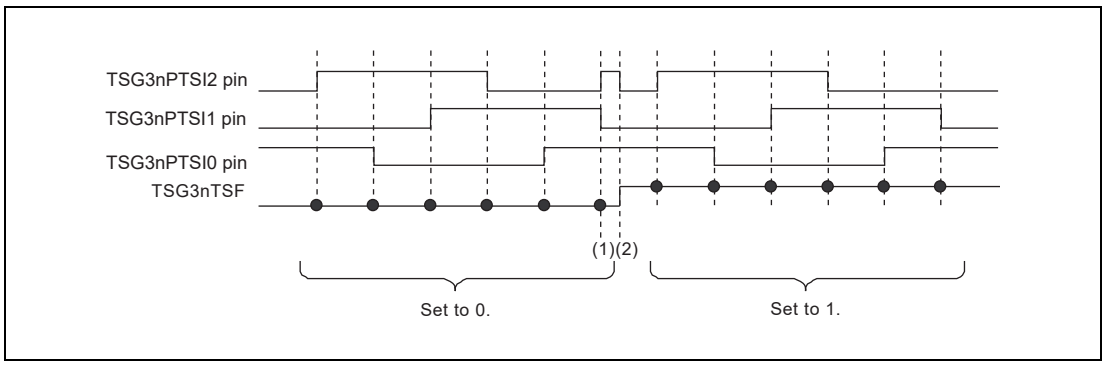


Figure 19.25 Example of Operation when Values Input to Two Pins of TSG3nPTSI2 to TSG3nPTSI0 Change (Abnormal Operation)

(1) TSG3nTSF does not change at this point because it expects the input pattern change to {0, 1, 0} or {0, 0, 1} (if values of two pins change, TSG3nTSF does not change).

(2) TSG3nPTSI2 to TSG3nPTSI0 pins are determined to have been changed from {1, 0, 1} to {0, 0,1}, and TSG3nTSF is set to 1.

Operation mode

Available in all operating modes.

19.4.3.6 Pattern Error Detection Flag (TSG3nPEF)

Name

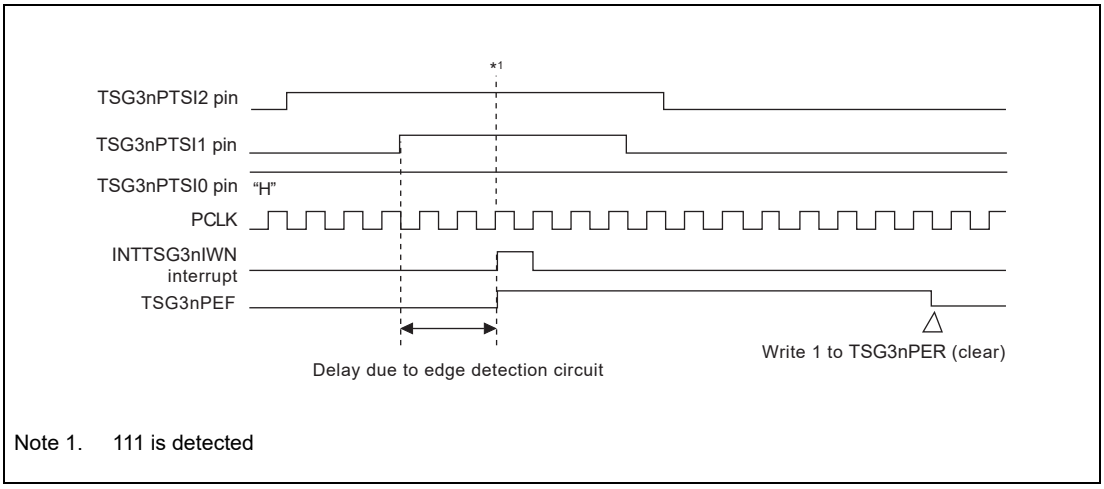
Pattern error detection flag (TSG3nSTR2.TSG3nPEF)

Description

TSG3nPEF can detect that 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nPEF is set to 1 when the levels of the TSG3nPTSI2 to TSG3nPTSI0 pins are 111 or 000, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPEF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPER.

Example of operation



Note 1. 111 is detected

Figure 19.26 Example of Pattern Error Detection Flag Operation (TSG3nPTSI2 to TSG3nPTSI0 Pins = 111)

Operation mode

Available in all operating modes.

CAUTIONS

TSG3nPEF is valid only when TSG3nCTL1.TSG3nPEC = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.3.7 Pattern Reversal Detection Flag (TSG3nPRF)

Name

Pattern reversal detection flag (TSG3nSTR2.TSG3nPRF)

Description

TSG3nPRF can detect that the pattern change order of the TSG3nPTSI2 to TSG3nPTSI0 pins have been reversed.

TSG3nPRF is set to 1 when the pattern order detection flag (TSG3nTSF) changes, and a warning interrupt (INTTSG3nIWN) is generated. However, immediately after TSG3nSTR0.TSG3nTE is set to 1, TSG3nPRF is valid at the timing of the second and subsequent change in TSG3nPTSI2 to TSG3nPTSI0 pins.

TSG3nPRF is cleared to 0 when 1 is written to the TSG3nSTC.TSG3nPRR bit.

Example of operation

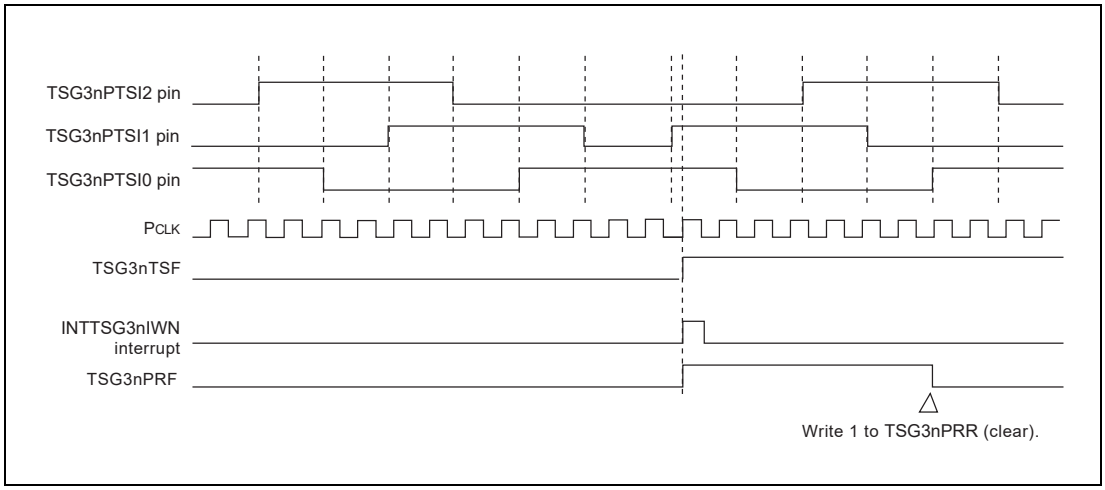
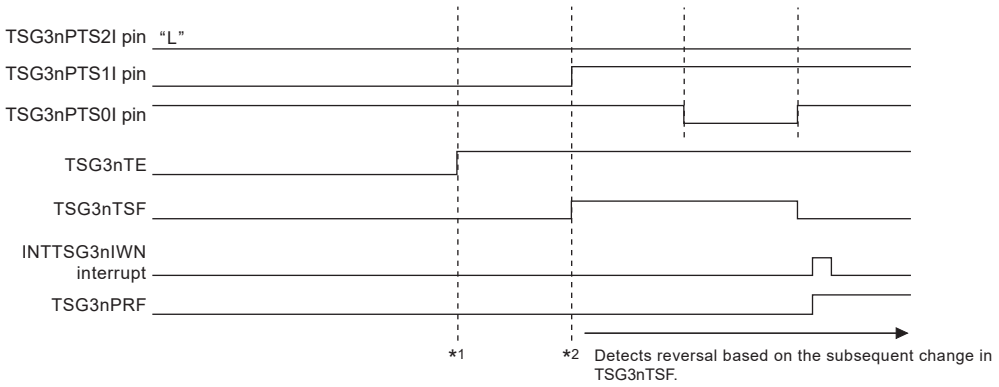


Figure 19.27 Example of Pattern Reversal Detection Flag Operation



Note 1. Operation starts.

Note 2. If TSG3nTSF is set to 1 by the first change in the TSG3nPTSI2 to TSG3nPTSI0 pins immediately after TSG3nTE is set to 1, reversal is not detected.

Figure 19.28 Example of Operation immediately after TSG3nTE Flag in TSG3nSTR0 is Set to 1

Operation mode

Available in all operating modes.

CAUTION

TSG3nPRF is valid only when TSG3nCTL1.TSG3nPRC = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.3.8 TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag (TSG3nPTF)

Name

TSG3nPTSI2 to TSG3nPTSI0 pin abnormal toggle detection flag (TSG3nSTR2.TSG3nPTF)

Description

TSG3nPTF can detect that the values of the TSG3nPTSI2 to TSG3nPTSI0 pins change three or more times during the TSG3nOPCI0 or TSG3nOPCI1 signal trigger.

TSG3nPTF is set to 1 when the third trigger of TSG3nOPCI0 or TSG3nOPCI1 signal occurs simultaneously with the change in TSG3nPTSI2 to TSG3nPTSI0, and a warning interrupt (INTTSG3nIWN) is generated.

TSG3nPTF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPTR.

Example of operation

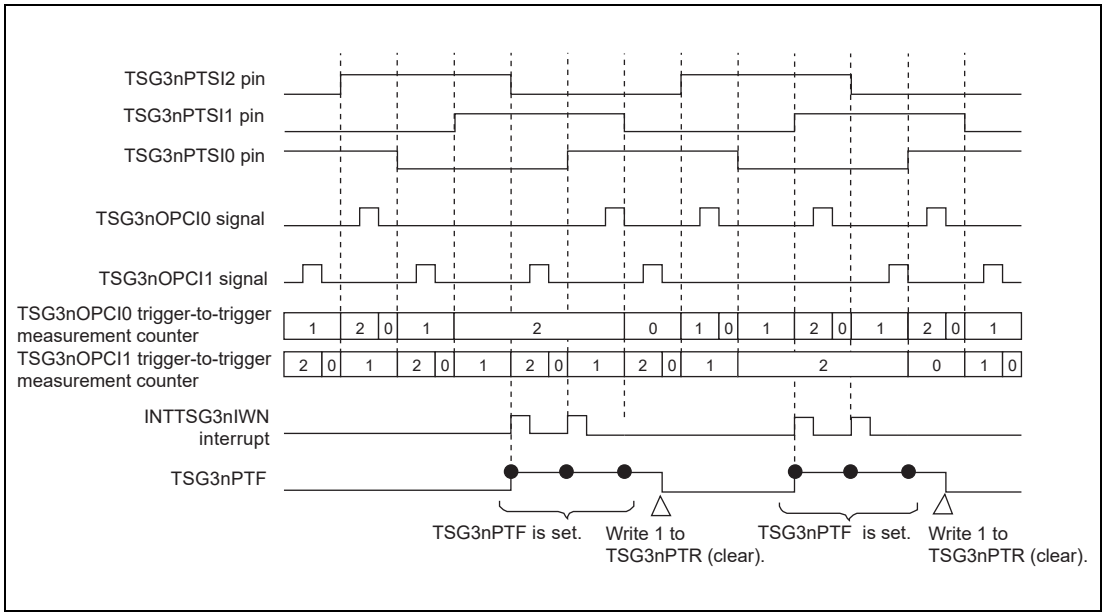


Figure 19.29 Operation of TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag Operation

Operating mode

Available in all operating modes.

NOTES

1. TSG3nPTF is valid only when TSG3nCTL1.TSG3nPTC1 bit = 1 and TSG3nSTR0.TSG3nTE = 1.
2. When TSG3nPTC0 bit = 1 and TSG3nPTC1 bit = 1, TSG3nO1 to TSG3nO6 pin output switch control is automatically switched to pattern switch method (TSG3nOPT0.TSG3nPOT bit = 0) if an abnormal toggle is detected.

19.4.3.9 TSG3nOPCI0, TSG3nOPCI1 Signal Simultaneous Trigger Detection Flag (TSG3nTDF)

Name

TSG3nOPCI0 and TSG3nOPCI1 signal simultaneous trigger detection flag (TSG3nSTR2.TSG3nTDF)

Description

TSG3nTDF can detect that TSG3nOPCI0 and TSG3nOPCI1 signals are generated simultaneously.

TSG3nTDF is set to 1 when the TSG3nOPCI0 and TSG3nOPCI1 signals are generated simultaneously, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nTDF is cleared to 0 when 1 is written to TSG3nSTC.TSG3nTDR.

Example of operation

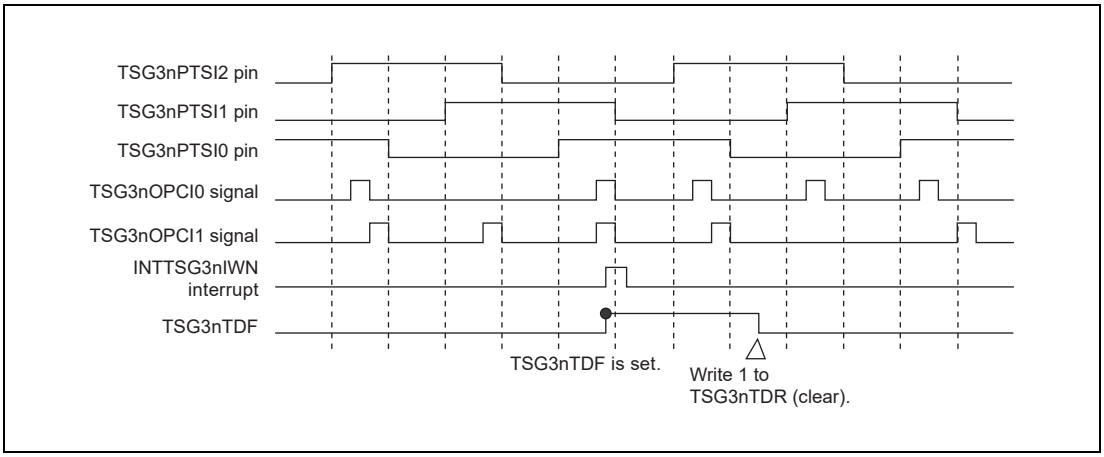


Figure 19.30 Operation of TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag Operation

Operating mode

Available in all operating modes.

CAUTION

TSG3nTDF is valid only when TSG3nCTL1.TSG3nTDC = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.3.10 Pattern Phase Difference Detection Flag (TSG3nPPF)

Name

Pattern phase difference detection flag (TSG3nSTR2.TSG3nPPF)

Description

TSG3nPPF can detect the phase difference between the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) and the output pattern (TSG3nSTR1.TSG3nOPF2 to TSG3nOPF0 flags).

TSG3nPPF is set to 1 when the pattern phase difference is detected when the TSG3nOPCI0 and TSG3nOPCI1 signal triggers are input, and a warning interrupt (INTTSG3nIWN) is generated. TSG3nPPF remains 1 until it is cleared to 0 when 1 is written to TSG3nSTC.TSG3nPPR by software. When the phase difference is detected, TSG3nPPF is set at each operation clock cycle (PCLK). TSG3nPPF should be cleared to 0 when no phase difference occurs.

Table 19.53 Correspondence between Normal Input Patterns and Output Patterns

TSG3nPTSI2 to TSG3nPTSI0 pins (input)	“1,0,1”	“1,0,0”	“1,1,0”	“0,1,0”	“0,1,1”	“0,0,1”
TSG3nOPF2 to TSG3nOPF0 flags (output)	“0,0,1” “1,0,1” “1,0,0”	“1,0,1” “1,0,0” “1,1,0”	“1,0,0” “1,1,0” “0,1,0”	“1,1,0” “0,1,0” “0,1,1”	“0,1,0” “0,1,1” “0,0,1”	“0,1,1” “0,0,1” “1,0,1”

Example of operation

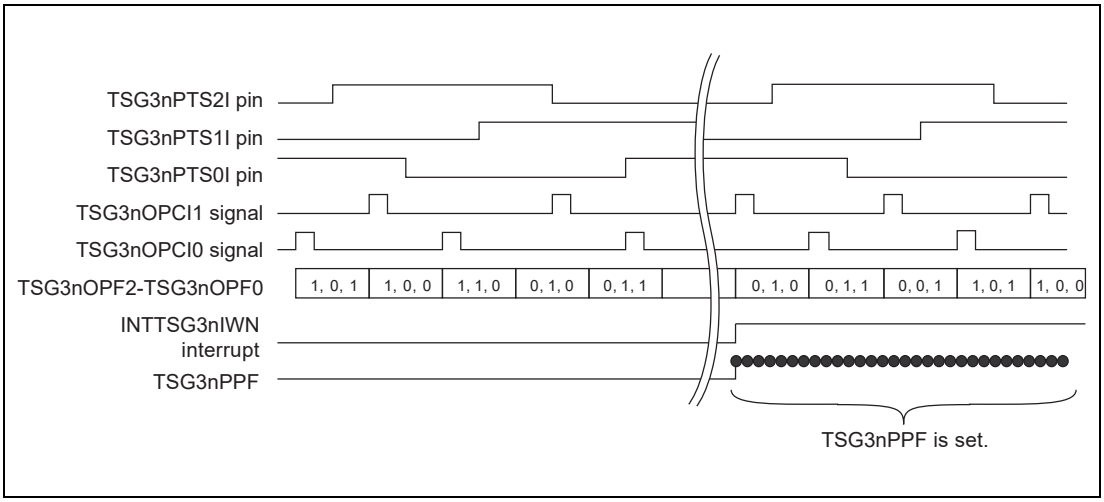


Figure 19.31 Example of Pattern Difference Detection Flag Operation

Operating mode

Available in all operating modes.

CAUTIONS

1. **TSG3nPPF is valid only when TSG3nCTL1.TSG3nPPC = 1 and TSG3nSTR0.TSG3nTE = 1.**
2. **When 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins, or when TSG3nOPF2 to TSG3nOPF0 are set to 000 or 111, TSG3nPPF is not set.**

19.4.3.11 Timer Output Pattern Flag (TSG3nOPF2-TSG3nOPF0)

Name

Timer output pattern flag (TSG3nSTR1.TSG3nOPF2 to TSG3nOPF0)

Description

TSG3nOPF2 to TSG3nOPF0 flags indicate the timer output patterns.

For details, see **Section 19.4.7.6, 120-DC Mode**, and **Section 19.4.7.10, Software Output Control Function**.

Operating mode

Available in all operating modes.

19.4.3.12 Pattern Switch Detection Signal (TSG3nPTE)

Name

Pattern switch detection signal (TSG3nPTE signal)

Description

The TSG3nPTE signal toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes.

The toggle pattern is determined by the TSG3nPSC bit (TSG3nOPT0.TSG3nPSS = 1).

Table 19.54 Change Timing of Pattern Switch Detection Signal (1/2)

- TSG3nPSC = 0

		TSG3nPTSI2-TSG3nPTSI0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

Table 19.54 Change Timing of Pattern Switch Detection Signal (2/2)

- TSG3nPSC = 1

		TSG3nPTSI2-TSG3nPTSI0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

Example of operation

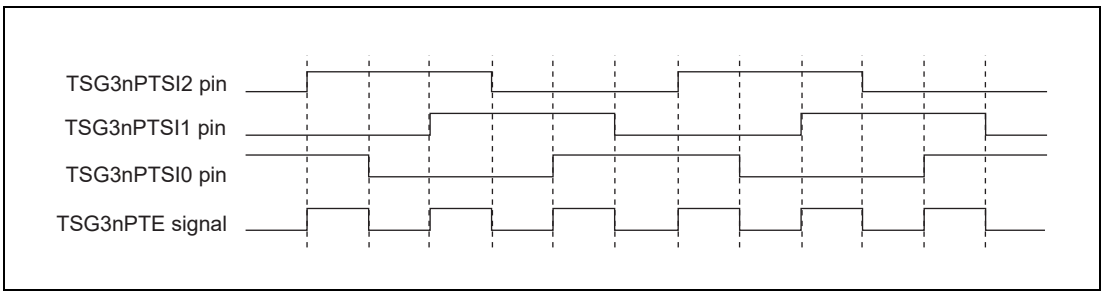


Figure 19.32 Example of Pattern Switch Detection Signal Operation

Operating mode

Available in all operating modes.

CAUTION

The TSG3nPTE signal is valid only when TSG3nIOC1.TSG3nPTS = 1 and TSG3nSTR0.TSG3nTE = 1.

19.4.4 Interrupt Skipping Function

Operation related to the interrupt skipping function is described below.

- Peak interrupts (INTTSG3nIPEK) and trough interrupts (INTTSG3nIVLY) can be skipped.
- TSG3nCTL4.TSG3nPIE enables outputting of the INTTSG3nIPEK interrupt and specifies whether to skip the interrupts.
- TSG3nCTL4.TSG3nVIE enables outputting of the INTTSG3nIVLY interrupt and specifies whether to skip the interrupts.

When TSG3nCTL3.TSG3nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

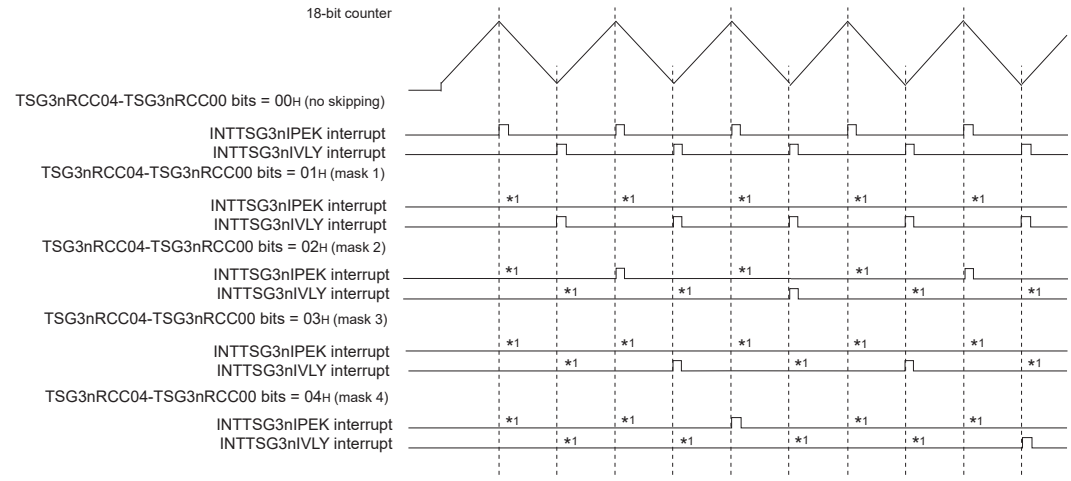
When TSG3nCTL3.TSG3nRIA is set to 0 (without reload skipping), reload is executed at the specified reload timing regardless of interrupt skipping.

CAUTION

When a value is written to TSG3nCTL4, and TSG3nRCC04 to TSG3nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG3nCTL3.TSG3nRIA = 1).

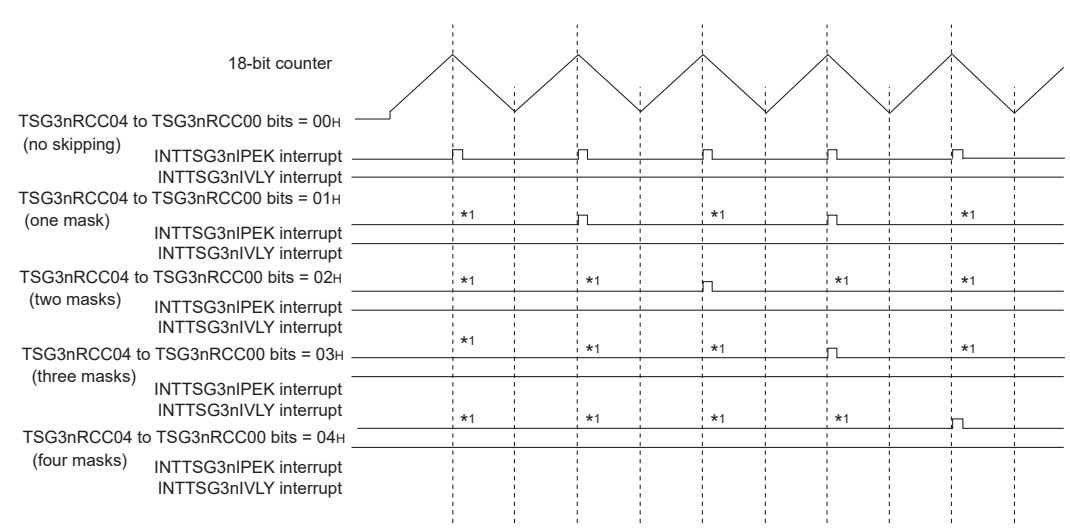
19.4.4.1 Operation of Interrupt Skipping Function

Timing diagram of interrupt skipping function in various conditions are shown in the following figures.



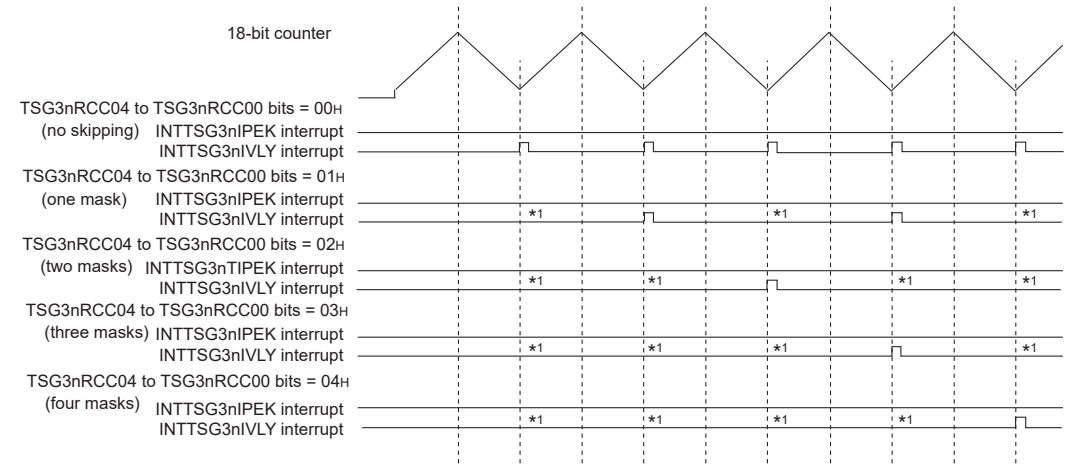
Note 1. Skipped interrupt request

Figure 19.33 Interrupt Skipping Operation when TSG3nPIE = 1 and TSG3nVIE = 1 in TSG3nCTL4 (Peak and Trough Interrupt Generation in HT-PWM Mode)



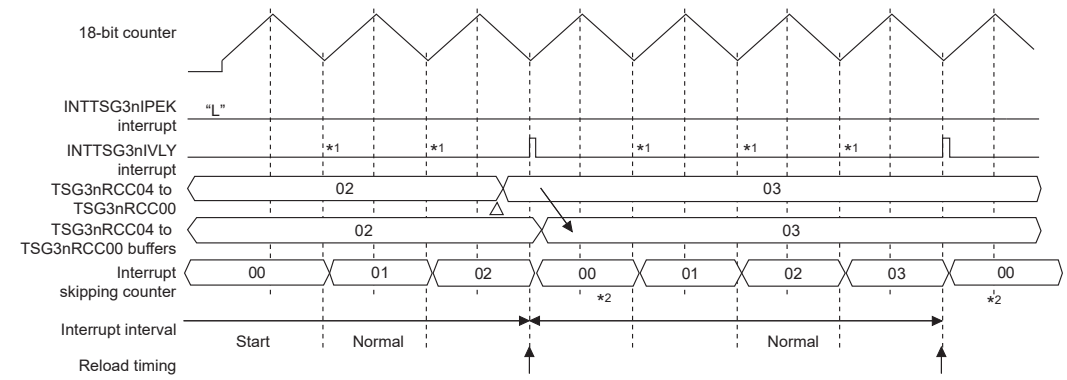
Note 1. Skipped interrupt request

Figure 19.34 Interrupt Skipping Operation when TSG3nPIE = 1 and TSG3nVIE = 0 in TSG3nCTL4 Register (only Peak Interrupt Generation in HT-PWM Mode)



Note 1. Skipped interrupt request

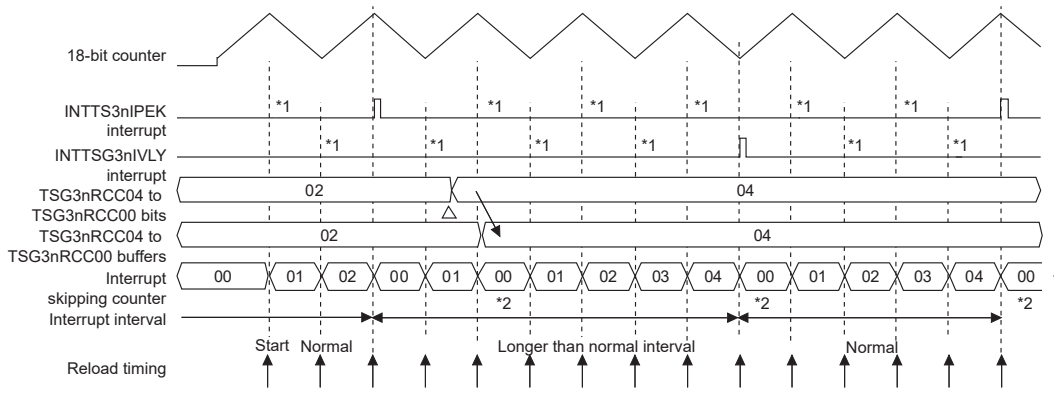
Figure 19.35 Interrupt Skipping Operation when TSG3nPIE = 0 and TSG3nVIE = 1 in TSG3nCTL4 Register (only Trough Interrupt Generation in HT-PWM Mode)



Note 1. Skipped interrupt request

Note 2. Interrupt skipping counter is cleared

Figure 19.36 When TSG3nRMC = 0, TSG3nRIA = 1 in TSG3nCTL3 (with Reload Skipping)

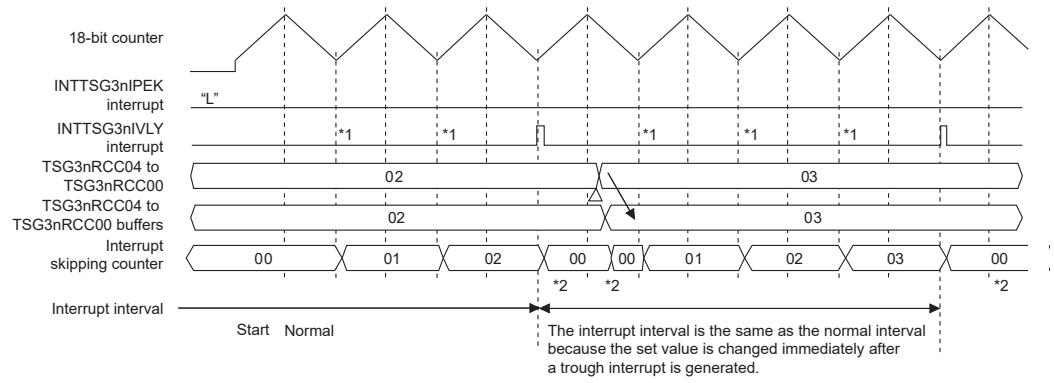


Note 1. Skipped interrupt request
Note 2. Interrupt skipping counter is cleared

Figure 19.37 When TSG3nRMC = 0, TSG3nRIA = 0 in TSG3nCTL3 (without Reload Skipping)

CAUTION

The interrupt interval might be longer.



Note 1. Skipped interrupt request
Note 2. Interrupt skipping counter is cleared

Figure 19.38 When TSG3nRMC = 1 in TSG3nCTL3 (Anytime Rewrite Mode)

NOTE

After rewriting, the value is reflected immediately regardless of the reload timing. The interrupt skipping counter is cleared when the value is transferred to the TSG3nRCC04 to TSG3nRCC00 buffers, not when the pertinent register is rewritten.

19.4.4.2 Example of Operation when Peak Interrupt is Generated (in PWM Mode)

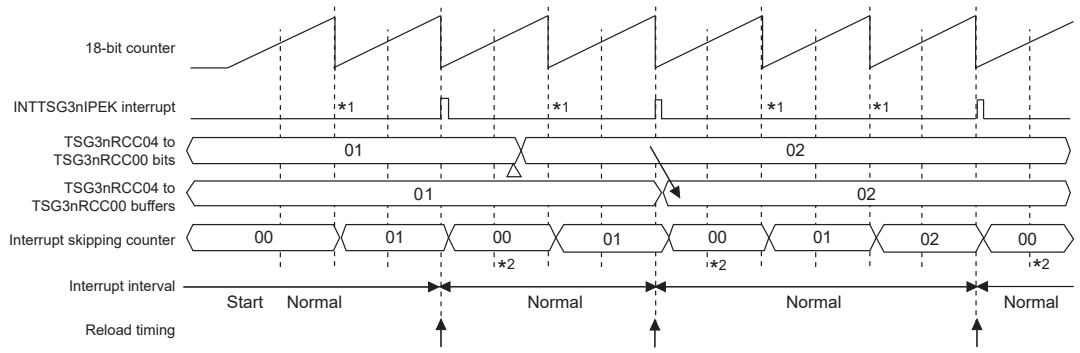
Operation related to the interrupt skipping function in PWM mode is described below.

- Trough interrupts (INTTSG3nIPEK) can be skipped. In PWM mode, it is generated by compare match of TSG3nCMP0E buffer register and 18-bit counter.
- TSG3nCTL4.TSG3nPIE enables outputting of the INTTSG3nIPEK interrupt and specifies whether to skip the interrupts.
- The setting of TSG3nCTL4.TSG3nVIE is disabled. At this time, the INTTSG3nIVLY interrupt is not generated.
  When TSG3nCTL3.TSG3nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

CAUTION

When a value is written to TSG3nCTL4, and TSG3nRCC04 to TSG3nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG3nCTL3.TSG3nRIA = 1).

(1) Example of operation



Note 1. Skipped interrupt request
Note 2. Interrupt skipping counter is cleared.

Figure 19.39 When TSG3nCTL3.TSG3nRMC = 0, TSG3nRIA = 1, TSG3nCTL4.TSG3nPRE = 1 (Recommended Setting)

NOTE

When TSG3nCTL3.TSG3nRIA = 1, reload is executed at the same timing as the interrupt after being skipped.

19.4.5 A/D Conversion Trigger Function

A/D conversion trigger operation is described below.

The TSG3nDCMP0E, TSG3nDCMP1E, and TSG3nDCMP2E registers are used as compare registers of the A/D conversion trigger function.

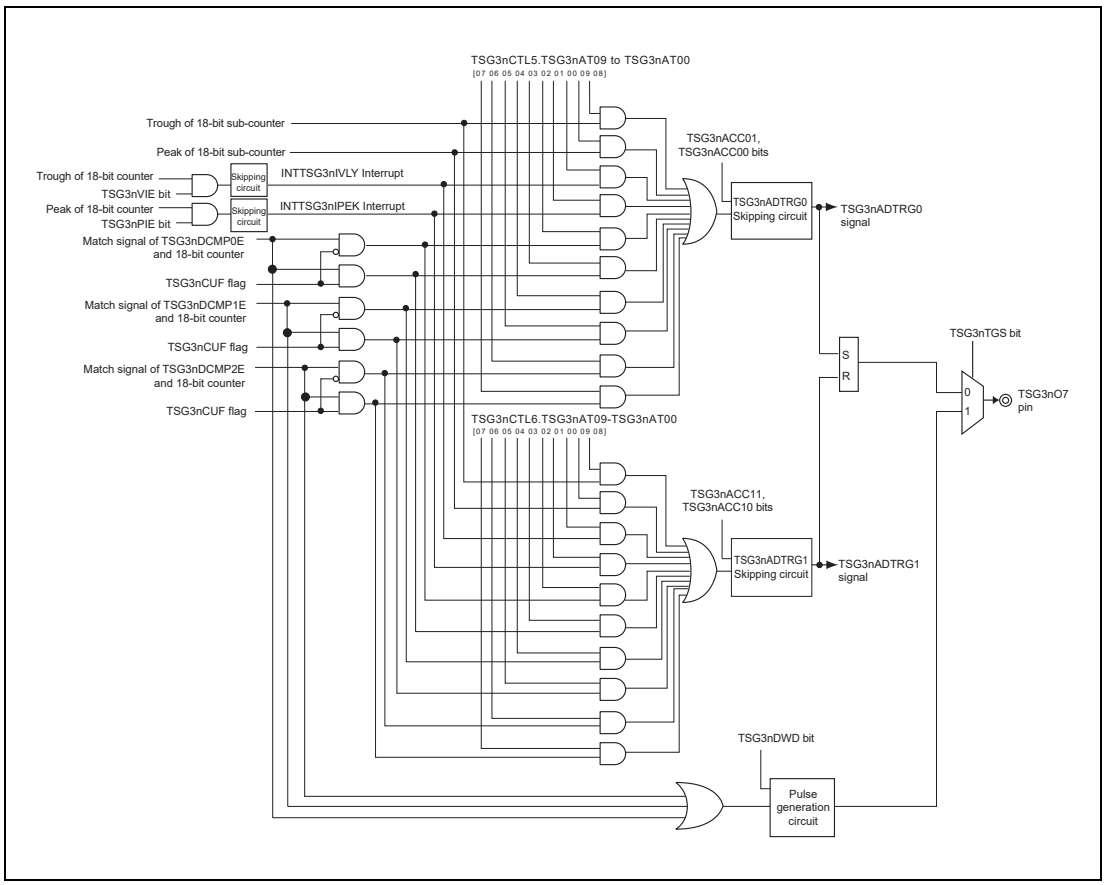


Figure 19.40 A/D Conversion Trigger and Diagnostic Output Control Circuit

As shown in **Figure 19.40**, a logical ORed signal can be generated by selecting the compare match of TSG3nDCMP0E to TSG3nDCMP2E with the 18-bit counter, a peak interrupt (INTTSG3nIPEK) and a trough interrupt (INTTSG3nIVLY) of the 18-bit counter, the peak timing of 18-bit sub-counter, and the trough timing of 18-bit sub-counter.

TSG3n has two channels of the identical A/D conversion trigger control circuits, which can be controlled independently. TSG3n also provides the A/D conversion trigger skipping function with the skipping rate of 1/1, 1/2, 1/4, or 1/8.

19.4.5.1 Operation of A/D Conversion Trigger

TSG3n has a function to generate A/D conversion start triggers (TSG3nADTRG0 and TSG3nADTRG1 signals) by selecting any of ten trigger sources as required. The trigger sources are selected by TSG3nAT09 to TSG3nAT00 in TSG3nCTL5 and TSG3nAT19 to TSG3nAT10 in TSG3nCTL6.

(1) TSG3nADTRG0/TSG3nADTRG1 Signal Output Control (TSG3nCTL5 and TSG3nCTL6)

[Trigger sources]

- TSG3nAT00/TSG3nAT10 = 1 : A trough interrupt (INTTSG3nIVLY) causes an A/D conversion trigger pulse to be generated.
- TSG3nAT01/TSG3nAT11 = 1 : A peak interrupt (INTTSG3nIPEK) causes an A/D conversion trigger pulse to be generated.
- TSG3nAT02/TSG3nAT12 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP0E compare match enables A/D conversion trigger to be generated.
- TSG3nAT03/TSG3nAT13 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP0E compare match enables A/D conversion trigger to be generated.
- TSG3nAT04/TSG3nAT14 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP1E compare match enables A/D conversion trigger to be generated.
- TSG3nAT05/TSG3nAT15 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP1E compare match enables A/D conversion trigger to be generated.
- TSG3nAT06/TSG3nAT16 = 1 : While the 18-bit counter is counting up, a TSG3nDCMP2E compare match enables A/D conversion trigger to be generated.
- TSG3nAT07/TSG3nAT17 = 1 : While the 18-bit counter is counting down, a TSG3nDCMP2E compare match enables A/D conversion trigger to be generated.
- TSG3nAT08/TSG3nAT18 = 1 : A trough timing of the 18-bit sub-counter (at a switch from decrementing to incrementing) enables A/D conversion trigger to be generated.
- TSG3nAT09/TSG3nAT19 = 1 : A peak timing of 18-bit sub-counter (at a switch from incrementing to decrementing) enables A/D conversion trigger to be generated.

[Skipping setting]

- TSG3nACC01, TSG3nACC00 and TSG3nACC11, TSG3nACC10 :
  Set the skipping rate of TSG3nADTRG0/TSG3nADTRG1

All A/D conversion triggers selected by TSG3nAT09 to TSG3nAT00 and TSG3nAT19 to TSG3nAT10 are logically ORed, and the resultant signals are subjected to skipping control specified by TSG3nACC01 and TSG3nACC00, and TSG3nACC11 and TSG3nACC10, and then the TSG3nADTRG0 and TSG3nADTRG1 signals are generated.

A peak interrupt (INTTSG3nIPEK) and a trough interrupt (INTTSG3nIVLY) selected by TSG3nAT00 and TSG3nAT01, and TSG3nAT10 and TSG3nAT11 are interrupt signals obtained after skipped. Therefore, they are output at the timing according to interrupt skipping control. If the interrupt output is not enabled by TSG3nCTL4.TSG3nPIE and TSG3nVIE, A/D conversion trigger is not output.

TSG3nACC01, TSG3nACC00, and TSG3nAT09 to TSG3nAT00, and TSG3nACC11, TSG3nACC10, and TSG3nAT19 to TSG3nAT10 can be rewritten during timer operation.

If A/D conversion trigger setting bits are rewritten during operation, the rewritten values are reflected on the A/D conversion trigger output status immediately. Such control bits are rewritten at anytime regardless of operating modes. If a write access is made to TSG3nCTL5 and TSG3nCTL6 (including a rewrite of the same value), the A/D conversion trigger skipping counter is cleared and starts counting from zero.

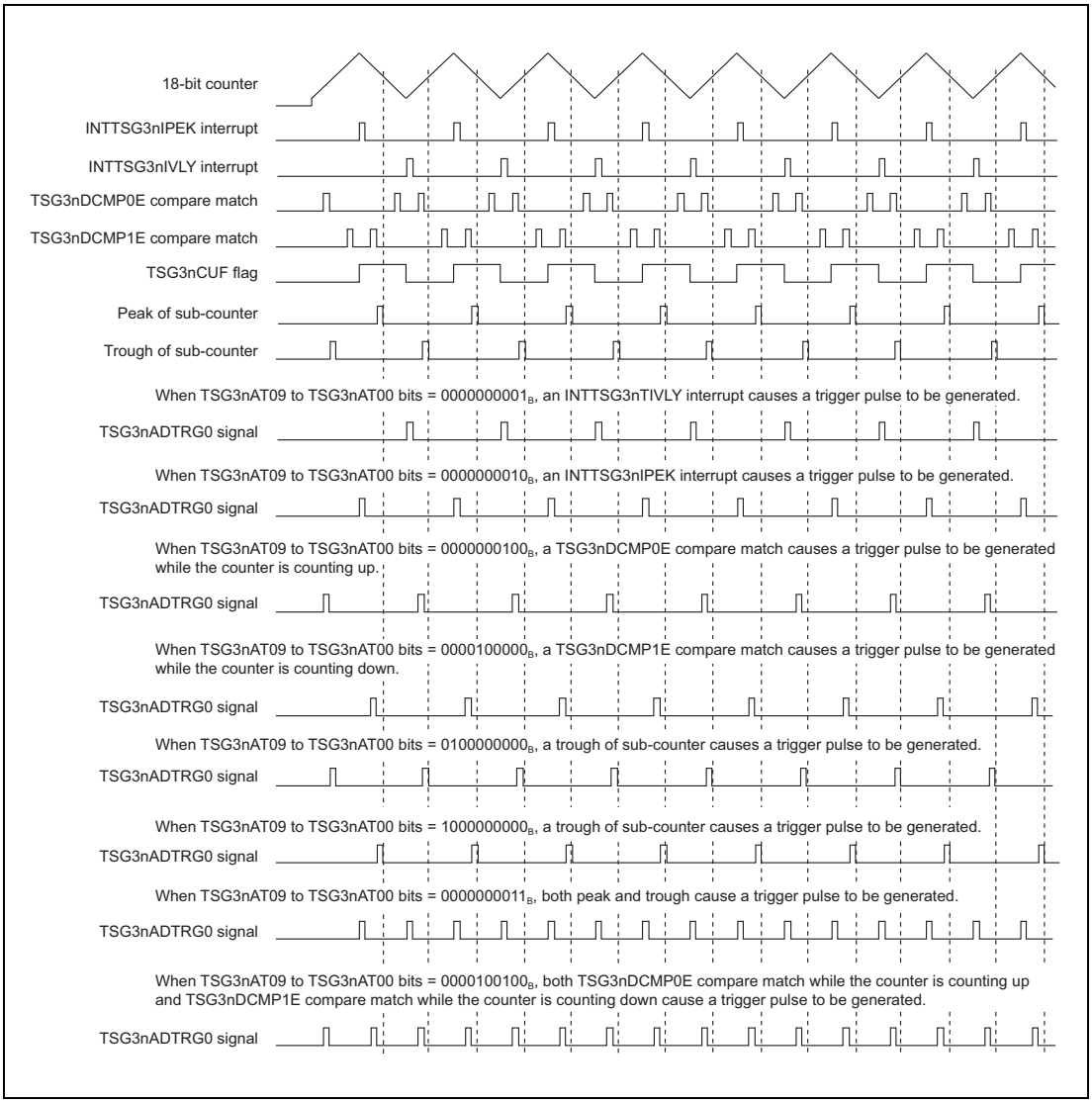
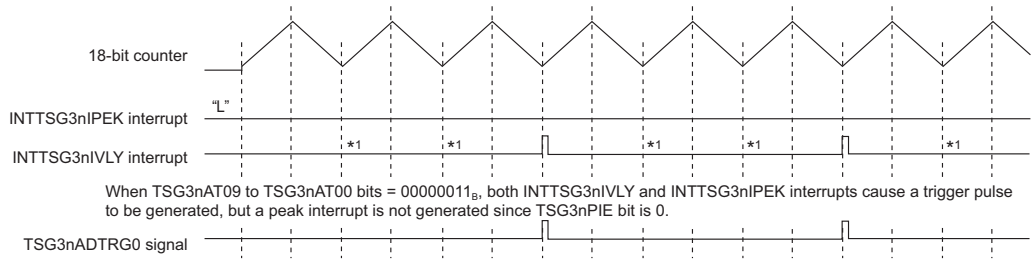
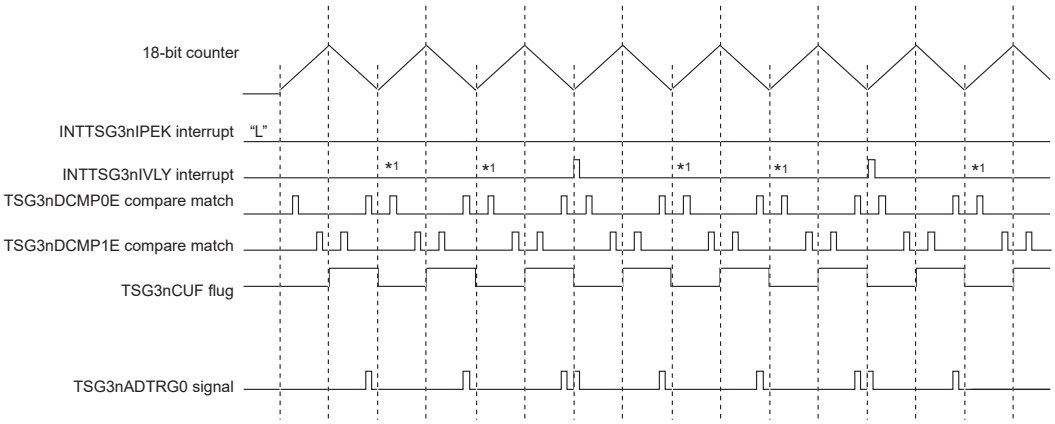


Figure 19.41 When TSG3nPIE = 1, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 00_H in TSG3nCTL4, and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode)



Note 1. Skipped interrupt request

Figure 19.42 When TSG3nPIE = 0, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 02_H in TSG3nCTL4 and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode)



Note 1. Skipped interrupt request

Figure 19.43 When TSG3nPIE = 0, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 02_H in TSG3nCTL4 and TSG3nACC01 and TSG3nACC00 = 00_B, and TSG3nAT09 to TSG3nAT00 = 0000 1001_B in TSG3nCTL5 (HT-PWM Mode)

(2) A/D Conversion Trigger Skipping Function

Example of operation of the A/D conversion trigger skipping function is shown in **Figure 19.44**.

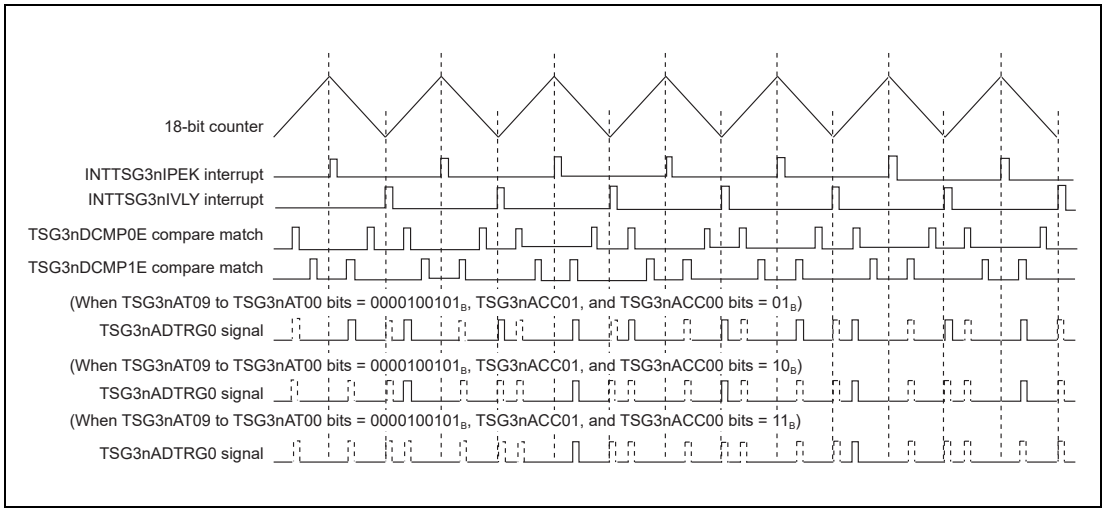


Figure 19.44 Example of Operation of A/D Conversion Trigger Skipping Function

NOTE

Broken-lined pulses indicate A/D conversion trigger pulses skipped by the A/D conversion trigger skipping function.

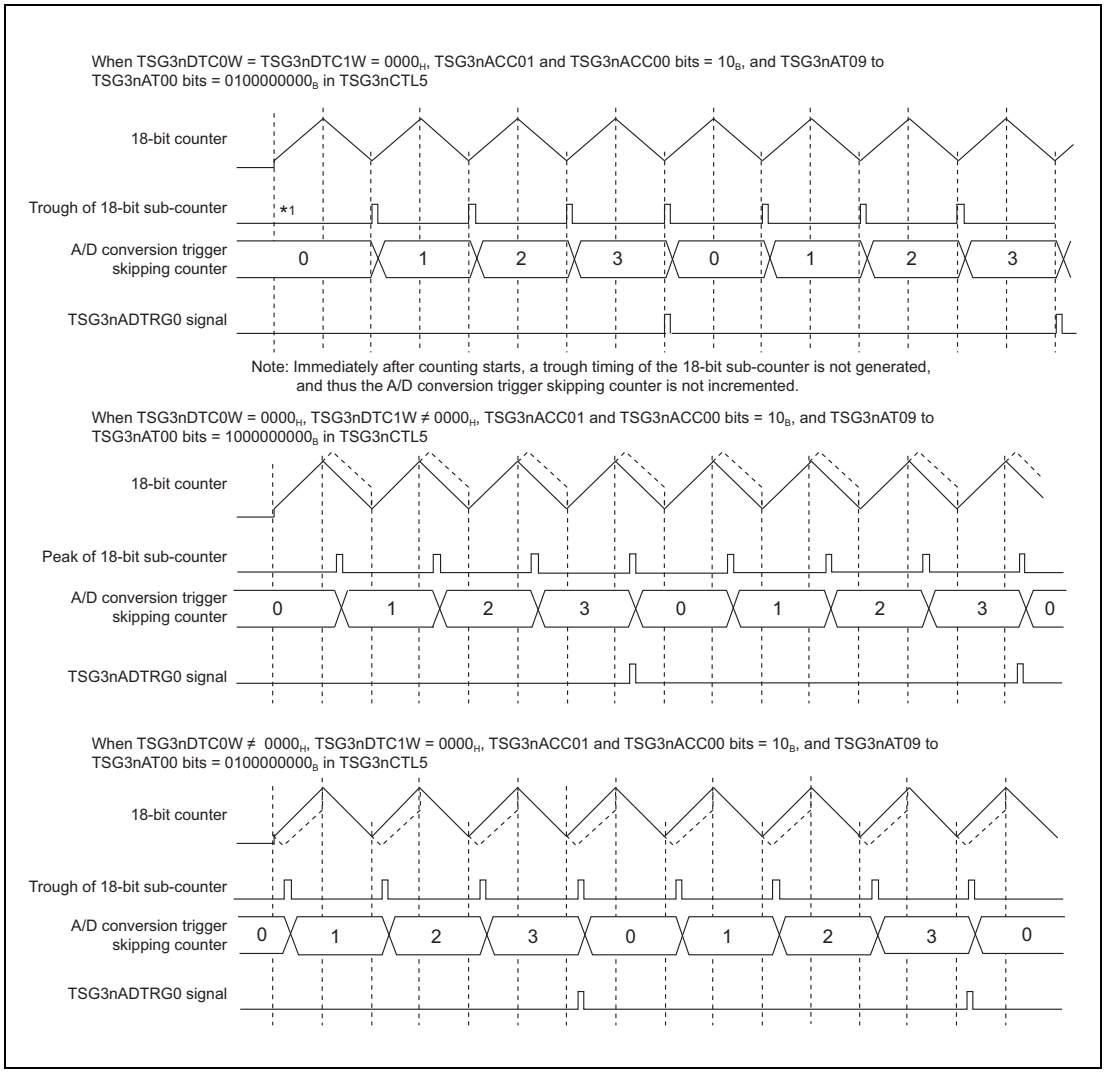


Figure 19.45 Example of Operation of A/D Conversion Trigger Skipping Function

(3) Notes on A/D Conversion Trigger

- If the same value is written to TSG3nDCMP0E and TSG3nDCMP1E or TSG3nDCMP2E, and the same condition (when the 18-bit counter increments or decrements) is set as the valid A/D conversion trigger, A/D conversion trigger skipping counter is incremented by one and one trigger pulse is output upon a match of the 18-bit counter with these registers.
- In PWM mode, SP-PWM mode, 120-DC mode, and HSP-PWM mode, a trough interrupt (INTTSG3nIVLY) is not generated. Only a peak interrupt (INTTSG3nIPEK) is valid.
- In 120-DC mode, when TSG3nS120DCO is set to 0, the 18-bit counter may be cleared during the carrier period due to switch of the output pattern. The A/D conversion trigger is not generated if TSG3nDCMP2E to TSG3nDCMP0E values do not match with the 18-bit counter value and a peak interrupt (INTTSG3nIPEK) is not generated.

19.4.6 Error/Warning Interrupt

19.4.6.1 Error Interrupt Function

If the simultaneous active state of the positive phase and inverse phase is detected after the error interrupt function is enabled (TSG3nIOC1.TSG3nEOC = 1), TSG3nSTR2.TSG3nTBF is set, and an error interrupt (INTTSG3nIER) of TSG3n is generated. Whether or not to detect an error of each phase (TSG3nO1 and TSG3nO2, TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins) can be selected by TSG3nCTL1.TSG3nTBA2 to TSG3nTBA0, respectively.

When an error occurs, outputs of the TSG3nO1 to TSG3nO6 pins can be set to high-impedance. For details, see **Section 20.4.1, Asynchronous Hi-Z Control Functions**.

The following table shows whether or not the simultaneous active state of the positive phase and inverse phase can be detected in each mode.

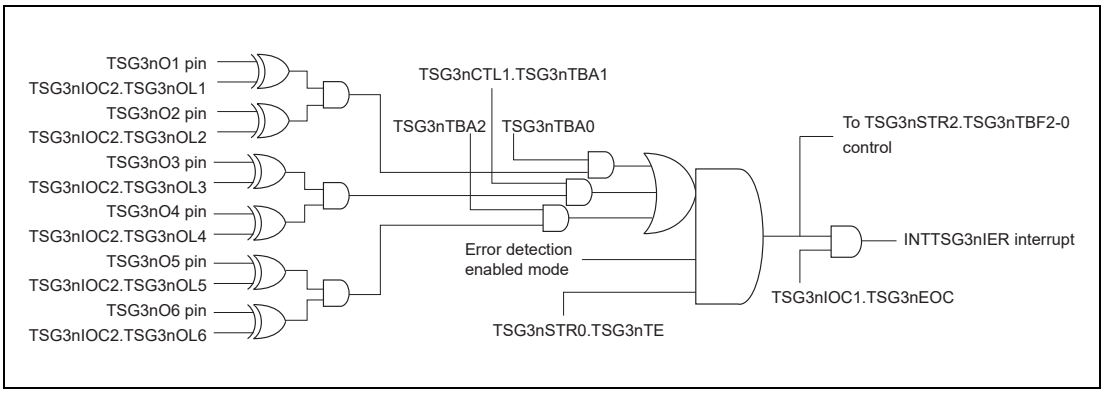


Figure 19.46 Error Interrupt (INTTSG3nIER) Generation Control Circuit

CAUTION

When an error interrupt is generated, the error status should be canceled (write 1 to TSG3nSTC.TSG3nTBR2 to TSG3nSTC.TSG3nTBR0) during an error interrupt handling. Otherwise, subsequent error interrupts are not generated.

(1) PWM Mode, 120-DC Mode and HSP-PWM Mode

In PWM mode, and HSP-PWM Mode if TSG3nCMP1E and TSG3nCMP2E, and TSG3nCMP3E and TSG3nCMP4E are set so that the TSG3nO1 and TSG3nO2 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated. With the same setting, if TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP9E, TSG3nCMP10E, TSG3nCMP11E, and TSG3nCMP12E are set so that the TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated.

In 120-DC mode, if TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E, TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E, TSG3nPAT0W, and TSG3nPAT1W are set so that the TSG3nO1 and TSG3nO2 pins output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated. With the same setting, the TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 pins also output the active level simultaneously, an error interrupt (INTTSG3nIER) is generated.

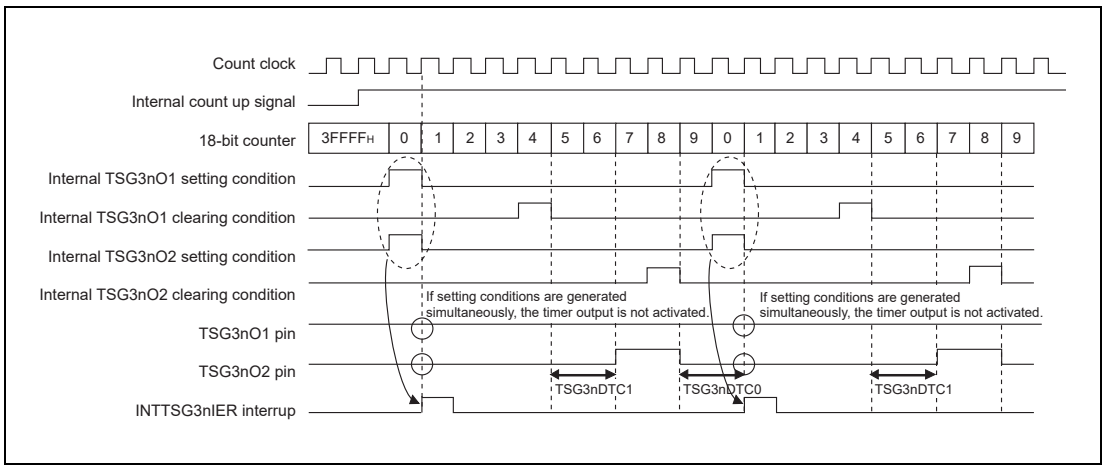


Figure 19.47 Example of Error Interrupt (INTTSG3nIER) Generation (PWM Mode)

NOTE

TSG3nO3 and TSG3nO4, and TSG3nO5 and TSG3nO6 behave the same.

When the active level of output is switched by operation TSG3nIOC2.TSG3nOL1 and TSG3nOL2, an error interrupt is generated as shown in the following figure.

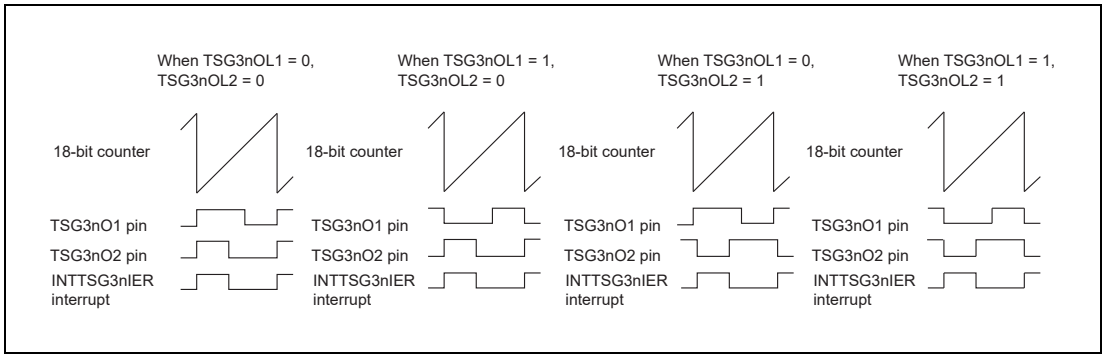


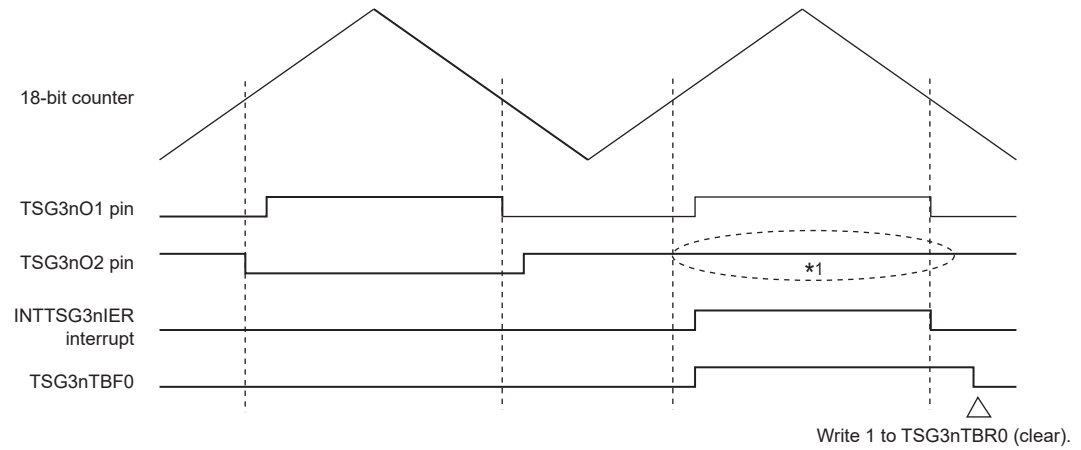
Figure 19.48 Example of Error Interrupt (INTTSG3nIER) Generation for each Active Level

(2) HT-PWM Mode and SP-PWM Mode

Either TSG3n dead time setting register 0 or 1 (TSG3nDTC0W or TSG3nDTC1W) is 0000_H, an error may occur.

NOTE

If an error occurs when the dead time control function is used (both TSG3nDTC0W and TSG3nDTC1W are not 0000_H), internal circuit failure may occur.



Note 1. TSG3nO2 pin control circuit failure occurs.

Figure 19.49 Example of Error Interrupt Operation

19.4.6.2 Warning Interrupt Function

TSG3n has a warning interrupt (INTTSG3nIWN).

Warning interrupt (INTTSG3nIWN) is generated when any of the following conditions is detected.

For details, see **Section 19.4.3, Flags**.

- When simultaneous change in two or more pins of TSG3nPTSI2 to TSG3nPTSI0 is detected:
  See **Section 19.4.3.4, Noise Detection Flag (TSG3nNDF)**.
- When reversal is detected of the TSG3nPTSI2 to TSG3nPTSI0 pins:
  See **Section 19.4.3.7, Pattern Reversal Detection Flag (TSG3nPRF)**.
- When 000 or 111 is detected from the TSG3nPTSI2 to TSG3nPTSI0 pins:
  See **Section 19.4.3.6, Pattern Error Detection Flag (TSG3nPEF)**.
- When a toggle of the TSG3nPTSI2 to TSG3nPTSI0 pins is generated three times or more between TSG3nOPCI0 and TSG3nOPCI1 signal triggers.:
  See **Section 19.4.3.8, TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag (TSG3nPTF)**.
- When the TSG3nOPCI0 and TSG3nOPCI1 signal triggers are detected simultaneously:
  See **Section 19.4.3.9, TSG3nOPCI0, TSG3nOPCI1 Signal Simultaneous Trigger Detection Flag (TSG3nTDF)**.
- When the phase difference between the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) and output pattern (TSG3nOPF2 to TSG3nOPF0) is detected:
  See **Section 19.4.3.10, Pattern Phase Difference Detection Flag (TSG3nPPF)**.

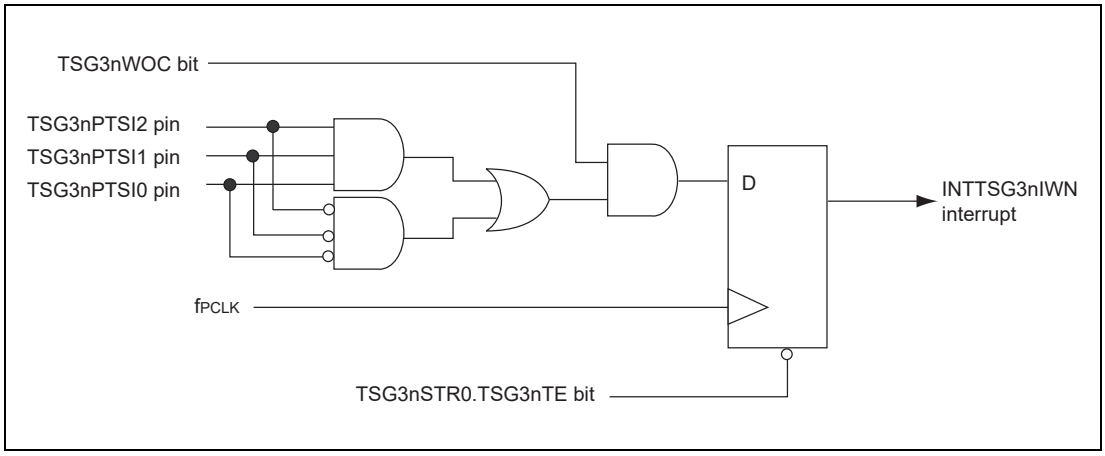


Figure 19.50 Detection of Abnormality of TSG3nPTSI2-TSG3nPTSI0 Pins

19.4.7 Operating Modes

19.4.7.1 PWM Mode

Overview

A PWM signal is output at the TSG3nO1 to TSG3nO6 pins according to set timing/clear timing of TSG3nCMP1E to TSG3nCMP12E registers with the PWM period set in the TSG3nCMP0E register.

Prerequisites

- Set the set timing to the compare register with an even number:
  TSG3nCMP2E (set timing of the TSG3nO1 output), TSG3nCMP4E (set timing of the TSG3nO2 output), TSG3nCMP6E (set timing of the TSG3nO3 output), TSG3nCMP8E (set timing of the TSG3nO4 output), TSG3nCMP10E (set timing of the TSG3nO5 output) and TSG3nCMP12E (set timing of the TSG3nO6 output)
- Set the clear timing to the compare register with an odd number:
  TSG3nCMP1E (clear timing of the TSG3nO1 output), TSG3nCMP3E (clear timing of the TSG3nO2 output), TSG3nCMP5E (clear timing of the TSG3nO3 output), TSG3nCMP7E (clear timing of the TSG3nO4 output), TSG3nCMP9E (clear timing of the TSG3nO5 output) and TSG3nCMP11E (clear timing of the TSG3nO6 output)

Functional description

Set the PWM period and set/clear timing of the TSG3nO1 to TSG3nO6 outputs. Set TSG3nTRG0.TSG3nTS = 1 to start the timer counter.

The TSG3nO1 to TSG3nO6 outputs are set to the inactive state at the same time the counting begins. The outputs are set to the active state by the match of the buffer registers TSG3nCMP2E, TSG3nCMP4E, TSG3nCMP6E, TSG3nCMP8E, TSG3nCMP10E, and TSG3nCMP12E with the 18-bit counter.

Next, the TSG3nO1 to TSG3nO6 outputs are set to the inactive state by the match of the buffer registers TSG3nCMP1E, TSG3nCMP3E, TSG3nCMP5E, TSG3nCMP7E, TSG3nCMP9E, and TSG3nCMP11E with the 18-bit counter.

During counting, a compare match interrupt (INTTSG3nI0 to INTTSG3nI12) is generated by the match of the buffer register TSG TSG3nCMP0E-TSG3nCMP12E.

CAUTION

Reload is executed when writing to the TSG3nCMP1E register at TSG3nCTL3.TSG3nRMC = 0. Therefore, even when it is needed to rewrite only the value of the TSG3nCMP0E register, a write operation to the TSG3nCMP1E register is necessary. When only the TSG3nCMP0E register is rewritten, reload is not done.

NOTE

The PWM mode is set when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 000_B.

(a) When TSG3nCMP0E and TSG3nCMP1E to TSG3nCMP12E are Not Rewritten during Timer Operation

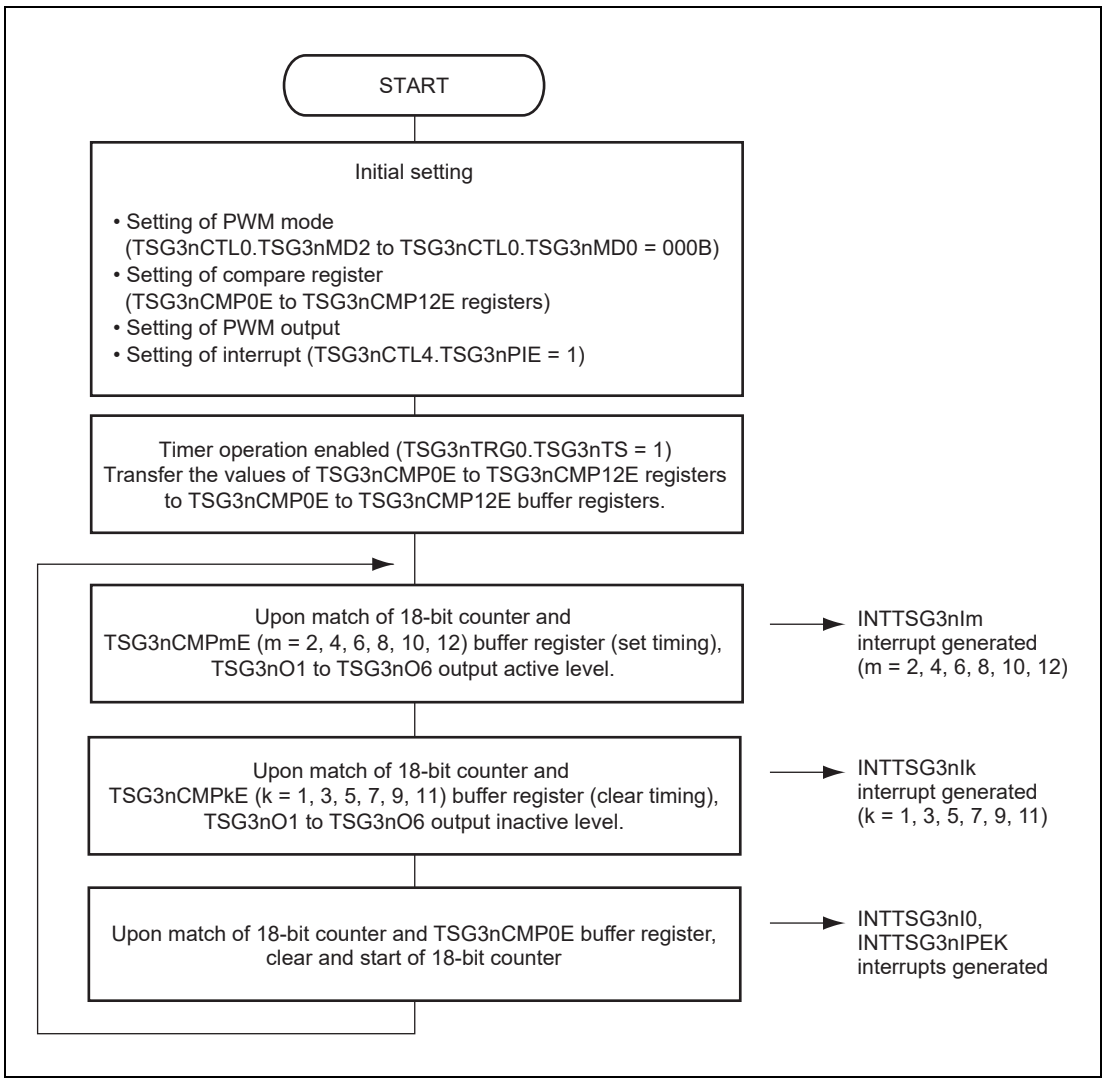
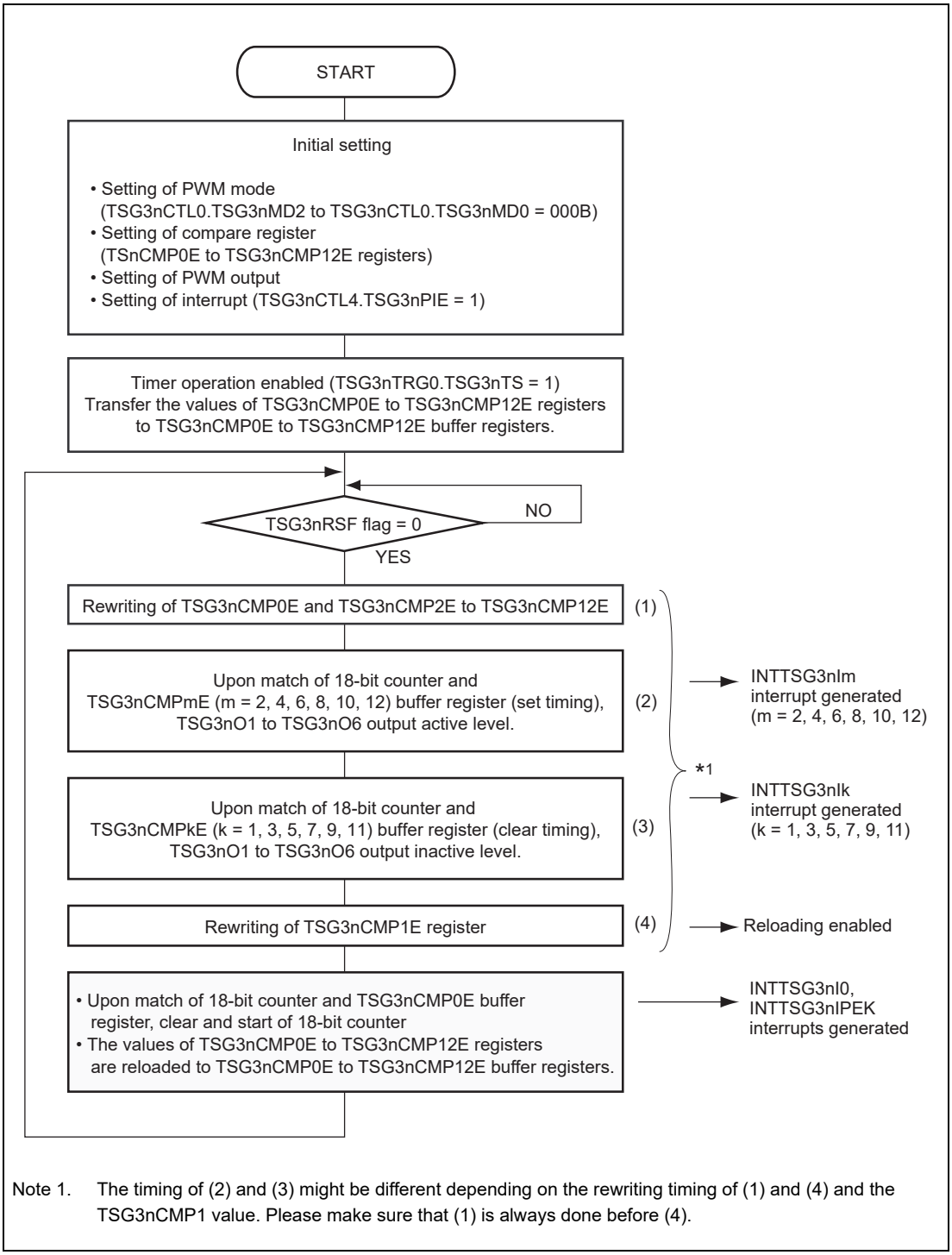


Figure 19.51 Basic Operation Flow of PWM Mode (1/2)

(b) When TSG3nCMP0E and TSG3nCMP1E to TSG3nCMP12E are Rewritten during Timer Operation



Note 1. The timing of (2) and (3) might be different depending on the rewriting timing of (1) and (4) and the TSG3nCMP1 value. Please make sure that (1) is always done before (4).

Figure 19.51 Basic Operation Flow of PWM Mode (2/2)

CAUTION

Please rewrite compare registers after confirming that the reload request flag TSG3nRSF is 0.

(1) List of Operations in PWM Mode

Table 19.55 Counter Functions in PWM Mode

Operation		Setting condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1, or a simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register and 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 19.56 Functions of Compare Registers and Dead Time Setting Register in PWM Mode

Register	Rewriting Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nCMPmE (m = 1 to 12)	Reload/Anytime rewrite	Possible	Setting set/clear timing
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible*1	Setting dead time

Note 1. For details, see **(3), Controlling Dead Time in PWM Mode**.

Table 19.57 Timer Output in PWM Mode

Pin	Function
TSG3nOm (m = 1 to 6)	PWM output by compare match of TSG3nCMPkE buffer register and 18-bit counter (k = 1 to 12)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 19.58 Interrupt Requests in PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of TSG3nCMPmE buffer register and 18-bit counter (m = 0 to 12)
INTTSG3nIER	Error (simultaneous active state of TSG3nO1 and TSG3nO2, or TSG3nO3 and TSG3nO4, or TSG3nO5 and TSG3nO6)
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Warning interrupt

Note: "—": Not available in PWM mode

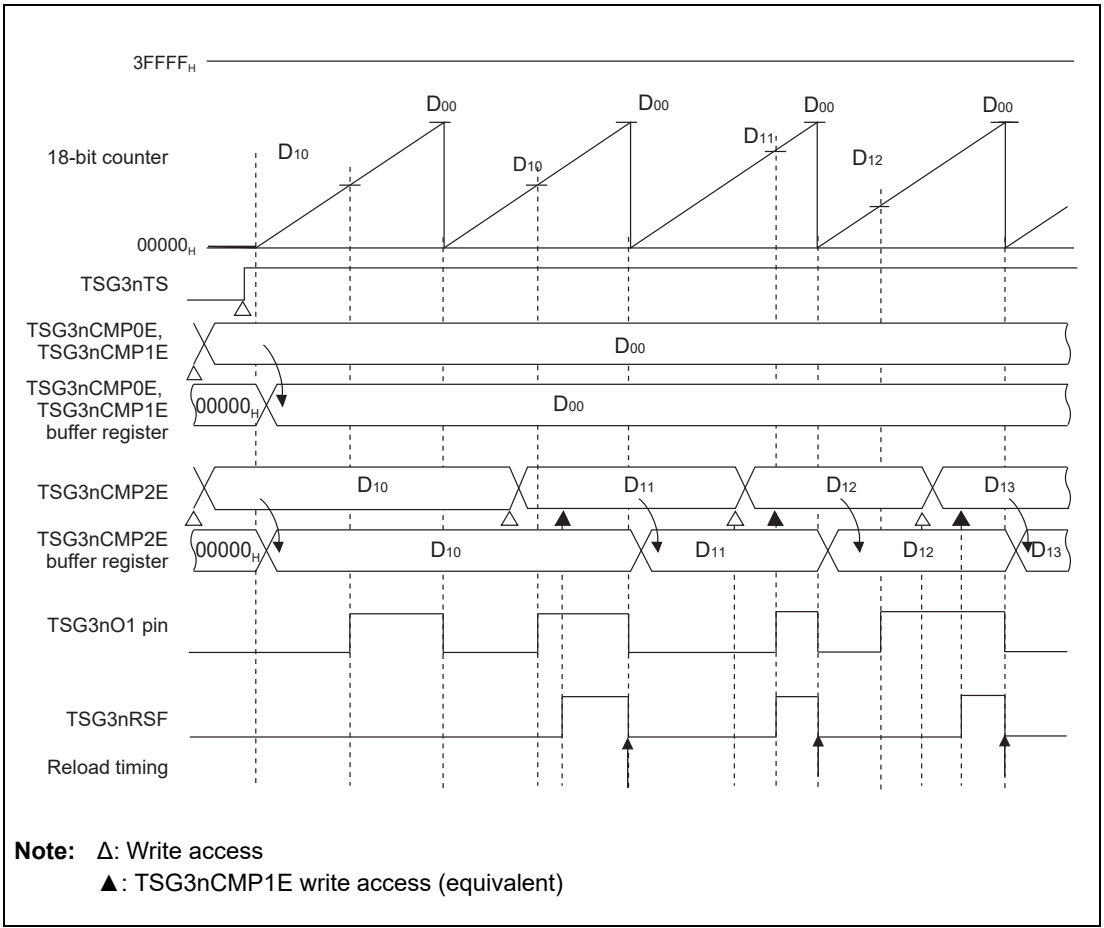
Table 19.59 Compare Match Timing in PWM Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After detecting the match of 18-bit counter and TSG3nCMPmE (m = 1 to 12)

Table 19.60 Example of Setting each Timer Output Condition in PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nOm (m = 1 to 6)	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty cycle 0%)	TSG3nCMPmE = TSG3nCMP (m + 1)E or TSG3nCMP (m + 1) E> TSG3nCMP0E (m = 1, 3, 5, 7, 9, 11)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1)E + 1 TSG3nCMP (m + 1)E = TSG3nCMPmE - 1 (m = 1, 3, 5, 7, 9, 11)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1)E - 1 TSG3nCMP (m + 1)E = TSG3nCMPmE + 1 (m = 1, 3, 5, 7, 9, 11)
			Output an active level throughout one period (duty cycle 100%)	TSG3nCMPmE > TSG3nCMP0E TSG3nCMP (m + 1)E ≤ TSG3nCMP0E (m = 1, 3, 5, 7, 9, 11)

When only TSG3nCMP2E is rewritten and the TSG3nO1 pin output is used (TSG3nIOC0.TSG3nTOE1 = 1, TSG3nIOC2.TSG3nOL1 = 0)



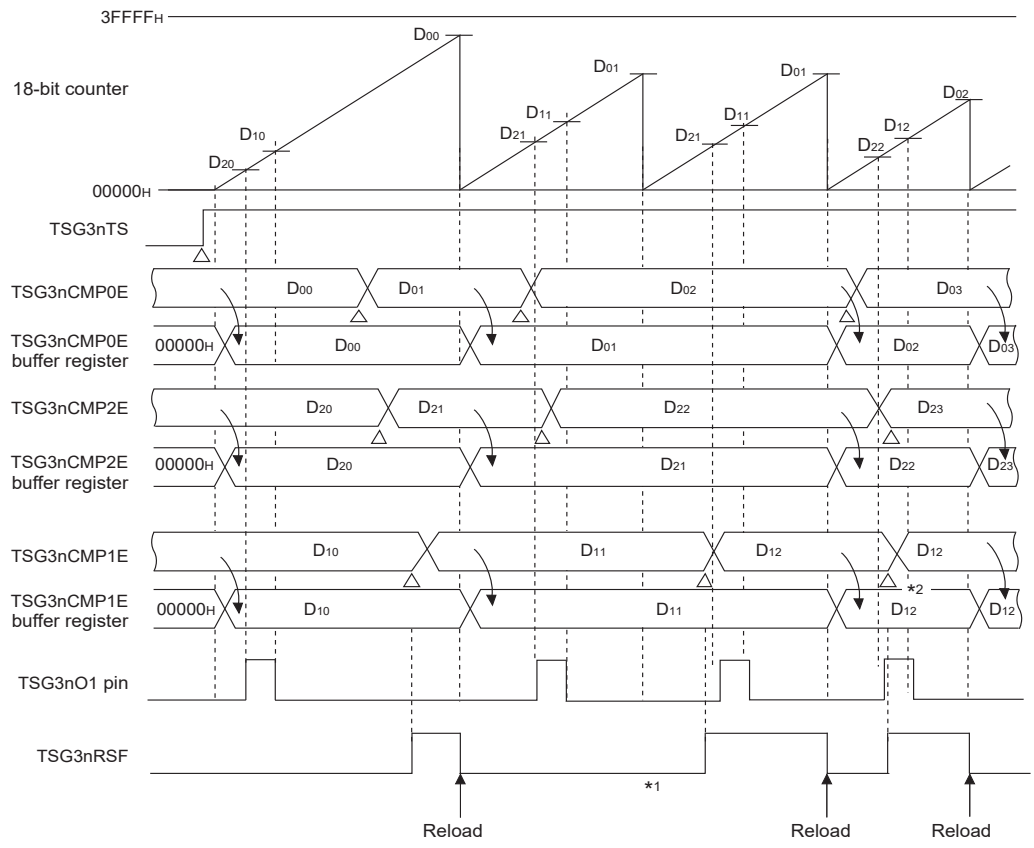
Note: Δ: Write access
▲: TSG3nCMP1E write access (equivalent)

Figure 19.52 Example of Basic Operation Timing in PWM Mode (1/2)

NOTES

1. D00: Set values of TSG3nCMP0E and TSG3nCMP1E (00000_H-$3FFFF_H$)
   D10, D11, D12 and D13: Set values of TSG3nCMP2E (00000_H-$3FFFF_H$)
2. TSG3nO1 (PWM) duty cycle = (TSG3nCMP1E-TSG3nCMP2E) (count lock)
   TSG3nO1 (PWM) period = (Set value of TSG3nCMP0E + 1) (count lock)
3. TSG3nO2-TSG3nO6 pins behave similarly to the TSG3nO1 pin

When TSG3nCMP0E-TSG3nCMP2E are rewritten, and the TSG3nO1 pin output is used (TSG3nIOC0.TSG3nTOE1 = 1, TSG3nIOC2.TSG3nOL1 = 0)



Note 1. The TSG3nCMP0E buffer register is not reloaded, because the TSG3nCMP1E register was not rewritten.

Note 2. Equivalent writing

Note: Δ: Write access

Figure 19.52 Example of Basic Operation Timing in PWM Mode (2/2)

NOTES

1. D00, D01, D02, D03: Set point of TSG3nCMP0E (00000_H-$3FFFF_H$)
   D10, D11, D12, D13: Set point of TSG3nCMP1E (00000_H-$3FFFF_H$)
   D20, D21, D22, D23: Set point of TSG3nCMP2E (00000_H-$3FFFF_H$)
2. Outputs from TSG3nO2 to TSG3nO6 behave similarly to the TSG3nO1 pin.

(2) Interrupt/Reload Skipping Function in PWM Mode

By setting TSG3nCTL4.TSG3nPRE and TSG3nPIE to 1 and setting TSG3nRCC04 to TSG3nRCC00 and TSG3nCTL3.TSG3nRIA, the reload and interrupt skipping function can be used.

By setting TSG3nPRE to 1 and setting TSG3nRCC04 to TSG3nRCC00, the interrupt skipping function can be used.

(3) Controlling Dead Time in PWM Mode

By setting the dead time value in the TSG3nDTC0W and TSG3nDTC1W registers in PWM mode, it is possible to control the dead time. The dead time is controlled according to the switch timing of the TSG3nO1 and TSG3nO2 pin outputs, the TSG3nO3 and TSG3nO4 pin outputs, and the TSG3nO5 and TSG3nO6 pin outputs.

Table 19.61 Dead Time in PWM Mode

Switch Timing	Dead Time
TSG3nO1: High level to low level TSG3nO2: Low level to high level	Value of TSG3nDTC1W register
TSG3nO2: High level to low level TSG3nO1: Low level to high level	Value of TSG3nDTC0W register
TSG3nO3: High level to low level TSG3nO4: Low level to high level	Value of TSG3nDTC1W register
TSG3nO4: High level to low level TSG3nO3: Low level to high level	Value of TSG3nDTC0W register
TSG3nO5: High level to low level TSG3nO6: Low level to high level	Value of TSG3nDTC1W register
TSG3nO6: High level to low level TSG3nO5: Low level to high level	Value of TSG3nDTC0W register

NOTE

Dead time counter keeps operating even when operation stop (TSG3nTE = 0) setting collided with dead time insert timing, and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

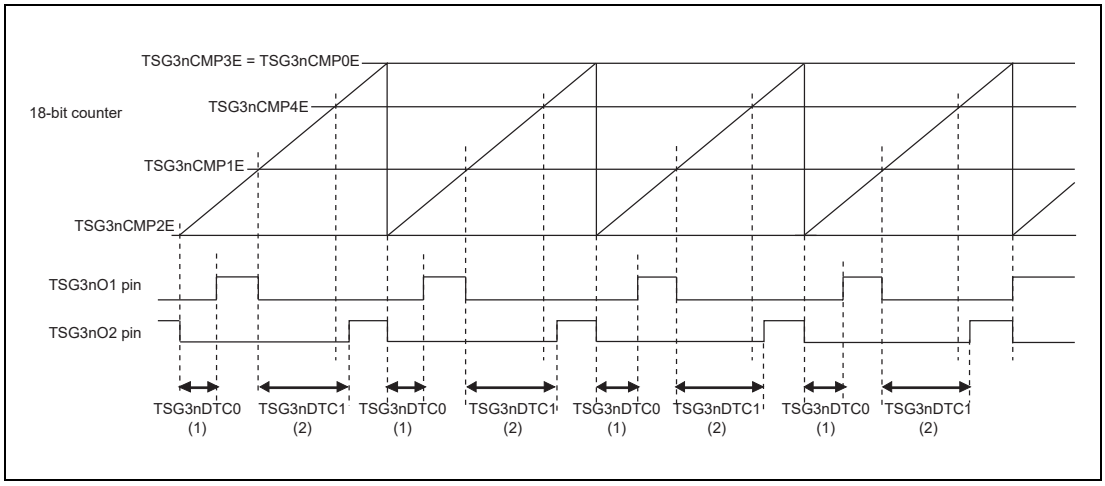


Figure 19.53 Example of Dead Time Control between TSG3nO1 and TSG3nO2 Outputs (1/2)

During (1), The dead time counter starts counting at the falling edge of the TSG3nO2 output. At this time, even after the 18-bit counter reaches 00000_H, the TSG3nO1 output stays inactive because the dead time counter is still operating. The TSG3nO1 output becomes active at the timing when the dead time count operation ends.

At (2), the dead time counter starts counting at the falling edge of the TSG3nO1 output. Even after the match of the 18-bit counter and TSG3nCMP4E, the TSG3nO2 output stays inactive because the dead time counter is still operating. The TSG3nO2 output becomes active at the timing when the dead time count operation ends.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

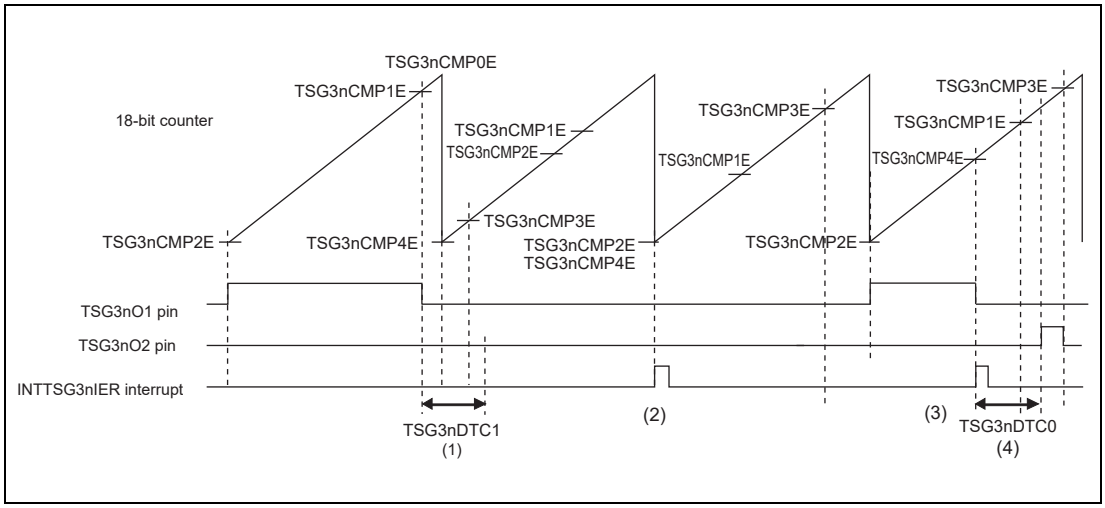


Figure 19.53 Example of Dead Time Control between TSG3nO1 and TSG3nO2 Outputs (2/2)

During (1), the dead time counter starts counting at the falling edge of the TSG3nO1 output. Even after the 18-bit counter reaches 00000_H and the match occurs between the 18-bit counter and TSG3nCMP4E, the TSG3nO2 output stays inactive because the dead time counter is still operating. Moreover, since the TSG3nCMP3E register compare match occurs before the operation of the dead time counter ends, the TSG3nO2 output stays inactive.

TSG3nCMP1E + TSG3nDTC1 ≥ TSG3nCMP0E + TSG3nCMP2E
(TSG3nO2 stays inactive)

TSG3nCMP2E + TSG3nDTC0 ≥ TSG3nCMP0E + TSG3nCMP1E
(TSG3nO1 stays inactive)

At (2), the INTTSG3nIER interrupt occurs because the TSG3nCMP2E register and the TSG3nCMP4E register are set so that the TSG3nO1 and TSG3nO2 outputs rise simultaneously. Here, the TSG3nO1 output and the TSG3nO2 output are inactive.

At (3), compare match with the TSG3nCMP4E register generates an INTTSG3nIER interrupt and both TSG3nO1 and TSG3nO2 outputs become inactive.

At (4), the falling edge (inactive) of the TSG3nO1 output is caused by the simultaneous active state detected and the dead time counter starts counting. After the end of the dead time counter operation, the TSG3nO2 output becomes active.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high.
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

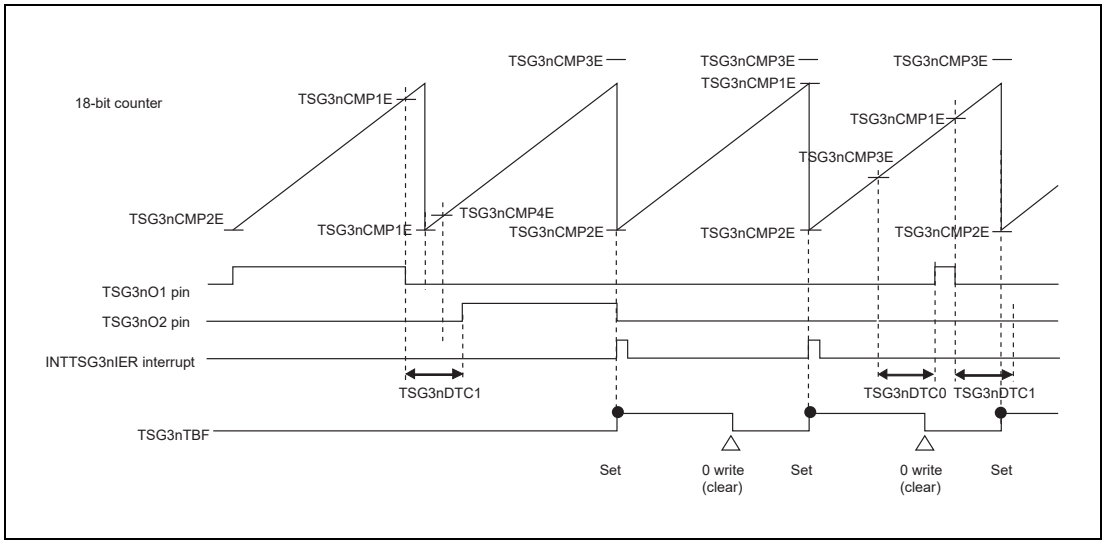


Figure 19.54 Example of 100% Duty Output at Dead Time Control

When the TSG3nO2 pin is set to duty cycle of 100% (TSG3nCMP3E ≥ TSG3nCMP0E + 1), the output of the TSG3nO1 pin is fixed to a low level. This control is intended to mask the active condition of TSG3nO1 output since the TSG3nO2 output is active before the TSG3nO1 output becomes active. In this case, the INTTSG3nIER interrupt is also generated because TSG3nO1 and TSG3nO2 outputs become high simultaneously.

NOTES

1. The TSG3nO1 and TSG3nO2 pins are set to active high.
2. The TSG3nO3 and TSG3nO4 pins and the TSG3nO5 and TSG3nO6 pins outputs behave similarly.

(4) Dead Time Rewriting during Timer Operation in PWM Mode

In PWM mode, it is possible to rewrite TSG3n dead time setting registers TSG3nDTC0W and TSG3nDTC1W while counting. The new settings are active at reload timing. It is not possible to change the dead time setting by rewriting at any time.

Please enable reloading by writing to the TSG3nCMP1E register.

19.4.7.2 HT-PWM mode (High accuracy Triangular - Pulse Width Modulation mode)

Overview

In this mode, the 18-bit counter (up/down count by ± 2 bits, practically 17 bits) and the 18-bit compare registers (LSB is used to control additional pulse) are used to generate a 6-phase PWM signal.

Prerequisites

- Set the carrier wave period to TSG3nCMP0E.
- Set the duty cycle of the voltage data signals of the U phase, V phase, and W phase with TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE (The values set to TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE are reflected immediately to the corresponding TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)).
- Symmetric triangular wave control is described in this section. Please refer to **Section 19.4.7.2 (10), Asymmetric Triangular Wave Control in HT-PWM Mode**, for asymmetric triangular wave control.

Functional description

Set the period of carrier wave and the duty cycle of the U phase, the V phase, and the W phase. Counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter counts up from TSG3nDTC0 as the minimum value, and counts down upon the match of the maximum value of TSG3nCMP0E + TSG3nDTC0.

The dead time is set with TSG3nDTC0 and TSG3nDTC1. TSG3nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch and TSG3nDTC1 sets the dead time of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG3nDTT1 to TSG3nDTT3) for dead time generation load the set values of TSG3nDTC0 and TSG3nDTC1 by the compare match of the 18-bit counter and the TSG3nCMPm buffer register (m = 1, 2, 5, 6, 9, 10), and start down-counting.

The INTTSG3nIm interrupts (m = 01, 02, 05, 06, 09, 10) are generated by the compare match of the 18-bit counter with the TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E buffer registers.

INTTSG3nIm interrupts (m = 3, 4, 7, 8, 11, 12) are generated by the compare match of the 18-bit counter with the TSG3nCMP3E, 7E, and 11E buffer registers when counting down (TSG3nCUF = 1), and with the TSG3nCMP4E, 8E, and 12E buffer registers when counting up (TSG3nCUF = 0).

NOTE

The HT-PWM mode is enabled when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 001_B.

(1) Block Diagram and Basic Timing Chart

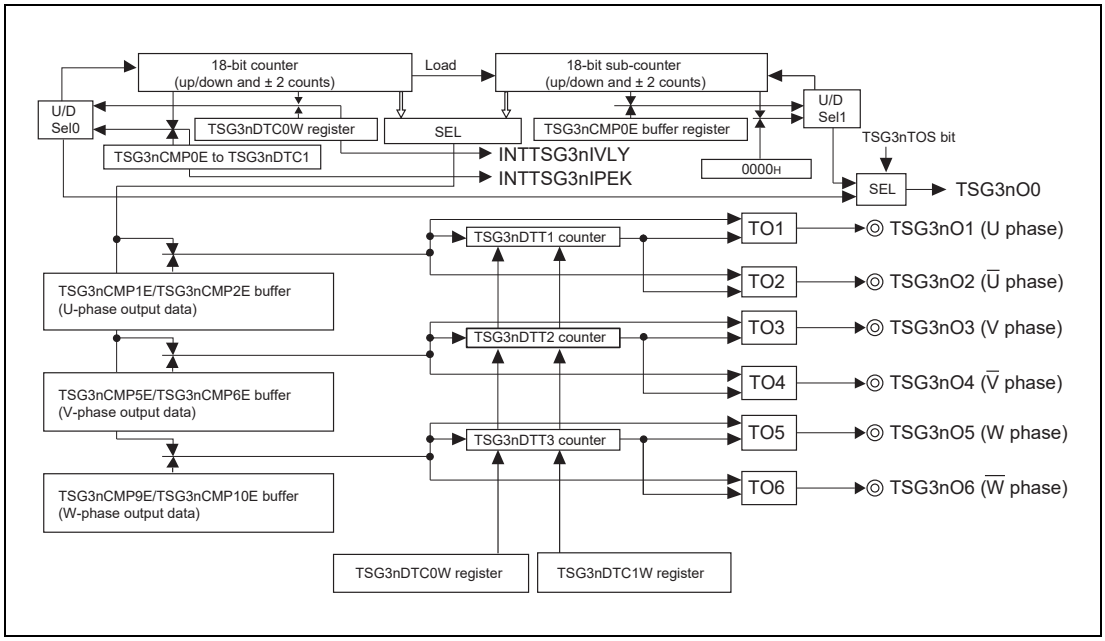


Figure 19.55 Block Diagram in HT-PWM Mode

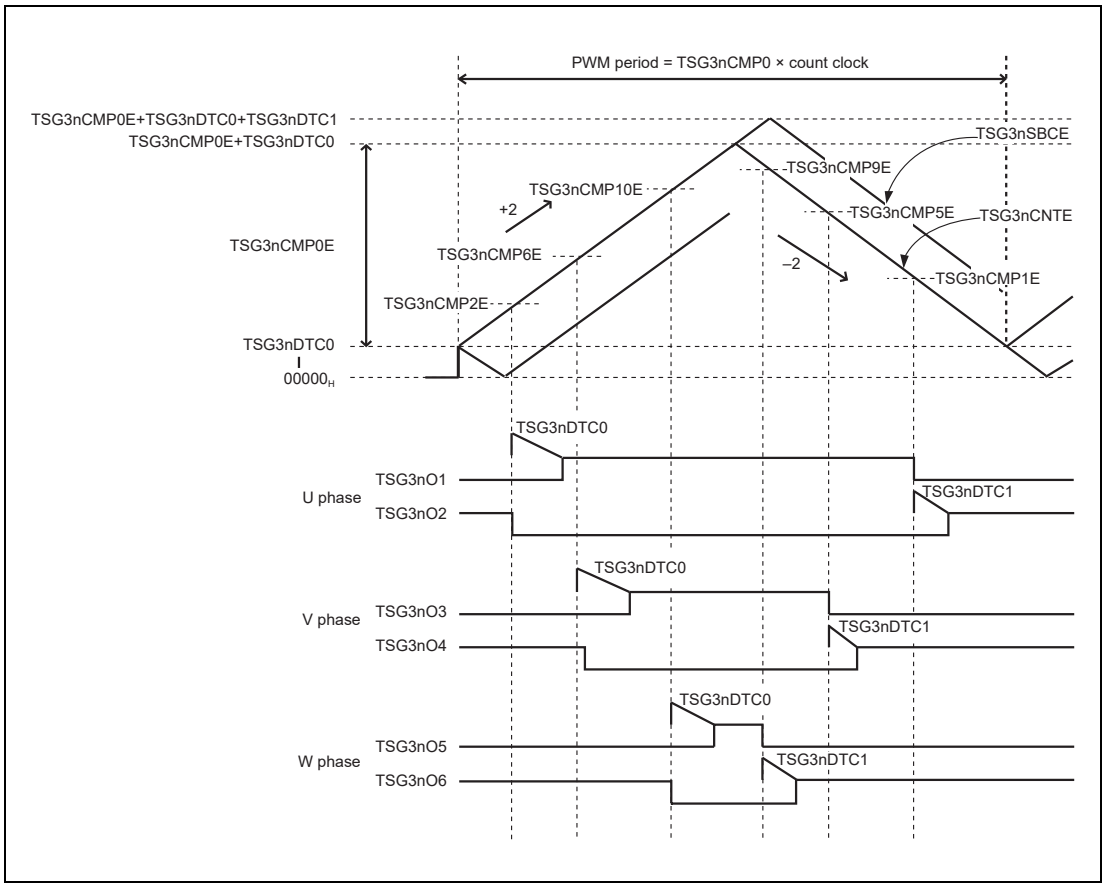


Figure 19.56 Basic Timing in HT-PWM Mode

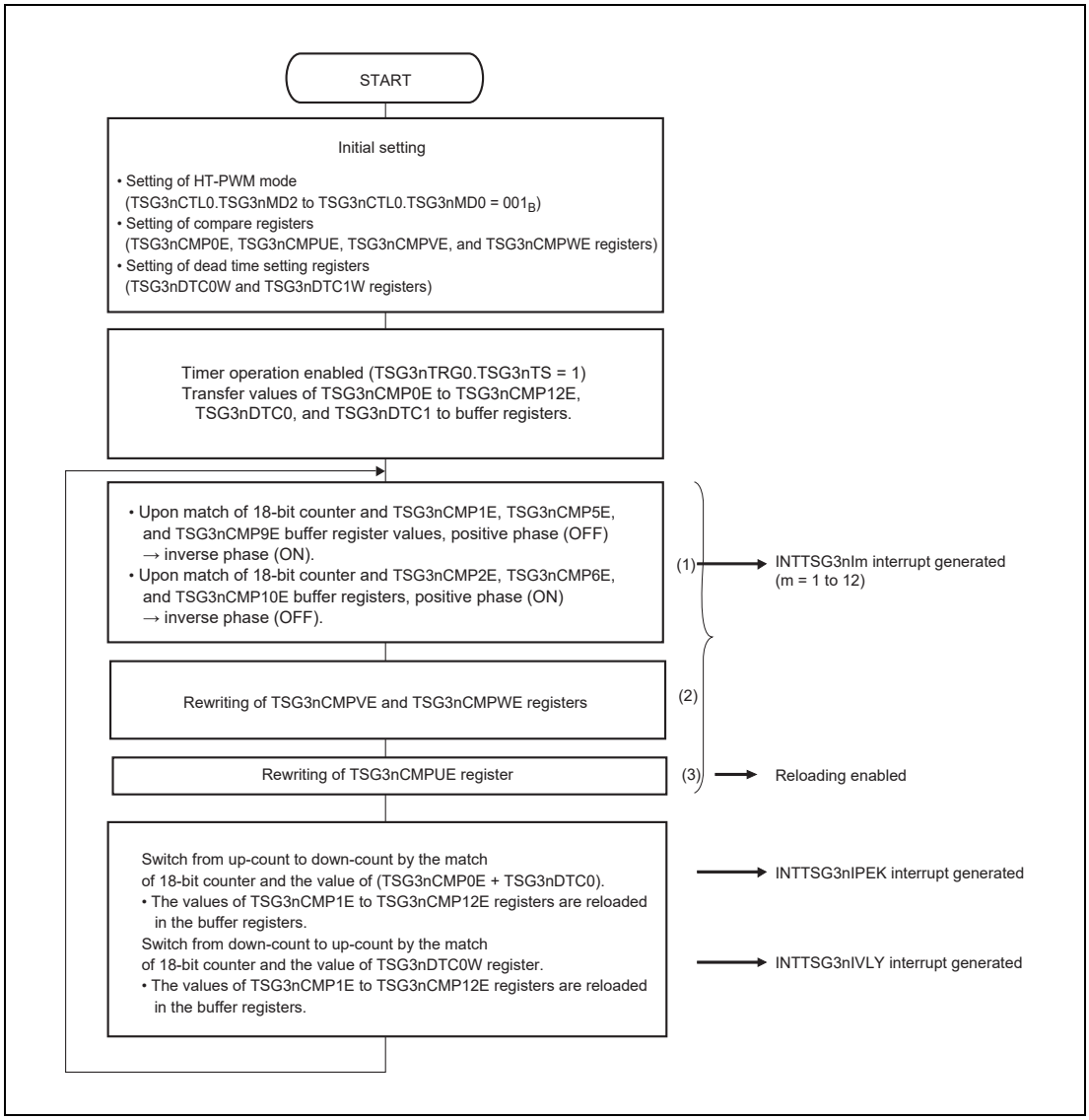


Figure 19.57 Basic Operation Flow in HT-PWM Mode

NOTE

- Write access to TSG3nCMPUE (TSG3nCMP1E) includes reloading enabling operation. Therefore, (3) must be done after (2).
- The INTTSG3nIPEK interrupt is generated only when TSG3nCTL4.TSG3nPIE = 1.
- The INTTSG3nIVLY interrupt is generated only when TSG3nCTL4.TSG3nVIE = 1.
- INTTSG3nI3, INTTSG3nI7, and INTTSG3nI11 outputs an interrupt at the match timing of TSG3nCMP3E, 7E, and 11E with TSG3nCNTE when counting down (TSG3nCUF = 1). INTTSG3nI4, INTTSG3nI8, and INTTSG3nI12 outputs an interrupt at the match timing of TSG3nCMP4E, 8E, and 12E with TSG3nCNTE when counting up (TSG3nCUF = 0).

(2) List of HT-PWM Mode Operations

Table 19.62 Counter Function in HT-PWM Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger (counting up from TSG3nDTC0)
	Up count	Compare match of TSG3nDTC0 buffer register and 18-bit counter
	Down count	Compare match of TSG3nCMP0E + TSG3nDTC0 buffer register and 18-bit counter
	Clear	—
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1
18-bit sub-counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger (counting down from TSG3nDTC0)
	Up count	Underflow
	Down count	Compare match of TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 buffer register and 18-bit sub-counter
	Load	• TSG3nCMP0E + TSG3nDTC0: When value of 18-bit counter matches the value of buffer register TSG3nCMP0E + TSG3nDTC0 • TSG3nDTC0: When value of 18-bit counter matches the value of the buffer register TSG3nDTC0
	Clear	—
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 19.63 Compare Register and Dead Time Setting Register Functions in HT-PWM Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nCMPUE	—	Possible	PWM control for U phase
TSG3nCMP1E, TSG3nCMP2E	Reload/Anytime rewrite		
TSG3nCMPVE	—	Possible	PWM control for V phase
TSG3nCMP5E, TSG3nCMP6E	Reload/Anytime rewrite		
TSG3nCMPWE	—	Possible	PWM control for W phase
TSG3nCMP9E, TSG3nCMP10E	Reload/Anytime rewrite		
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic signal output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible conditionally	Period and dead time setting

NOTE

- The rewritten values of TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE are set to TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E and TSG3nCMP10E.
- For rewriting method of TSG3nDTC0 and TSG3nDTC1, see **Section 19.4.7.2**, **(8)**, **(a) TSG3nDTC0 and TSG3nDTC1 Rewriting**.

Table 19.64 Timer Output Function in HT-PWM Mode

Pin	Function
TSG3nO0	Active level output during up count, inactive level output at down count of the 18-bit counter/sub-counter
TSG3nO1	PWM output with dead time by compare match of TSG3nCMP1E buffer register and 18-bit counter (down count) and compare match of TSG3nCMP2E buffer register and 18-bit counter (up count) PWM output by compare match of TSG3nCMP1E buffer register and 18-bit sub-counter (during down counting) and compare match of TSG3nCMP2E buffer register and 18-bit sub-counter (during up counting) while TSG3nCMP1E < DTC0
TSG3nO2	Inverse phase output to TSG3nO1
TSG3nO3	PWM output with dead time by compare match of TSG3nCMP5E buffer register and 18-bit counter (down count) and compare match of TSG3nCMP6E buffer register and 18-bit counter (up count). PWM output by compare match of TSG3nCMP3E buffer register and 18-bit sub-counter (during down counting) and compare match of TSG3nCMP6E buffer register and 18-bit sub-counter (during up counting) while TSG3nCMP5E < DTC0
TSG3nO4	Inverse phase output to TSG3nO3
TSG3nO5	PWM output with dead time by compare match of TSG3nCMP9E buffer register and 18-bit counter (down count) and TSG3nCMP10E buffer register and 18-bit counter (up count). PWM output by compare match of TSG3nCMP5E buffer register and 18-bit sub-counter (during down counting) and compare match of TSG3nCMP10E buffer register and 18-bit sub-counter (during up counting) while TSG3nCMP9E < DTC0
TSG3nO6	Inverse phase output to TSG3nO5
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

NOTES

1. State of TSG3nO0 output can be switched with TSG3nIOC1.TSG3nTOS.
2. When the peak and trough values of the 18-bit sub-counter are set in TSG3nCMP1E and TSG3nCMP2E, clearing takes precedence.

Table 19.65 Interrupt Request in HT-PWM Mode

Interrupt	Function
INTTSG3nI0	Compare match of TSG3nDTC0 buffer register and 18-bit counter (periodic interrupt)
INTTSG3nIm (m = 1, 2, 5, 6, 9, 10)	Compare match of TSG3nCMPmE buffer register and 18-bit counter (m = 1, 2, 5, 6, 9, 10)
INTTSG3nIER	Error interrupt
INTTSG3nIVLY	Trough interrupt
INTTSG3nIPEK	Peak interrupt
INTTSG3nIWN	Warning interrupt

Table 19.66 Compare Match Timing in HT-PWM Mode

Compare Match	Timing
TSG3nCMP0E	When the 18-bit counter changes from TSG3nDTC0 to TSG3nDTC0 + 2
TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)	When 18-bit counter changes from TSG3nCMPmE to TSG3nCMPmE ± 2 (m = 1, 2, 5, 6, 9, 10)

Table 19.67 Example of Setting Each Timer Output Condition in HT-PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nO0	Toggle output	TSG3nCMP0E × count clock	Output an inactive level when counting up, and an active level when counting down.	—
TSG3nO1, TSG3nO3, TSG3nO5	PWM output	TSG3nCMP0E × count clock	Output an inactive level throughout one period (0% duty)	TSG3nCMP0E ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP0E – 1 (m = U, V, W)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = 0001_H (m = U, V, W)
			Output an active level throughout one period(100% duty)	TSG3nCMPmE = 0000_H (m = U, V, W)
TSG3nO2, TSG3nO4, TSG3nO6	PWM output		Output an inactive level throughout one period (0% duty)	TSG3nCMPmE ≤ TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
			Output an active level of one count clock in one period	TSG3nCMPm = TSG3nDTC0 + TSG3nDTC1 + 1 (m = U, V, W)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 – 1 (m = U, V, W)
			Output an active level throughout one period (100% duty)	TSG3nCMPmE = TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 (m = U, V, W)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	TSG3nCMP0E × count clock	See **Section 19.4.5, A/D Conversion Trigger Function**	

(3) Various Settings of HT-PWM Mode

Mode setting

HT-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 001_B.

Setting timer output

The output pins TSG3nO1 to TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The output pin TSG3nO0 indicates the up/down count status of the 18-bit counter or the 18-bit sub-counter. Switch between the 18-bit sub-counter and the 18-bit counter is done with the TSG3nIOC1.TSG3nTOS bit.

The TSG3nO7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of the positive phase and inverse phase is enabled by setting TSG3nIOC1.TSG3nEOC to 1. In HT-PWM mode, with any value set in the compare register, the simultaneous active state of the positive phase and inverse phase is not possible. For the detail, see **Section 19.4.6, Error/Warning Interrupt**.

Setting register rewriting timing with reload function

With TSG3nCTL3.TSG3nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG3nCTL4.TSG3nPRE or TSG3nVRE to 1.

The reload timing is not generated if both the TSG3nPRE bits and TSG3nVRE bits are set to 0.

When “anytime rewrite” is specified, the unintended output may be generated depending on the rewrite timing.

Setting interrupts and skipping function

Interrupts and the skipping function are set with TSG3nCTL4. TSG3nPIE should be set to 1 when peak interrupt (INTTSG3nIPEK) is necessary and TSG3nVIE to 1 when trough interrupt (INTTSG3nIVLY) is necessary. To use the skipping function for peak/trough interrupts, the TSG3nRCC4 to TSG3nRCC0 must be set.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSG3nADTRG0 signal), use TSG3nCTL5.TSG3nAT09 to TSG3nAT00. With TSG3nAT09 to TSG3nAT00, A/D conversion trigger output is enabled or disabled at the match of 18-bit counter (during up count) with TSG3nDCMP2E to TSG3nDCMP0E, the match of the 18-bit counter (during down count) with TSG3nDCMP2E to TSG3nDCMP0E, the 18-bit counter peak interrupt (INTTSG3nIPEK), the 18-bit counter trough interrupt (INTTSG3nIVLY), the 18-bit sub-counter peak timing, and 18-bit sub-counter trough timing.

To set A/D conversion trigger 1 (TSG3nADTRG1 signal), use TSG3nCTL6.TSG3nAT19 to TSG3nAT10.

To set the match timing of 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to the pertinent register.

The skipping function can be used for TSG3nADTRG0 and the TSG3nADTRG1 signals. Use TSG3nACC01, TSG3nACC00 of TSG3nCTL5, TSG3nACC10, and TSG3nACC11 of TSG3nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

Set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E-TSG3nDCMP0E correctly when using the TSG3nO7 output for the A/D conversion trigger timing pulse.

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

The dead time is calculated by the following expressions:

PCLK × TSG3nDTC0

PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, TSG3nO5 to the inactive state and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state, respectively.

TSG3nDTC0 and TSG3nDTC1 can only be set to an even value.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

TSG3nCMP0E = Carrier period/count clock period (PCLK)

Satisfy the following requirements when setting the TSG3nCMP0E register regarding the dead time:

- TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 ≤ 3FFFEH
- TSG3nCMP0E > TSG3nDTC0
- TSG3nCMP0E > TSG3nDTC1
- TSG3nCMP0E > 3 × MAX (TSG3nDTC0, TSG3nDTC1)
- TSG3nCMP0E: Even number

NOTE

MAX (A, B) indicates the larger value of A and B.

Duty (PWM width) setting

The duty of the U phase, the V phase, and the W phase is set with TSG3nCMPmE (m = U, V, W, or 1, 2, 5, 6, 9, and 10), respectively. The setting range of the compare registers is as follows:

00000_H ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1

LSB (least significant bit) of TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE indicates the setting of an additional pulse. When TSG3nCMPUE = 00003_H, the change in the inverse phase (TSG3nO2 output) is done one count clock later compared to the TSG3nCMPUE = 00002_H setting (when the 18-bit counter is up-counting). The additional pulse cannot be set to TSG3nCMP1E,

TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, or TSG3nCMP10E (only even numbers can be set to these registers).

(4) H18-bit counter Operation in HT-PWM Mode

The 18-bit counter is initialized to 00000_H and the value of TSG3nDTC0 is loaded immediately after starting the TSG3n timer operation (TSG3nTRG0.TSG3nTS = 1). Afterwards, counting is done by +2. After 18-bit counter reaches the value of TSG3nCMP0E + TSG3nDTC0, counting is done by -2.

The following figure shows 18-bit counter operation.

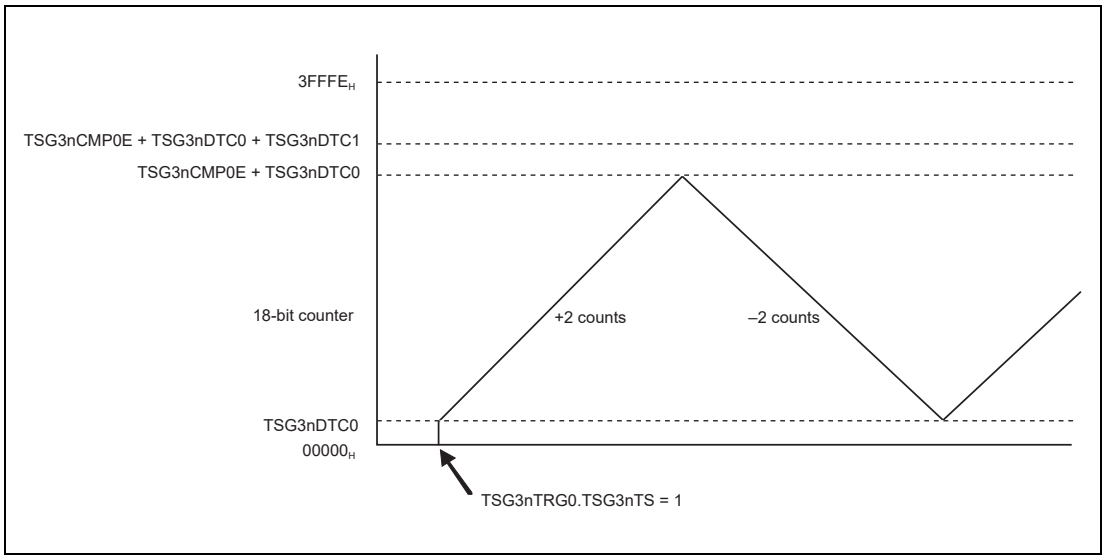


Figure 19.58 Example of 18-bit counter Operation in HT-PWM Mode

NOTE

Minimum 18-bit counter value: TSG3nDTC0
Maximum 18-bit counter value: TSG3nCMP0E + TSG3nDTC0
Carrier period: TSG3nCMP0E × count clock period (PCLK)

The 18-bit sub-counter is initialized to 00000_H and the value of TSG3nDTC0 is loaded immediately after starting the TSG3n timer operation (TSG3nTRG0.TSG3nTS = 1). Afterwards, counting by -2 is done until 00000_H is reached and counting by +2 begins. Next, the value of the 18-bit counter is loaded into the 18-bit sub-counter at a change timing of the 18-bit counter into the down count. Counting up by the 18-bit sub-counter continues until the value reaches the value of TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1, and then counting by -2 begins. Similarly, when the 18-bit counter value matches the TSG3nDTC0 value, the 18-bit counter value is loaded to the 18-bit sub-counter and the down count is continued.

The following figure shows the 18-bit sub-counter operation.

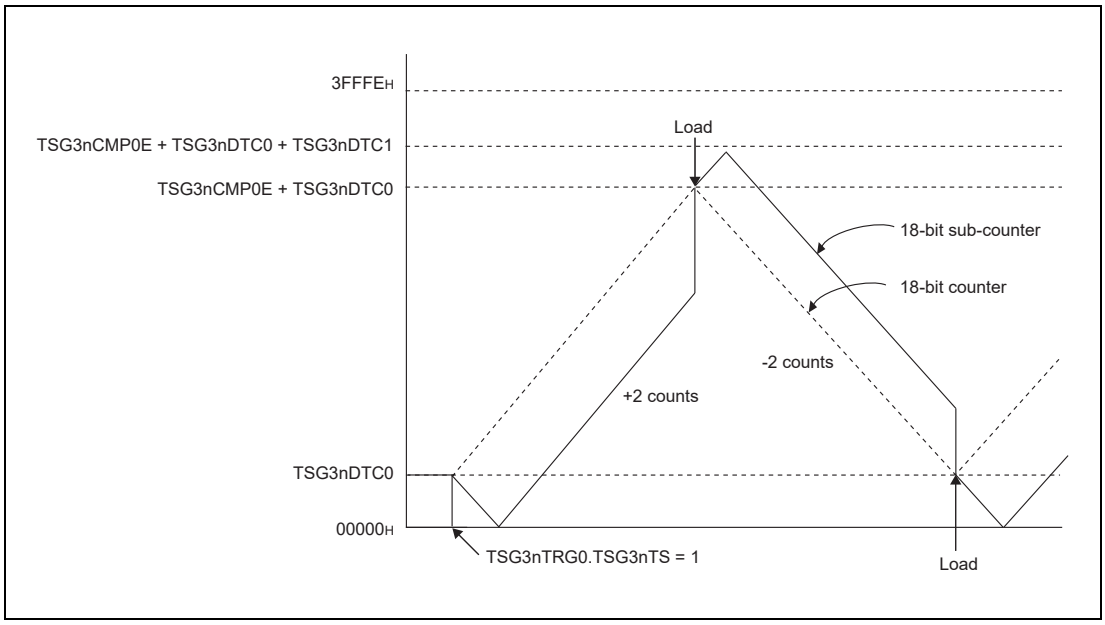


Figure 19.59 Example of 18-bit Sub-Counter Operation in HT-PWM Mode

NOTE

Minimum 18-bit sub-counter value: 00000_H
Maximum 18-bit sub-counter value: TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1

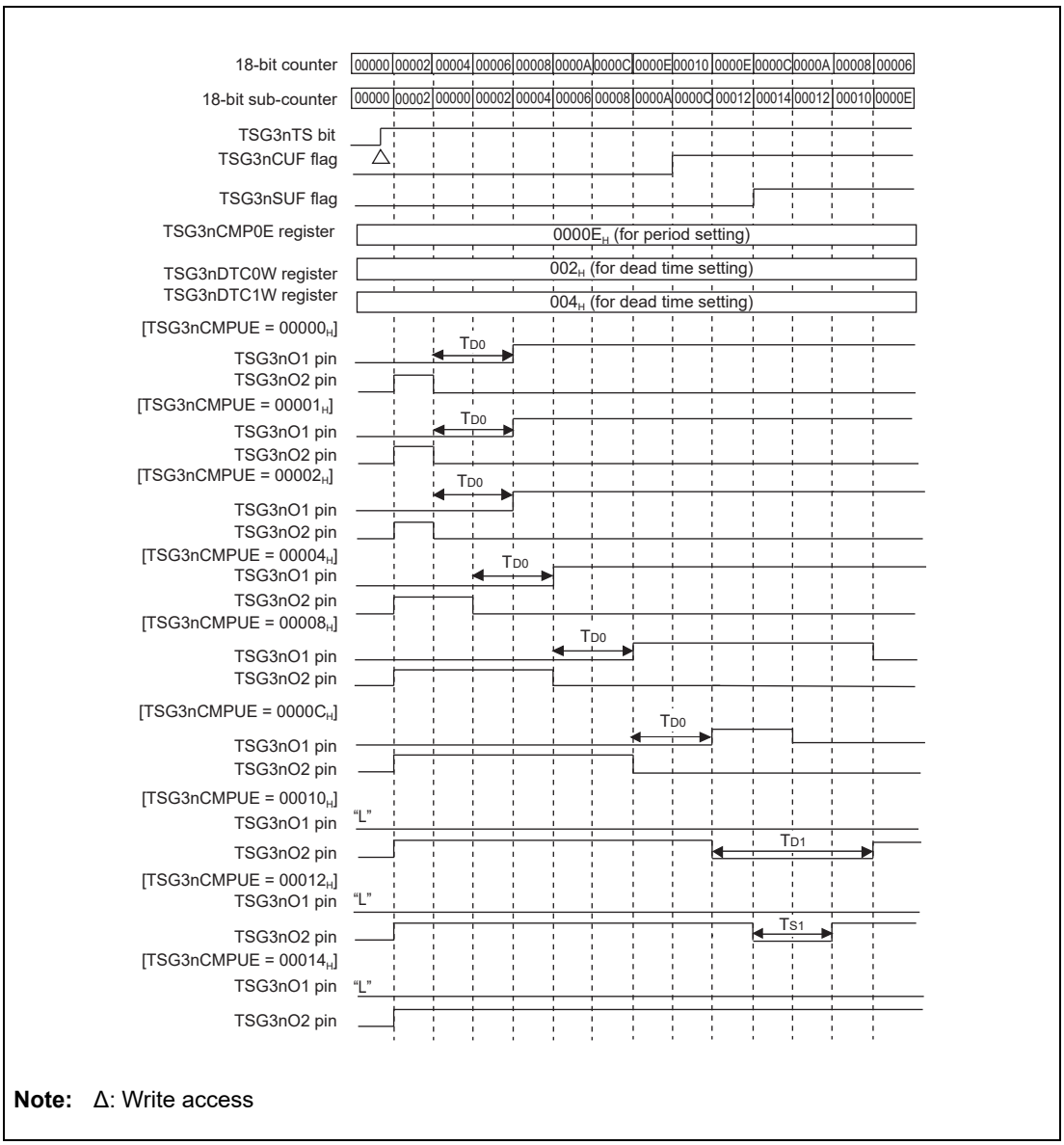
(5) Basic Operation of HT-PWM Mode

(a) Example of Timer Output Immediately after the Start of the TSG3n Timer Operation

The following figure shows the timing chart when TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H and TSG3nCMPUE = 00000_H to 00014_H (excerpt). In this example, TSG3nIOC2.TSG3nOL1 to TSG3nOL6 = 000000_B.

When operation starts (TSG3nTRG0.TSG3nTS = 1), the level of the TSG3nO2 pin changes to active. Afterwards, if TSG3nCMPUE ≤ TSG3nDTC0, the TSG3nO2 pin is cleared after 1 count clock cycle.

The TSG3nO2 pin is cleared upon a match of the 18-bit counter and the compare register (TSG3nCMP2E), or a match of the 18-bit sub-counter and the compare register (TSG3nCMP2E) if TSG3nCMPUE ≥ TSG3nDTC0. Afterwards, the TSG3nO1 pin is set after the set dead time period (the TSG3nO1 pin is not set if TSG3nCMPUE ≥ TSG3nCMP0E).



Note: Δ: Write access

Figure 19.60 Example of Timer Output when TSG3nTS is Set to 1 (Initial) in HT-PWM Mode)

NOTES

1. TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H
2. T_{D0}: Time depending on setting of the dead time in the TSG3nDTC0W register
   T_{D1}: Time depending on setting of the dead time in the TSG3nDTC1W register
   T_{S1}: Time decided by compare match of the 18-bit sub-counter and TSG3nCMPUE register, when TSG3nCMPUE >18-bit counter maximum value

(b) Example of Timer Output during TSG3n Timer Operation

The following figure shows the timing chart when TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H, and TSG3nCMPUE is set to 00000_H-00014_H (excerpt). In this example, TSG3nIOC2.TSG3nOL1-TSG3nOL6 = 000000_B.

The range of the active (high level) width of a positive phase (TSG3nO1) output is 00000_H ≤ TSG3nCMPUE ≤ TSG3nCMP0E (for the additional pulse). The range of the active (high level) width of an inverse phase (TSG3nO2) output is TSG3nDTC0 + TSG3nDTC1 ≤ TSG3nCMPUE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1.

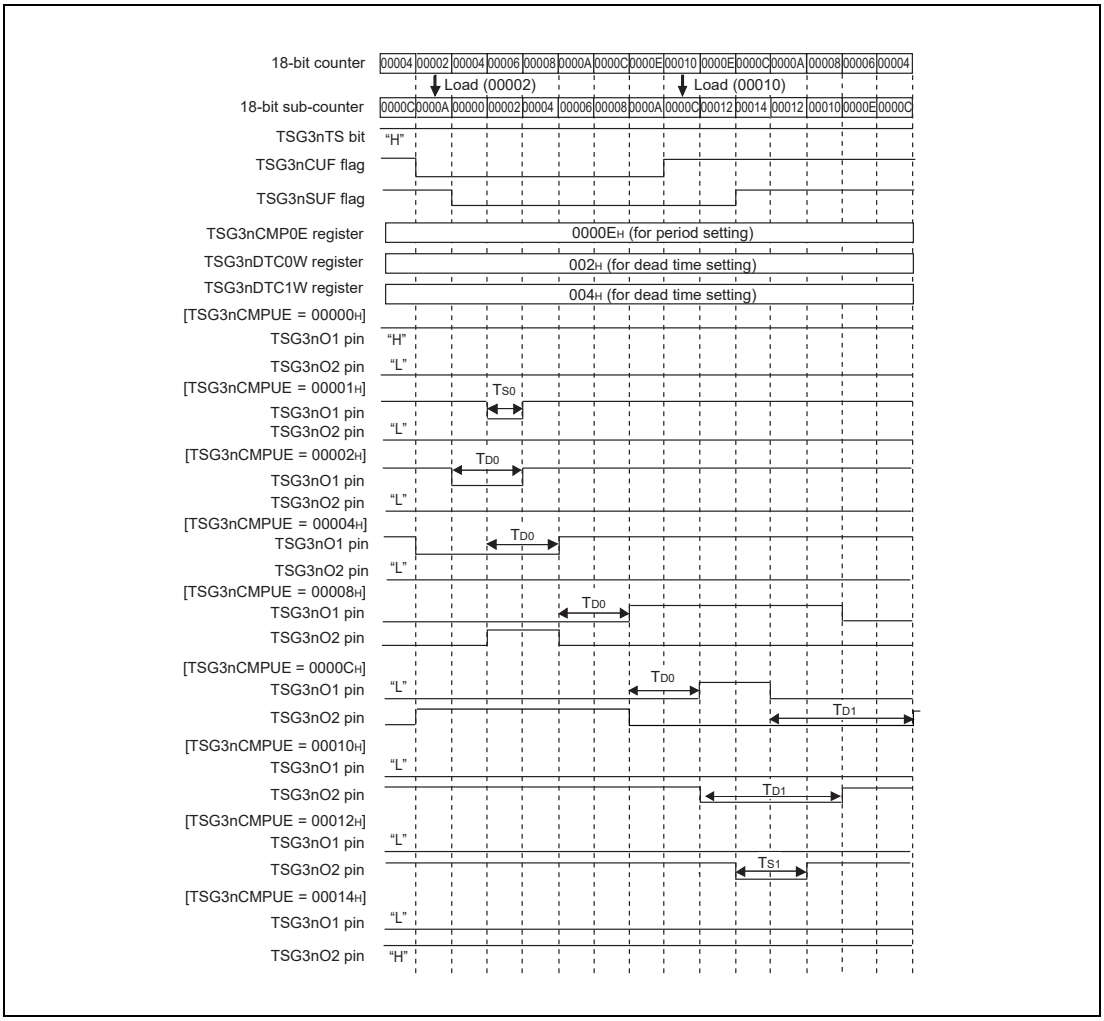


Figure 19.61 Example of Timer Output during TSG3n Operation in HT-PWM Mode

NOTES

1. TSG3nCMP0E = $0000E_H$, TSG3nDTC0 = 002_H, TSG3nDTC1 = 004_H
2. T_{D0}: Time depending on setting of the dead time in the TSG3nDTC0 register
   T_{D1}: Time depending on setting of the dead time in the TSG3nDTC1 register
   T_{S0}: Time decided by compare match of 18-bit sub-counter and the TSG3nCMPUE register, when TSG3nCMPUE < 18-bit counter minimum value
   T_{S1}: Time decided by compare match of the 18-bit sub-counter and the TSG3nCMPUE register, when TSG3nCMPUE > 18-bit counter maximum value

(6) Additional Pulse Control in HT-PWM Mode

The HT-PWM mode can generate an additional pulse by setting 1 to the LSB of the duty setting registers (TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE). This allows more precise control of the pulse duty than standard pulse control.

The following sections describe two examples of pulse output of TSG3nO1: additional pulse control is used in one example and additional pulse control is not used in another.

(a) Example of Pulse Output when Additional Pulse Control Is Used

Figure 19.62 shows the additional pulse control when an odd value is set to TSG3nCMPUE.

The arrows and numerical values show the width of the duty cycle of the TSG3nO1 output in one period.

When the additional pulse control is used as shown in **Figure 19.62**, the width of the output of the TSG3nO1 (duty cycle) can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in one-clock-cycle step.

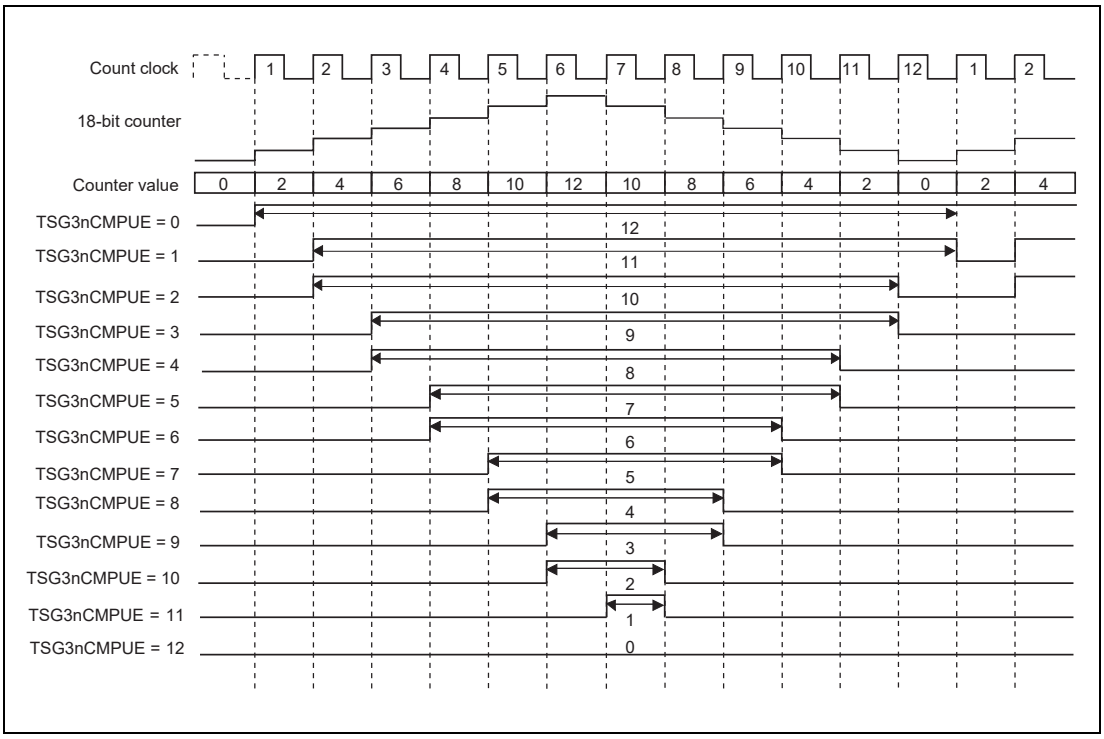


Figure 19.62 Example of TSG3nO1 Output when Additional Pulse Control Is Used in HT-PWM Mode

NOTE

TSG3nCMP0E = 12, TSG3nDTC0 = 0, TSG3nDTC1 = 0

(b) Example of Pulse Output when Additional Pulse Control Is Not Used

The arrows and numerical values in **Figure 19.63** show the width of the duty cycle of the TSG3nO1 output in one period.

When the additional pulse control is not used, the width of the TSG3nO1 output can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in two-clock-cycle step. In this case, the change in duty cycle is larger than that in the case when the additional pulse control is used.

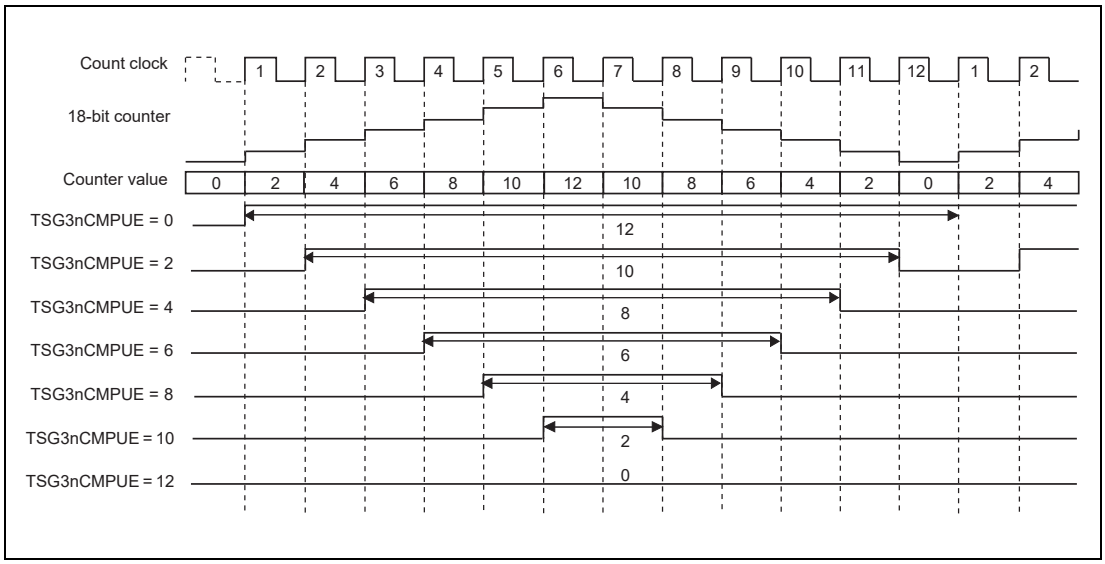


Figure 19.63 Example of Output when Additional Pulse Control Is Not Used in HT-PWM Mode

NOTE

TSG3nCMP0E = 12, TSG3nDTC0 = 0, TSG3nDTC1 = 0

(7) Dead Time Control in HT-PWM Mode

Duty setting registers are TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE in HT-PWM mode. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are two dead time setting registers (TSG3nDTC0W and TSG3nDTC1W) and six 10-bit down counters that operate synchronously with the count clock of the 18-bit counter. TSG3nDTC0 is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG3nDTC1 is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state.

The following figure shows the output waveform when TSG3nDTC0 = x and TSG3nDTC1 = y.

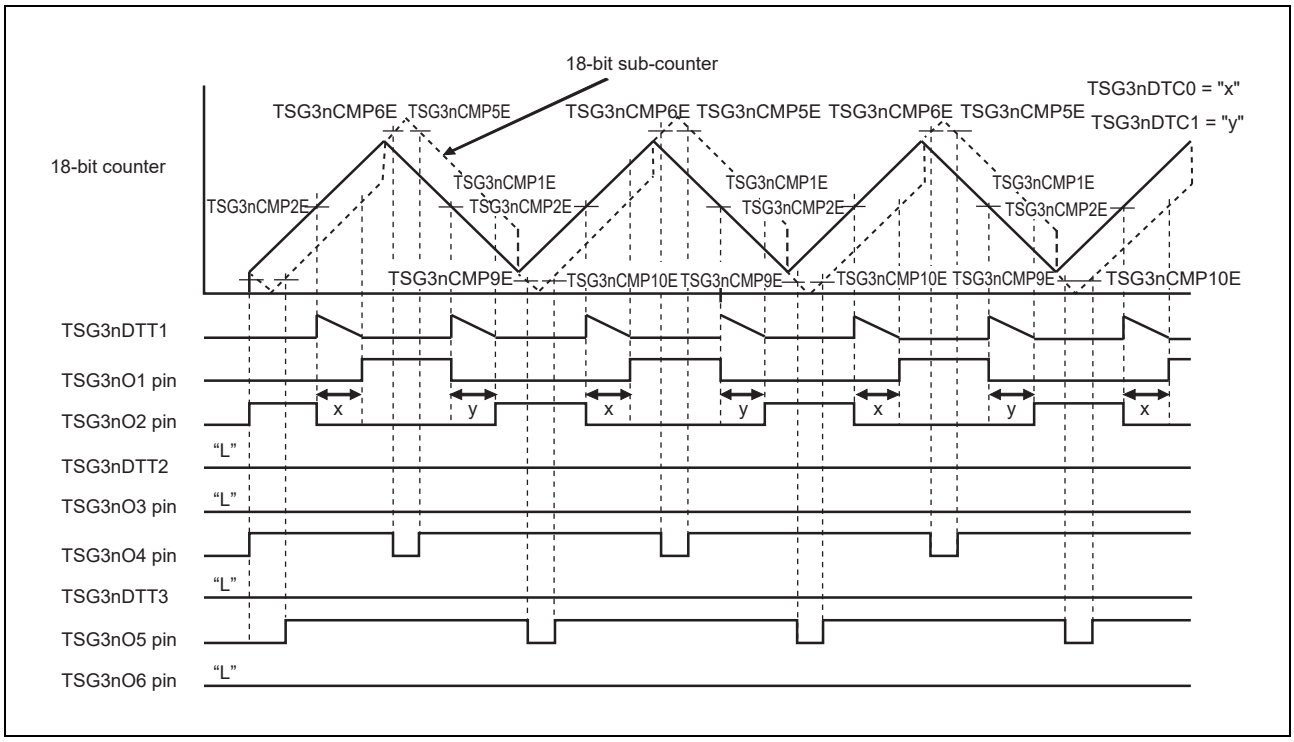


Figure 19.64 Example of Output Waveform with Dead Time in HT-PWM Mode

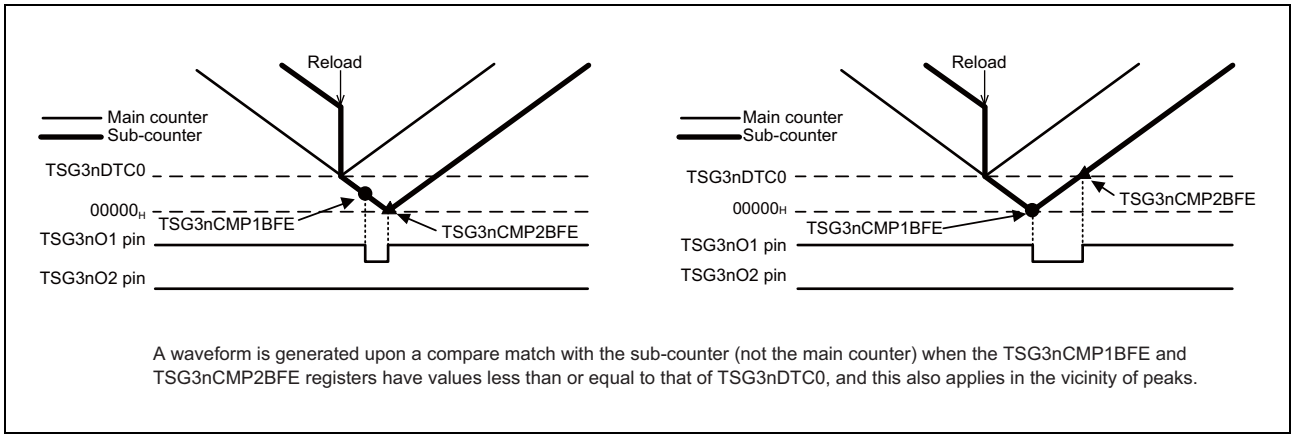


Figure 19.65 Example of Output Waveform near Trough after Reloading

(8) Notes Concerning Dead Time Control in HT-PWM Mode

(a) TSG3nDTC0 and TSG3nDTC1 Rewriting

It is possible to rewrite the dead time setting in TSG3nDTC0 and TSG3nDTC1 registers during timer operation.

CAUTIONS

1. **Rewrite TSG3nDTC0 and TSG3nDTC1 when the reload function is used (TSG3nRMC = 0).**
2. **The write protection code check function is applied when TSG3nDTC0 and TSG3nDTC1 are rewritten. For the detail, see the pertinent register descriptions (Section 19.3.44, Section 19.3.45, Section 19.3.64).**
3. **When the TSG3nCMP0E and TSg3nDTC1 are updated at the peak of the 18-bit counter:**
   When the set value of TSG3nCMPmE is greater than the updated TSG3nCMP0E + TSG3nDTC0 (new maximum value of the main counter), the match interrupt (INTTSG3nIm) is not generated immediately after reloading (m = 2, 6, or 10).

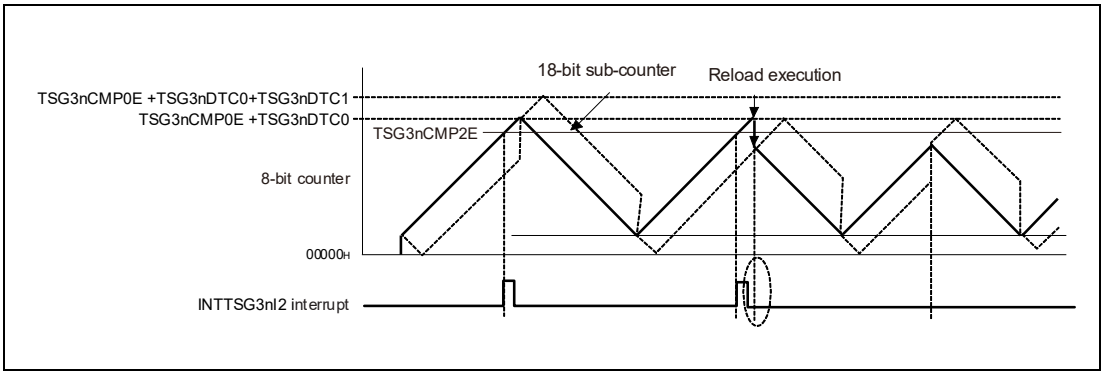


Figure 19.66 Example of Operation During Reloading at 18-Bit Counter Peak Timing

4. **When the TSG3nDTC0 is updated at the trough of the 18-bit counter:**
   When the TSG3nCMPmE set value is smaller than the updated TSG3nDTC0 (new minimum value of the main counter), the match interrupt (INTTSG3nIm) is not generated immediately after reloading (m = 1, 5, or 9).

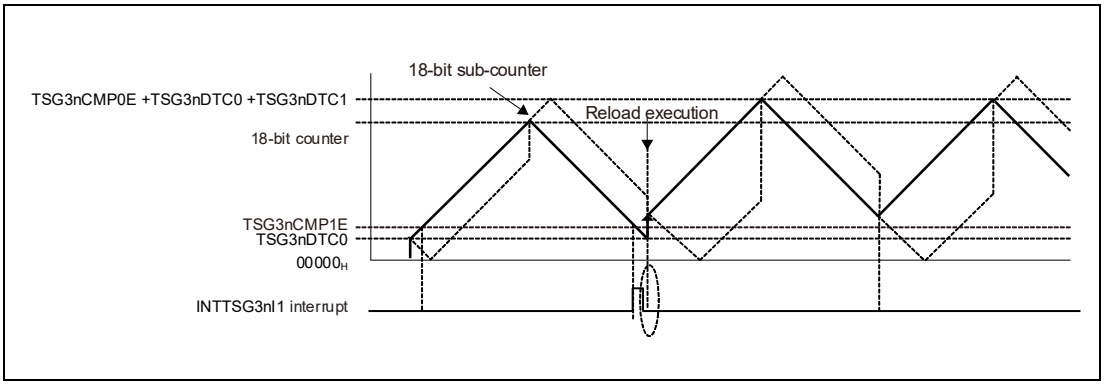


Figure 19.67 Example of Operation During Reloading at 18-Bit Counter Trough Timing

(9) Software Output Control Function in HT-PWM Mode

TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT1.TSG3nSPC2-TSG3nSPC0 are used in HT-PWM mode for software control of timer output control.

As shown in **Figure 19.68**, with TSG3nSTE = 0, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG3nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to HT-PWM mode output control.

For details, refer to **Section 19.4.7.10, Software Output Control Function**.

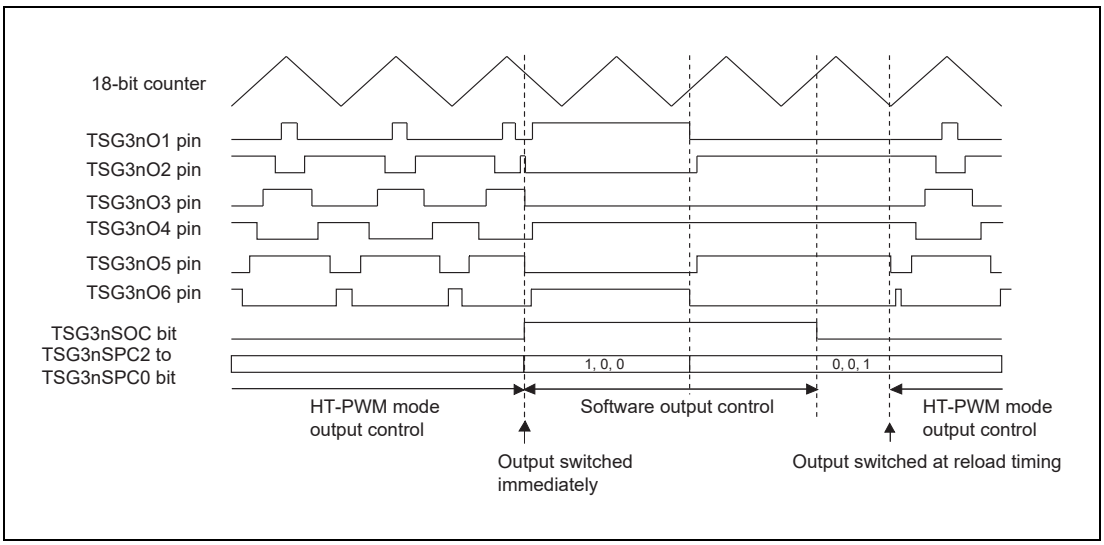


Figure 19.68 Example of Software Output Control Switching in HT-PWM Mode

CAUTION

Use reload (simultaneous rewrite) mode (TSG3nCTL3.TSG3nRMC = 0) when software output control function is used.

(a) Procedure for Software Output Control

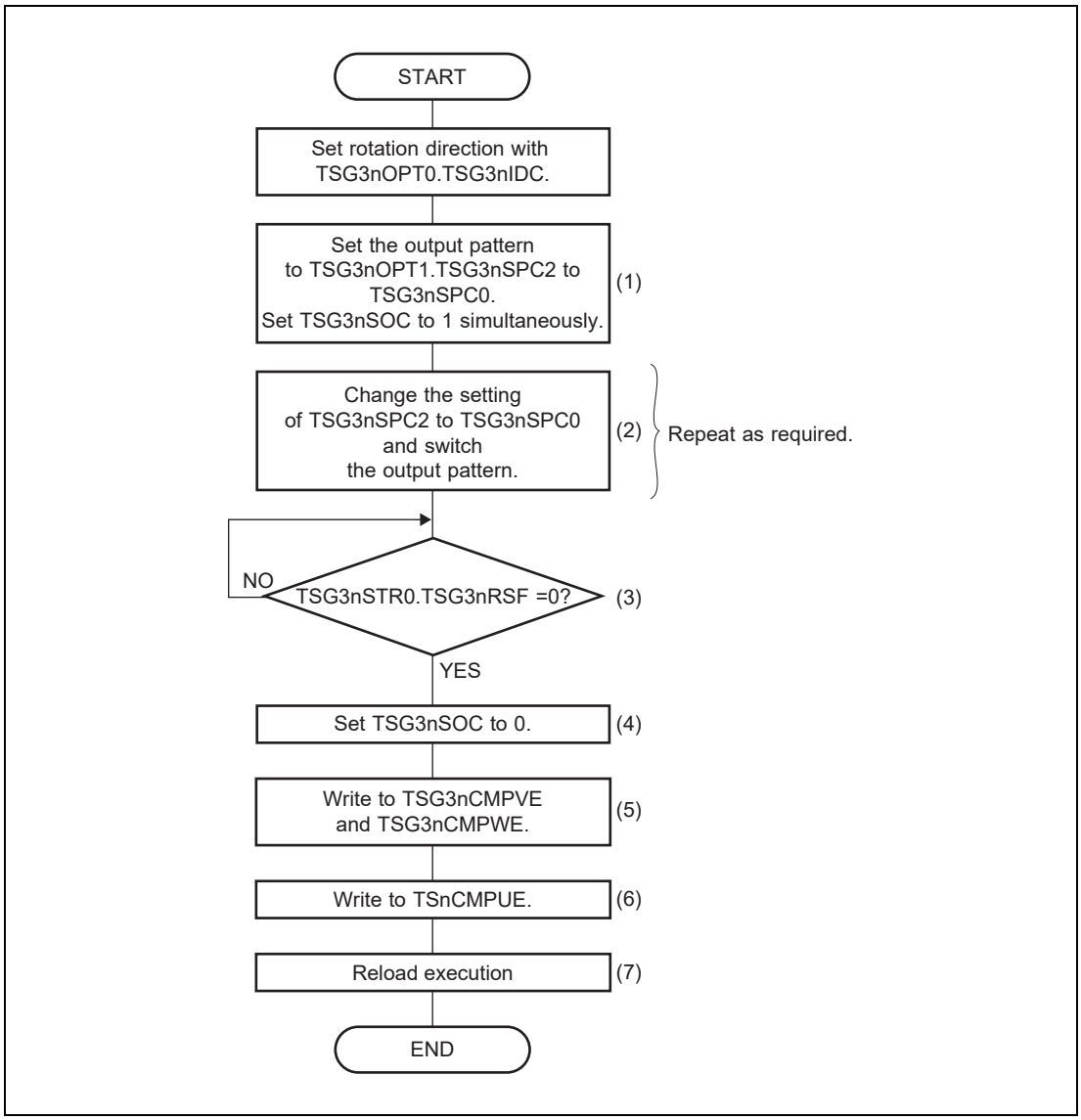


Figure 19.69 Flow of Software Output Control in HT-PWM Mode

The procedure for software output control is described below.

(1) Set the output pattern to the TSG3nOPT1.TSG3nSPC2-TSG3nSPC0. To enable software output control, set TSG3nOPT0.TSG3nSOC to 1 simultaneously.

(2) Change the output pattern setting of the TSG3nSPC2-TSG3nSPC0 to change the timer output.

(3) Confirm that reload request flag TSG3nSTR0.TSG3nRSF = 0. In case TSG3nRSF = 1, do not proceed to the following step until TSG3nRSF = 0.

(4) By setting TSG3nSOC = 0, the software control starts to be released (it is not released here yet).

(5) After releasing the software output control, set the necessary compare registers. Proceed to the following step when the register setting is not required. Here, change the registers with the reload function.

(6) Write to TSG3nCMPUE (TSG3nCMP1E) to start reloading.

(7) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (3), (4), (5), and (6). When reload cannot be executed, the software output cannot be released.

(10) Asymmetric Triangular Wave Control in HT-PWM Mode

In HT-PWM mode, it is possible to control output by an asymmetric triangular waveform by setting the different timings for setting and clearing the U, V, and W phases.

The following describes the differences of the asymmetric triangular wave control from the symmetric triangular wave control.

(a) PWM Setting

When a symmetric triangular wave is used, the output control of each of the U phase, V phase, and W phase is done by setting the set timing and the clear timing to the same value to the TSG3nCMPUE, TSG3nCMPVE, and TSG3nCMPWE. When an asymmetric triangular wave is used, the output control of each phase is done by setting TSG3nCMPmE as follows (m = 1, 2, 5, 6, 9, 10).

Prerequisites

- The clear timing of PWM of the voltage data signal of U, V and W phases is set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E.
- The set timing of PWM of the voltage data signal of U, V, and W phases is set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E.
- The set and clear timings of each phase can also be set with TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP5E, TSG3nCMP9E, and TSG3nCMP10E.
- TSG3nCMPmE can only be set to an even value (m = 1, 2, 5, 6, 9, 10).

(b) Timer Output

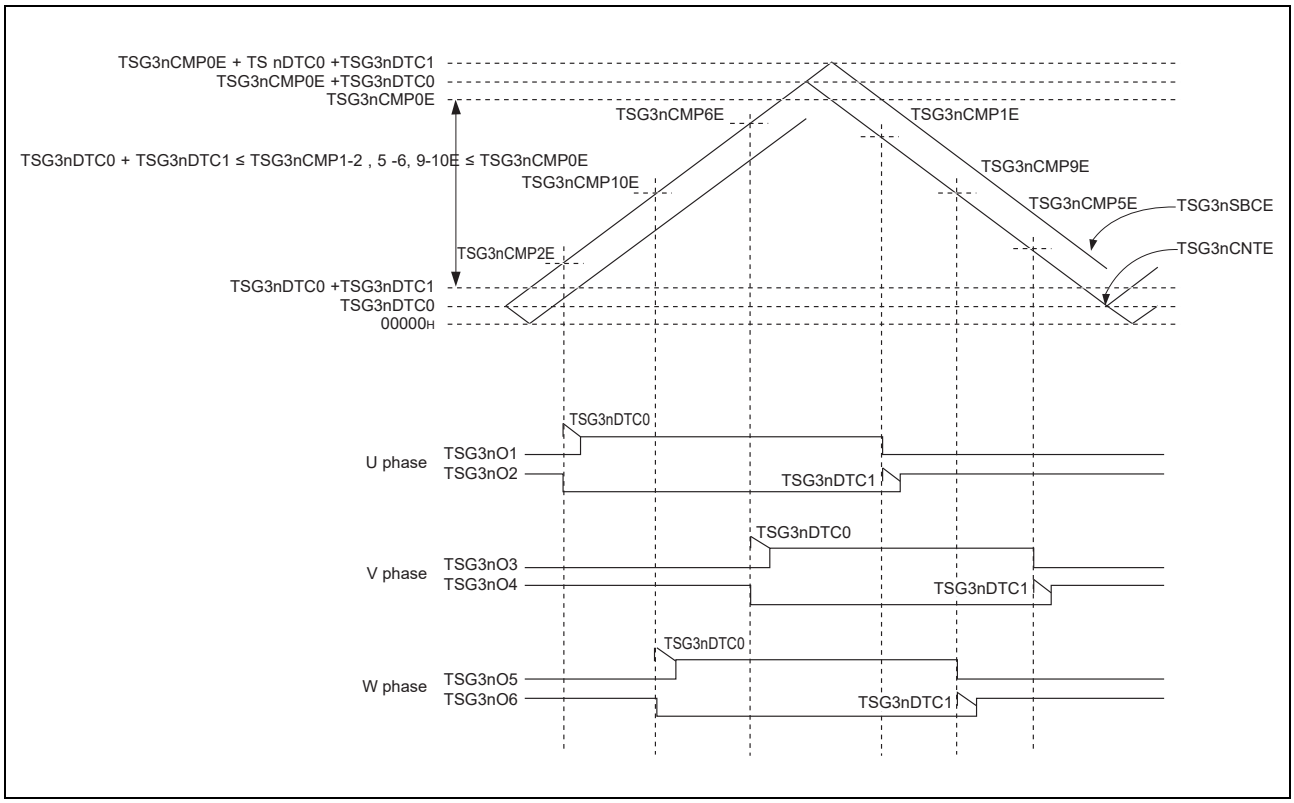


Figure 19.70 Example of Timer Output Waveform in HT-PWM Mode

NOTE

When output is controlled by the asymmetric triangular waveform, the following conditions apply to setting of TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10).

- TSG3nDTC0+TSG3nDTC1 ≤ TSG3nCMPmE ≤ TSG3nCMP0E
- Only when TSG3nCMPmE = TSG3nCMP (m + 1) E, or TSG3nCMPmE = TSG3nCMP (m + 1) E + 2, it is possible to set TSG3nCMPmE under the condition 00000_H ≤ TSG3nCMPmE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1, which also applies to the case where the symmetric triangular wave is used.

19.4.7.3 Data Transfer from EMU2

TSG3 can directly reflect the values of carrier cycle set in EMU2 and the U, V, and W phase duty calculated by EMU2 to the compare registers TSG3nCMP0E, 1E, 2E, 5E, 6E, 9E, and 10E.

When the carrier cycle value EMU2nCARR from EMU2 and the compare values EMU2nPWMUIP, EMU2nPWMVIP, and EMU2nPWMWIP of U, V, and W phases are input to TSG3 and the EMU2n register write signal is set to 1, the TSG3 compare registers TSG3nCMP0E, 1E, 2E, 5E, 6E, 9E, and 10E are updated.

When the calculation of the PWM IP in EMU2n is completed, the EMU2n register write signal is set to 1. In addition, the signal can also be controlled by software.

For details, see **Section 24.4.7, PWM IP** in **Section 24, Enhanced Motor Control Unit (EMU2)**.

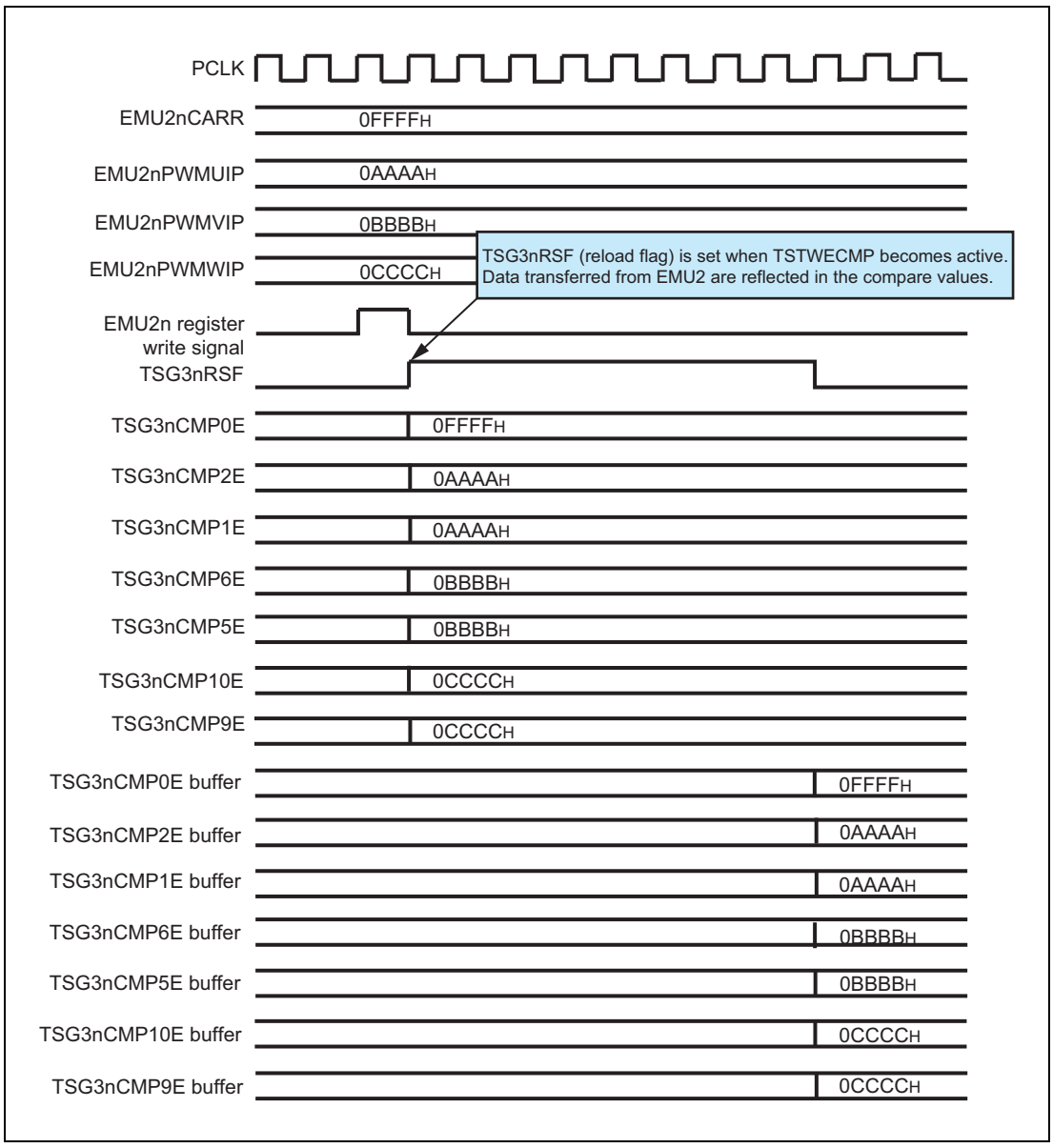


Figure 19.71 Transfer Timing of Data from EMU2

CAUTION

Transfer from EMU2 is enabled only in HT-PWM mode (TSG3nMD2-0 = 001) and in reload mode operation (TSG3nRMC = 0). Do not transfer data in PWM mode, SP-PWM mode, 120-DC mode, HSP-PWM mode or in anytime rewrite mode (TSG3nRMC = 1).

If transfers of data from EMU2 are performed sequentially before a reload timing is generated, the last transferred data is effective.

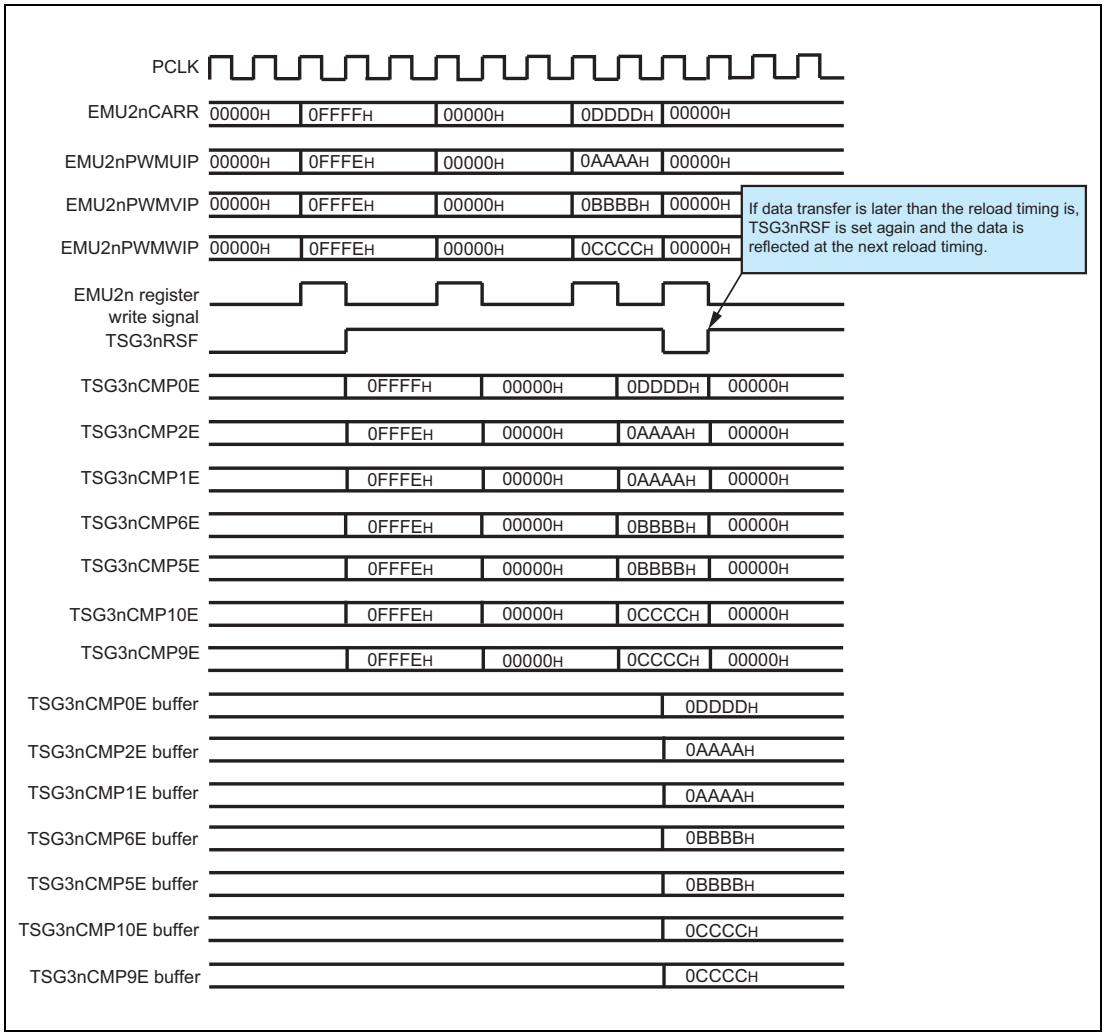


Figure 19.72 Transfer Timing of Data from EMU2

NOTES

1. If a transfer of data from EMU2 occurs at the same time as reload timing, TSG3nRSF is not cleared and the transferred data conflicted is reflected at the next reload timing.
2. Transfer of data from EMU2 occurs at the same time as a write access to a register, the write access is effective. For example, in the case of writing to TSG3nCMP0E and transfer of data from EMU2, writing makes effects to the TSG3nCMP0E register and transferred data is reflected in the other registers TSG3nCMP1E, 2E, 5E, 6E, 9E, and 10E.

19.4.7.4 ESW Function

ESW function outputs PWM which is not generated using the counter/compare value of TSG3 but adding the set dead time to the rectangular waveform from EMU2 to TSG3.

(1) PWM Output Using ESW Function

Setting 1 to the TSG3nOPT2.TSG3nESSC bit enables the ESW function. PWM output is switched from the one using TSG3 counter/compare values to the one derived from the rectangular waveform input from EMU2.

With the ESW function, the U-phase output pattern value, V-phase output pattern value, and W-phase output pattern value input from EMU2 are positive output (TSG3nO1, 3, and 5), and the inverse of the U-phase output pattern value, V-phase output pattern value, and W-phase output pattern value signals are inverse output (TSG3nO2, 4, and 6).

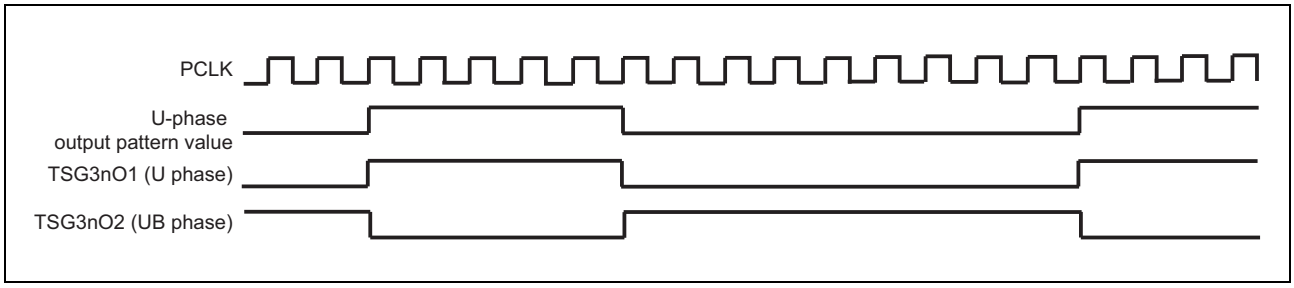


Figure 19.73 PWM Output Using ESW Function (With 0 Dead Time)

The setting of dead time is also effective in ESW function. The dead time set to TSG3nDTC0 is inserted to the positive phase (TSG3nO1, 3, and 5), and the dead time set to TSG3nDTC1 is inserted to the negative phase (TSG3nO2, 4, and 6).

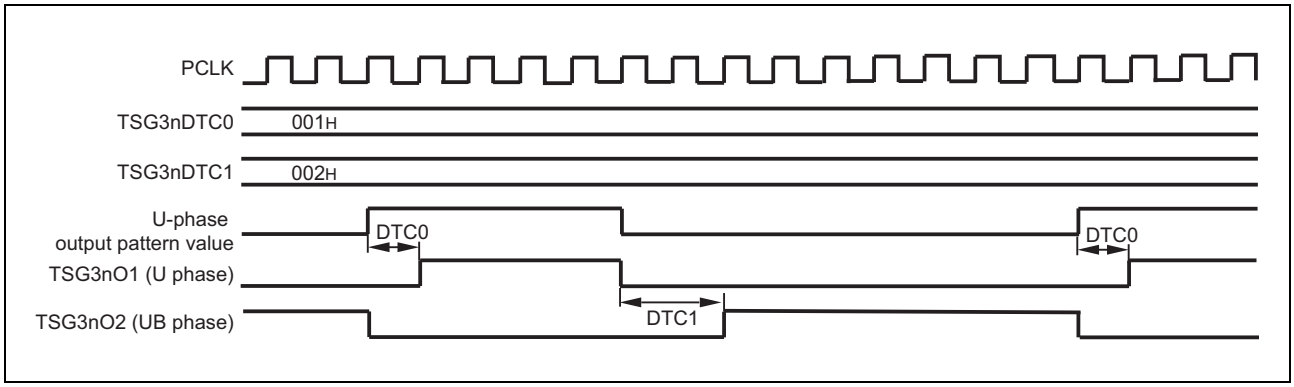


Figure 19.74 PWM Output Using ESW Function (TSG3nDTC0 = 1, TSG3nDTC1 = 2)

(2) Switching Operation to ESW Function

Changing the TSG3nOPT2.TSG3nESSC bit from 0 to 1 immediately enables the ESW function and PWM output is switched from to one derived from rectangular waveform input from EMU2. When the TSG3nESSC bit is changed from 1 to 0, time output is synchronized with the counter and switched to HT-PWM mode at reload timing.

(3) Dead Time Insertion when Switching to ESW unction

The set dead time is always inserted when switching to ESW function. If PWM output is switched from positive-phase high-level output to positive-phase high-level output of rectangular waveform, output at high level is continued (not to become inactive by switching to ESW function).

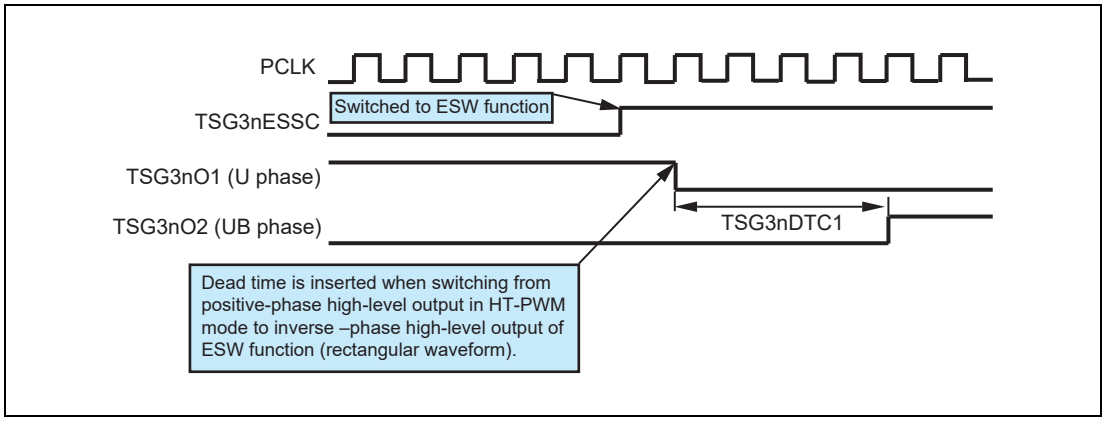


Figure 19.75 Switching Operation to ESW Function (From Positive-Phase High-Level to Inverse-Phase High-Level)

Dead time setting is also effective when ESW function is used. The dead time set in TSG3nDTC0 is inserted to the positive phase (TSG3nO1, 3, and 5), and the dead time set in TSG3nDTC1 is inserted to the inverse phase (TSG3nO2, 4, and 6).

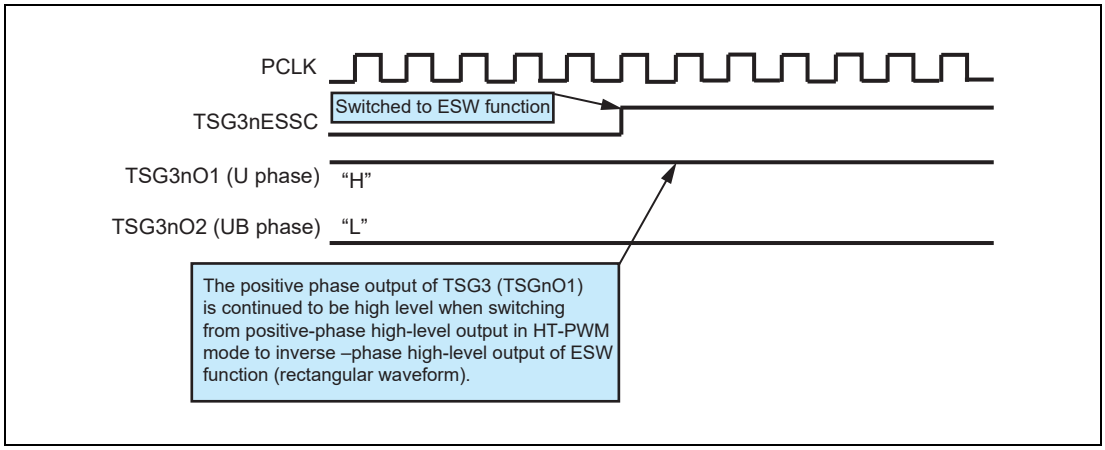


Figure 19.76 Switching Operation to ESW Function (From Positive-Phase High-Level to Positive-Phase High-Level)

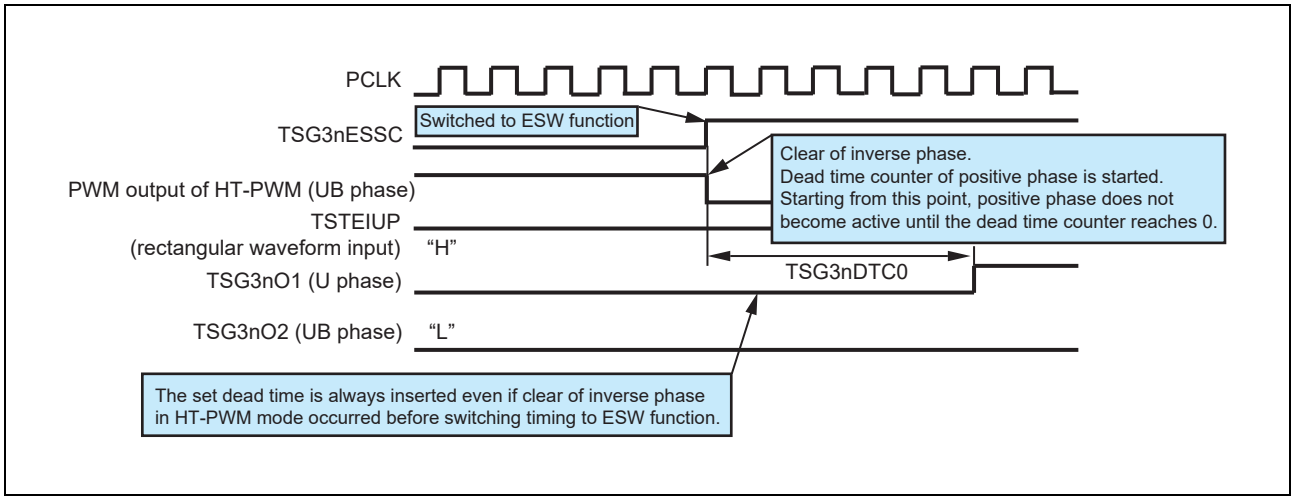


Figure 19.77 Switching Operation to ESW Function (Inverse Phase Cleared Before Switching)

(4) Settings for Operation using ESW Function

Use ESW function with the following settings of bits. Do not modify settings during operation (TSG3nTE = 1).

Table 19.68 List of Settings for Operation using ESW Function

Bit Name	Setting Value	Description
TSG3nCTL0.TSG3nMD2-0	001_B	Switchable only in HT-PWM mode.
TSG3nCTL3.TSG3nRMC	0	Available only in reload mode.
TSG3nIOC3.TSG3nTOL6-1	000000_B	Setting of logical inverse of PWM set/clear is prohibited (HT-PWM mode limitation)
TSG3nOPT0.TSG3nSOC	0	Switching to software control function is prohibited.
TSG3nOPT0.TSG3nSTE	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPOT	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSS	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nIDC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT1.TSG3nSPC2-0	000_B	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nPAT0W	000000_H	Operation setting of 120-DC mode (value after reset).
TSG3nPAT1W	000000_H	Operation setting of 120-DC mode (value after reset).

19.4.7.5 SP-PWM Mode (Shifted-pulse - Pulse Width Modulation Mode)

Overview

In this mode, a 6-phase PWM can be generated using the 18-bit counter and the 18-bit compare registers.

Prerequisites

- The PWM signal cycle is set in TSG3nCMP0E.
- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, while the clear timings of these phases are set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E (when set timing and clear timing are used for control).
- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E while the active periods of these phases are set with TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE. The sum of the set values of TSG3nCMP2E, TSG3nCMP6E, TSG3nCMP10E, and the set values of TSG3nUPWE, TSG3nVPWE, TSG3nWPWE are set to TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E respectively (when set timing and active period are used for control). The value after addition must be no greater than $3FFFF_H$. Truncate values having 19 or more bits.

Functional description

Set the carrier period and the set timings and duty of U phase, V phase, and W phase. The counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter counts up from 00000_H and is cleared by match with TSG3nCMP0E.

The dead time is set with TSG3nDTC0 and TSG3nDTC1. TSG3nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch while TSG3nDTC1 sets that of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG3nDTT1 to TSG3nDTT3) for dead time generation load the set values of TSG3nDTC0 and TSG3nDTC1 at compare match of the 18-bit counter with the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) and start down-counting.

INTTSG3nIm interrupts (m = 1, 2, 5, 6, 9, 10) are generated by the compare match of the 18-bit counter with the TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E buffer registers.

NOTE

SP-PWM mode is enabled when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 010_B.

(1) Basic Timing Chart

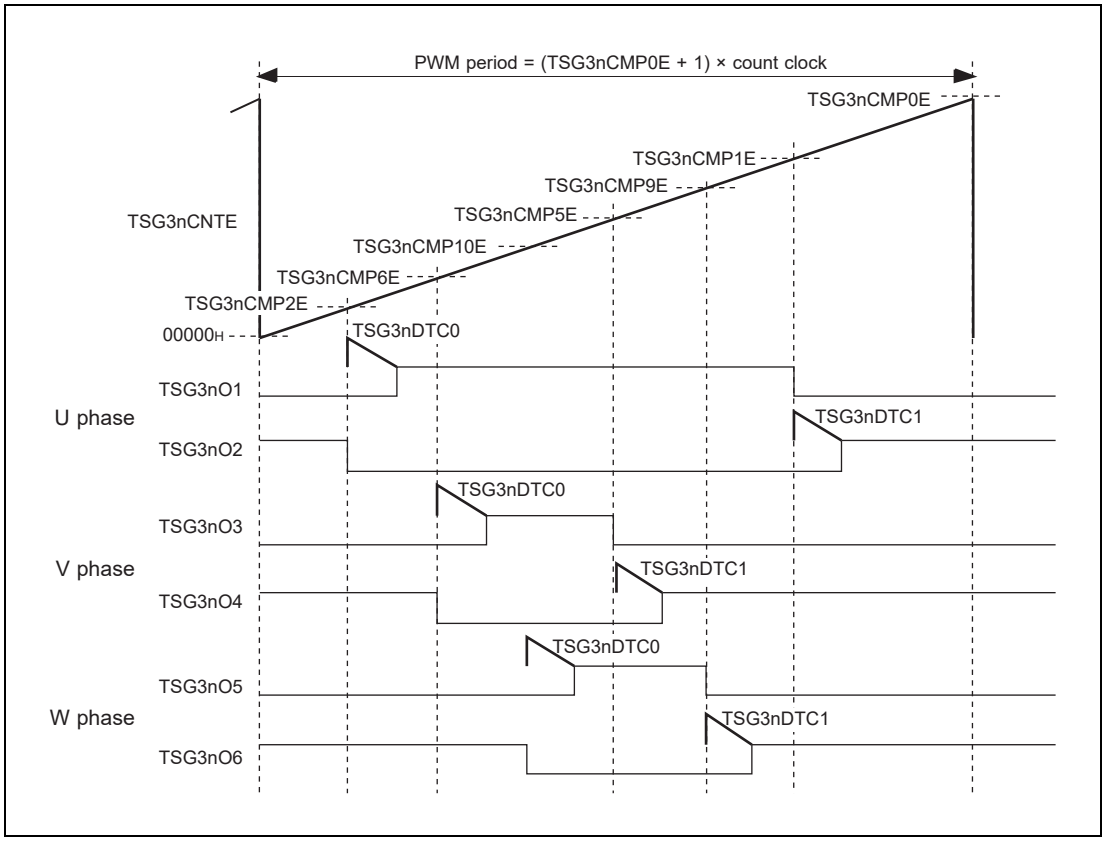


Figure 19.78 Basic Timing in SP-PWM Mode

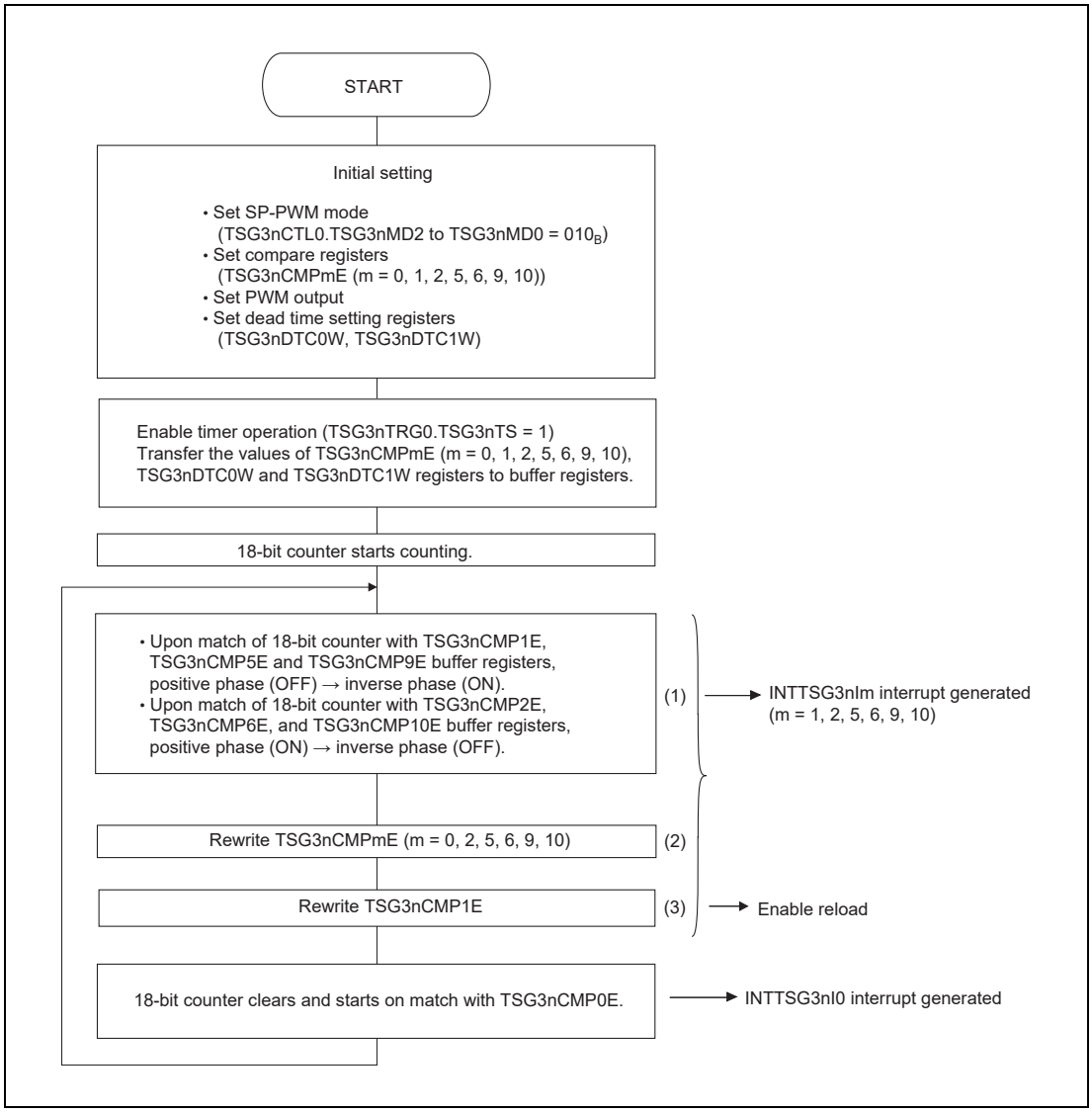


Figure 19.79 Basic Operation Flow in SP-PWM Mode

NOTE

The timing of (1) may be different depending on the rewriting timing of (2) and (3) and the TSG3nCMP1E value. Be sure to rewrite (2) followed by rewriting (3).

(2) List of SP-PWM Mode Operations

Table 19.69 Counter Functions in SP-PWM Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register with 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 19.70 Compare Registers and Dead Time Setting Register Functions in SP-PWM Mode

Register	Rewriting Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload/Anytime rewrite	Possible	Setting period
TSG3nUPWE	Reload/Anytime rewrite	Possible	PWM control for U phase
TSG3nCMP1E, TSG3nCMP2E	Reload/Anytime rewrite		
TSG3nVPWE	Reload/Anytime rewrite	Possible	PWM control for V phase
TSG3nCMP5E, TSG3nCMP6E	Reload/Anytime rewrite		
TSG3nWPWE,	Reload/Anytime rewrite	Possible	PWM control for W phase
TSG3nCMP9E, TSG3nCMP10E	Reload/Anytime rewrite		
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload/Anytime rewrite	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible	Period and dead time

Table 19.71 Output Functions in SP-PWM Mode

Pin	Function
TSG3nO1	PWM output with dead time by compare match of TSG3nCMP1E buffer register (clear timing) or TSG3nCMP2E buffer register (set timing) with 18-bit counter
TSG3nO2	Output inverse phase with respect to TSG3nO1 (with dead time)
TSG3nO3	PWM output with dead time by compare match of TSG3nCMP5E buffer register (clear timing) or TSG3nCMP6E buffer register (set timing) with 18-bit counter
TSG3nO4	Output inverse phase with respect to TSG3nO3 (with dead time)
TSG3nO5	PWM output with dead time by compare match of the TSG3nCMP9E buffer register (clear timing) or the TSG3nCMP10E buffer register (set timing) with the 18-bit counter
TSG3nO6	Output inverse phase with respect to TSG3nO5 (with dead time)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 19.72 Interrupt Requests in SP-PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0, 1, 2, 5, 6, 9, 10)	Compare match of TSG3nCMPmE buffer register with 18-bit counter (m = 0, 1, 2, 5, 6, 9, 10)
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0 interrupt)
INTTSG3nIWN	Warning

Table 19.73 Compare Match Timing in SP-PWM Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10)	After match of 18-bit counter and TSG3nCMPmE is detected (m = 1, 2, 5, 6, 9, 10)

Table 19.74 Example of Setting Each Timer Output Condition in SP-PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nO1, TSG3nO3, TSG3nO5	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG3nCMPmE = TSG3nCMP (m + 1) E or TSG3nCMP (m + 1) E > TSG3nCMP0E (m = 1, 5, 9)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1) E + 1 TSG3nCMP (m + 1) E = TSG3nCMPmE – 1 (m = 1, 5, 9)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m + 1) E - 1 TSG3nCMP (m + 1) E = TSG3nCMPmE + 1 (m = 1, 5, 9)
			Output an active level throughout one period (duty 100%)	TSG3nCMPmE > TSG3nCMP0E TSG3nCMP (m + 1) E ≤ TSG3nCMP0E (m = 1, 5, 9)
TSG3nO2, TSG3nO4, TSG3nO6	PWM output	(TSG3nCMP0E + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG3nCMP (m – 1) E > TSG3nCMP0E (m = 2, 6, 10)
			Output an active level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m – 1) E - 1 TSG3nCMP (m – 1) E = TSG3nCMPmE + 1 (m = 2, 6, 10)
			Output an inactive level of one count clock in one period	TSG3nCMPmE = TSG3nCMP (m – 1) E + 1 TSG3nCMP (m – 1) E = TSG3nCMPmE – 1 (m = 2, 6, 10)
			Output an active level throughout one period (duty 100%)	TSG3nCMPmE = TSG3nCMP(m – 1)E or TSG3nCMPmE > TSG3nCMP0E (m = 2, 6, 10)
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG3nCMP0E + 1) × count clock	Please refer to **Section 19.4.5, A/D Conversion Trigger Function**.	

(3) Various Settings of SP-PWM Mode

Mode setting

SP-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 010_B.

Setting timer output

The output pins TSG3nO1-TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin provides output pulse as diagnostic output or A/D conversion trigger. The pin should be set as necessary.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of the positive and inverse phases is enabled by setting TSG3nIOC1.TSG3nEOC to 1. For details, refer to **Section 19.4.6, Error/Warning Interrupt**.

Setting rewriting timing of register with reload function

With the TSG3nCTL3.TSG3nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG3nCTL4.TSG3nPRE to 1.

No reload timing is generated when TSG3nPRE = 0.

When “anytime rewrite” is specified, the unintended output may be generated depending on the rewrite timing.

Setting A/D conversion trigger output

The A/D conversion trigger 0 (TSG3nADTRG0 signal) is set with TSG3nCTL5.TSG3nAT09 to TSG3nAT00.

TSG3nAT09 to TSG3nAT00 is used to enable or disable the A/D conversion trigger output on timing match of TSG3nDCMP2E to TSG3nDCMP0E with the 18-bit counter (up count).

TSG3nCTL6.TSG3nAT19 to TSG3nAT10 is used to set the A/D conversion trigger 1 (TSG3nADTRG1 signal).

To set the match timing of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to each register.

The skipping function can be used for TSG3nADTRG0 and TSG3nADTRG1 signals. TSG3nACC00 and TSG3nACC01 of TSG3nCTL5, and TSG3nACC10 and TSG3nACC11 of TSG3nCTL6 can be used to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

- **Be sure to set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E to TSG3nDCMP0E correctly when the timing pulse of A/D conversion trigger is output to TSG3nO7.**
- **In SP-PWM mode, no trough interrupt (INTTSG3nIVLY) is generated. Therefore, TSG3nCTL5.TSG3nAT00 and TSG3nCTL6.TSG3nAT10 must be set to 0.**
- **In SP-PWM mode, the 18-bit sub-counter does not operate. Therefore, TSG3nAT09, and TSG3nAT08 in TSG3nCTL5, and TSG3nAT19, and TSG3nAT18 in TSG3nCTL6 must be set to 0.**

- **In SP-PWM mode, down counting by the 18-bit counter is not generated. Therefore, TSG3nAT07, TSG3nAT05, and TSG3nAT03 in TSG3nCTL5 and TSG3nAT17, TSG3nAT15, and TSG3nAT13 in TSG3nCTL6 should be set to 0.**

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

The dead time is calculated by the following expressions:

PCLK × TSG3nDTC0

PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state, and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state, respectively.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

TSG3nCMP0E = (carrier period/count clock period) – 1

CAUTION

PWM output with 100% duty cannot be produced when TSG3nCMP0E = $3FFFF_H$.

Setting duty (PWM width)

The duty of U phase, V phase, and W phase is set with TSG3nCMPmE, TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE (m = 1, 2, 5, 6, 9, 10), respectively.

- The set timings of the U phase, V phase, and W phase are set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, and the clear timings are set with TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E. (The set and clear timing setting is used for control.)
- The set timings of U phase, V phase, and W phase is set with TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E while the active periods of these phases are set with TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE.
  The sum of the set values of TSG3nCMP2E, TSG3nCMP6E, and TSG3nCMP10E, and the set values of TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE are set to TSG3nCMP1E, TSG3nCMP5E, and TSG3nCMP9E respectively (when set timing and active period are used for control).

(4) Dead Time Control in SP-PWM mode

Duty setting registers are TSG3nCMPmE (m = 1, 2, 5, 6, 9, 10), TSG3nUPWE, TSG3nVPWE, and TSG3nWPWE and register for setting the period is TSG3nCMP0E. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are six 10-bit down counters that operate synchronously with the count clock of the 18-bit counter and two dead time setting registers (TSG3nDTC0W and TSG3nDTC1W). TSG3nDTC0W is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG3nDTC1W is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state.

Dead time counter keeps operating even when operation stop (TSG3nTE = 0) setting collided with dead time insert timing, and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

The following figure shows an example of the output waveform.

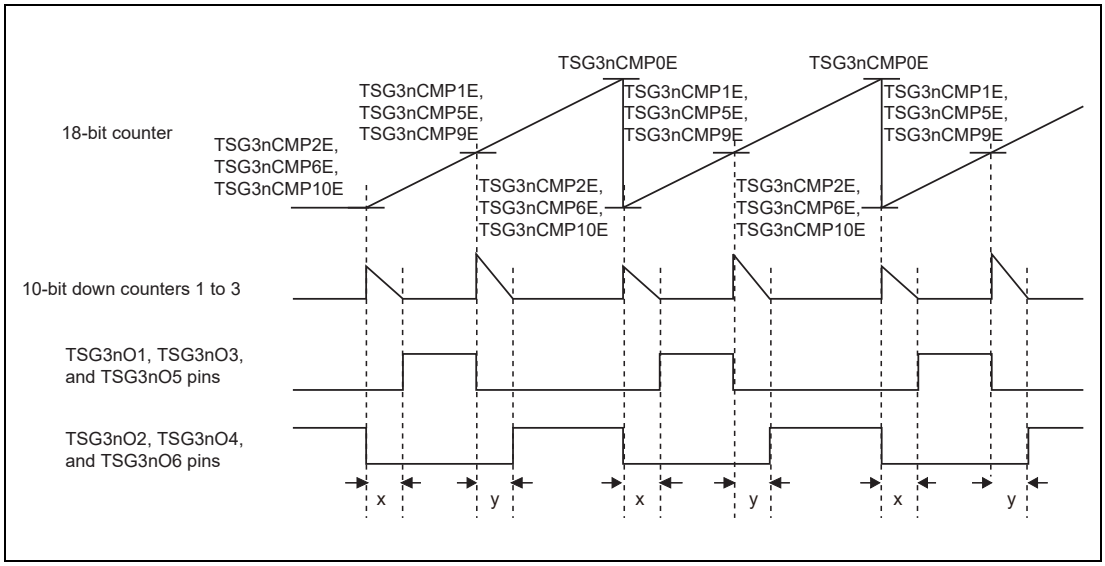


Figure 19.80 Example of Output Waveform in SP-PWM Mode

NOTE

x: TSG3nDTC0 register values, y: TSG3nDTC1 register values

(5) Software Output Control Function in SP-PWM Mode

TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are used to control timer output by software.

As shown in **Figure 19.81**, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG3nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to SP-PWM mode output control.

For details, refer to **Section 19.4.7.10, Software Output Control Function**.

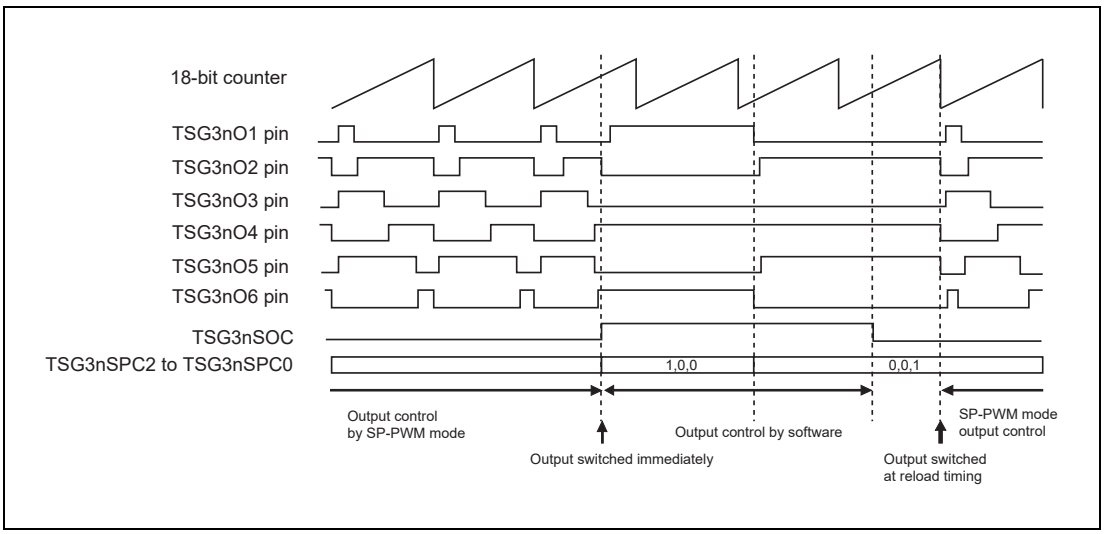


Figure 19.81 Example of Output Control Switching from SP-PWM Mode Control to Software Control

(a) Procedure on Software Output Control Processing

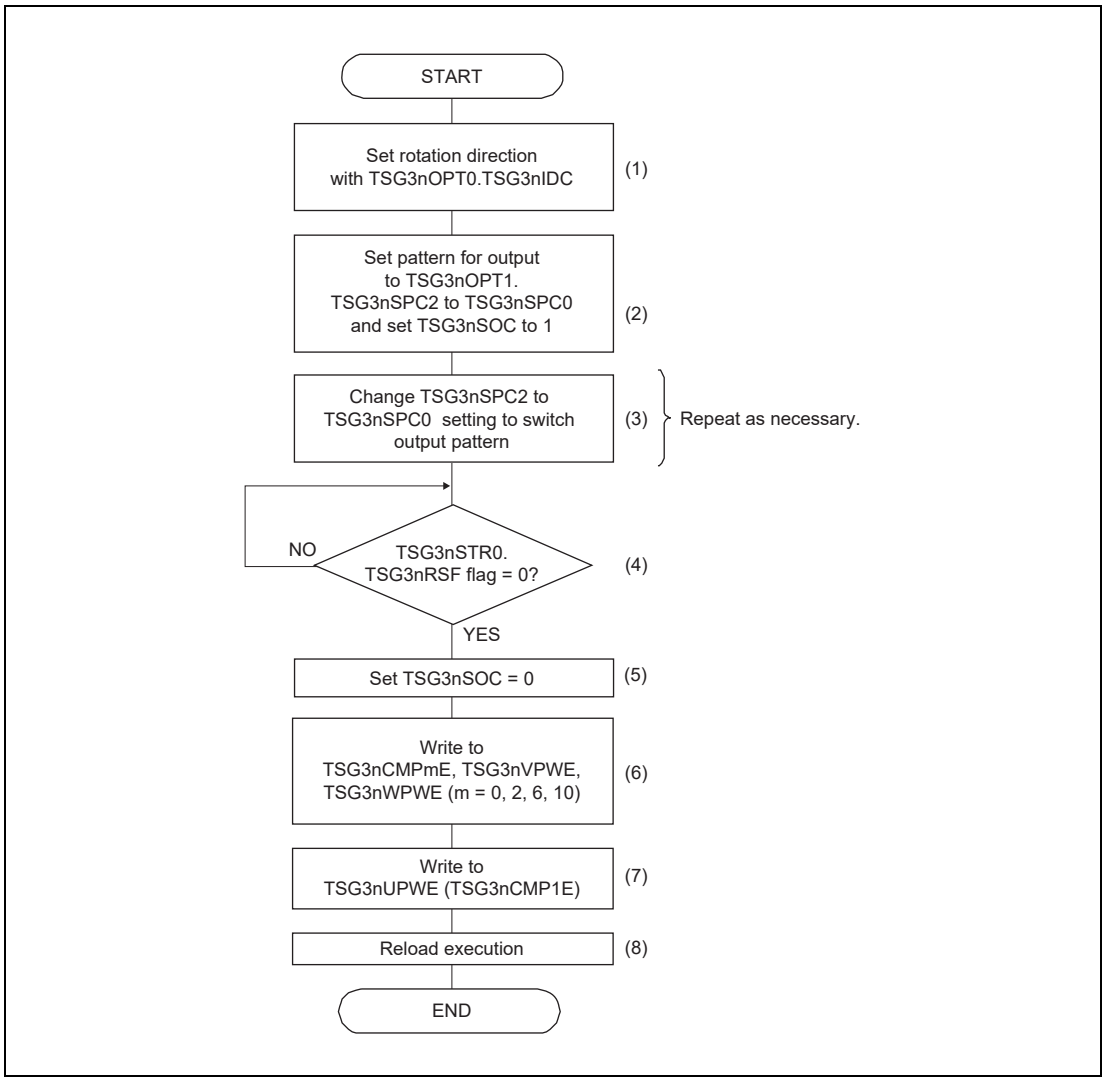


Figure 19.82 Flow of Software Output Control

The procedure for software output control is described below.

(1) Set TSG3nIDC to determine the electric current direction. The timer output has a 180-degree phase shift between when TSG3nIDC = 0 and when TSG3nIDC = 1. When this bit is rewritten with the software output control function, the output pattern will change according to the new setting at the next timer cycle.

(2) Set the pattern for output to TSG3nSPC2-0. At the same time, set TSG3nSOC to 1 to switch to software output control mode.

(3) Change the output pattern setting for TSG3nSPC2-0 to change the timer output.

(4) Ensure that the reload request flag TSG3nRSF is 0. If TSG3nRSF is 1, do not shift to the following procedure until it goes 0.

(5) By clearing TSG3nSOC to 0, software control starts to be released (not yet released here).

(6) After software output control is released, set the compare registers if necessary. Move to the following procedure if no setting is required. In addition, change the registers with the reload function if necessary.

(7) Write TSG3nUPWE (TSG3nCMP1E) to start reloading.

(8) Reload is executed and software output control is released.

CAUTION

Be sure to execute reload after execution of steps (4), (5), (6), and (7). Unless reload can be executed, software output control cannot be released

19.4.7.6 120-DC Mode

Overview

In this mode, PWM output period set to TSG3nCMP0E and timer output (TSG3nO1 to TSG3nO6) according to the duty cycle set to TSG3nCMP1E to TSG3nCMP12E are controlled with three types of pattern inputs (software output control method, pattern switch method, and trigger switch method) to perform 120-DC control.

Prerequisites

- Set the PWM period to TSG3nCMP0E.
- Set the PWM duty to TSG3nCMP1E to TSG3nCMP12E and set the output pattern to TSG3nPAT0W and TSG3nPAT1W.

Functional description

Set the PWM period, set the duty cycle to individual compare register, and set the pattern to be output to the pattern register. Setting TSG3nTRG0.TSG3nTS to 1 starts counting.

The 18-bit counter starts counting from 00000_H, and is cleared by the match with TSG3nCMP0E.

INTTSG3nI1 to INTTSG3nI12 interrupts are generated by a compare match of the 18-bit counter and TSG3nCMP1E to TSG3nCMP12E buffer registers, respectively.

NOTE

120-DC mode is valid when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 011_B.

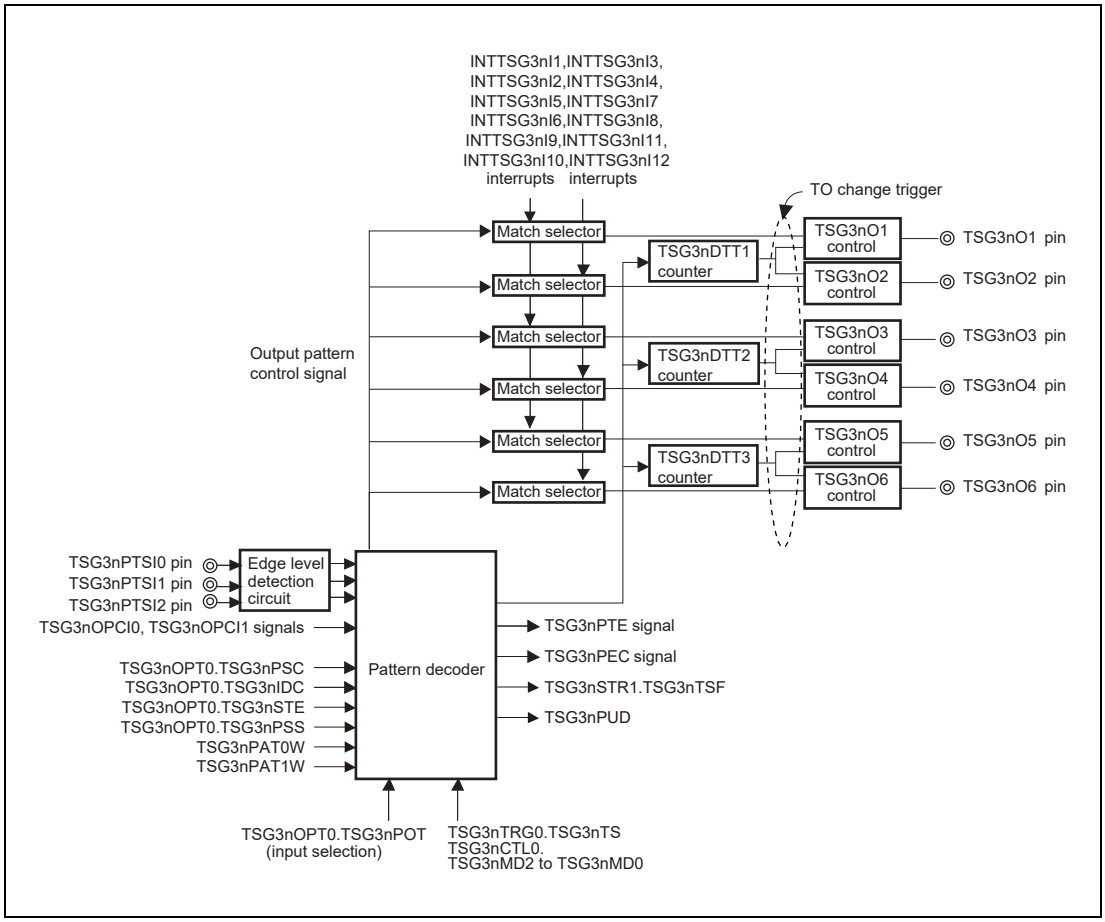


Figure 19.83 Block Diagram in 120-DC Mode

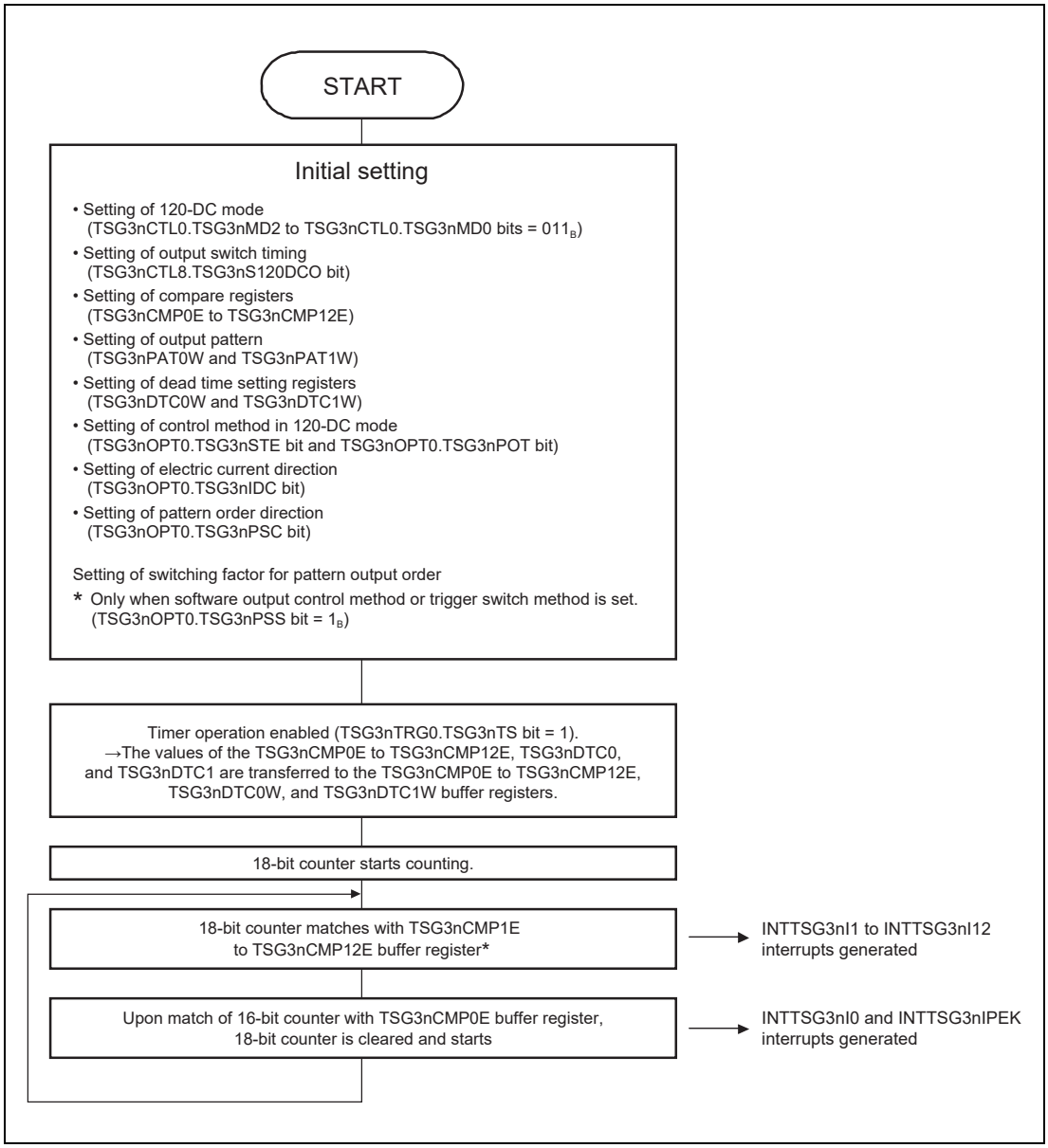


Figure 19.84 Basic Operation Flow in 120-DC Mode

NOTE

The 18-bit counter is not cleared by match of the 18-bit counter with the TSG3nCMP1E to TSG3nCMP12E buffer registers

(1) List of Operations in 120-DC Mode

Table 19.75 Counter Functions in 120-DC Mode

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	When TSG3nCTL8.TSG3nS120DCO is 0: timing of a match of TSG3nCMP0E value and 18-bit counter value or output pattern switch When TSG3nCTL8.TSG3nS120DCO is 1: timing of a match of TSG3nCMP0E value with 18-bit counter value
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 19.76 Functions of Compare Registers and Dead Time Setting Registers in 120-DC Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload	Possible	Setting period
TSG3nCMPmE (m = 1-12)	Reload	Possible	Setting PWM duty
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload	Possible	Outputting diagnostic signal or setting A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Possible	Setting dead time

Table 19.77 Timer Input Function in 120-DC Mode

Pin/Signal	Function
TSG3nPTSI2-TSG3nPTSI0 pins	Pattern input (3 phases)
TSG3nOPCI0, TSG3nOPCI1 signals	Trigger input

Table 19.78 Timer Output Function in 120-DC Mode

Pin/Signal	Function
TSG3nO1 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO2 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO3 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO4 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO5 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 1, 2, 5, 6, 9, 10) with the 18-bit counter and output pattern selected by TSG3nPAT0W setting.
TSG3nO6 pin	PWM output (with dead time) according to a compare match of the TSG3nCMPmE buffer register (m = 3, 4, 7, 8, 11, 12) with the 18-bit counter and output pattern selected by TSG3nPAT1W setting.
TSG3nO7 pin	Diagnostic signal output or pulse output by A/D conversion trigger
TSG3nPTE signal	Toggle signal by change in input pattern

Table 19.79 Interrupt Requests in 120-DC Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of TSG3nCMPmE buffer register and 18-bit counter (m = 0 to 12)
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Warning interrupt

Table 19.80 Compare Match Timing in 120-DC Mode

Compare Match	Timing
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After detecting the match of 18-bit counter and TSG3nCMPmE (m = 1 to 12)

Table 19.81 Example of Setting Each Timer Output Condition in 120-DC Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSG3nOm (m = 1 to 6)	PWM output	(TSG3nCMP0E + 1) × count clock	See **(6), List of Output Patterns in 120-DC Mode**.	—
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG3nCMP0E + 1) × count clock	See **Section 19.4.5, A/D Conversion Trigger Function**.	—

(2) Various Settings of 120-DC Mode

Mode setting

120-DC mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 011_B.

Setting timer output

The output pins TSG3nO1 to TSG3nO6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

With TSG3nIOC1.TSG3nEOC = 1, the error interrupt (INTTSG3nIER) generation is enabled when the simultaneous active state of the positive phase and inverse phase is detected. For details, see **Section 19.4.6, Error/Warning Interrupt**.

Setting register rewrite timing

Reloading the registers with the reload function is activated with TSG3nCTL3.TSG3nRMC (simultaneous rewrite; default setting is 0 = reload). Set TSG3nCTL4.TSG3nPRE to 1 when reload is used.

The reload timing is not generated if TSG3nPRE is 0.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSG3nADTRG0 signal), use TSG3nCTL5.TSG3nAT09 to TSG3nAT00.

With TSG3nAT09 to TSG3nAT00, A/D conversion trigger output is enabled or disabled at the match of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E (during up count).

To set A/D conversion trigger 1 (TSG3nADTRG1 signal), use TSG3nCTL6.TSG3nAT19 to TSG3nAT10.

To set the match timing of the 18-bit counter and TSG3nDCMP2E to TSG3nDCMP0E, set the compare value to the pertinent register.

The skipping function can be used for TSG3nADTRG0 and TSG3nADTRG1 signals. Use TSG3nACC01, TSG3nACC00 of TSG3nCTL5 and TSG3nACC11, and TSG3nACC10 in TSG3nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

- **Set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E to TSG3nDCMP0E correctly when using the TSG3nO7 output for the A/D conversion trigger timing pulse.**
- **In 120-DC mode, a trough interrupt (INTTSG3nIVLY) is not generated. Therefore, set TSG3nAT00 and TSG3nAT10 in TSG3nCTL5 and TSG3nCTL6 to 0.**
- **In 120-DC mode, the 18-bit sub-counter does not operate. Therefore, set TSG3nAT09, TSG3nAT08, TSG3nAT19, and TSG3nAT18 in TSG3nCTL5 and TSG3nCTL6 to 0.**
- **In 120-DC mode, the 18-bit counter does not decrement. Therefore, set TSG3nAT07, TSG3nAT05, TSG3nAT03, TSG3nAT17, TSG3nAT15, and TSG3nAT13 in TSG3nCTL5 and TSG3nCTL6 to 0.**

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

The dead time is calculated by the following expressions

PCLK × TSG3nDTC0

PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state, respectively.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state to a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state.

Carrier period

Set the carrier period with TSG3nCMP0E according to the following expression:

TSG3nCMP0E = (carrier period/count clock period) – 1

Duty (PWM width) setting

The duty of PWM output is set with TSG3nCMP1E to TSG3nCMP12E. The setting range of the compare registers is as follows:

$00000_H \leq$ TSG3nCMPmE $\leq$ TSG3nCMP0E + 1

CAUTION

Do not set TSG3nCMPmE = TSG3nCMP0E + 1 (m = 1 to 12) only when TSG3nCMP0E + 1 < TSG3nCMPmE, and TSG3nCMP0E = $3FFFF_H$.

Output PWM setting

In 120-DC mode, the output pins TSG3nO1, TSG3nO3, and TSG3nO5 are controlled by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E, and the output pins TSG3nO2, TSG3nO4, TSG3nO6 are controlled by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E. The duty cycle of a PWM period (TSG3nCMP0E) can be set with TSG3nCMP1E to TSG3nCMP12E. Setting TSG3nCMP1E to TSG3nCMP12E to 00000_H sets the PWM duty cycle to 0%. Setting TSG3nCMP1E to TSG3nCMP12E to TSG3nCMP0E + 1 value sets the PWM duty cycle to 100%. This allows chopping output control and rectangular wave output control.

(3) Control Methods in 120-DC Mode

Control methods in 120-DC mode are listed below.

Table 19.82 Control Method in 120-DC Mode

Control Method	Function
Software output control method	Switches the output pattern according to the TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 setting made by software.
Pattern switch method	Switches the output pattern by the pattern input signal of TSG3nPTSI0 to TSG3nPTSI2.
Trigger switch method	Switches the output pattern by the trigger switch method using the trigger input signals TSG3nOPCI0 and TSG3nOPCI1 or by the pattern input setting of TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 in the constant order.

Switch timing of timer output when changing input pattern of 120-DC mode can be set by TSG3nCTL8.TSG3nS120DCO.

Table 19.83 Setting of TSG3nS120DCO and Operation in 120-DC Mode

TSG3nS120DCO	Function
0	When input patterns are changed, the main counter (TSG3nCNTE) is cleared and the change of patterns is immediately reflected to timer output.
1	When input patterns are changed, the change of input patterns is reflected to timer output from the next timer cycle (after a match of the main counter (TSG3nCNTE) with TSG3nCMP0E).

Setting software output control method

Setting TSG3nOPT0.TSG3nSTE = 0 switches the output pattern by software output control. The TSG3nO1 to TSG3nO6 pin output is switched according to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0.

The pattern output order at the beginning of operation is set with TSG3nOPT0.TSG3nIDC and TSG3nOPT0.TSG3nPSC.

Operation of software output control method

The PWM output of TSG3nO1 to TSG3nO6 pins (PWM output defined by TSG3nCMP1E to TSG3nCMP12E) is selected by TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 by software. To control the dead time, the dead time counter is activated at the falling edge of the signals in each phase and the dead time is inserted.

The 18-bit counter counts based on the carrier period set in TSG3nCMP0E. The 18-bit counter is cleared by match of the 18-bit counter and TSG3nCMP0E or by a write access to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 (when TSG3nS120DCO = 0).

In this method, the pattern is output, which is decoded using information on the output pattern (TSG3nSPC2 to TSG3nSPC0), the electric current direction control bit (TSG3nOPT0.TSG3nIDC), and order detection control bit (TSG3nOPT0.TSG3nPSC). **Figure 19.105** shows the timer output when the output pattern is changed by software output control.

Immediately after the operation starts (TSG3nTRG0.TSG3nTS = 1), the output pattern of TSG3nSPC2 to TSG3nSPC0 and the pattern set with TSG3nIDC and TSG3nPSC (TSG3nOPT0.TSG3nPSS = 1) are output.

Setting pattern switch method

Setting TSG3nOPT0.TSG3nSTE to 1 and TSG3nPOT to 0 selects the pattern switch method. The TSG3nO1 to TSG3nO6 pin output pattern is changed at the change timing of the TSG3nPTSI2 to TSG3nPTSI0 pins.

The output pattern at the beginning of operation is set with TSG3nOPT0.TSG3nIDC and with TSG3nOPT0.TSG3nPSC. However, after determining the rotation direction (after the value is set to TSG3nSTR1.TSG3nTSF), the setting of TSG3nPSC is disabled.

Operation of pattern switch method

After level detection is performed for the pins (three inputs from the hall sensor), the level-detected signals are decoded. From the decoding result, the PWM output of TSG3nO1 to TSG3nO6 pins (PWM output defined by TSG3nCMP1E to TSG3nCMP12E) is selected. To control the dead time, the dead time counter is activated at the falling timing of signals in each phase and the dead time is inserted.

The 18-bit counter is cleared by a match of the 18-bit counter value with the TSG3nCMP0E value or by a change of the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) while TSG3nS120DCO is 0.

In this method, the pattern, which is decoded by using information on input pattern (TSG3nPTSI2 to TSG3nPTSI0), the electric current direction control bit (TSG3nOPT0.TSG3nIDC), and TSG3nPTSI2 to TSG3nPTSI0 pattern order detection flag (TSG3nSTR1.TSG3nTSF), is output. **Figure 19.86** to **Figure 19.89** show the timer output when TSG3nPTSI2 to TSG3nPTSI0 pin inputs change.

Immediately after the operation starts (TSG3nTRG0.TSG3nTS = 1), the output pattern set by the levels input on the TSG3nPTSI2 to TSG3nPTSI0 pins and by the TSG3nIDC and TSG3nPSC bits is produced. If a level on any of the TSG3nPTSI2 to TSG3nPTSI0 pins is changed, TSG3nTSF is determined by the direction of the change to the sequence. After the TSG3nTSF value is determined, the pattern set by the TSG3nTSF bit replaces that set by the TSG3nPSC bit.

CAUTION

When connecting the three-phase pulse input signal to the TSG3nPTSI2 to TSG3nPTSI0 pins, confirm that the three-phase pulse input value and the patterns output from the TSG3nO1 to TSG3nO6 pins satisfy the expected conditions.
If the expected conditions are not satisfied, change the connection between the three-phase pulse input signal and the TSG3nPTSI2 to TSG3nPTSI0 pins.

Setting trigger switch method

Setting TSG3nOPT0.TSG3nSTE and TSG3nPOT to 1 selects the trigger switch method. The output patterns of the pins TSG3nO1 to TSG3nO6 are changed at a rising edge of an external input (TSG3nOPCI1 and TSG3nOPCI0 signals).

For pattern output order, see **Section 19.4.7.6**, **(5) Operation in 120-DC Mode**.

The initial output pattern can be controlled with TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0.

When starting the TSG3n operation (TSG3nTRG0.TSG3nTS = 1) after setting TSG3nSPC2 to TSG3nSPC0, the initial pattern is output. For details, see **Section 19.4.7.6**, **(6) List of Output Patterns in 120-DC Mode**.

Operation of trigger switch method

With the trigger input switch method, the rising edges of the TSG3nOPCI0 and TSG3nOPCI1 signals are detected and the output switch timing is generated. The initial timer output pattern is set with TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0. The subsequent output patterns are switched after rising of the TSG3nOPCI0 and TSG3nOPCI1 signals is detected. Furthermore, the output patterns can be switched by setting TSG3nOPT1.TSG3nSPC2 to TSG3nOPT1.TSG3nSPC0.

The 18-bit counter counts based on the carrier period set in TSG3nCMP0E. The 18-bit counter is cleared by match of the 18-bit counter and TSG3nCMP0E, by a write access to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0, or by detecting a rising of TSG3nOPCI0 and TSG3nOPCI1 signals. (When TSG3nS120DCO = 0)

For examples of operation in 120-DC mode when trigger input switch method is used, see **Figure 19.87** to **Figure 19.90**.

CAUTION

The initial pattern should be set according to the read input level of the port to which TSG3nPTSI2 to TSG3nPTSI0 pins are connected.

(4) Timer Output in 120-DC Mode

In 120-DC mode, the PWM output is controlled with TSG3nPAT0W, TSG3nPAT1W, and TSG3nCMP1E to TSG3nCMP12E. TSG3nPAT0W, TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, and TSG3nCMP10E are set to control the output of TSG3nO1, TSG3nO3, and TSG3nO5 pins. TSG3nPAT1W, TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, and TSG3nCMP12E are set with the output of SG2nO2, TSG3nO4, and TSG3nO6 pins.

With PWM output control, eight types of output patterns can be selected for each of TSG3nO1, TSG3nO3, and TSG3nO5 pins and TSG3nO2, TSG3nO4, and TSG3nO6 pins.

Table 19.84 TSG3nPAT0W Set Value and Output Control

PATmT Value	Output Control
000	Fixed to low
001	PWM output set with TSG3nCMP1E
010	PWM output set with TSG3nCMP2E
011	PWM output set with TSG3nCMP5E
100	PWM output set with TSG3nCMP6E
101	PWM output set with TSG3nCMP9E
110	PWM output set with TSG3nCMP10E
111	Fixed to high

Note: m = 0, 1, 2, 3, 4, 5

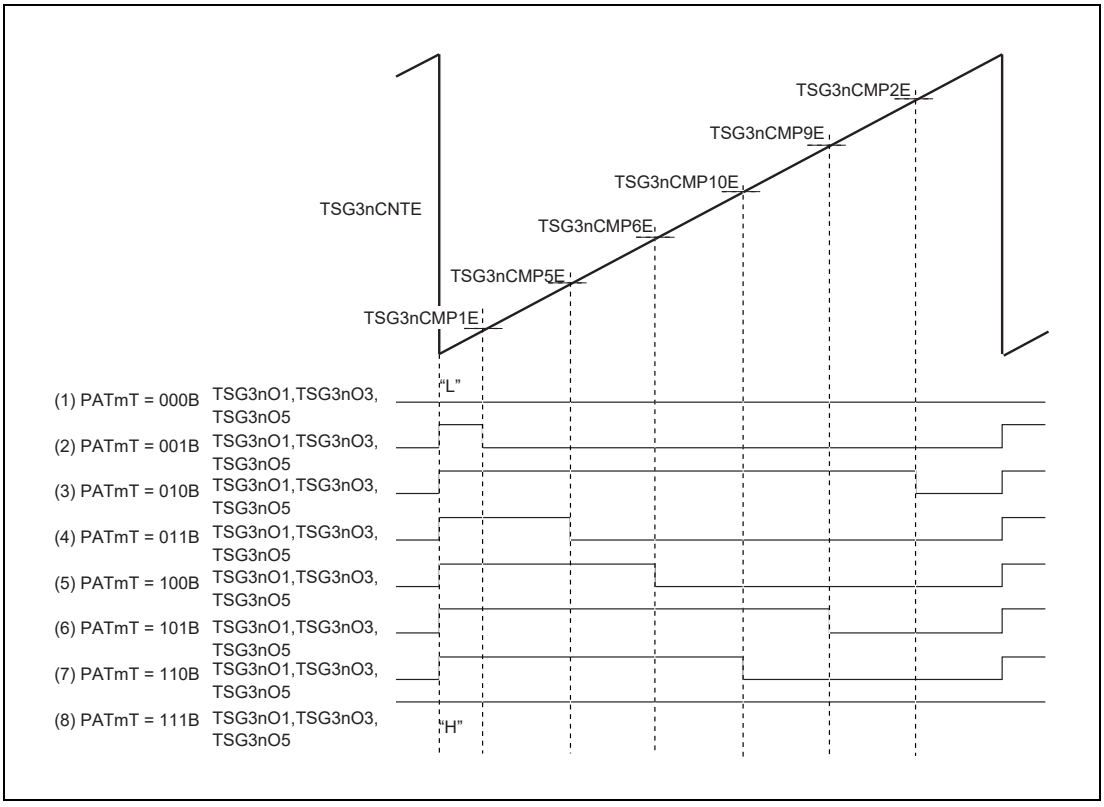


Figure 19.85 TSG3nO1, TSG3nO3, TSG3nO5 Pin Output of Each Output Pattern

Table 19.85 TSG3nPAT1W Set Value and Output Control

PATmB Value	Output Control
000	Fixed to low
001	PWM output set with TSG3nCMP3E
010	PWM output set with TSG3nCMP4E
011	PWM output set with TSG3nCMP7E
100	PWM output set with TSG3nCMP8E
101	PWM output set with TSG3nCMP11E
110	PWM output set with TSG3nCMP12E
111	Fixed to high

Note: m = 0, 1, 2, 3, 4, 5

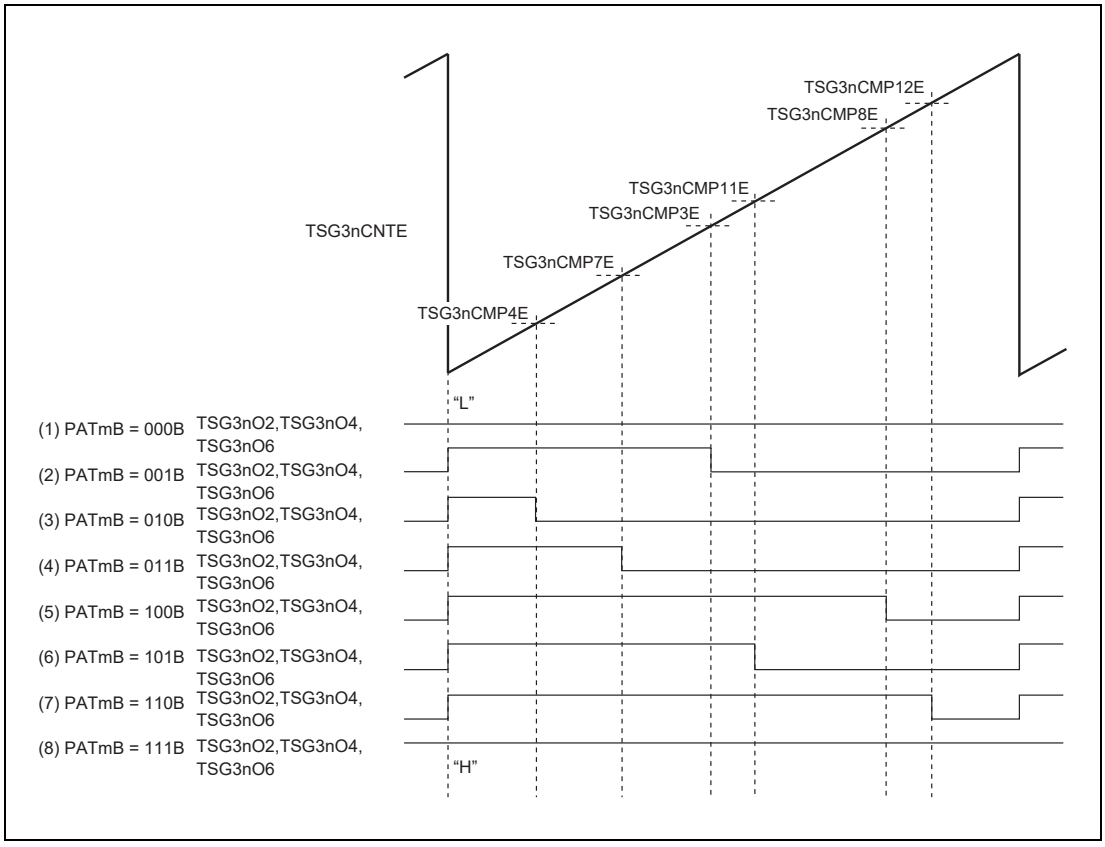


Figure 19.86 TSG3nO2, TSG3nO4, TSG3nO6 Pin Output of Each Output Pattern

(5) Operation in 120-DC Mode

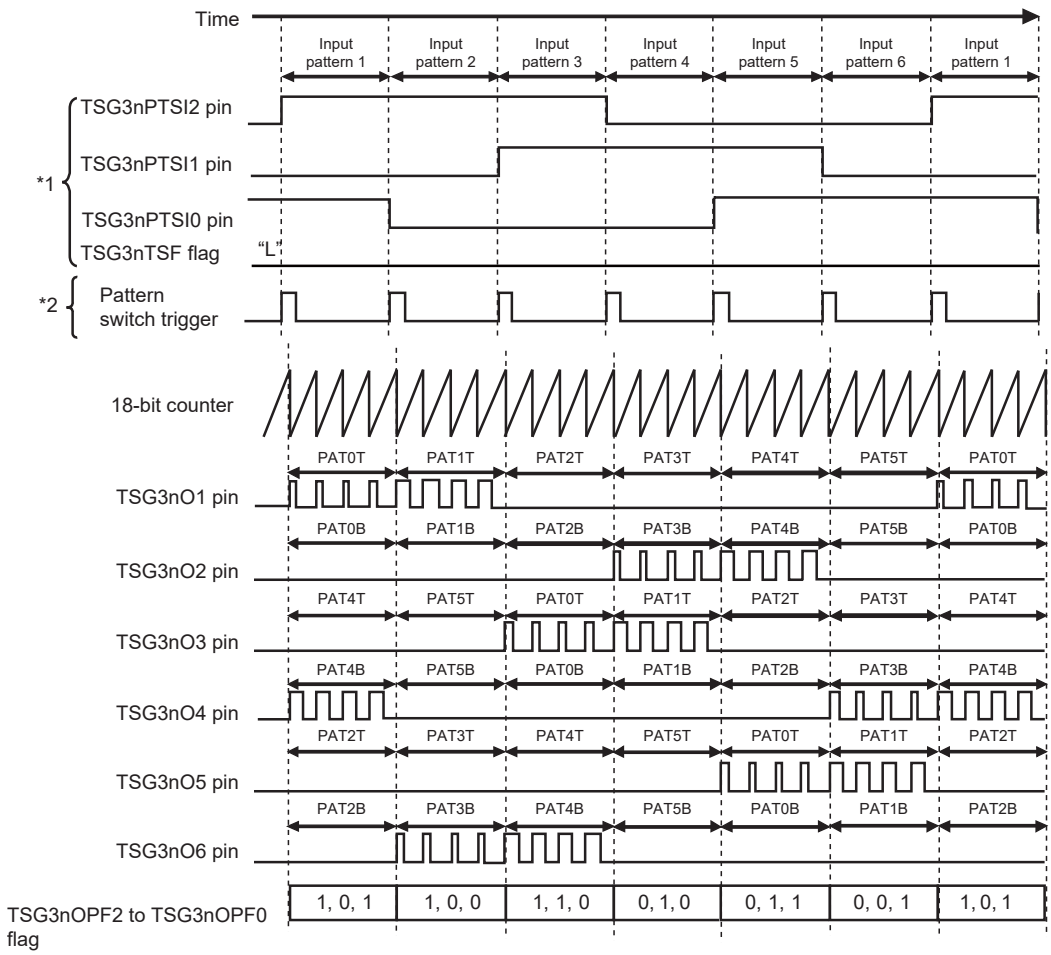
Figure 19.87 to **Figure 19.90** show examples of operation in 120-DC mode.

The TSG3nO1 to TSG3nO6 pins detect the input level change of the TSG3nPTSI2 to TSG3nPTSI0 pins, and then change the output pattern. The 18-bit counter produces sawtooth waveform and TSG3nCMP0E to TSG3nCMP12E output PWM signal. When TSG3nS120DCO = 0, the 18-bit counter is cleared to 00000_H each time the counter value matches with TSG3nCMP0E or a change in the TSG3nPTSI2 to TSG3nPTSI0 pins is detected. The timer output pattern is switched each time a change in the TSG3nPTSI2 to TSG3nPTSI0 pins is detected.

When TSG3nS120DCO = 1, the 18-bit counter is cleared to 00000_H when the counter value matches with TSG3nCMP0E but not cleared with a change in the TSG3nPTSI2 to TSG3nPTSI0 pins. The timer output pattern is switched to the one corresponding to the patterns of TSG3nPTSI2 to TSG3nPTSI0 pins at the next match timing of TSG3nCNTE and TSG3nCMP0E.

NOTE

PAT0T to PAT5T and PAT0B to PAT5B show PWM operation set by TSG3nCMP1E to TSG3nCMP12E, respectively.

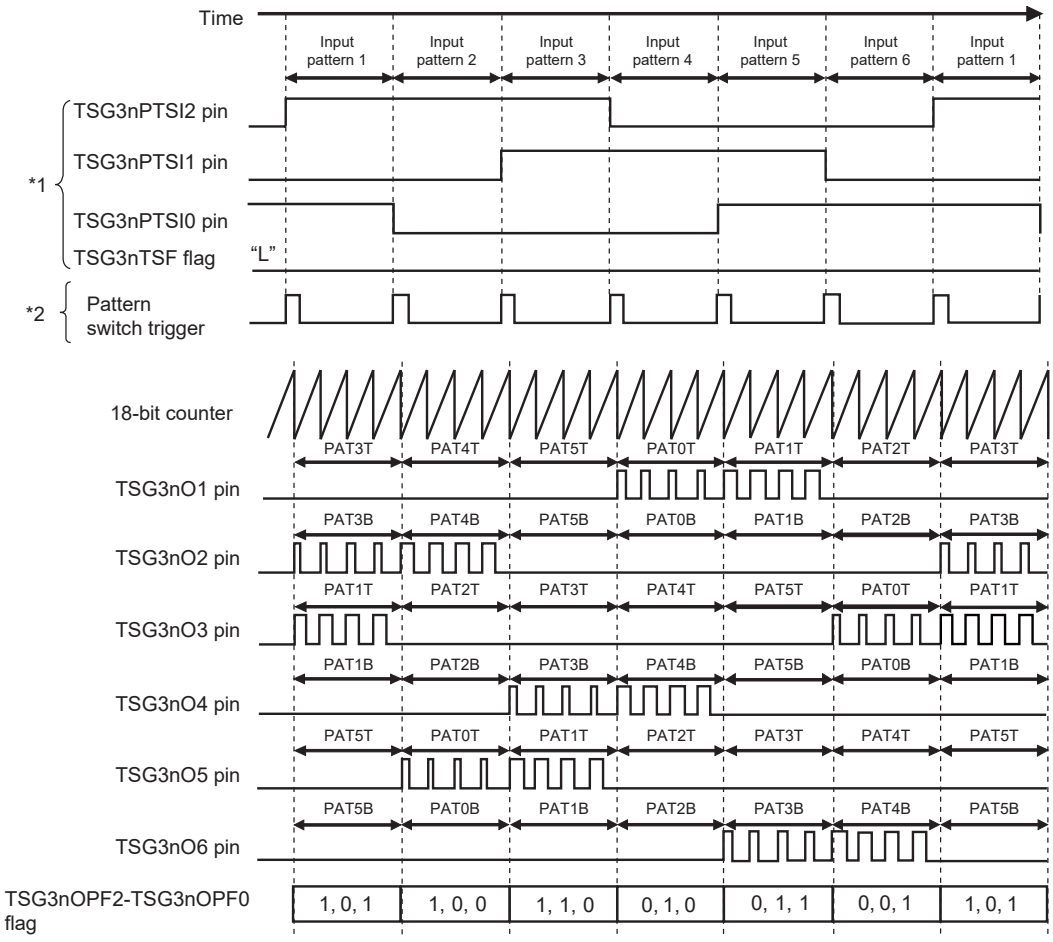


Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)
Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Figure 19.87 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 0)

NOTE

TSG3nOPT0.TSG3nSOC = 0

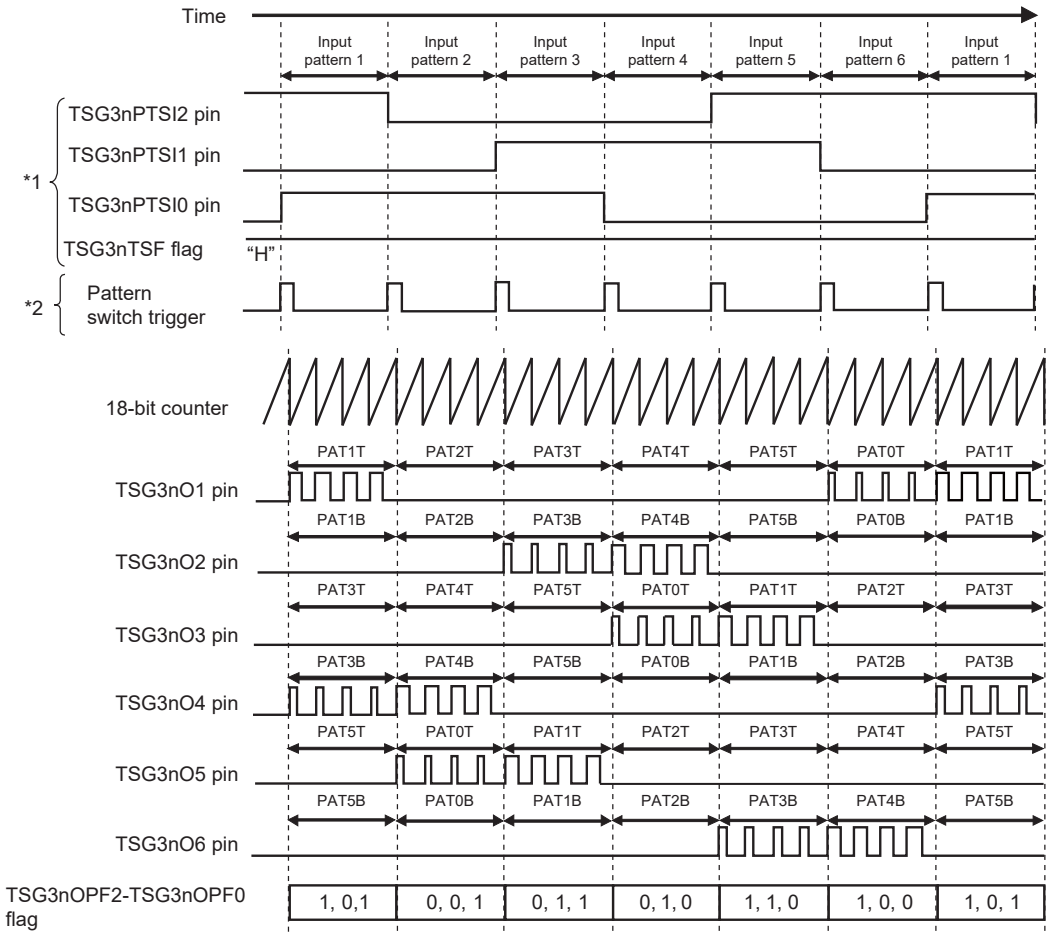


Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)
Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Figure 19.88 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 1)

NOTE

TSG3nOPT0.TSG3nSOC = 0



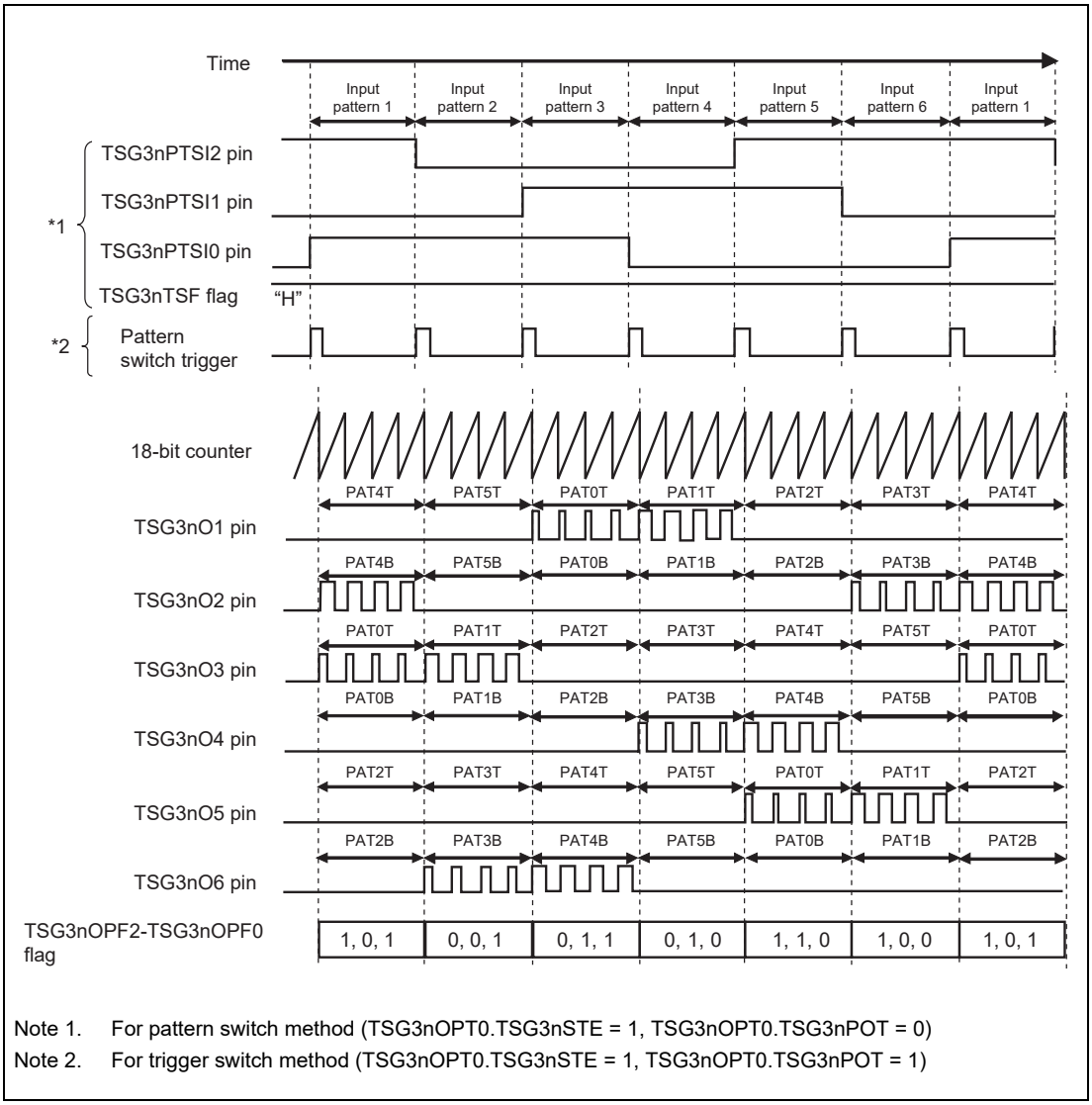
Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Figure 19.89 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 0)

NOTE

TSG3nOPT0.TSG3nSOC = 0



Note 1. For pattern switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0)

Note 2. For trigger switch method (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1)

Figure 19.90 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 1)

NOTE

TSG3nOPT0.TSG3nSOC = 0

(6) List of Output Patterns in 120-DC Mode

In 120-DC mode, the output pattern is determined according to the electric current direction (TSG3nOPT0.TSG3nIDC) and the pattern order direction.

Table 19.86 Selection of Pattern Order Direction in 120-DC Mode

TSG3nOPT0			Pattern Order Direction
TSG3nSTE	TSG3nPOT	TSG3nPSS	
0	—	1	TSG3nPSC
1	0	—	TSG3nTSF (TSGnPSC only for initial setting at operation start)
1	1	1	TSG3nPSC

Table 19.87 List of Output Patterns in 120-DC Mode (1/4)

Electric current direction: normal (TSG3nIDC = 0)
Pattern order direction: normal (TSG3nTSF = 0 or TSG3nPSC = 0)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	Low	Low
TSG3nO2	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	Low	Low
TSG3nO3	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	Low	Low
TSG3nO4	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	Low	Low
TSG3nO5	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	Low	Low
TSG3nO6	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	Low	Low

Table 19.87 List of Output Patterns in 120-DC Mode (2/4)

Electric current direction: reverse (TSG3nIDC = 1)
Pattern order direction: normal (TSG3nTSF = 0 or TSG3nPSC = 0)

	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
Output pins	101	100	110	010	011	001	000	111
TSG3nO1	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	Low	Low
TSG3nO2	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	Low	Low
TSG3nO3	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	Low	Low
TSG3nO4	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	Low	Low
TSG3nO5	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	Low	Low
TSG3nO6	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	Low	Low

Note 1. When TSG3nSPC2 to TSG3nSPC0 are written to while the values of TSG3nSPC2 to TSG3nSPC0, TSG3nSTE, and TSG3nPOT are 1, the output pattern changes. Thereafter, when a pattern switch trigger is generated on rising of the TSG3nOPCI0 and TSG3nOPCI1 signals, the output is switched according to the order of pattern switching. When TSG3nS120DCO = 0, the output is switched immediately. When TSG3nS120DCO = 1, the output is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle). TSG3nSPC2 to TSG3nSPC0 remain unchanged even if the output pattern is switched.

NOTES

1. PAT0T to PAT5T: PWM output set by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E
2. PAT0B to PAT5B: PWM output set by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E

Table 19.87 List of Output Patterns in 120-DC Mode (3/4)

Electric current direction: normal (TSG3nIDC = 0)
Pattern order direction: reverse (TSG3nTSF = 1 or TSG3nPSC = 1)

Output pins	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
	101	100	110	010	011	001	000	111
TSG3nO1	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	Low	Low
TSG3nO2	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	Low	Low
TSG3nO3	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	Low	Low
TSG3nO4	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	Low	Low
TSG3nO5	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	Low	Low
TSG3nO6	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	Low	Low

Table 19.87 List of Output Patterns in 120-DC Mode (4/4)

Electric current direction: reverse (TSG3nIDC = 1)
Pattern order direction: reverse (TSG3nTSF = 1 or TSG3nPSC = 1)

Output pins	TSG3nOPT1.TSG3nSPC2-TSG3nSPC0*1/TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
	101	100	110	010	011	001	000	111
TSG3nO1	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	Low	Low
TSG3nO2	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	Low	Low
TSG3nO3	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	Low	Low
TSG3nO4	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	Low	Low
TSG3nO5	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	Low	Low
TSG3nO6	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	Low	Low

Note 1. When TSG3nSPC2 to TSG3nSPC0 are written to while the values of TSG3nSPC2 to TSG3nSPC0, TSG3nSTE, and TSG3nPOT are 1, the output pattern changes. Thereafter, when a pattern switch trigger is generated on rising of the TSG3nOPCI0 and TSG3nOPCI1 signals, the output is switched according to the order of pattern switching. When TSG3nS120DCO = 0, the output is switched immediately.
When TSG3nS120DCO = 1, the output is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle). TSG3nSPC2 to TSG3nSPC0 remain unchanged even if the output pattern is switched.

NOTES

1. PAT0T to PAT5T: PWM output set by TSG3nCMP1E, TSG3nCMP2E, TSG3nCMP5E, TSG3nCMP6E, TSG3nCMP9E, TSG3nCMP10E
2. PAT0B to PAT5B: PWM output set by TSG3nCMP3E, TSG3nCMP4E, TSG3nCMP7E, TSG3nCMP8E, TSG3nCMP11E, TSG3nCMP12E

(7) Operation Start Timing in 120-DC Mode

In 120-DC mode, when trigger switch control (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1) is used, pattern set with TSG3nOPT1.TSG3nSPC2 to TSG3nOPT1.TSG3nSPC0, TSG3nOPT0.TSG3nPSC, and TSG3nOPT0.TSG3nIDC can be output. However, when pattern switch control (TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 0) is used, the pattern of the TSG3nPTSI2 to TSG3nPTSI0 pins can be detected but the pattern order direction (TSG3nSTR1.TSG3nTSF) cannot be determined. Therefore, set the pattern order direction in TSG3nPSC when TSG3nTE is 0. The TSG3nPSC set value is loaded to TSG3nTSF, and the value can be used for the initial pattern setting.

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 0, TSG3nPOT = 0, TSG3nIDC = 0, TSG3nSTE = 1

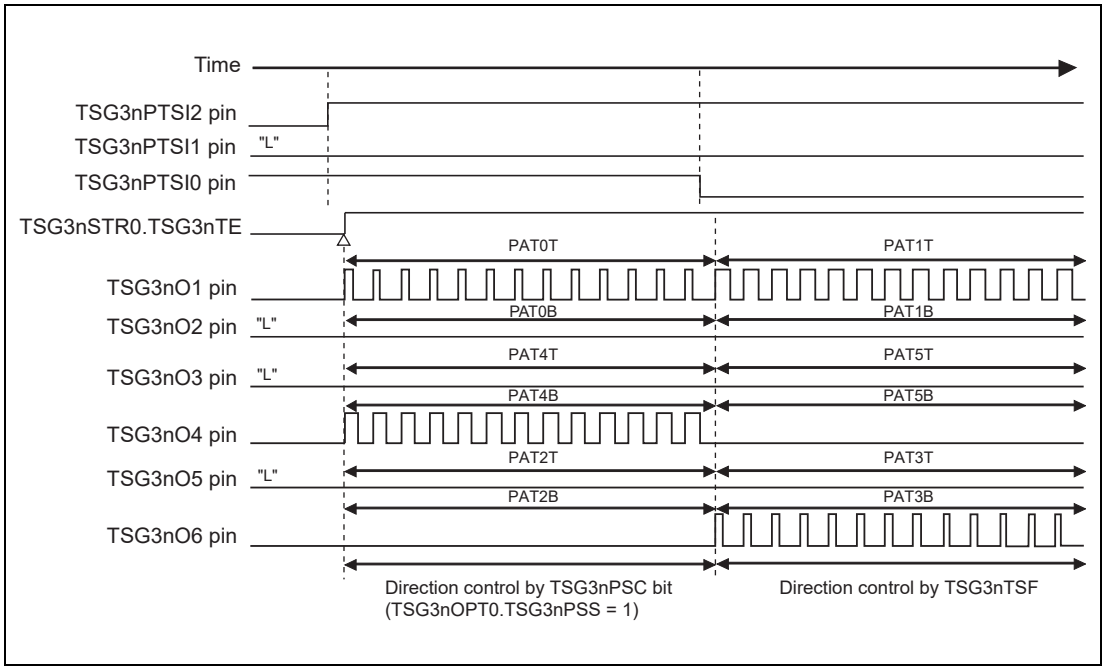


Figure 19.91 Control when Timer Output Starts in Normal Rotation (when Normal Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 1, TSG3nPOT = 0, TSG3nIDC = 1, TSG3nSTE = 1

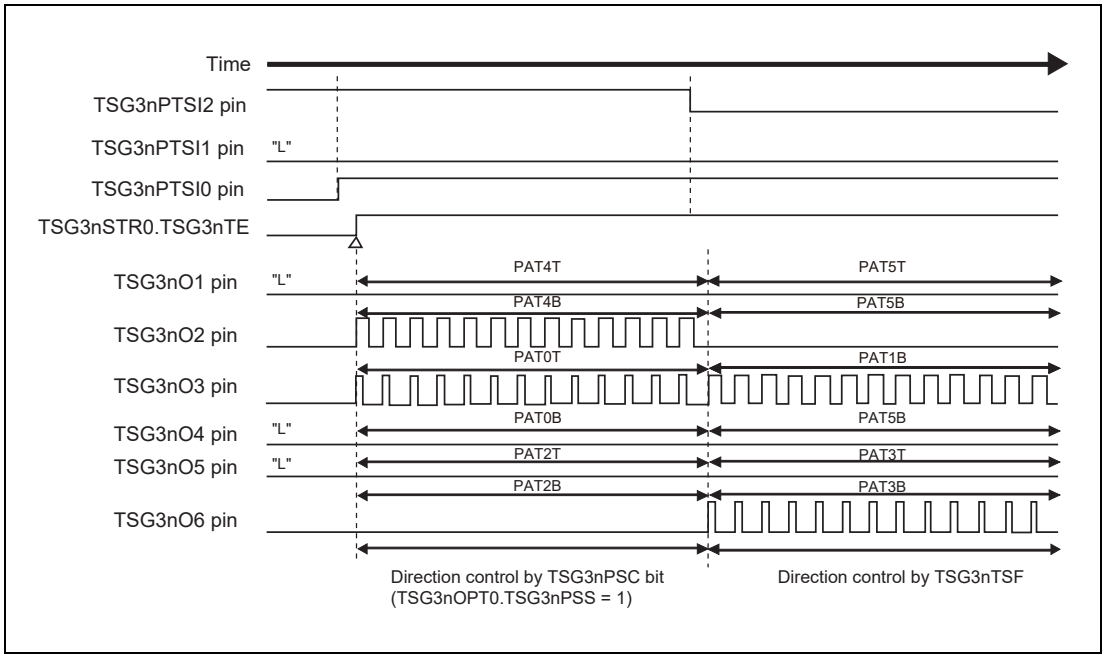


Figure 19.92 Control when Timer Output Starts in Reverse Rotation (when Normal Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 0, TSG3nPOT = 0, TSG3nIDC = 0, TSG3nSTE = 1

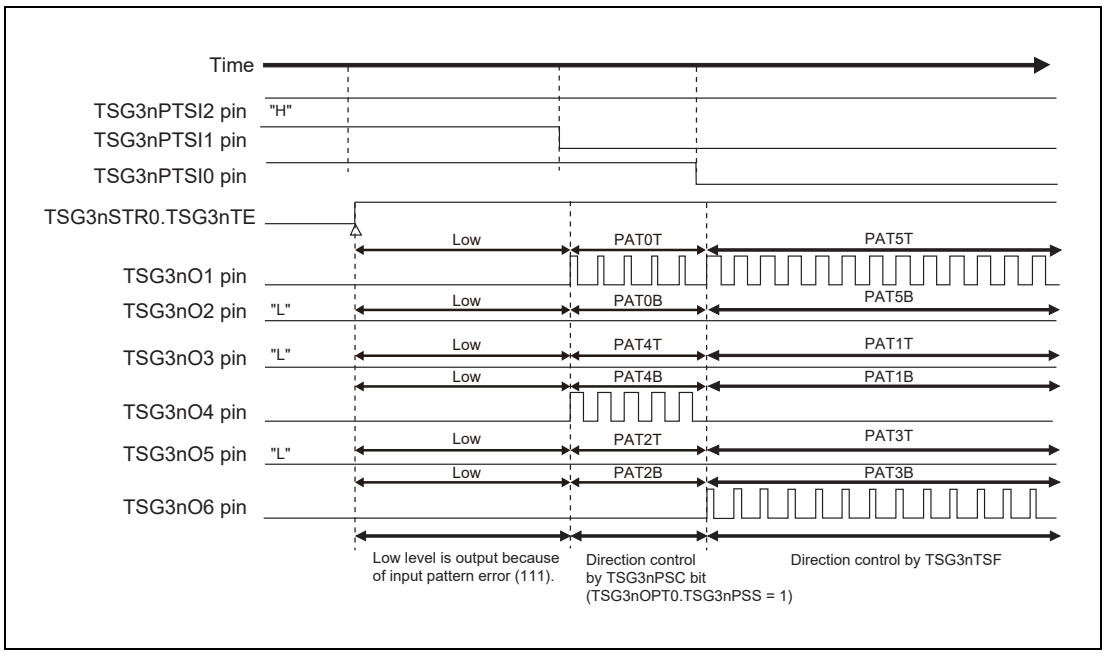


Figure 19.93 Control when Timer Output Starts in Normal Rotation (when Error Pattern is Input)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nPSC = 1, TSG3nPOT = 0, TSG3nIDC = 1, TSG3nSTE = 1

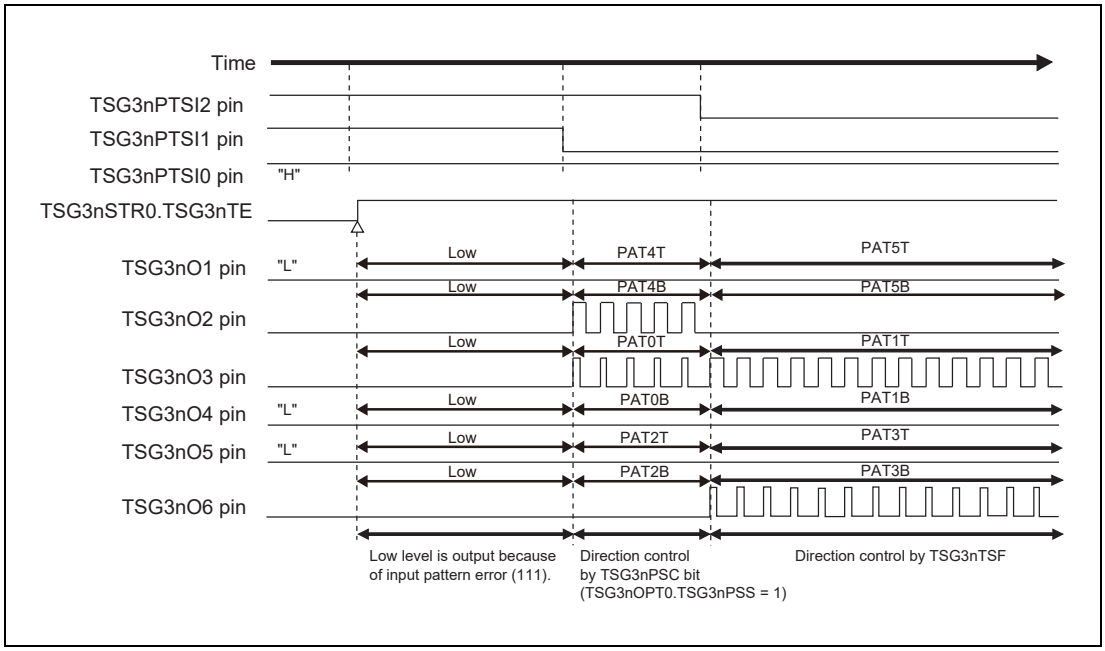


Figure 19.94 Control when Timer Output Starts in Reverse Rotation (when Error Pattern is Input)

(8) Output Switch Timing in 120-DC Mode (TSG3nS120DCO = 0)

As shown in **Figure 19.94** to **Figure 19.97**, in 120-DC mode, the external switch timing for output pattern (TSG3nOPCI0 and TSG3nOPCI1 signals, and TSG3nPTSI2 to TSG3nPTSI0 pins) is input irrespective of the 18-bit counter operation. When TSG3nS120DCO is 0, the output is switched to the new pattern by clearing the 18-bit counter using the pattern switch timing signal applied from outside.

In the pattern switch method, if a change in TSG3nPTSI2 to TSG3nPTSI0 pins occurs several times within one period, the output pattern is switched by clearing the 18-bit counter at each change. In the trigger switch method, if TSG3nOPCI0 and TSG3nOPCI1 signal trigger is input for several times within one period, the output pattern is switched by clearing the 18-bit counter each time the trigger is accepted.

If TSG3nSPC2 to TSG3nSPC0 are rewritten several times within one period, the output pattern is switched by clearing the 18-bit counter each time TSG3nSPC2 to TSG3nSPC0 are rewritten.

In case of a conflict between a rewrite to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 and TSG3nOPCI0 and TSG3nOPCI1 trigger, rewriting of TSG3nSPC2 to TSG3nSPC0 takes precedence.

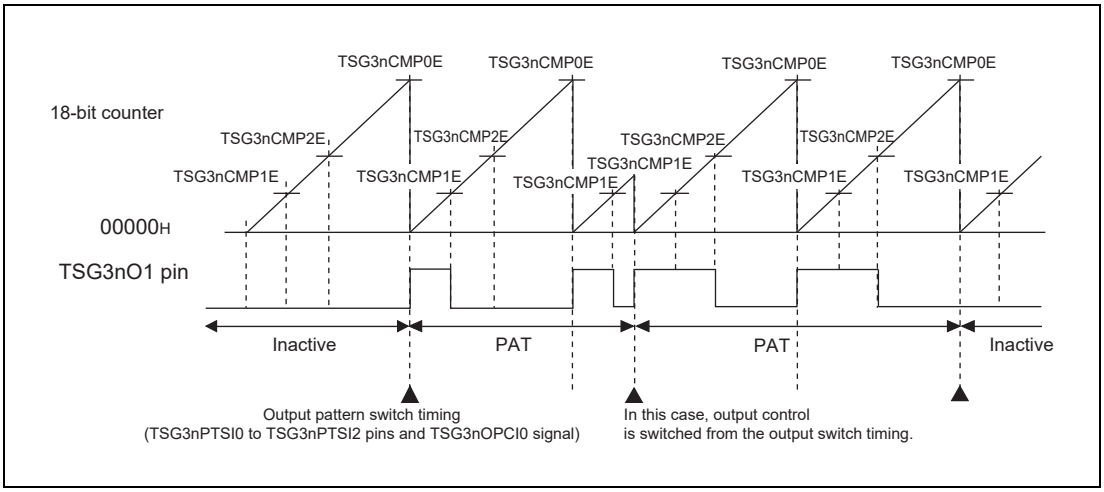
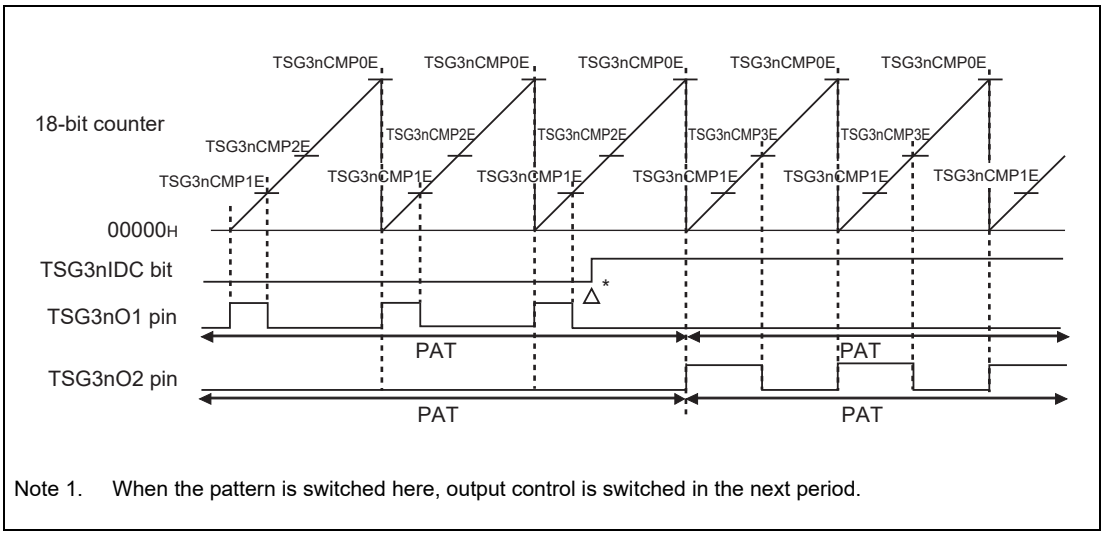


Figure 19.95 Output Switch Example (TSG3nPTSI2 to TSG3nPTSI0 Pins and TSG3nOPCI0 and TSG3nOPCI1 Signal Trigger Input)



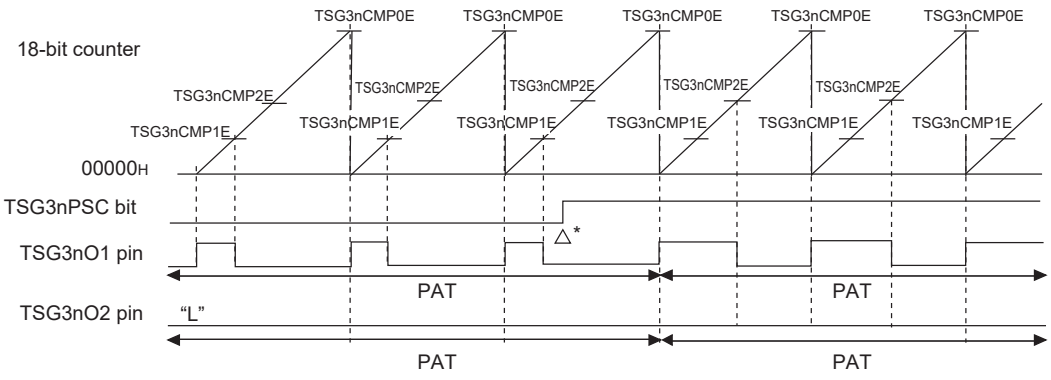
Note 1. When the pattern is switched here, output control is switched in the next period.

Figure 19.96 Output Switch Example (Switched by TSG3nOPT0.TSG3nIDC)

NOTE

If a change in the TSG3nPTSI2-0 pins occurs by the time next period when output control is switched by the TSG3nIDC bit, the 18-bit counter is cleared and output control is switched.

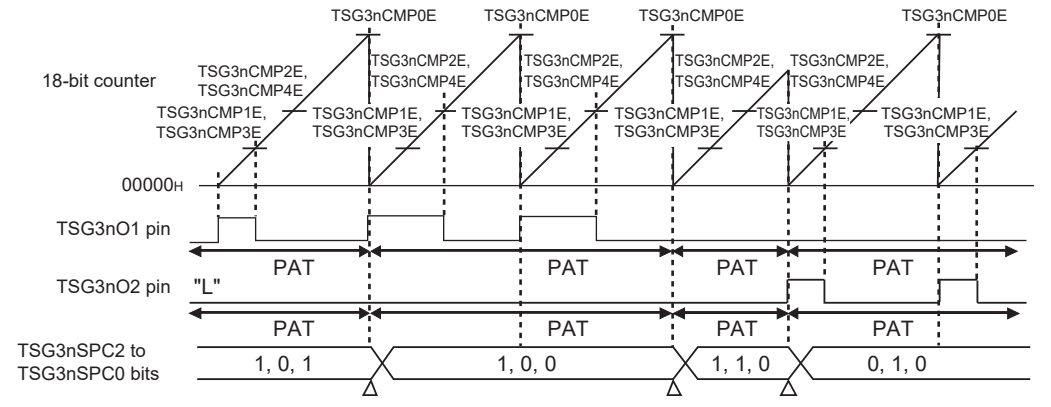
- TSG3nOPT0.TSG3nSTE =1, TSG3nOPT0.TSG3nPOT = 1



Note 1. When the pattern is switched here, output control is switched from the next period.

Figure 19.97 Output Switch Example (Switched by TSG3nOPT0.TSG3nPSC)

- TSG3nOPT0.TSG3nSOC = 0, TSG3nOPT0.TSG3nSTE =1, TSG3nOPT0.TSG3nPOT = 1



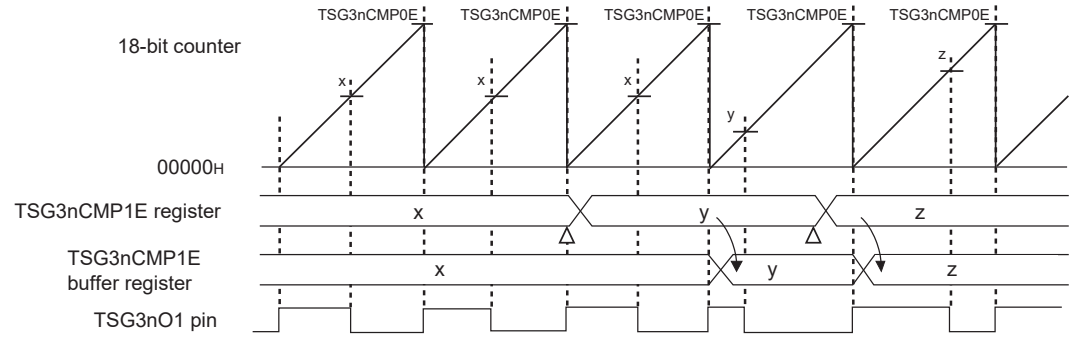
Note: Δ: Write access

Figure 19.98 Output Switch Example (Switched by TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0)

(9) Compare Register Rewrite Timing in 120-DC Mode

Example of operation when TSG3nCMP1E is reloaded (rewritten simultaneously) is shown below.

Figure 19.99 shows an output example when TSG3nCMP1E is rewritten. After TSG3nCMP1E is changed, data is not transferred to the TSG3nCMP1E buffer register (changed data is not valid) until the next reload timing; therefore, the specified output waveform can be obtained. However, do not write to TSG3nCMP1E again while the reload is suspended (period from when TSG3nCMP1E is changed to when simultaneous rewrite is executed). Be sure to read the reload request flag (TSG3nRSF) to confirm that the flag is 0, and write data to TSG3nCMP1E.



Note: Δ: Write access

Figure 19.99 Output Example when TSG3nCMP1E is Rewritten

(10) Dead Time Control in 120-DC Mode

In 120-DC mode, the dead time is controlled on falling of each phase, and the dead time is added.

The dead time set in TSG3nDTC1W is inserted on falling of the positive phase, and the dead time set in TSG3nDTC0W is inserted on falling of the inverse phase.

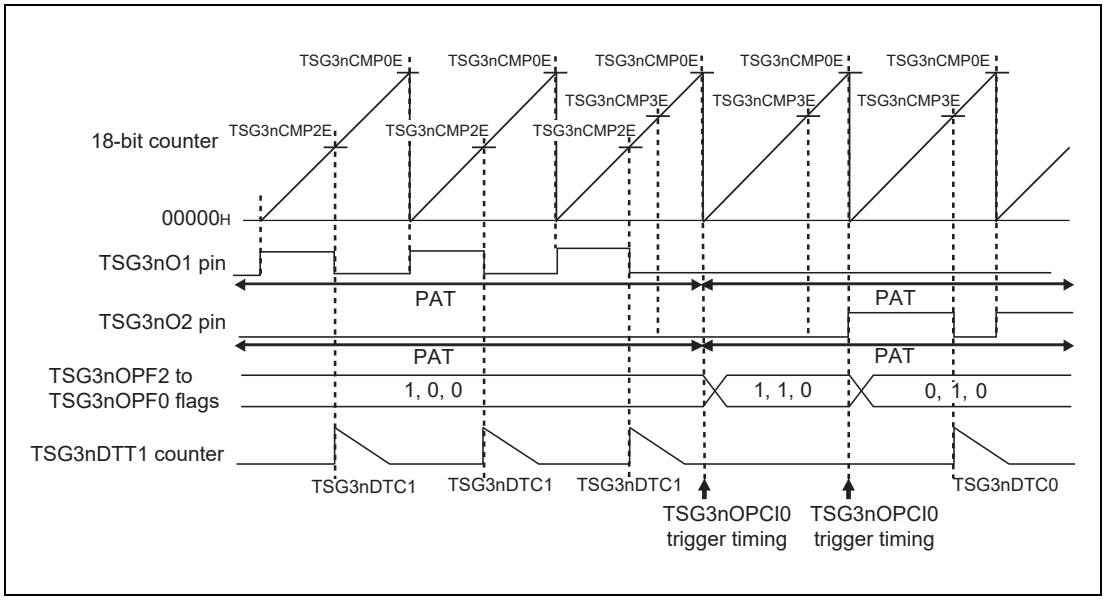


Figure 19.100 Output Switch Example

CAUTION

The dead time control method may affect the timer output. The timer output may not have the specified active level width due to the dead time control under the following conditions:

- **When noise is generated on the input pattern in the pattern switch method**
- **When a change in the input pattern occurs earlier than the PWM period in the pattern switch method**
- **When TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are changed and the output pattern is forcibly changed in the trigger switch method**
- **When switch method is changed**
- **When the current direction control bit (TSG3nOPT0.TSG3nIDC) is changed**
- **When the software output control function is used**

(11) Output Switch in 120-DC Mode

In 120-DC mode, the output pattern can be controlled by writing values to TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 when the trigger switch method (TSG3nOPT0.TSG3nSTE = 1, and TSG3nPOT = 1) is used. The dead time is secured by hardware at the switch timing.

CAUTION

When 111_B or 000_B is written to TSG3nSPC2 to TSG3nSPC0, the TSG3nO1 to TSG3nO6 pins are driven low.

(12) Operation when Noise is Generated in TSG3nPTSI2 to TSG3nPTSI0 Pins in 120-DC Mode

Input to the TSG3nPTSI2 to TSG3nPTSI0 pins is assumed to be the hall sensor signals of the brushless DC motor. Depending on the system, a noise may be generated on the TSG3nPTSI2 to TSG3nPTSI0 pins. Operation when a noise is generated is described below.

For system product design, be sure to insert a noise filter circuit between the hall sensor and the TSG3nPTSI2 to TSG3nPTSI0 pins.

Figure 19.102 shows a case when a noise is generated on the TSG3nPTSI2 to TSG3nPTSI0 pins during operation with the pattern switch method used.

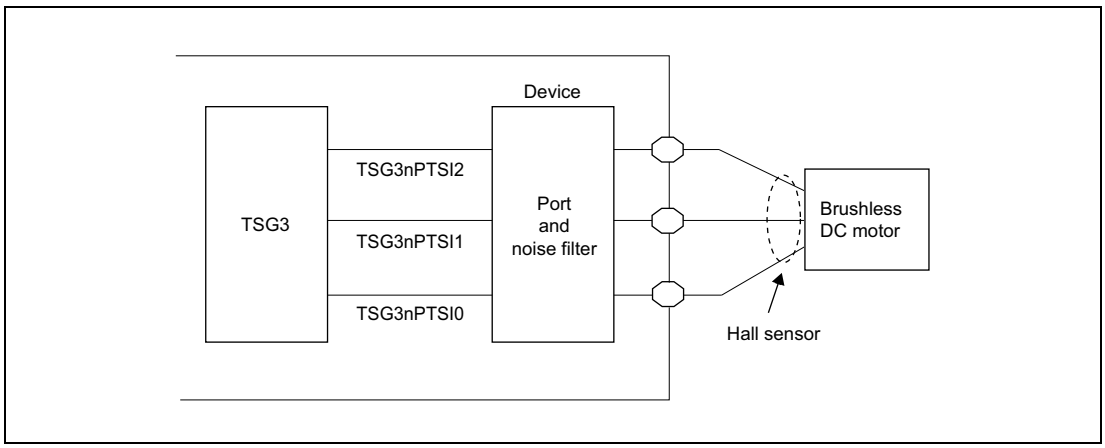


Figure 19.101 Example of Noise Filter Circuit Connection

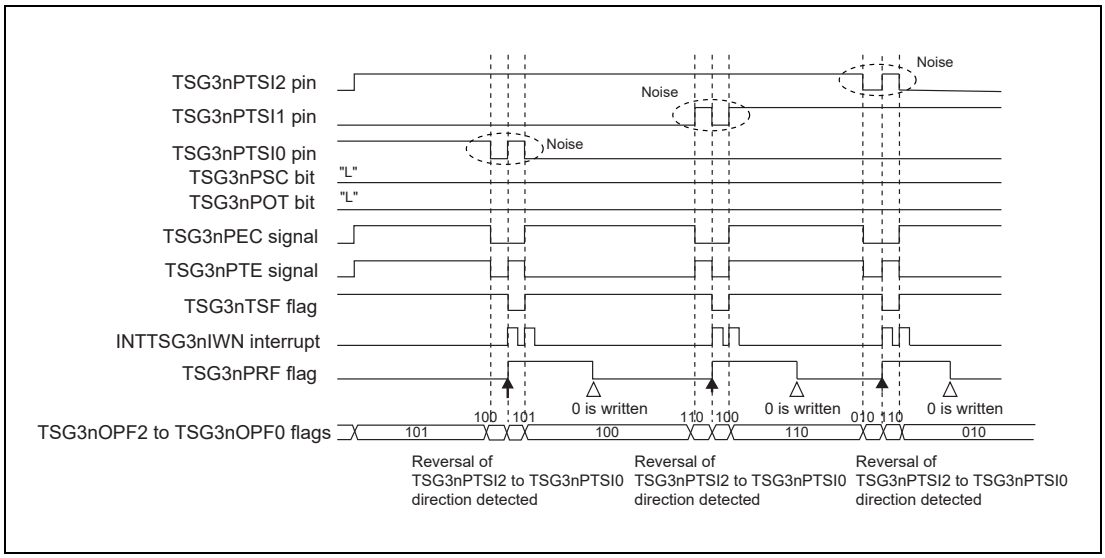


Figure 19.102 Example of Noise Generation at Level Change in TSG3nPTSI2 to TSG3nPTSI0 Pins (Pattern Switch Method)

(a) Change Timing of Input Pattern Change Detection Signal (TSG3nPTE)

The TSG3nPTE signal toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes.

CAUTION

Be sure to specify the pattern order direction by the TSG3nPSC bit (when the TSG3nPSS bit in the TSG3nOPT0 register is 1) in the TSG3nOPT0 register.

When TSG3nPSC = 0:

Table 19.88 TSG3nPTE Toggle Operation when TSG3nPSC is 0

		TSG3nPTSI2-TSG3nPTSI0 Pins after change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

When TSG3nPSC = 1:

Table 19.89 TSG3nPTE Toggle Operation when TSG3nPSC is 1

		TSG3nPTSI2-TSG3nPTSI0Pins after change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

(b) Change Timing of Three-Phase Encode Signal (TSG3nPEC)

he TSG3nPEC signal toggles when input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes

Table 19.90 TSG3nPEC Toggle Operation

		TSG3nPTSI2-TSG3nPTSI0after change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	Toggle
	100	—	—	Toggle	—	Toggle	—	—	—
	110	—	—	—	Toggle	—	Toggle	—	—
	010	—	—	—	—	Toggle	—	Toggle	—
	011	—	—	—	—	—	Toggle	—	Toggle
	001	—	—	Toggle	—	—	—	Toggle	—

(c) Change Timing of TSG3nO1 to TSG3nO6 Pins

- When the pattern switch method is used, the output pattern changes when the input signal of the TSG3nPTSI2 to TSG3nPTSI0 pins*1 changes. The output is also switched when two or more pins change simultaneously.
- When the trigger switch method is used, the output pattern changes at the rising edge of the TSG3nOPCI0 and TSG3nOPCI1 signals. The output also changes when data is written to TSG3nSPC2 to TSG3nSPC0*1 in TSG3nOPT0.

Note 1. When the input pattern changes to 000 or 111, the TSG3nO1-TSG3nO6 pins are driven low. The output pattern of TSG3nO1-TSG3nO6 changes immediately only when TSG3nS120DCO = 0. When TSG3nS120DCO = 1, the output pattern is switched when the main counter (TSG3nCNTE) is matched with TSG3nCMP0E (from the next timer cycle).

(d) Change Timing of TSG3nTSF Flag

The TSG3nTSF flag toggles when the input pattern (TSG3nPTSI2 to TSG3nPTSI0 pins) changes

Table 19.91 Setting and Clearing of TSG3nTSF

		TSG3nPTSI2-TSG3nPTSI0 Pins after change							
		000	111	101	100	110	010	011	001
Current TSG3nPTSI2 to TSG3nPTSI0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	0	—	—	—	1
	100	—	—	1	—	0	—	—	—
	110	—	—	—	1	—	0	—	—
	010	—	—	—	—	1	—	0	—
	011	—	—	—	—	—	1	—	0
	001	—	—	0	—	—	—	1	—

(e) Set Timing of TSG3nNDF Flag

This flag is set when two or more pins of the TSG3nPTSI2 to TSG3nPTSI0 pins change simultaneously, and cleared when 1 is written to the TSG3nNDR bit. The TSG3nNDF flag is valid when 1 is set to the TSG3nNDC bit.

(f) Set Timing of TSG3nPRF Flag

This flag is set when the TSG3nTSF flag changes, and cleared when 1 is written to the TSG3nPRR bit. The TSG3nPRF flag is valid when 1 is set to the TSG3nPRC bit.

(g) Set Timing of TSG3nPEF Flag

This flag is set when 000 or 111 is input to the TSG3nPTSI2 to TSG3nPTSI0 pins, and cleared when 1 is written to the TSG3nPER bit. The TSG3nPEF flag is valid when 1 is set to the TSG3nPRC bit.

(13) Basic Control Flow in 120-DC Mode

In 120-DC mode, there are eight control states as listed in **Table 19.92**.

When TSG3nOPT0.TSG3nSTE = 1 and TSG3nPOT = 0, the pattern switch method is used for 120-DC control.

This is defined as fixed phase control. The fixed phase control should be performed considering the factors such as delay from the hall sensor and delay from sensor level detection to timer output. However, acceleration or deceleration can be performed simply by changing the PWM duty.

When TSG3nOPT0.TSG3nSTE = 1 and TSG3nPOT = 1, the trigger switch method is selected for 120-DC control.

This is defined as variable phase control. With the variable phase control, the timer output pattern is set prior to the hall sensor; therefore, acceleration or deceleration control according to the phase difference can be performed. However, control is more complex than the fixed phase control because offset width with respect to the hall sensor and the predicted value with respect to the hall sensor should be considered. For details, **Section 23.2.3.10, Three-Phase Pulse Input Control Function**.

When TSG3nOPT0.TSG3nSTE = 1, TSG3nOPT0.TSG3nPOT = 1, and TSG3nPSS = 1, the pattern order direction of the motor can be set with the TSG3nPSC bit in the TSG3nOPT0 register. Set TSG3nPSC to 0 to set normal rotation, and 1 to set reverse rotation.

The TSG3nIDC bit in the TSG3nOPT0 register sets the electric current direction. If the same value as the rotation direction of the motor (TSG3nPSC set value) is set, acceleration control is set. If the different value from the rotation direction of the motor is set, deceleration control is set.

Table 19.92 Timer Control Status

Status	TSG3nPSC in TSG3nOPT0	TSG3nTSF in TSG3nSTR1	TSG3nIDC in TSG3nOPT0	TSG3nPOT in TSG3nOPT0	Control
A	—	0	0	0	Normal rotation, acceleration, and fixed phase
B	0	—	0	1	Normal rotation, acceleration, and variable phase
C	0	—	1	1	Normal rotation, deceleration, and variable phase
D	—	0	1	0	Normal rotation, deceleration, and fixed phase
E	—	1	1	0	Reverse rotation, acceleration, and fixed phase
F	1	—	1	1	Reverse rotation, acceleration, and variable phase
G	1	—	0	1	Reverse rotation, deceleration, and variable phase
H	—	1	0	0	Reverse rotation, deceleration, and fixed phase

Generally, the state, when the motor rotation stops, is assumed to be a state of the start and the control begins. First the fixed phase control is used to rotate the motor from the stopped state. Afterwards, to accelerate the motor speed to the fast rotation, the variable phase control is switched on. In combination with encoder timer (ENCA) and the variable phase control, the timer output is changed according to timing that is earlier than the change point of the hall sensors (leading).

To decelerate the motor speed from fast rotation, the direction of control is switched to deceleration control by rewriting only TSG3nIDC in TSG3nOPT0. When the rotation count can be reduced to low-speed rotation, the rotation can be placed in the stopped state by decreasing the PWM duty.

State transition is shown in **Figure 19.103** and **Figure 19.104**.

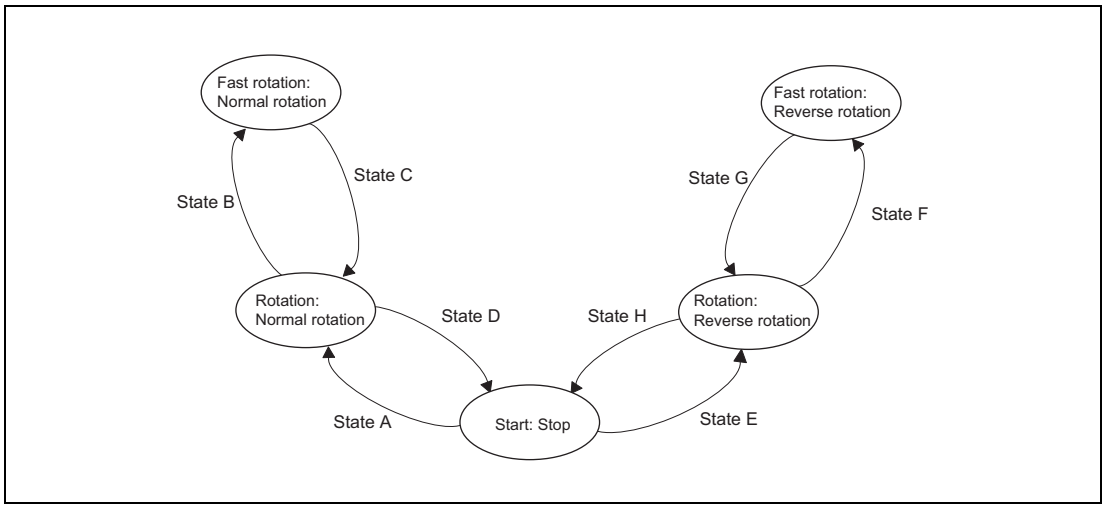


Figure 19.103 State Transition Diagram

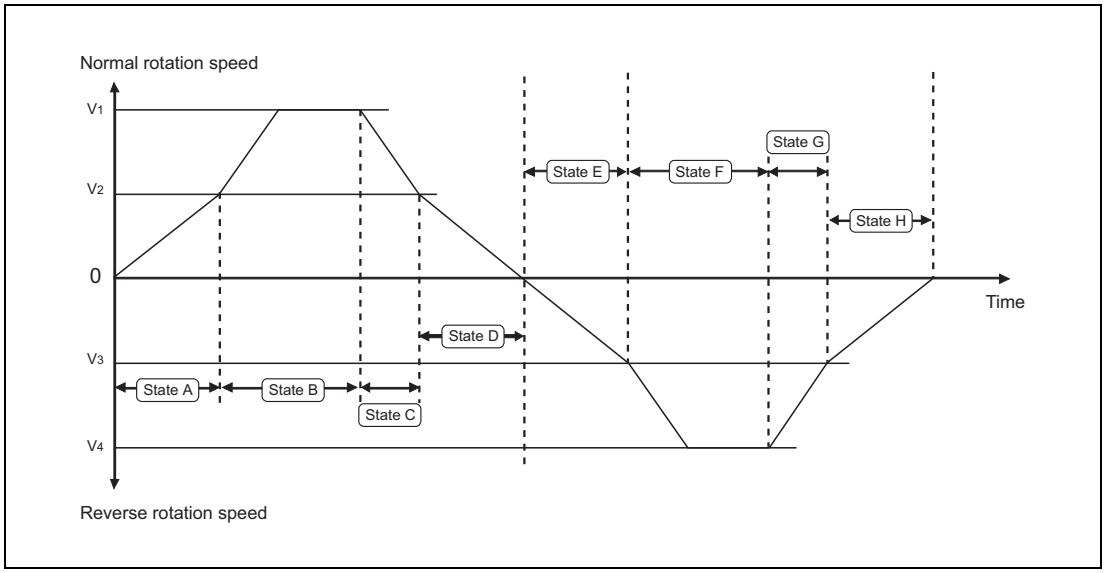


Figure 19.104 Relationship between State Transition and Rotation Speed of Motor

NOTE

V1 and V4: Fast rotation speed of normal rotation and reverse rotation

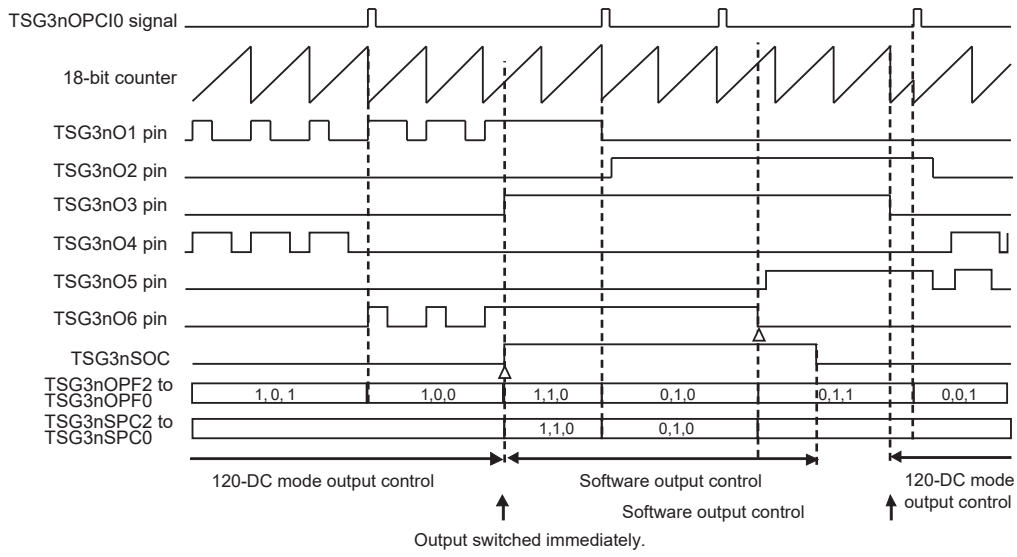
V2 and V3: Low rotation speed of normal rotation and reverse rotation

(14) Software Output Control Function in 120-DC Mode

TSG3nOPT0.TSG3nSOC and TSG3nIDC, and TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0 are used in 120-DC mode for timer output control by software.

As shown in **Figure 19.105**, the output control is switched immediately when TSG3nSOC is set to 1. If the dead time is set, the period of the dead time period is guaranteed. After that, to switch the output control from software output control to 120-DC control, set TSG3nSOC to 0. At this timing, output control is retained. When the reload timing is generated, output control is switched to 120-DC mode.

For details on software output control function, see **Section 19.4.7.10, Software Output Control Function**.



Note: Δ: Write access

Figure 19.105 Example of Switching from 120-DC Mode to Software Output Control Function

(a) Procedure for Software Output Control

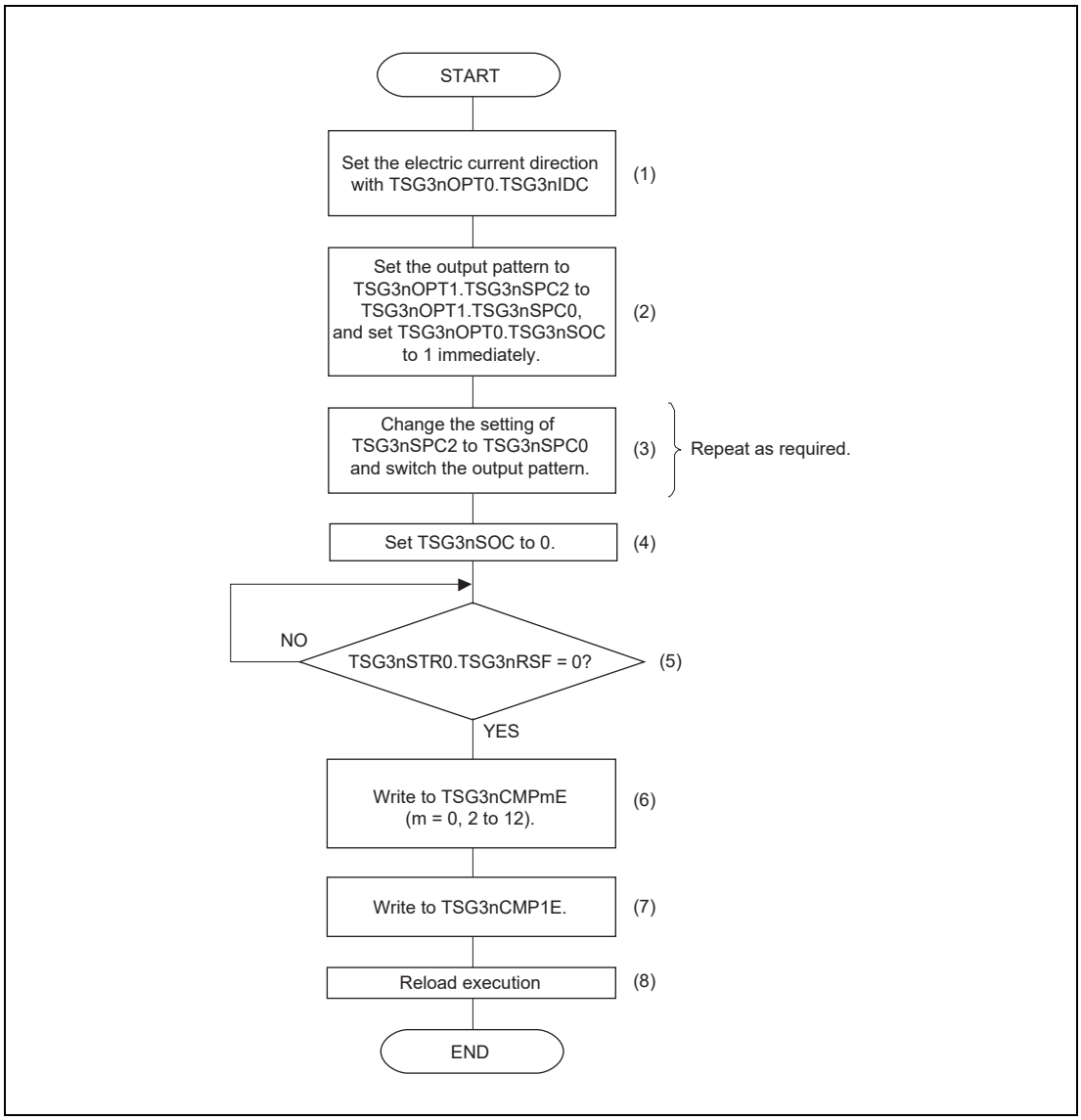


Figure 19.106 Process Flow of Software Output Control

The procedure for software output control is described below.

(1) Set the TSG3nIDC bit. The phase of the timer output with TSG3nIDC = 0 is different by 180 degrees from that with TSG3nIDC = 1. When this bit is rewritten with the software output control function, the output pattern will change according to the new setting at the next timer cycle. However, if the period match occurs before step (2), the output pattern of 120-DC control changes. So, schedule so as to prevent the period match from occurring before step (2).

(2) Set the output pattern to TSG3nSPC2 to TSG3nSPC0. To enable software output control, set TSG3nSOC to 1 simultaneously.

(3) Change the output pattern setting of TSG3nSPC2 to TSG3nSPC0 to change the timer output.

(4) Confirm that the reload request flag (TSG3nRSF) = 0. If TSG3nRFS = 1, do not proceed to the following step until TSG3nRSF = 0.

(5) By setting TSG3nSOC = 0 the software control starts to be released (it is not released here yet).

(6) After releasing the software output control, set the necessary compare registers. Proceed to the following step when the register setting is not required. Here, change the registers with the reload function.

(7) Write to TSG3nCMP1E to start reloading.

(8) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (4) to (7). When reload cannot be executed, the software output cannot be released.

19.4.7.7 HSP-PWM Mode (High accuracy Shifted-pulse - Pulse Width Modulation Mode)

Overview

In this mode, the 18-bit counter and the 18-bit compare register are used to generate a high accuracy sawtooth waveform PWM signal.

Prerequisites

- Set the PWM period to TSG3nCMP0E.
- Set PWM output width with the TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE registers. Set PWM shift width with the TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE registers. Set dead time with the TSG3nDTC0W and TSG3nDTC1W registers. The values set in these registers are immediately reflected in the corresponding TSG3nCMPmE (m = 1 to 12) based on the calculation described later.

Functional description

Set the PWM period, the duty cycle, and the PWM sift width. Then set the PWM output width. Counting up begins when TSG3nTRG0.TSG3nTS is set to 1.

The 18-bit counter starts counting from 00000_H and is cleared by the match with TSG3nCMP0E.

During counting, a compare match interrupt (INTTSG3nI0 to INTTSG3nI12) is generated by the match of the buffer register TSG3nCMP0E to TSG3nCMP12E with the 18-bit counter.

NOTE

The HSP-PWM mode is set when TSG3nCTL0.TSG3nMD2-TSG3nMD0 = 100_B.

(1) Basic Timing Chart

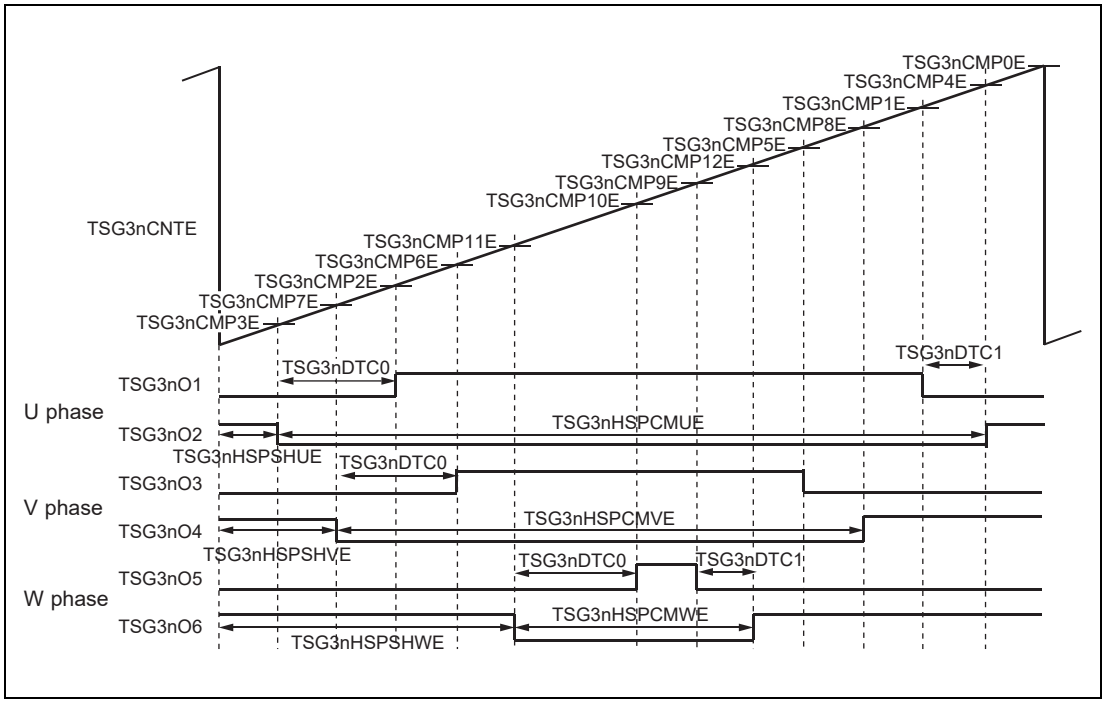


Figure 19.107 Basic Timing in HSP-PWM Mode

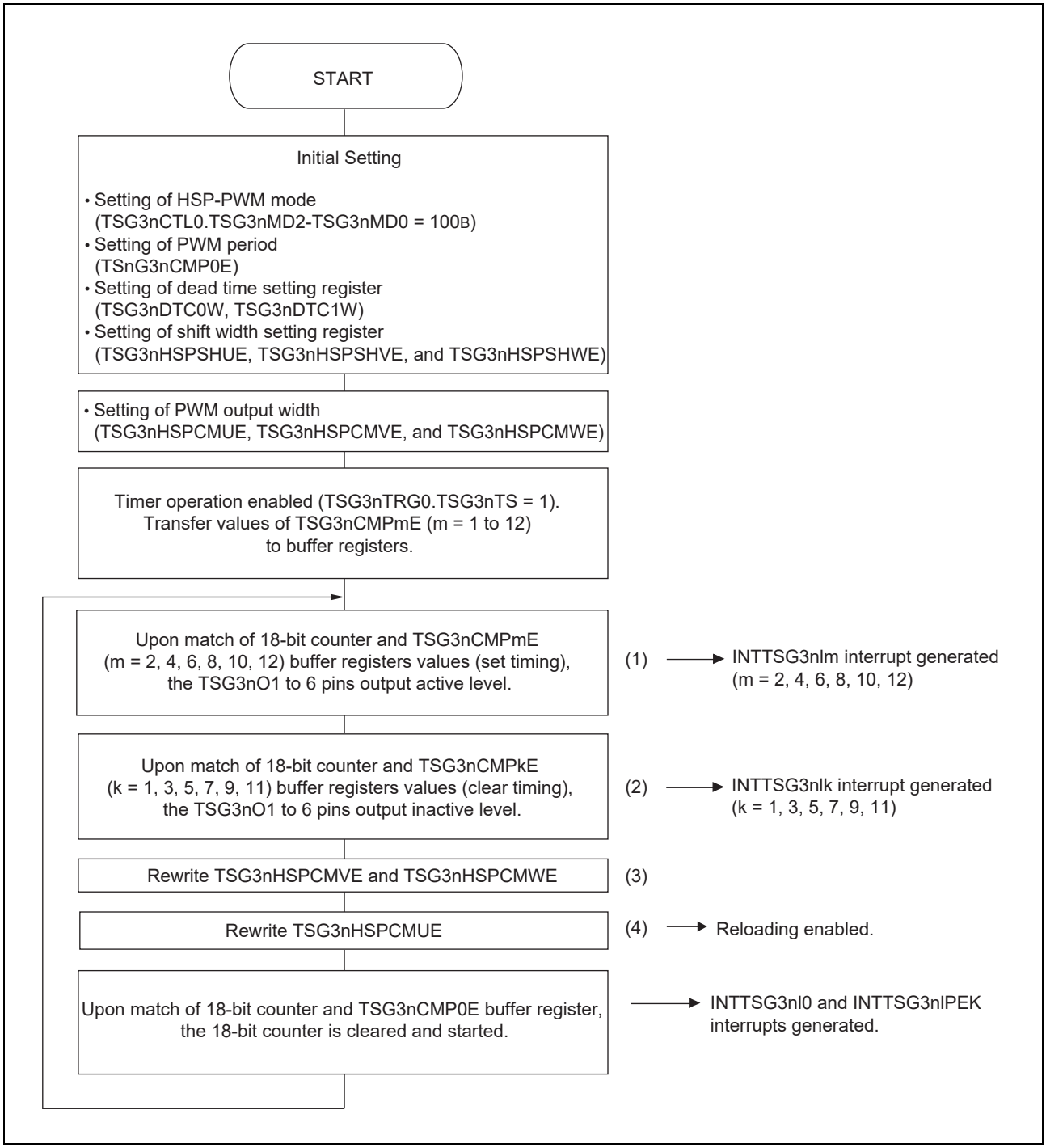


Figure 19.108 Basic Operation Flow in HSP-PWM Mode

NOTE

The timing may be different from the orders described above but be sure to execute (4) after (3).

CAUTIONS

1. **When changing the settings for PWM output width in the TSG3nHSPCMUE, VE, and WE registers during operation, set the TSG3nHSPCMUE register last. When only the settings for the V phase and W phase are changed but the PWM output width in the U phase is not, the existing value should be written back to the TSG3nHSPCMUE register.**
2. **After changing the PWM cycle by using the TSG3nCMP0E register, the PWM output width settings in the TSG3nHSPCMUE, VE, and WE registers must be remade. Changing the settings for the PWM shift width in the TSG3nHSPSHUE, VE, and WE registers and for the PWM cycle in the TSG3nCMP0E register at the same time is not allowed.**
3. **Remake the settings for PWM output width in the TSG3nHSPCMUE, VE, and WE registers after changing the settings for PWM shift width in the TSG3nHSPSHUE, VE, and WE registers.**

(2) List of Operation in HSP-PWM

Table 19.93 Counter Function in HSP-PWM

Operation		Setting Condition
18-bit counter	Start	TSG3nTRG0.TSG3nTS = 0 → 1 or simultaneous start trigger
	Clear	Compare match of TSG3nCMP0E buffer register and 18-bit counter
	Stop	TSG3nTRG1.TSG3nTT = 0 → 1

Table 19.94 Functions of Compare Registers, Sift Width Setting Register, and Dead Time Setting Register in HSP-PWM

Register	Rewriting Method	Rewrite during Operation	Function
TSG3nCMP0E	Reload	Possible	Setting period
TSG3nHSPCMUE	Reload	Possible	PWM control for U phase
TSG3nHSPSHUE			
TSG3nHSPCMVE	Reload	Possible	PWM control for V phase
TSG3nHSPSHVE			
TSG3nHSPCMWE	Reload	Possible	PWM control for W phase
TSG3nHSPSHWE			
TSG3nDCMP0E, TSG3nDCMP1E, TSG3nDCMP2E	Reload	Possible	Diagnostic output or A/D conversion trigger
TSG3nDTC0W, TSG3nDTC1W	Reload	Prohibited	Dead time

Table 19.95 Timer Output in HSP-PWM Mode

Pins	Function
TSG3nO1	PWM output by compare match of the TSG3nCMP1E buffer register (clear timing) or the TSG3nCMP2E buffer register (set timing) with the 18-bit counter.
TSG3nO2	PWM output by compare match of the TSG3nCMP3E buffer register (clear timing) or the TSG3nCMP4E buffer register (set timing) with the 18-bit counter.
TSG3nO3	PWM output by compare match of the TSG3nCMP5E buffer register (clear timing) or the TSG3nCMP6E buffer register (set timing) with the 18-bit counter.
TSG3nO4	PWM output by compare match of the TSG3nCMP7E buffer register (clear timing) or the TSG3nCMP8E buffer register (set timing) with the 18-bit counter.
TSG3nO5	PWM output by compare match of the TSG3nCMP9E buffer register (clear timing) or the TSG3nCMP10E buffer register (set timing) with the 18-bit counter.
TSG3nO6	PWM output by compare match of the TSG3nCMP11E buffer register (clear timing) or the TSG3nCMP12E buffer register (set timing) with the 18-bit counter.
TSG3nO7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 19.96 Interrupt Request in HSP-PWM Mode

Interrupt	Function
INTTSG3nIm (m = 0 to 12)	Compare match of the TSG3nCMPmE buffer register with 18-bit counter
INTTSG3nIER	Error
INTTSG3nIVLY	—
INTTSG3nIPEK	Peak interrupt (generated at the same timing as INTTSG3nI0)
INTTSG3nIWN	Error

Table 19.97 Compare Match Timing in HSP-PWM Mode

Compare Match	Function
TSG3nCMP0E	When 18-bit counter changes from TSG3nCMP0E to 00000_H
TSG3nCMPmE (m = 1 to 12)	After match of 18-bit counter and TSG3nCMP0E

(3) Various Settings of HSP-PWM Mode

Mode setting

HSP-PWM mode is entered by setting TSG3nCTL0.TSG3nMD2-TSG3nMD0 to 100_B.

Setting timer output

The output pins TSG3nO1-6 are controlled by setting TSG3nIOC0, TSG3nIOC2, and TSG3nIOC3.

The TSG3nO7 pin provides output pulse as diagnostic output or A/D conversion trigger. The pin should be set as necessary.

Enabling error interrupt generation

Error interrupt (INTTSG3nIER) due to the detection of the simultaneous active state of positive and inverse phases is enabled by setting TSG3nIOC1.TSG3nEOC to 1. For details, see **Section 19.4.6, Error/Warning Interrupt**.

Setting rewriting timing of register with reload function

This function is only available in reload mode. Set TSG3nCTL3.TSG3nRMC to 0.

Setting A/D conversion trigger output

The A/D conversion trigger 0 (TSG3nADTRG0 signal) is set with TSG3nCTL5.TSG3nAT09-TSG3nAT00.

TSG3nAT09-TSG3n00 is used to enable or disable the A/D conversion trigger output on timing match of TSG3nDCMP2E-0E with 18-bit counter (up count).

TSG3nCTL6.TSG3nAT19-TSG3nAT10 is used to set the A/D conversion trigger 1 (TSG3nADTRG1 signal).

To set the match timing of the 18-bit counter and TSG3nDCMP2E-TSG3nDCMP0E, set the compare value to each register.

The skipping function can be used for TSG3nADTRG0, TSG3nADTRG1 signals. TSG3nACC01 and TSG3nACC00 of TSG3nCTL5, and TSG3nACC11 and TSG3nACC10 of TSG3nCTL6 can be used to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

- **Be sure to set TSG3nCTL5, TSG3nCTL6, and TSG3nDCMP2E-0E correctly when the timing pulse of A/D conversion trigger is output to TSG3nO7.**
- **In HSP-PWM mode, no trough interrupt (INTTSG3nIVLY) is generated. Therefore, SG3nCTL5.TSG3nAT00 and TSG3nCTL6.TSG3nAT10 must be set to 0.**
- **In HSP-PWM mode, the 18-bit sub-counter does not operate. Therefore, TSG3nCTL5.TSG3nAT09 and TSG3nAT08, and TSG3nCTL6.TSG3nAT19 and TSG3nAT18 must be set to 0.**
- **In HSP-PWM mode, down counting by the 18-bit counter is not generated. Therefore, TSG3nCTL5.TSG3nAT07, TSG3nAT05, and TSG3nAT03, and TSG3nCTL6.TSG3nAT17, TSG3nAT15, and TSG3nAT13 should be set to 0.**

Setting PWM period

Set the PWM period with TSG3nCMP0E according to the following expression:

PCLK × (TSG3nCMP0E + 1)

Setting PWM output width

The PWM output width is set with TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE (TSG3nCMP1E-12E).

Satisfy the following requirements when setting the TSG3nHSPCMUE, TSG3nHSPCMVE, and TSG3nHSPCMWE registers:

0 ≤ TSG3nHSPCMUE,TSG3nHSPCMVE, TSG3nHSPCMWE ≤ TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

Set the PWM output width after setting TSG3nCMP0E, TSG3nHSPSHUE, TSG3nHSPSHVE, TSG3nHSPSHWE, TSG3nDTC0, and TSG3nDTC1.

Setting of PWM sift width

PWM shift width is set with TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE.

Satisfy the following requirements when setting TSG3nHSPSHUE, TSG3nHSPSHVE, and TSG3nHSPSHWE.

TSG3nHSPSHUE, TSG3nHSPSHVE, TSG3nHSPSHWE ≤ TSG3nCMP0E

Setting dead time

The dead time can be set with TSG3nDTC0 and TSG3nDTC1.

PCLK × TSG3nDTC0

PCLK × TSG3nDTC1

TSG3nDTC0 can set the time between a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the inactive state and a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the active state.

TSG3nDTC1 can set the time between a change of TSG3nO1, TSG3nO3, and TSG3nO5 to the inactive state and a change of TSG3nO2, TSG3nO4, and TSG3nO6 to the active state.

Satisfy the following requirements when setting TSG3nDTC0 and TSG3nDTC 1.

(TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1) < $3FFFF_H$

TSG3nCMP0E > 3 × TSG3nDTC0

TSG3nCMP0E > 3 × TSG3nDTC1

CAUTION

Do not modify the settings of TSG3nDTC0 and TSG3nDTC1 during timer operation in HSP-PWM (TSG3nTE = 1). Set TSG3nDTC0 and TSG3nDTC1 while TSG3nTE = 0.

Set dead time in HSP-PWM mode. Do not set 0 to TSG3nDTC0 and TSG3nDTC1.

Settings for operation in HSP-PWM Mode

Use HSP-PWM mode with the following settings of control registers and bits.

Do not modify the settings during operation (TSG3nTE = 1).

Table 19.98 Setting Prohibited in HSP-PWM Mode

Bit Name	Setting Value	Description
TSG3nCTL3.TSG3nRMC	0	Available only in reload mode.
TSG3nIOC3.TSG3nTOL6-1	000000_B	Logical inverse of set/clear of PWM is prohibited (HSP-PWM mode limitation)
TSG3nOPT0.TSG3nSOC	0	Switching to software control function is prohibited.
TSG3nOPT0.TSG3nSTE	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPOT	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSS	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nIDC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT0.TSG3nPSC	0	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nOPT1.TSG3nSPC2-0	000_B	Operation setting of 120-DC mode and software control function (value after reset).
TSG3nPAT0W	0000000_H	Operation setting of 120-DC mode (value after reset).
TSG3nPAT1W	0000000_H	Operation setting of 120-DC mode (value after reset).
TSG3nOPT2.TSG3nESSC	0	Switching to ESW function is prohibited.

CAUTION

In HSP-PWM mode, any setting should not be made directly to the TSG3nCMPmE register (m = 1 to 12).

The PWM output width and the PWM shift width should be set with TSG3nHSPCMUE, VE, and WE registers and TSG3nHSPSHUE, VE, and WE registers.

19.4.7.8 Compare Register Set Value in HSP-PWM Mode

In HSP-PWM mode, the PWM output width is set with TSG3nHSPCMUE, VE and WE.

With a write access to TSG3nHSPCMUE, VE, and WE, TSG3 calculates and sets the values to TSG3nCMP1E-12E based on the values written to the followings:

- TSG3nCMP0E (PWM period setting)
- TSG3nDTC0 (dead time setting 0)
- TSG3nDTC1 (dead time setting 1)
- TSG3nHSPSHUE/VE/WE (PWM shift width setting)
- TSG3nHSPCMUE/VE/WE (PWM output width setting)

The high accuracy PWM is realized by these values.

The algorithm to set compare register values are listed in the following table.

Table 19.99 Algorithm for Compare Setting in HSP-PWM Mode

Value Set In HSPCMUE					CMP4E	CMP3E	CMP2E	CMP1E
HSPCMUE = 0					if (HSPSHUE = 0) 0 else HSPCMUE – 1 + HSPSHUE	CMP0E + 1	0	0
0	<	HSPCMUE	≤	DTC0 + DTC1	HSPCMUE – 1 + HSPSHUE	if (HSPSHUE = 0) CMP0E else HSPSHUE – 1	0	0
DTC0 + DTC1	<	HSPCMUE	≤	CMP0E	HSPCMUE – 1 + HSPSHUE	if (HSPSHUE = 0) CMP0E else HSPSHUE – 1	DTC0 – 1 + HSPSHUE	HSPCMUE – DTC1 – 1 + HSPSHUE
CMP0E	<	HSPCMUE	≤	CMP0E + DTC1 + 1	0	0	DTC0 – 1 + HSPSHUE	HSPCMUE – DTC1 – 1 + HSPSHUE
CMP0E + DTC1 + 1	<	HSPCMUE	<	CMP0E + DTC0 + DTC1 + 1	0	0	DTC0 – 1 + HSPSHUE	HSPCMUE – CMP0E – DTC1 – 2 + HSPSHUE
HSPCMUE = CMP0E + DTC0 + DTC1 + 1					0	0	DTC0 – 1 + HSPSHUE	CMP0E + 1

: For the colored sells, subtract CMP0E + 1 when the calculated result is greater than CMP0E.

NOTE

"TSG3n" is omitted from the register names used in the calculation.

19.4.7.9 Timer Output Operation in HSP-PWM Mode

TSG3nO1-6 output is set by compare match of TSG3nCNTE with each TSG3nCMP2E, 4E, 6E, 8E, 10E, and 12E, and cleared by compare match of TSG3nCNTE with each TSG3nCMP1E, 3E, 5E, 7E, 9E, and 11E.

When PWM output width is set in TSG3nHSPCMUE,VE, and WE, the value is set in the TSG3nCMP1E-12E registers based on the calculation described in **Section 19.4.7.8, Compare Register Set Value in HSP-PWM Mode** that enables a high accuracy PWM output from 0% to 100%.

PWM output timing can be shifted by setting desired width to TSG3nHSPSHUE, VE, WE.

(1) When TSG3nHSPCMUE, VE, and WE (PWM output width setting) set value is 0

When 0 is set to TSG3nHSPCMUE (U phase PWM output width setting), 0 is set to TSG3nCMP1E, 2E, and 4E, and "TSG3nCMP0E+1" is set to TSG3nCMP3E.

Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E and clearing by the match with TSG3nCMP1E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO1 is fixed to inactive.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E occurs when TSG3nCNTE = 0.

"TSG3nCMP0E + 1" is set to TSG3nCMP3E. Here, with no match of TSG3nCNTE with TSG3nCMP3E occurs, TSG3nO2 is fixed to active.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when 0 is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

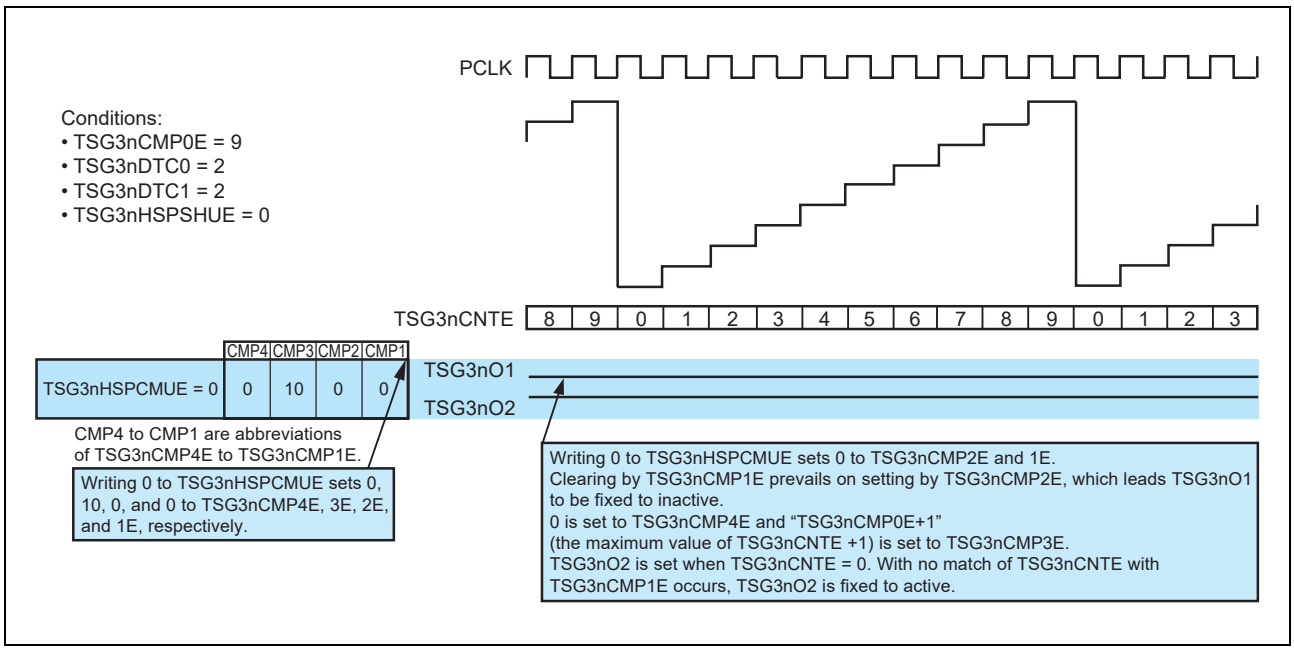


Figure 19.109 Waveform in HSP-PWM Mode (0 is set to TSG3nHSPCMUE)

(2) When TSG3nHSPCMUE/VE/WE (PWM output width setting) set value is 0 < TSG3nHSPCMUE ≤ TSG3nDTC0 + TSG3nDTC1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), 0 is set to TSG3nCMP1E and 2E, “the TSG3nHSPCMUE set value - 1” is set to TSG3nCMP4E, and the value same as TSG3nCMP0E is set to TSG3nCMP3E.

Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E and clearing by the match with TSG3nCMP1E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO1 is fixed to inactive.

TSG3nO2 is cleared by the match of TSG3nCNTE with TSG3nCMP3E and set by the match with TSG3nCMP4E. Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for one cycle during PWM period when 1 is set, two cycles when 2 is set, and three cycles when 3 is set.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO2 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

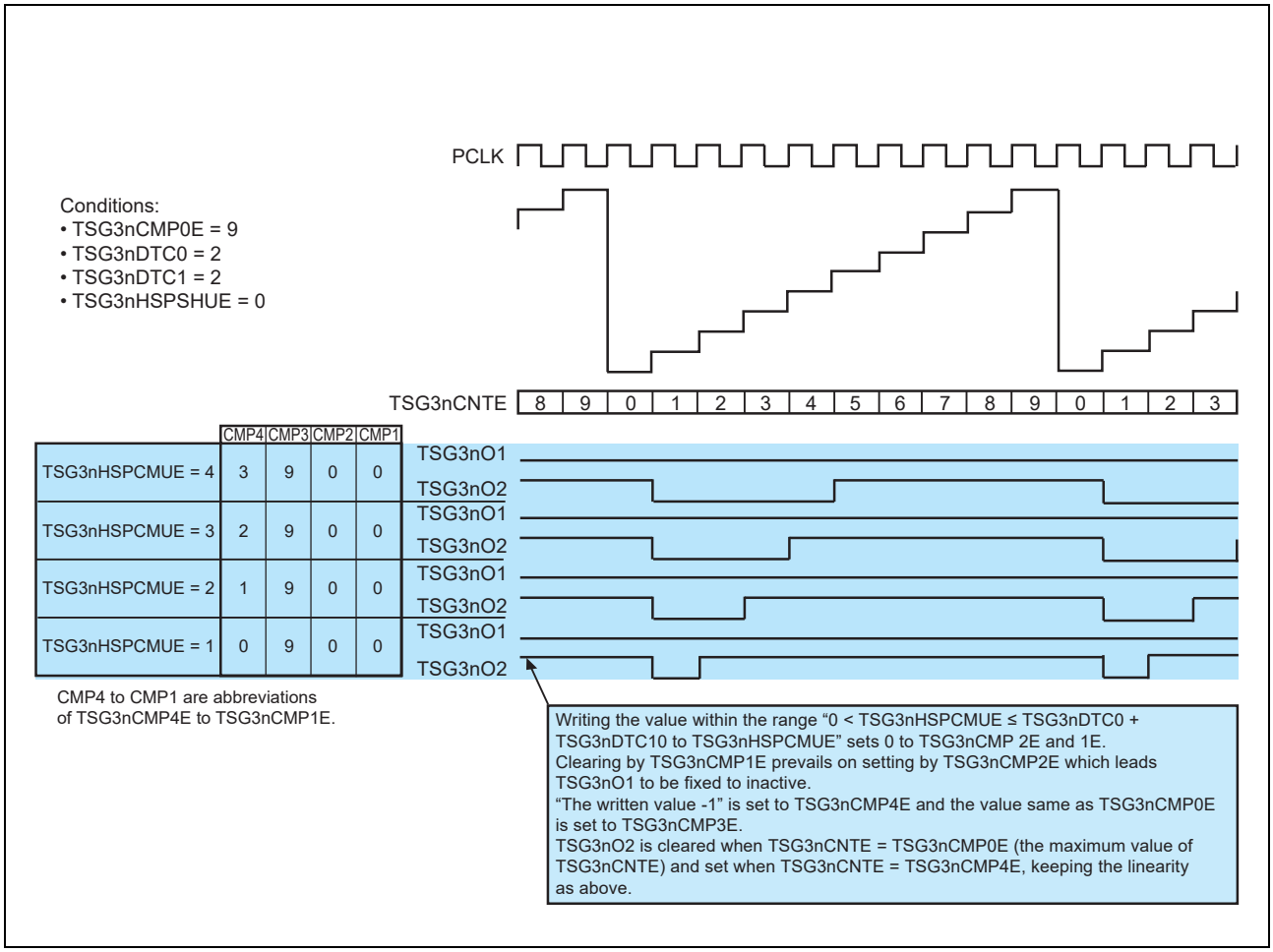


Figure 19.110 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to 0 < TSG3nHSPCMUE ≤ TSG3nDTC0 + TSG3nDTC1)

(3) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nDTC0 + TSG3nDTC1 <TSG3nHSPCMUE/VE/WE ≤ TSG3nCMP0E

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTCD0 -1” is set to TSG3nCMP2E, “TSG3nHSPCMUE - TSG3nDTC1 - 1” is set to TSG3nCMP1E, “TSG3nHSPCMUE - 1” is set to TSG3nCMP4E, and the value same as TSG3nCMP0E is set to TSG3nCMP3E.

TSG3nO1 is set by the match of TSG3nCNTE with TSG3nCMP2E and cleared by the match with TSG3nCMP1E while TSG3nO2 is set by the match of TSG3nCNTE with TSG3nCMP4E and cleared by the match with TSG3nCMP3E

Here, the outputs are shifted according to the set value in TSG3nHSPCMUE, that is, when “TSG3nDTC0 + TSG3nDTC1 + 1” is set, TSG3nO1 is active for one cycle during PWM period and TSG3nO2 is inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1+ 1” during PWM period. On the other hand, when “TSG3nDTC0 + TSG3nDTC1 + 2” is set, TSG3nO1 is active for two cycles during PWM period and TSG3nO2 is inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1+ 2” during PWM period.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 and TSG3nO2 are shifted for the number of cycles set in TSG3nHSPSHUE to the right.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

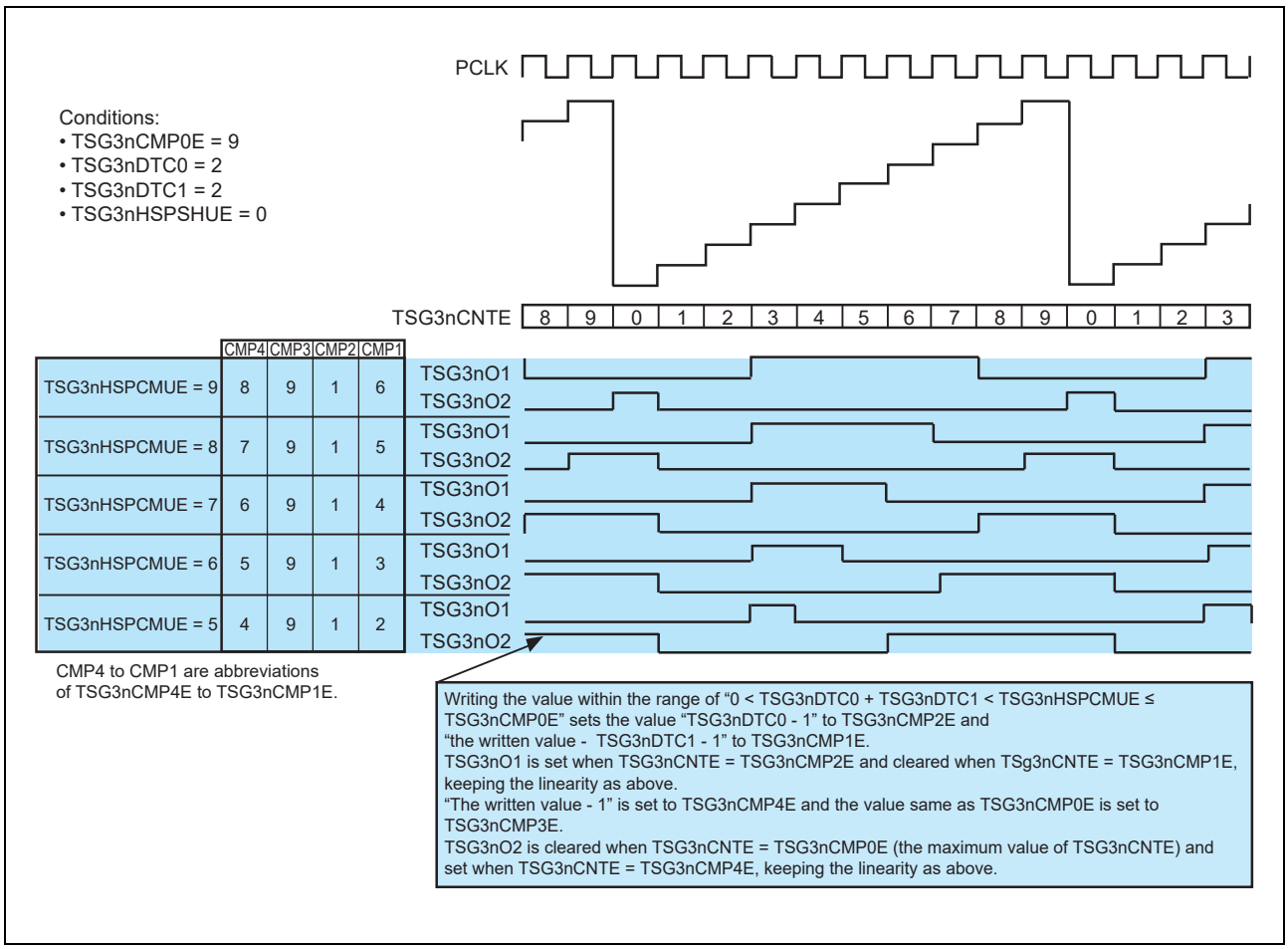


Figure 19.111 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nDTC0 + TSG3nDTC1 < TSG3nHSPCMUE ≤ TSG3nCMP0E)

(4) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E < TSG3nHSPCMUE/VE/ WE < TSG3nCMP0E + TSG3nDTC1 + 1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nHSPCMU - TSG3nDTC1 - 1” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

TSG3nO1 is cleared by the match of TSG3nCNTE with TSG3nCMP1E and set by the match with TSG3nCMP2E.

Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1” during PWM period when “TSG3nCMP0E + 1” is set, inactive for the cycles of “TSG3nDTC0 + TSG3nDTC1 - 1” during PWM period when “TSG3nCMP0E + 2” is set.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

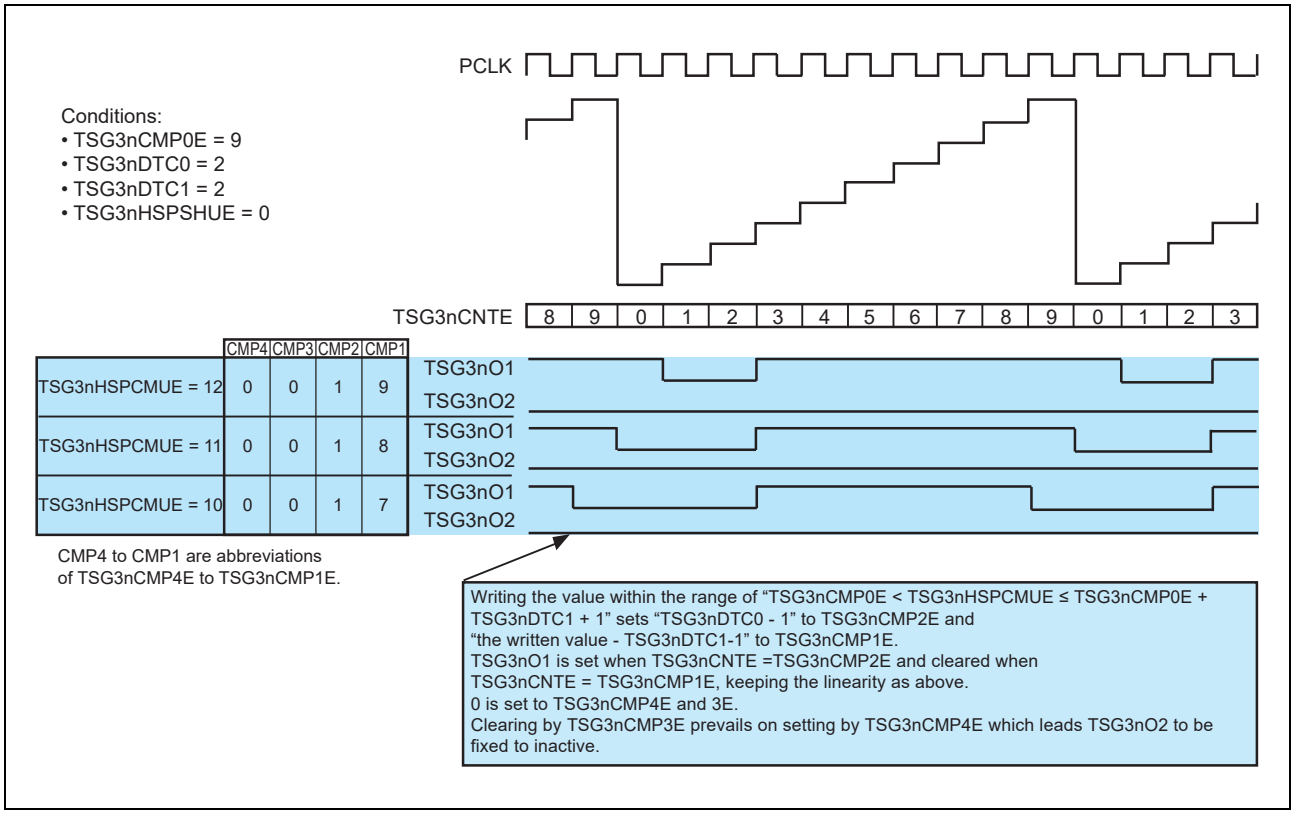


Figure 19.112 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nCMP0E < TSG3nHSPCMUE < TSG3nCMP0E + TSG3nDTC1 + 1)

(5) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E + TSG3nDTC1 + 1 < TSG3nHSPCMUE/VE/WE < TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

When the value within this range is set to TSG3nHSPCMUE (U phase PWM output width setting), “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nHSPCMUE - TSG3nCMP0E - TSG3nDTC1 - 2” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

TSG3nO1 is cleared by the match of TSG3nCNTE with TSG3nCMP1E and set by the match with TSG3nCMP2E.

Here, the output is shifted according to the set value in TSG3nHSPCMUE, that is, inactive for one cycle during PWM period when “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1” (the maximum value of PWM output width - 1) is set, inactive for two cycles during PWM period when “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 - 1” (the maximum value of PWM output width - 2) is set.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE (PWM shift width), set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

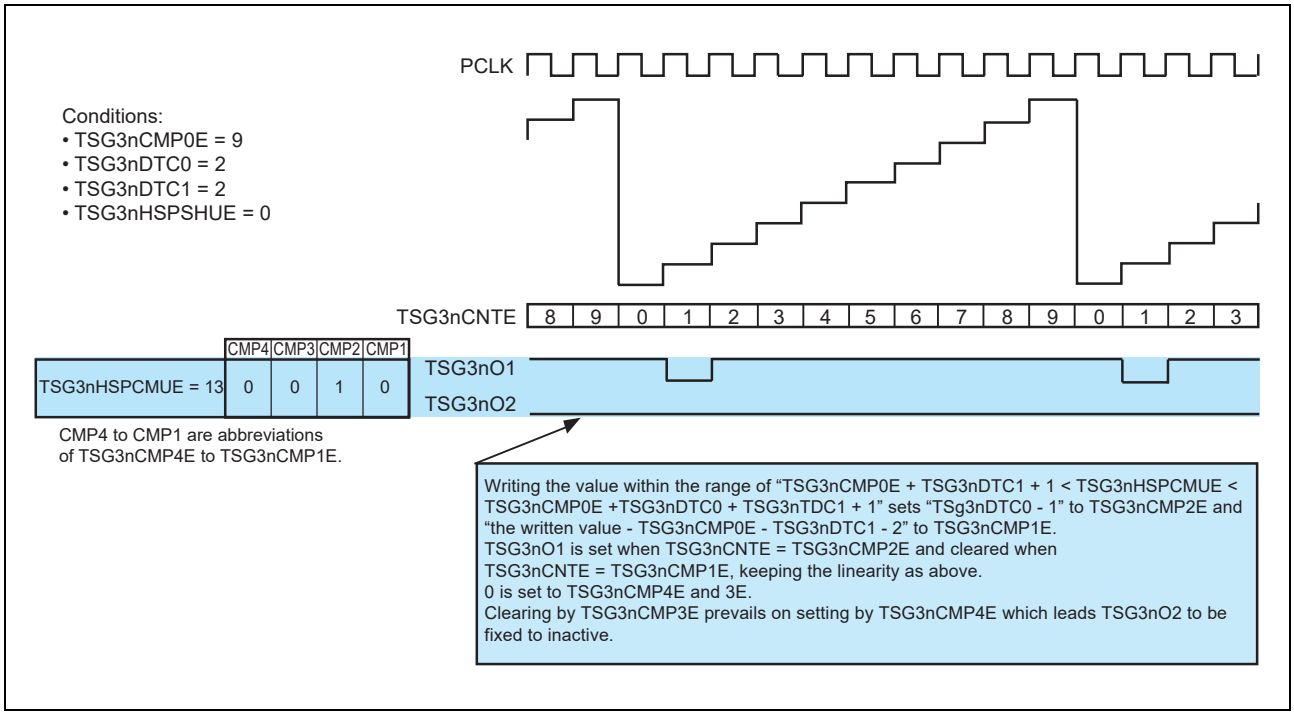


Figure 19.113 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nCMP0E + TSG3nDTC1 + 1 < TSG3nHSPCMUE < TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1)

(6) When TSG3nHSPCMUE/VE/WE (PWM Output Width Setting) set value is TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1

When “TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1” (the maximum value of PWM output width) is set to TSG3nHSPCMUE, “TSG3nDTC0 - 1” is set to TSG3nCMP2E, “TSG3nCMP0E + 1” is set to TSG3nCMP1E, and 0 is set to TSG3nCMP3E and 4E.

Setting of TSG3nO1 by the match of TSG3nCNTE with TSG3nCMP2E occurs when TSG3nCNTE = TSG3nDTC0 - 1.

“TSG3nCMP0E + 1” is set to TSG3nCMP3E. Here, with no match of TSG3nCNTE with TSG3nCMP3E occurs, TSG3nO1 is fixed to active.

Setting of TSG3nO2 by the match of TSG3nCNTE with TSG3nCMP4E and clearing by the match with TSG3nCMP3E occurs simultaneously. Here, clearing prevails on setting, therefore, TSG3nO2 is fixed to inactive.

When the value other than 0 is set to TSG3nHSPSHUE, set/clear timing of TSG3nO1 is shifted to the right for the number of cycles set in TSG3nHSPSHUE. Note that the set timing is shifted only at operation start (TSG3nTE = 0) because TSG3nO1 output is fixed to active. For the operation when operation start, see **Section 19.4.7.9**, **(8) Notes Concerning Dead Time Control in HT-PWM Mode**.

The same condition applies to TSG3nO3-6, which are the output pins for V phase and W phase when the value of above range is set to TSG3nHSPCMVE (V phase PWM output width setting) and TSG3nHSPCMWE (W phase PWM output width setting).

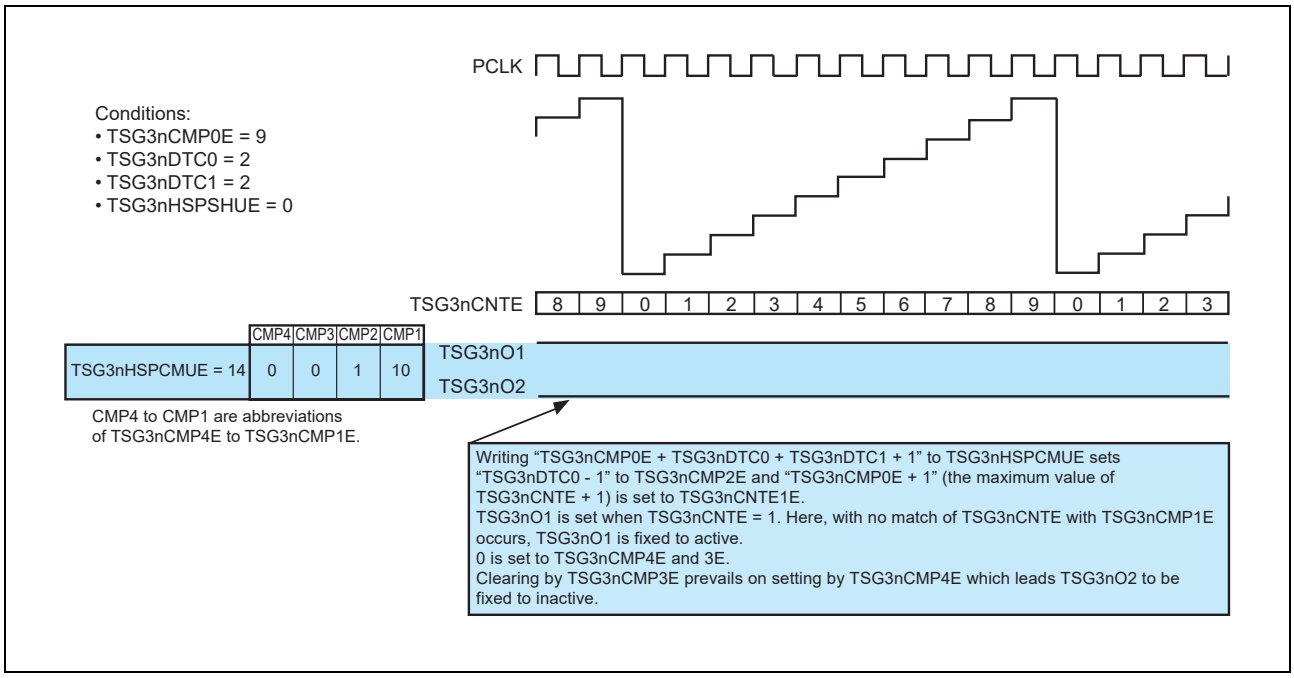


Figure 19.114 Waveform in HSP-PWM (TSG3nHSPCMUE is set to TSG3nCMP0E + TSG3nDTC0 + TSG3nDTC1 + 1)

NOTES

1. In HSP-PWM mode, TSG3nO1 and TSG3nO2 can be set at the same timing by setting the same value to TSG3nCMP4E and 2E. In this case, both TSG3nO1 and TSG3nO2 go inactive.
2. In HSP-PWM mode, dead time counter keeps operating even when TSG is stopped (TSG3nTE = 0) and the dead time set in TSG3nO1 and TSG3nO2 are always inserted.

(7) PWM Adjustment at Reload Timing in HSP-PWM Mode

In HSP-PWM mode, PWM output width is adjusted when TSG3nHSPCMUE/VE/WE (PWM output width setting) is modified during operation. TSG3nO1-6 are set/cleared at reload timing and immediately switched to the output reflecting the new PWM output width.

TSG3nO1-6 are forcibly set/cleared according to the following formula. Even for adjustment, the dead time set to TSG3nDTC0 and 1 is always inserted to TSG3nO1-6. If values are directly written to TSG3nCMP1-12E, output adjustment is not performed at reload timing.

Formula for reload adjustment operation

Table 19.100 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is 0 (PWM sift width is set to 0)

Pin	Set	Clear
TSG3nO1/3/5	CMP0E + DTC1 + 1 < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ CMP0E + DTC1 + 1
TSG3nO2/4/6	HSPCMUE/VE/WE = 0	0 < HSPCMUE/VE/WE

Table 19.101 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is not 0 (PWM sift width is not set to 0)

Pin	Set		Clear
TSG3nO1/3/5	(i)	CMP0E + DTC1 + 1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ CMP0E + DTC1 + 1 – HSPSHUE/VE/WE
	(ii)	(CMP0E + 1) × 2 + DTC1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE	HSPCMUE/VE/WE ≤ (CMP0E + 1) × 2 + DTC1 – HSPSHUE/VE/WE
TSG3nO2/4/6	HSPCMUE/VE/WE ≤ CMP0E + 1 – HSPSHUE/VE/WE		CMP0E + 1 – HSPSHUE/VE/WE < HSPCMUE/VE/WE

If the value other than 0 is set as PWM shift width, set/clear condition for positive phase is determined whether the set shift width is greater than “CMP0E + DTC0 – 1” or not.

(i) HSPSHUE/VE/WE ≤ CMP0 – DTC0 + 1

(ii) HSPSHUE/VE/WE > CMP0 – DTC0 + 1

“TSG3n” is omitted from the register names used in the calculation.

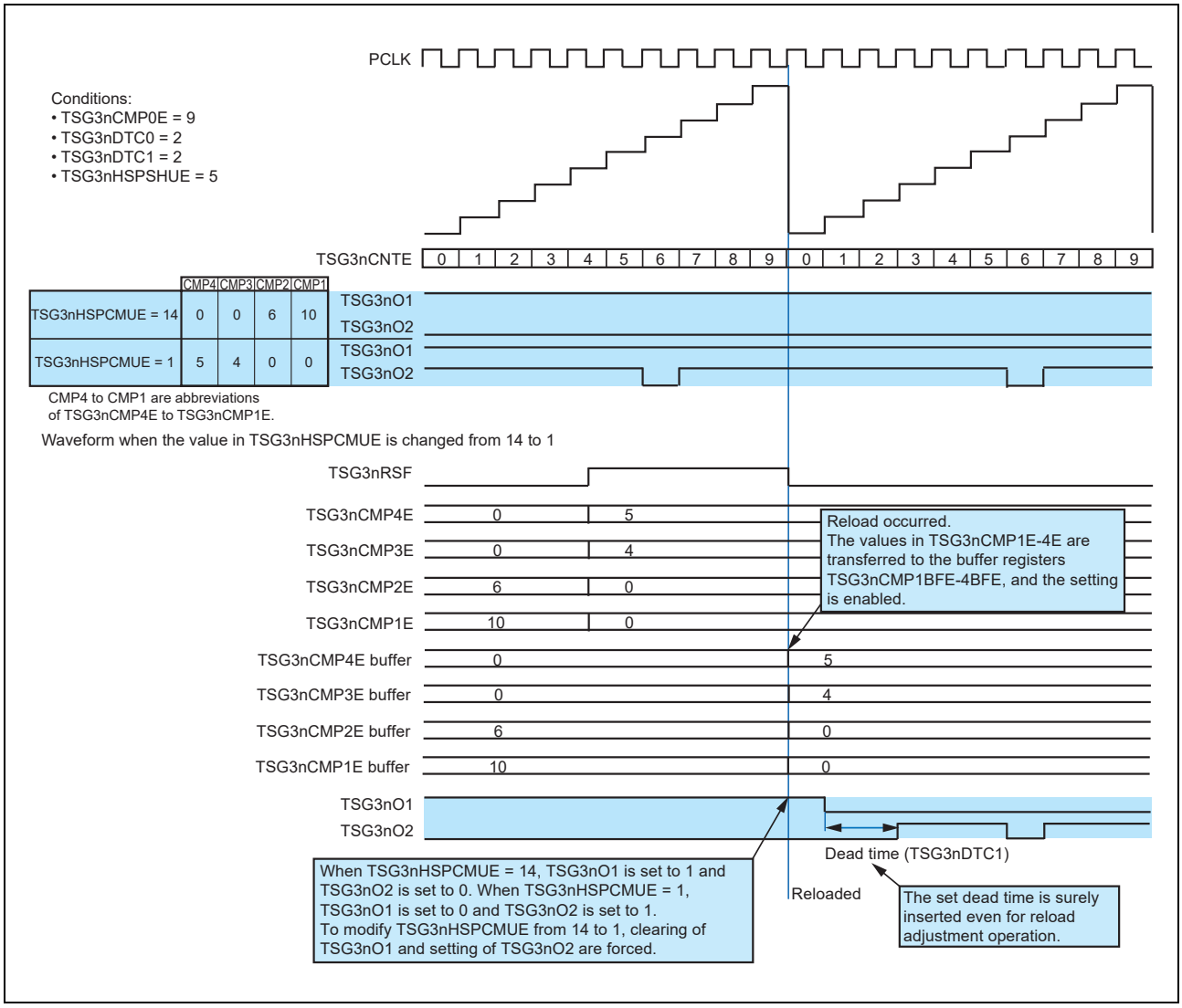


Figure 19.115 Detailed Timing Diagram of Reload Adjustment (Modification of TSG3nHSPCMUE from 14 to 1)

(8) TO Operation at Operation Start in HSP-PWM Mode

In HSP-PWM mode, TSG3nO1-6 are cleared when operation starts.

Then, TSG3nO1-6 are set/cleared as TSG3nCNTE counts up depending on the values set in TSG3nHSPCMUE/VE/WE (TSG3nCMP1E to 12E).

Even if TSG3nO1-6 are set before operation start and cleared at operation start, and then set again, the set dead time is always inserted.

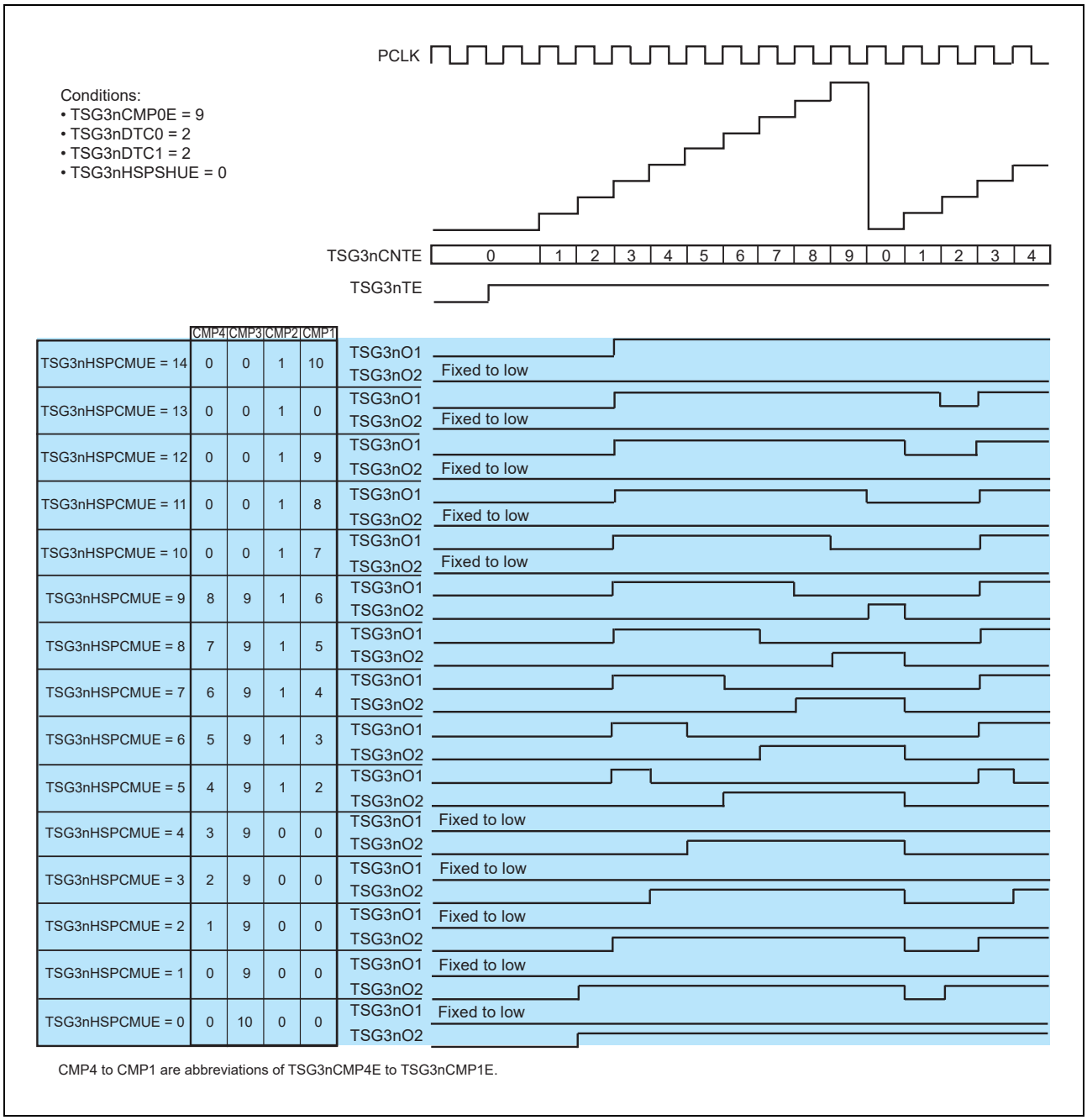


Figure 19.116 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 0 (shift width is set to 0))

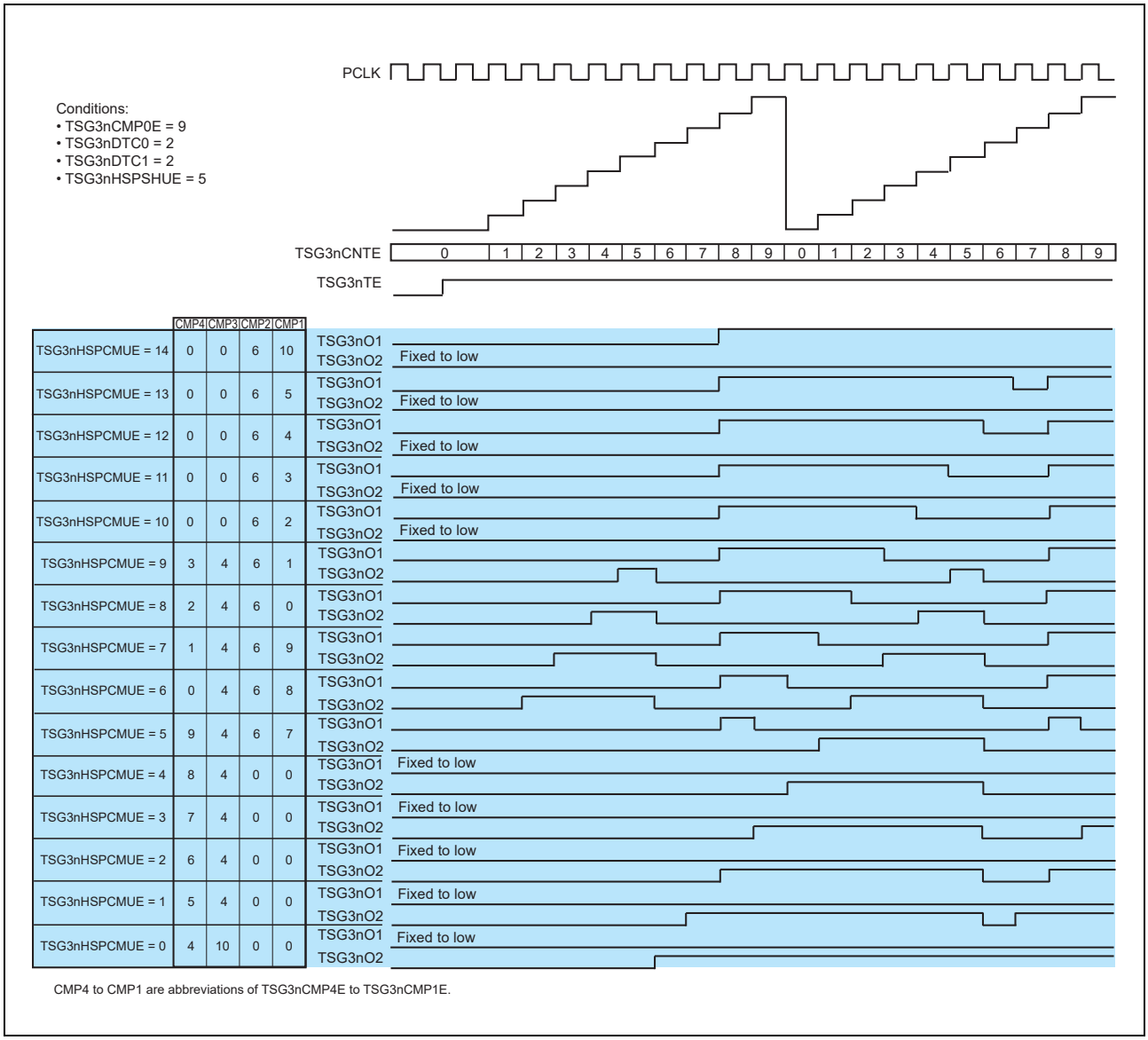


Figure 19.117 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 5 (shift width is set to five clock cycles))

19.4.7.10 Software Output Control Function

Software output control function is available in any mode except HSP-PWM mode. This function can switch six output patterns for the TSG3nO1 to TSG3nO6 pins using TSG3nOPT0.TSG3nSOC, TSG3nIDC, and TSG3nOPT0.TSG3nSPC2-0.

When TSG3nSOC is switched from 0 to 1, the output control method of the TSG3nO1 to TSG3nO6 pins is switched to the software output control immediately. On the contrast, when TSG3nSOC is switched from 1 to 0, the software output control is released at the reload timing.

Table 19.102 Registers Associated with Software Output Control Function

Register	Operation
TSG3nOPT0.TSG3nSOC	TSG3nSOC = 1
TSG3nOPT0.TSG3nSTE	TSG3nSTE = 0
TSG3nOPT1.TSG3nSPC2-TSG3nSPC0	Sets output patterns listed in the following **Table 19.93** and **Table 19.94**.
TSG3nOPT0.TSG3nIDC	Sets output pattern (electric current direction).

Table 19.103 Output Patterns by Software Output Control (TSG3nOPT0.TSG3nIDC = 0)

TSG3nOPT0.TSG3nSOC = 1, TSG3nSTE = 0, TSG3nIDC = 0

	TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
output Pin	101	100	110	010	011	001	000	111
TSG3nO1	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSG3nO2	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSG3nO3	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSG3nO4	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSG3nO5	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT
TSG3nO6	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT

Note: ACT: Indicates active level is output.
INACT: Indicates inactive level is output.

Table 19.104 Output Patterns by Software Output Control (TSG3nOPT0.TSG3nIDC = 1)

TSG3nOPT0.TSG3nSOC = 1, TSG3nSTE = 0, TSG3nIDC = 1

	TSG3nSTR1.TSG3nOPF2-TSG3nOPF0							
output Pins	101	100	110	010	011	001	000	111
TSG3nO1	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSG3nO2	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSG3nO3	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSG3nO4	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSG3nO5	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT
TSG3nO6	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT

Note: ACT: Indicates active level is output.
INACT: Indicates inactive level is output.

Section 20 Timer Option Module (TAPA)

This section contains a generic description of the timer option module (TAPA).

The first part of this section describes RH850/C1x specific properties, such as the number of units, register base addresses, etc. The remainder of this section describes TAPA functions and registers.

20.1 Features of RH850/C1x TAPA

20.1.1 Units

This LSI has the following number of TAPA units.

Table 20.1 Units

Product	RH850/C1x
Number of units	4
Name	TAPAn (n = 0 to 3)

Table 20.2 Index

Index	Meaning
n	Throughout this section, the individual TAPA units are identified by the index “n” (n = 0 to 3); for example, TAPAnFLG is the TAPAn flag register.

20.1.2 Register Base Address

TAPA base addresses are listed in the table below.

TAPA register addresses are given as offsets from the base addresses in general.

Table 20.3 Register Base Address

Base Address Name	Base Address
<TAPA0_base>	FFE9 0000_H
<TAPA1_base>	FFE9 1000_H
<TAPA2_base>	FFE9 2000_H
<TAPA3_base>	FFE9 3000_H

20.1.3 Clock Supply

Clock supply by and to TAPA is listed in the following table.

Table 20.4 Clock Supply

Unit Name	Clock for the Unit	Internal Clock Name
TAPAn	PCLK	CLKC_HSB (unmodulated high-speed peripheral clock)

20.1.4 Interrupt Requests

TAPA interrupt requests are listed in the following table.

Table 20.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMA Trigger Number	DTS Trigger Number
TAPA0				
TAPA0TIPEK0	TAPA0 peak interrupt	126	40	50
TAPA0TIVLY0	TAPA0 trough interrupt	127	41	51
TAPA1				
TAPA1TIPEK0	TAPA1 peak interrupt	128	42	52
TAPA1TIVLY0	TAPA1 trough interrupt	129	43	53

20.1.5 Reset Sources

TAPA reset sources are listed in the following table.

Table 20.6 Reset Sources

Unit Name	Reset Source
TAPAn	Any reset source

20.1.6 Peripheral Configuration

The following figure shows the peripheral configuration of TAPA.

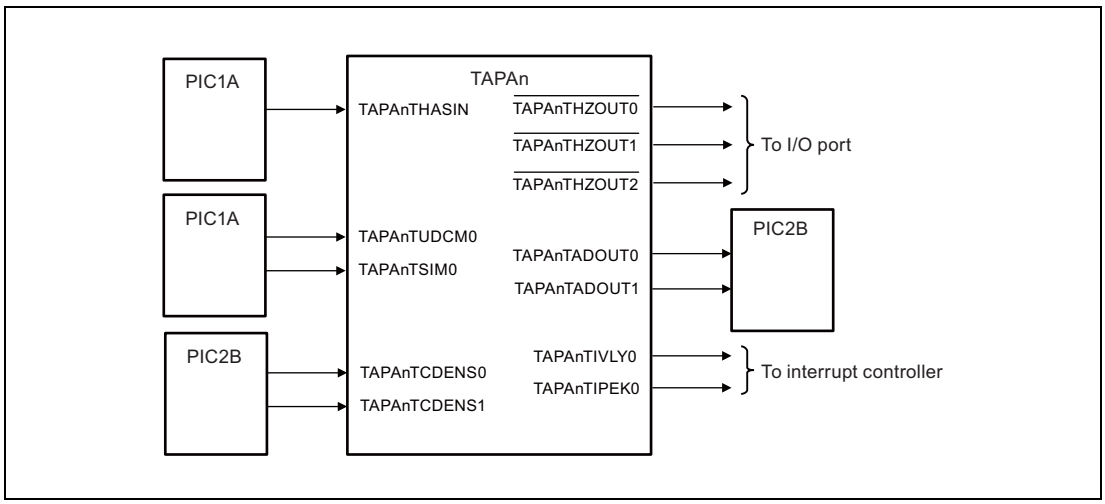


Figure 20.1 Peripheral Configuration of TAPA

The following describes the peripheral configuration of TAPA.

- TAPAnTHASIN: Hi-Z control asynchronous input signal
  This signal controls Hi-Z by the source selected in PIC1A.
  For the sources that can be selected in PIC1A, see **23.2.3.15, Hi-Z Control Function**.
- TAPAnTUDCM0: TAUD master channel up/down input
- TAPAnTSIM0: TAUD master channel INT input
  Peak and trough interrupts can be generated on TAUDn channels selected in PIC1A.
  For TAUDn channels that can be selected in PIC1A, see **23.2.2.22, PIC1AREG200 — Timer Input/Output Control Register 200** and **23.2.2.23, PIC1AREG210 — Timer Input/Output Control Register 210**.
  Only TAPA0 and TAPA1 have this connection.
- TAPAnTCDENS0, TAPAnTCDENS1: TAUD slave channel match detection input
  The TAUDn channel interrupts selected in PIC2B can be handled as AD conversion trigger outputs 0 and 1 (TAPAnTADOUT1 and TAPAnTADOUT0). For the TAUDn channel interrupts that can be selected in PIC, see **23.3.3.2, TAUD Trigger Output Function**.
  Only TAPA0 and TAPA1 have this connection.
- TAPAnTADOUT1 and TAPAnTADOUT0: A/D conversion trigger outputs 1 and 0
  The A/D conversion trigger signals generated on TAPAn are output to PIC2B. The setting of the PIC2B can also be used to set these signals as triggers for A/D conversion.
  For the register specifications in the PIC2B, see **23.3.3.1, ADCC Trigger Select Function**.
  Only TAPA0 and TAPA1 have this connection.

20.2 Overview

20.2.1 Functional Overview

The timer option module (TAPA) is for use with the timer array unit D (TAUD) and TSG3 modules.

- Asynchronous Hi-Z control to each TAUD and TSG3 output is enabled by TAPA input signals.
- Output of the INTn signal from the TAUD as a peak or trough interrupt is selectable.
- The INTn signal output by the TAUD provides the basis for the output of two conversion-trigger signals for the A/D converter.

20.2.2 Terms

In this section, the following terms are used.

Peak and Trough, and Peak and Trough Interrupts

In this document, the period from a TAUD down status (counting-down status) to generation of INT from the master channel is defined as a trough period, and this INT is defined as a trough interrupt (INT-VLY).

In contrast, the period from a TAUD up status (counting-up status) to generation of INT from the master channel is defined as a peak period, and this INT is defined as a peak interrupt (INT-PEK).

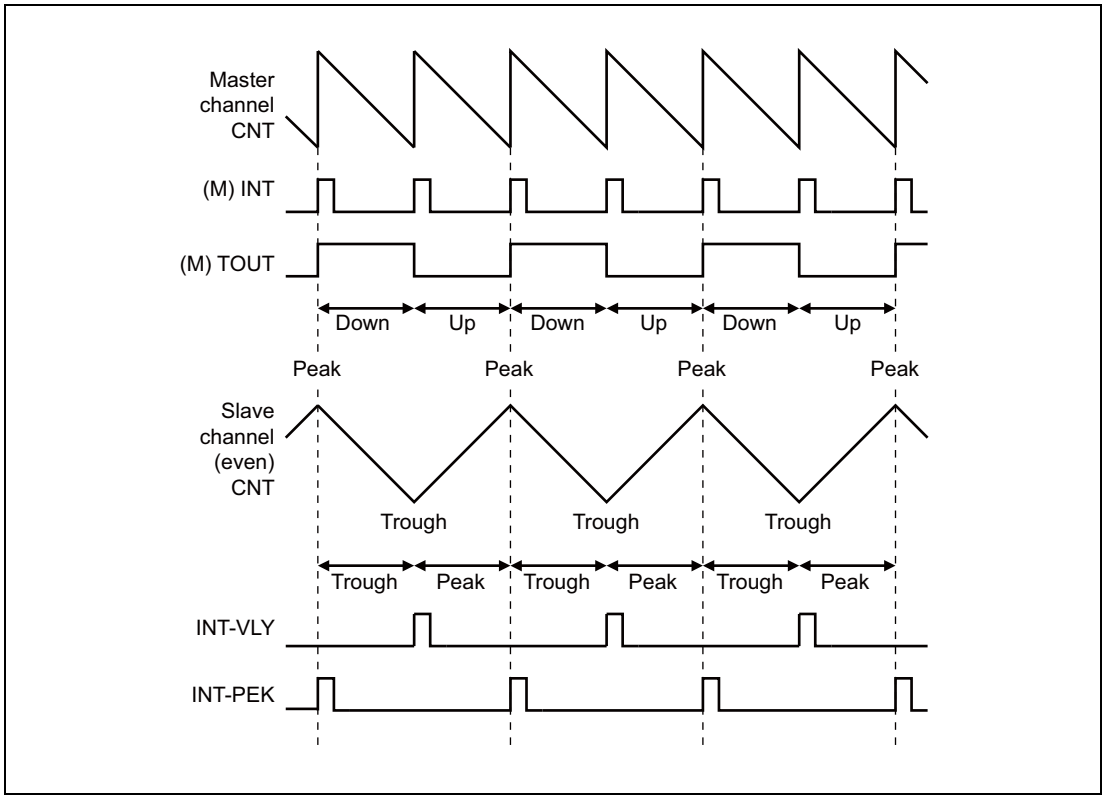


Figure 20.2 Peak and Trough of Timer Counter, and Peak and Trough Interrupts

20.2.3 Block Diagram

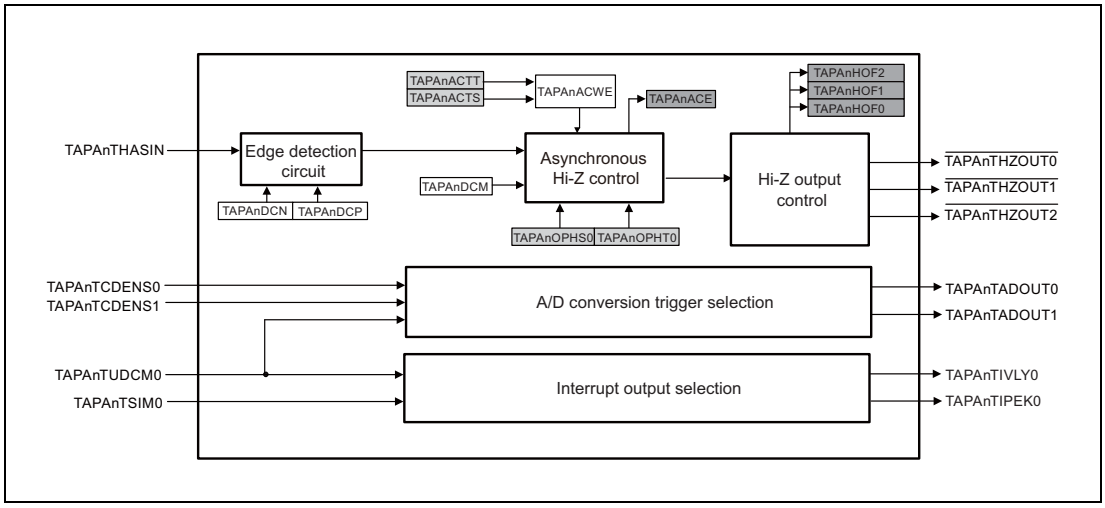


Figure 20.3 Block Diagram

20.3 Registers

20.3.1 Registers Overview

TAPAn (n = 0 to 3) registers are listed in the following table.

For information on <TAPAn_base>, see **Section 20.1.2, Register Base Address**.

Module Name	Register Name	Symbol	Address
TAPAn	TAPAn control register 0	TAPAnCTL0	<TAPAn_base> + 20_H
TAPAn	TAPAn control register 1*[1]	TAPAnCTL1	<TAPAn_base> + 24_H
TAPAn	TAPAn flag register	TAPAnFLG	<TAPAn_base> + 00_H
TAPAn	TAPAn asynchronous control write enable register	TAPAnACWE	<TAPAn_base> + 04_H
TAPAn	TAPAn asynchronous control start trigger register	TAPAnACTS	<TAPAn_base> + 08_H
TAPAn	TAPAn asynchronous control stop trigger register	TAPAnACTT	<TAPAn_base> + $0C_H$
TAPAn	TAPAn Hi-Z start trigger register	TAPAnOPHS	<TAPAn_base> + 14_H
TAPAn	TAPAn Hi-Z stop trigger register	TAPAnOPHT	<TAPAn_base> + 18_H

Note 1. TAPAnCTL1 is valid only when TAPAn(n = 0, 1).

20.3.2 TAPAnCTL0 — TAPAn Control Register 0

Control register 0 is used to control Hi-Z.

A value in this register can only be rewritten in the following conditions.

- TAPAnFLG.TAPAnACE = 0 while TAPAn (n = 0, 1) and TAUDnTEm = 0 (m = 10 to 15) at the corresponding TAUDn master channel.
- TAPAnFLG.TAPAnACE = 0 while TAPAn (n = 2, 3).

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 20_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TAPAn DCM	TAPAn DCN	TAPAn DCP	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Table 20.7 TAPAnCTL0 Register Contents

Bit Position	Bit Name	Function
4	TAPAnDCM	Clearing Condition Specification This control bit specifies the condition for clearing of the Hi-Z control outputs. 0: Manipulation of TAPAnOPHT0 is enabled regardless of the TAPAnTHASIN signal input level. 1: Manipulation of TAPAnOPHT0 is disabled when the TAPAnTHASIN input signal is at the active level. Manipulation of TAPAnOPHT0 is enabled when the TAPAnTHASIN signal input is inactive.
3, 2	TAPAnDCN, TAPAnDCP	Hi-Z Input Edge Selection These control bits specify the effective edge of TAPAnTHASIN.

TAPAnDCN	TAPAnDCP	Description
0	0	Does not detect effective edges
0	1	Detects a rising edge as the effective edge (active level = high)
1	0	Detects a falling edge as the effective edge (active level = low)
1	1	Setting is prohibited.

20.3.3 TAPAnCTL1 — TAPAn Control Register 1

TAPAn control register 1 is only valid when n = 0 or 1.

Access: This register can be read/written in 8-bit units.

Address: <TAPAn_base> + 24_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TAPAnATS3	TAPAnATS2	TAPAnATS1	TAPAnATS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 20.8 TAPAnCTL1 Register Contents

Bit Position	Bit Name	Function
3, 2	TAPAn ATS[3:2]	A/D Converter Trigger 1 Select These control bits specify the signal for output as A/D converter conversion trigger output 1 (TAPAnTADOUT1).

TAPA0ATS3	TAPA0ATS2	Description
0	0	Outputs INT while the master channel is in the down state.
0	1	Outputs INT while the master channel is in the up state.
1	0	Outputs INT while the master channel is in the up/down state.
1	1	Outputs INT and a trough interrupt of the master channel (TAPAnTIVLYn) while the master channel is in the up/down state.

Bit Position	Bit Name	Function
1, 0	TAPAn ATS[1:0]	A/D Converter Trigger 0 Select These control bits specify the signal for output as A/D converter conversion trigger output 0 (TAPAnTADOUT0).

TAPA0ATS1	TAPA0ATS0	Description
0	0	Outputs INT while the master channel is in the down state.
0	1	Outputs INT while the master channel is in the up state.
1	0	Outputs INT while the master channel is in the up/down state.
1	1	Outputs INT and a trough interrupt of the master channel (TAPAnTIVLYn) while the master channel is in the up/down state.

20.3.4 TAPAnFLG — TAPAn Flag Register

Control register is used to control Hi-Z.

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 00_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TAPAn HOF2	TAPAn HOF1	TAPAn HOF0	—	—	—	—	—	—	—	TAPAn ACE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 20.9 TAPAnFLG Register Contents

Bit Position	Bit Name	Function
10 to 8	TAPAnHOFm	$\overline{\text{TAPAnTHZOUTm}}$ Output Monitor (m = 0 to 2) These bits monitor the $\overline{\text{TAPAnTHZOUTm}}$ output. 0: The $\overline{\text{TAPAnTHZOUTm}}$ output is at the high level. 1: The $\overline{\text{TAPAnTHZOUTm}}$ output is at the low level.*1
0	TAPAnACE	Asynchronous Hi-Z Control Enable This bit indicates the state of asynchronous Hi-Z control. 0: Asynchronous Hi-Z control is stopped. 1: Asynchronous Hi-Z control is enabled. The conditions for setting and clearing this bit are as follows. Clearing condition: Writing 1 to TAPAnACTT while TAPAnACWE = 1. Setting condition: Writing 1 to TAPAnACTS while TAPAnACWE = 1.

Note 1. TAPAnHOFm (m=1, 2) is valid only when TAPAn(n = 0, 1).

20.3.5 TAPAnACWE — TAPAn Asynchronous Control Write Enable Register

This register enables writing for asynchronous Hi-Z control.

Access: This register can be read/written in 8-bit units.

Address: <TAPAn_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACWE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 20.10 TAPAnACWE Register Contents

Bit Position	Bit Name	Function
0	TAPAnACWE	Asynchronous Control Write Enable This is a write-enable bit for asynchronous Hi-Z control. After 1 has been written to this bit, it is automatically cleared to 0 by writing 1 to TAPA0ACTS or TAPA0ACTT. 0: Disables writing to TAPAnACTS and TAPAnACTT. 1: Enables writing to TAPAnACTS and TAPAnACTT.

20.3.6 TAPAnACTS — TAPAn Asynchronous Control Start Trigger Register

This register enables the start trigger for asynchronous Hi-Z control.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.11 TAPAnACTS Register Contents

Bit Position	Bit Name	Function
0	TAPAnACTS	Asynchronous Control Start Trigger This bit enables the start trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Enables asynchronous Hi-Z control if TAPAnACWE = 1.

20.3.7 TAPAnACTT — TAPAn Asynchronous Control Stop Trigger Register

This register enables the stop trigger for asynchronous Hi-Z control.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnACTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.12 TAPAnACTT Register Contents

Bit Position	Bit Name	Function
0	TAPAnACTT	Asynchronous Control Stop Trigger This bit enables the stop trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Stops asynchronous Hi-Z control if TAPAnACWE = 1.

20.3.8 TAPAnOPHS — TAPAn Hi-Z Start Trigger Register

This register sets the start trigger for a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*1)).

Note 1. TAPAn (n = 2, 3) are not available when m = 1, 2.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnOPHS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.13 TAPAnOPHS Register Contents

Bit Position	Bit Name	Function
0	TAPAnRSF0	Hi-Z Control Signal Start Trigger 0 This bit sets the start trigger for a Hi-Z control signal. 0: The read value is 0 and writing 0 to this bit is ignored (no function). 1: Sets the corresponding Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*1)) to the low level.

20.3.9 TAPAnOPHT — TAPAn Hi-Z Stop Trigger Register

This register sets the start trigger for a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*1)).

Note 1. TAPAn (n = 2, 3) are not available when m = 1, 2.

Access: This register can be written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAnOPHT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.14 TAPAnOPHT Register Contents

Bit Position	Bit Name	Function
0	TAPAnRDT0	Hi-Z Control Signal Stop Trigger 0 This bit sets the stop trigger for a Hi-Z control signal. 0: The read value is 0 and writing 0 to this bit is ignored (no function). 1: Sets a Hi-Z control signal ($\overline{\text{TAPAnTHZOUTm}}$ (m = 0 to 2*1)) to the high level.

20.4 Function

20.4.1 Asynchronous Hi-Z Control Functions

Abnormal operation in timer motor-control under CPU control leads to rotation of the externally connected motor also becoming abnormal. In such a case, this function forcibly sets the motor control output to the Hi-Z state, independently of control by the CPU.

20.4.1.1 Overview

The following method is available for controlling Hi-Z.

- Asynchronous Hi-Z control for TAPA input signals (TAPAnTHASIN)
  - Controls the Hi-Z control output signals $\overline{\text{TAPAnTHZOUT0}}$ (phase U), $\overline{\text{TAPAnTHZOUT1}}$ (V phase), $\overline{\text{TAPAnTHZOUT2}}$ (W phase) asynchronously at TAPAn (n = 0, 1).
  - Controls the Hi-Z control output signals $\overline{\text{TAPAnTHZOUT0}}$ asynchronously at TAPAn (n = 2, 3).

Asynchronous Hi-Z Control and Its Operation

Hi-Z Control	Function and Operation
Asynchronous Hi-Z control for TAPA input signals (TAPAnTHASIN)	Asynchronous Hi-Z control This function forcibly places the output from the corresponding timer module (TAUD, TSG3) into Hi-Z. Device port outputs become Hi-Z while TAPAnTHASIN is active and until software sends a stop request (when TAPAnCTL0.TAPAnDCM = 0)

NOTE

The following timer output pins are controlled by this function:

■ TAPAn (n = 0, 1)

- TAPAnTHZOUT0 (U-phase): TAPAnUP, TAPAnUN
- TAPAnTHZOUT1 (V-phase): TAPAnVP, TAPAnVN
- TAPAnTHZOUT2 (W-phase): TAPAnWP, TAPAnWN

■ TAPAn (n = 2)

- TAPAnTHZOUT0: TSG30O1 to TSG30O6

■ TAPAn (n = 3)

- TAPAnTHZOUT0: TSG31O1 to TSG31O6

20.4.1.2 An Example of System Configuration

When effective edges of the external error detection signal are detected, an interrupt is generated and, at the same time, the motor-driving signal output is set to the Hi-Z state.

This module assumes that microcontroller operation may hang when an error occurs. To handle such situations, external error detection signals are continuously processed so that the motor-driving signal can be set to the Hi-Z state even if no clock signal is being supplied.

This module only detects an error as an edge of the error-detection signal. A fixed output level is not detected as an error (the signal has no edge).

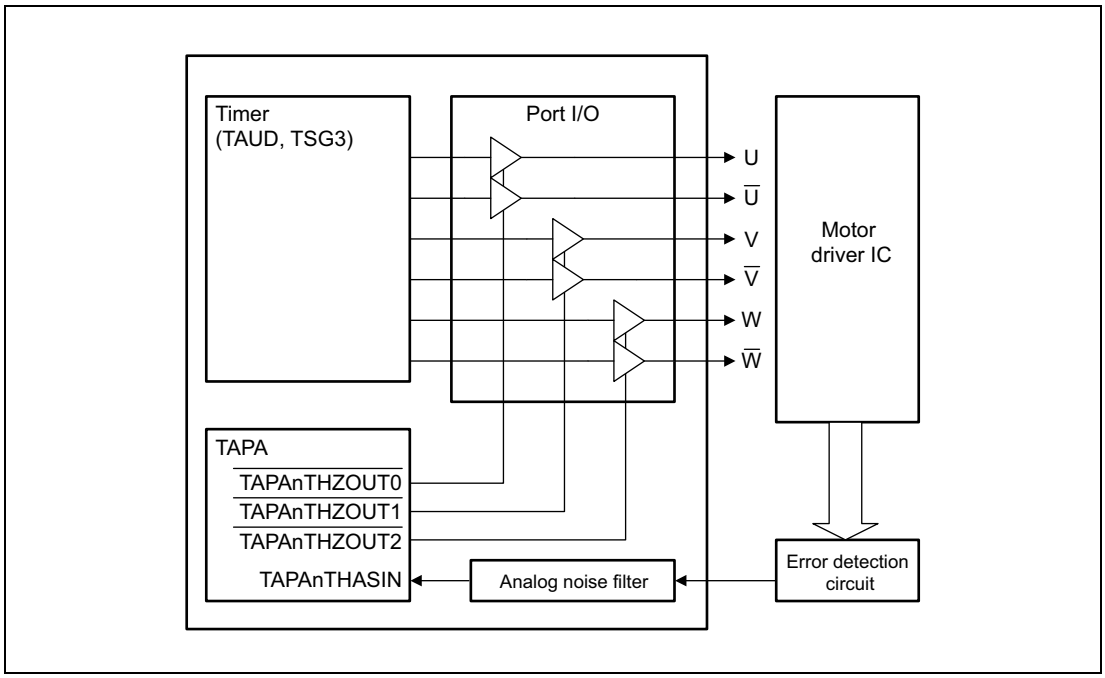


Figure 20.4 An Example of System Configuration of Asynchronous Hi-Z Control for Analog Inputs

20.4.1.3 Basic Operation

Setting examples are described as follows.

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 0, TAPAnDCP = 0, and TAPAnDCN = 0

$\overline{\text{TAPAnTHZOUT0}}$ goes to the low level on detection of an effective edge of the asynchronous input (TAPAnTHASIN).

Output is forcibly stopped (by port control for Hi-Z output) as long as the $\overline{\text{TAPAnTHZOUT0}}$ output is at the low level.

$\overline{\text{TAPAnTHZOUT0}}$ goes to the high level in response to writing 1 to Hi-Z stop trigger 0 (TAPAnOPHT0), regardless of the level of TAPAnTHASIN.

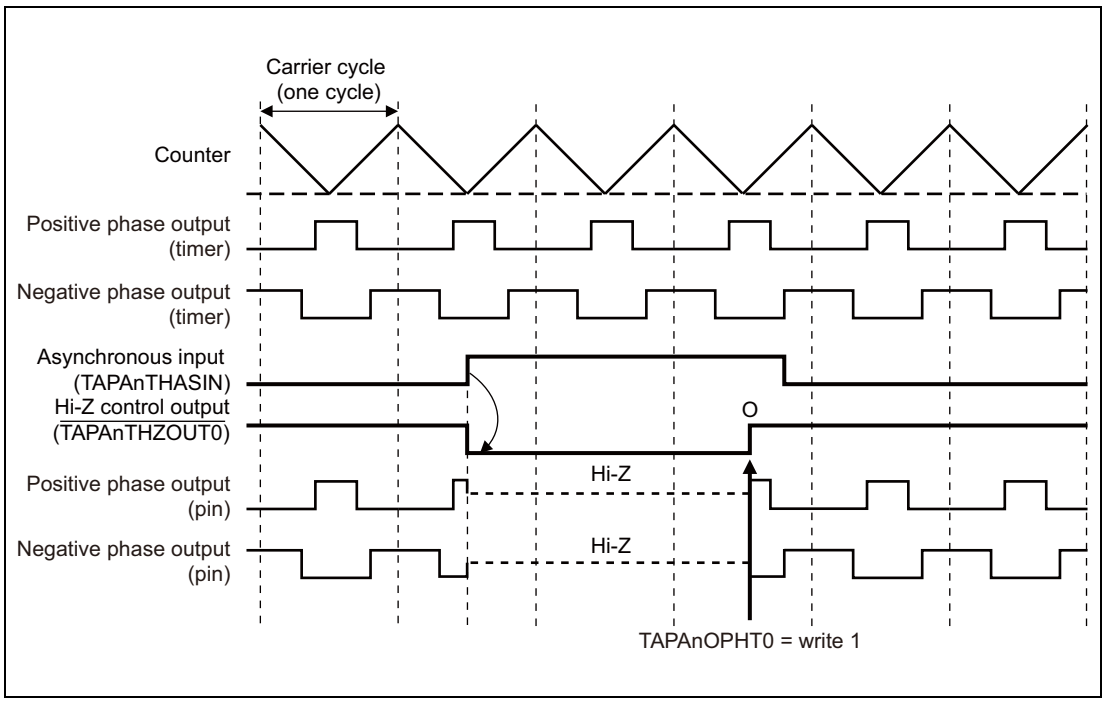


Figure 20.5 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM = 0, TAPAnDCP = 1, and TAPAnDCN = 0

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 1, TAPAnDCP = 1, TAPAnDCN = 0

$\overline{\text{TAPAnTHZOUT0}}$ goes to the low level in response to detection of an effective edge of the asynchronous input signal (TAPAnTHASIN).

Output is forcibly stopped (by port control for Hi-Z output) as long as the $\overline{\text{TAPAnTHZOUT0}}$ output is at the low level.

Writing of 1 to Hi-Z stop trigger 0 (TAPAnOPHT0) is ignored as long as the asynchronous input signal (TAPAnTHASIN) is at the active level (high because TAPAnDCP = 1).

After the asynchronous input signal (TAPAnTHASIN) is switched to the inactive level (low because TAPAnDCP = 1), $\overline{\text{TAPAnTHZOUT0}}$ goes to the high level when 1 is written to Hi-Z stop trigger 0 (TAPAnOPHT0).

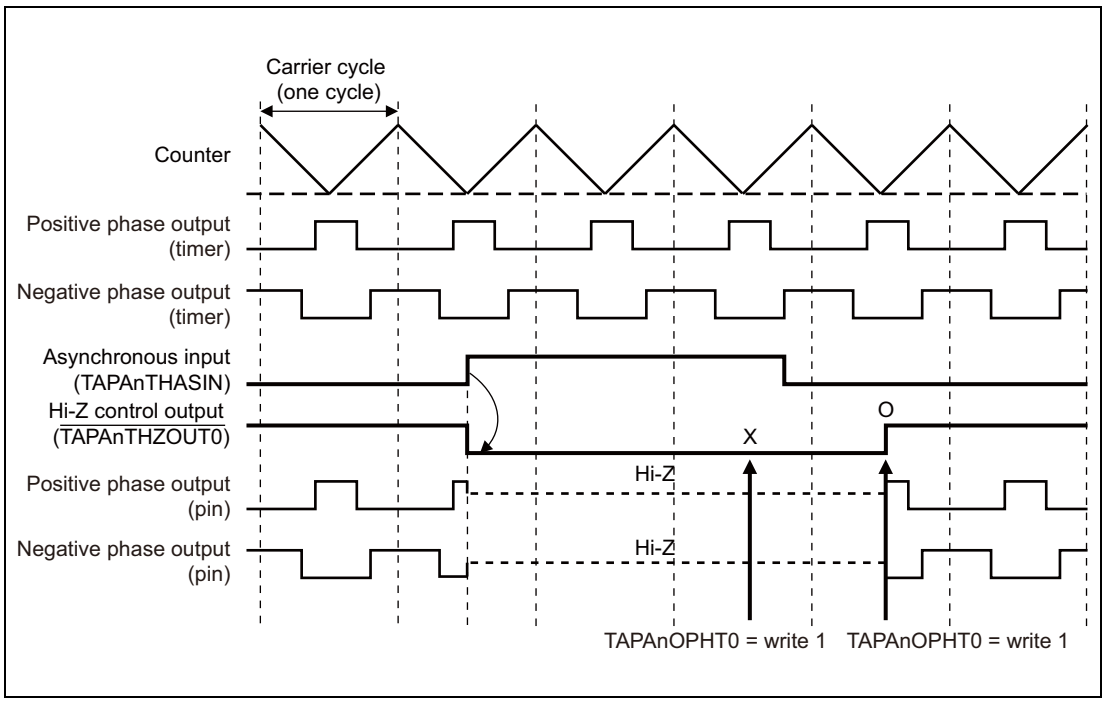


Figure 20.6 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM = 1, TAPAnDCP = 1, and TAPAnDCN = 0

20.4.1.4 Asynchronous Hi-Z Control Using Software Trigger

This module allows software control of the output of Hi-Z control signals.

Hi-Z start trigger 0 (TAPAnOPHS0) and Hi-Z stop trigger 0 (TAPAnOPHT0) are used to control $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$*1, and $\overline{\text{TAPAnTHZOUT2}}$*1.

Note 1. Not available when TAPAn (n = 2, 3).

Operation of the Hi-Z Start Trigger (TAPAnOPHS)

TAPAnDCM	Operation
0/1	Writing 1 to the TAPAnOPHS0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ and $\overline{\text{TAPAnTHZOUT2}}$ signals at the low level.

Operation of the Hi-Z Stop Trigger (TAPAnOPHT) during Hi-Z Control in Response to Asynchronous Input

The Hi-Z stop trigger operates as follows.

TAPAnDCM	Operation
0	Writing 1 to the TAPAnOPHT0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ and $\overline{\text{TAPAnTHZOUT2}}$ signals at the high level.
1	If TAPAnTHASIN is at the inactive level, writing 1 to the TAPAnOPHT0 bit places the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$ and $\overline{\text{TAPAnTHZOUT2}}$ signals at the high level. If TAPAnTHASIN is at the active level, writing of 1 to the TAPAnOPHT0 bit is ignored.

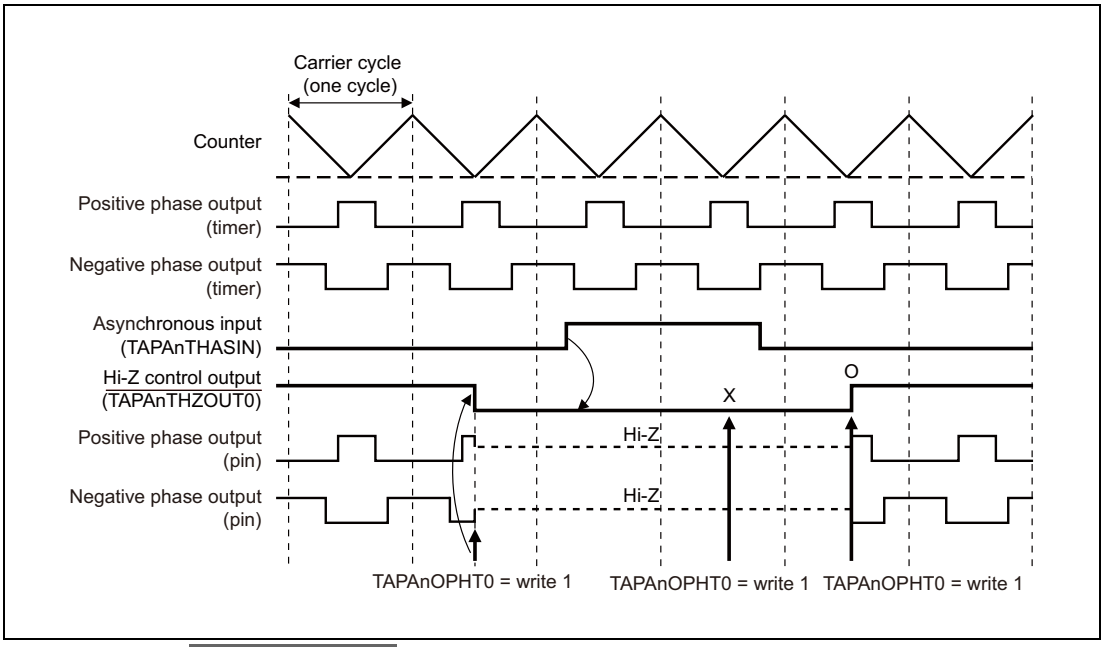


Figure 20.7 $\overline{\text{TAPAnTHZOUT0}}$ Operation when TAPAnDCM = 1, TAPAnDCP = 1, and TAPAnDCN = 0

20.4.1.5 Operating Procedure

An example of the operating procedure for Hi-Z control in response to asynchronous input is as follows (the table only covers settings for the timer option module because this operation does not depend on timer operations).

	Operation	State of TAPA
Initial settings	Setting in the TAPAnCTL0 register. Set TAPAnDCP and TAPAnDCN (input edge selection). Set TAPAnDCM (clearing mode selection).	Hi-Z control in response to asynchronous input is stopped (TAPAnFLG.TAPAnACE = 0).
Starting operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTS register: Set the TAPAnACTS bit to 1.	Writing to the TAPAnASTS bit is enabled. TAPAnFLG.TAPAnACE = 1, enabling Hi-Z control in response to asynchronous input.
During operation	To start Hi-Z control of an output from the timer: - Control is by the TAPAnOPHS0 bit of TAPA - Control is by the Hi-Z input signal (TAPAnTHASIN) for TAPA To stop Hi-Z control of output from the timer: - Control is by the TAPAnOPHT0 bit of TAPA (if TAPAnDCM = 0) - TAPAnOPHT0 is used if the Hi-Z input signal for TAPA (TAPAnTHASIN) is at the inactive level (if TAPAnDCM = 1) The state of TAPA operations can be read from the TAPAnFLG register at all times.	On detection of input of the starting edge of the Hi-Z input signal (TAPAnTHASIN) or setting of the start trigger bit (TAPAnOPHS0 = 1), the Hi-Z controller switches the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$*1 and $\overline{\text{TAPAnTHZOUT2}}$*1 pins to low-level output. In accord with the operating mode settings in TAPAnDCM, the Hi-Z controller switches the $\overline{\text{TAPAnTHZOUT0}}$, $\overline{\text{TAPAnTHZOUT1}}$*1 and $\overline{\text{TAPAnTHZOUT2}}$*1 pins to high-level outputs in response to setting of the stop trigger bit (TAPAnOPHT0 = 1).
Stopping operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTT register. Set the TAPAnACTT bit to 1.	Writing to the TAPAnACTT bit is enabled. TAPAnFLG.ACE = 0, stopping asynchronous Hi-Z control

Restart operation

Note 1. Not available when TAPAn (n = 2, 3).

20.4.2 Selection of INT Signal Output

20.4.2.1 Configuration

This function outputs of peak or trough interrupts by using the INT and TAPAnTUDCMm signals output from the triangle-wave carrier-cycle generation channel (master) of the TAUD module.

Signals generated at cycle 0 and 1 of TAUD are used as input signals so that two sets of peak or trough interrupts are output.

20.4.2.2 Basic Operation

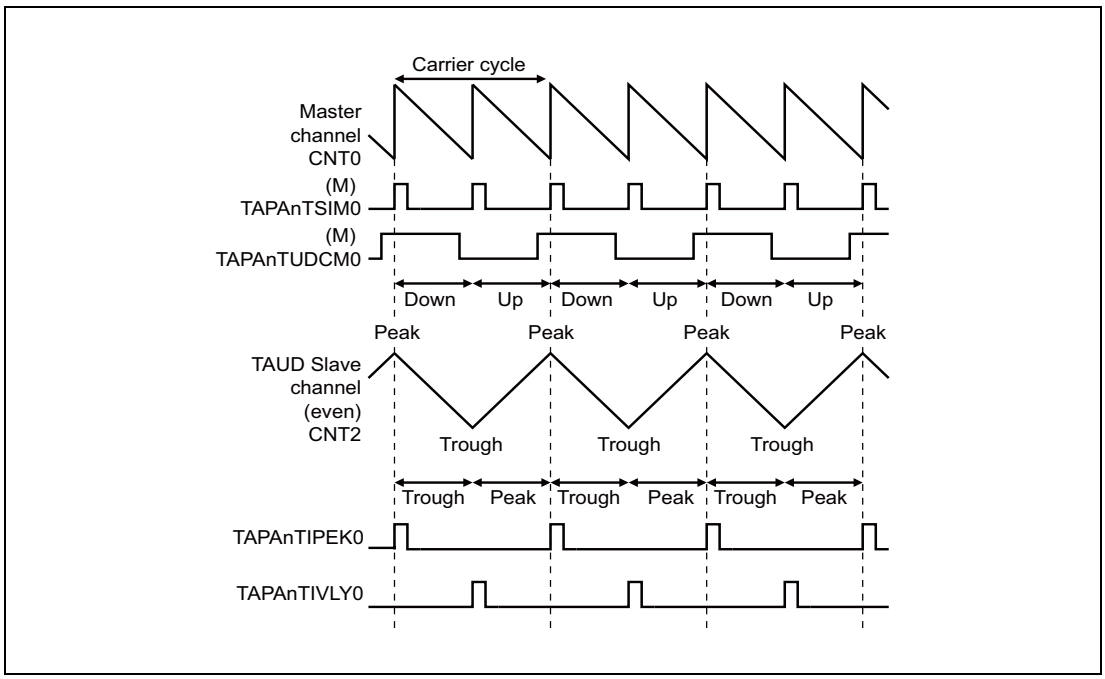


Figure 20.8 Example of Operations in Response to Master Channel 0 of TAUD

TAUD

The master channel of TAUD generates a triangle-wave carrier-cycle.

The TAUD module outputs an INT0 signal and toggles the up/down signal in accord with the setting of TAPAnTUDCM0 per half cycle of this carrier wave.

TAPA

- The TAPA module handles an TAPAnTSIM0 signal arriving while TAPAnTUDCM0 is high as a peak interrupt and outputs the TAPAnTIPEK0 signal in response.
- The TAPA module handles an TAPAnTSIM0 signal arriving while TAPAnTUDCM0 is low as a trough interrupt and outputs the TAPAnTIVLY0 signal in response.

CAUTION

Combined circuits handle output of the TAPAnTIPEK0/TAPAnTIVLY0 signals and operate regardless of the operating mode.

When the TAPAnTIPEK0/TAPAnTIVLY0 signals are not in use, they must be controlled by Section 6.2.3, EI Level Interrupt Mask Registers 0 to 7 (IMR0 to IMR7).

20.4.2.3 Operating Procedure

The procedure for selecting interrupt output is as follows.

		Operation	States of the TAUD and TAPA Modules
	Initial settings	The TAPA module does not require initial settings.	TAUD and TAPA are stopped.
		Initialize the TAUD module. Determine the timer operating mode.	
	Starting operation	Start the TAUD module.	TAUD starts counting.
	During operation	TAUD runs in accord with the settings for the various functions.	The interrupt output selector outputs a peak interrupt (TAPAnTIPEK0) or a trough interrupt (TAPAnTIVLY0) for control cycle 0. This is based on interrupt input (TAPAnTSIM0) and up/down input (TAPAnTUDCM0) from TAUD.
Restart operation	Stopping operation	Stop the TAUD module.	TAUD stops counting.

20.4.3 Selecting a Trigger to Start Conversion by the A/D Converter

The TAPA is capable of producing triggers to start conversion by the A/D converter (TAPAnTADOUT0 and TAPAnTADOUT1). A trigger signal is produced from the INT and TOUT signals output from the triangle-wave carrier-cycle generation channel (master) of the TAUD and the INT signal output from the channel selected for operation with the trigger to start conversion by the A/D converter.

20.4.3.1 Configuration

Table 20.15 Signals Used in Generating the TAPAnTADOUT Signals

Output Signal	Up/Down Input	Timer Enable Input	Slave Match Detection Signal	Trough Interrupt Signal
TAPAnTADOUT0	TAPAnTUDCM0	TAPAnTTOEM0	TAPAnTCDENS0	TAPAnTIVLY0
TAPAnTADOUT1	TAPAnTUDCM0	TAPAnTTOEM0	TAPAnTCDENS1	TAPAnTIVLY0

Table 20.16 Operation of TAPAnCTL1.TAPAnATS[1:0] and TAPAnTADOUT0

TAPAnATS1	TAPAnATS0	Description
0	0	The INT signal from slave 0 is output as TAPATADOUT0 while master 0 of the TAUD module is in the down state.
0	1	The INT signal from slave 0 is output as TAPATADOUT0 while master 0 of the TAUD module is in the up state.
1	0	The INT signal from slave 0 of TAUD is output as TAPATADOUT0.
1	1	The INT and TAPATIVLY0 (trough interrupt signal 0) signals from slave 0 of TAUD are output as TAPAnTADOUT0.

Table 20.17 Operation of TAPAnCTL1.TAPAnATS[3:2] and TAPAnTADOUT1

TAPAnATS3	TAPAnATS2	Description
0	0	The INT signal from slave 1 is output as TAPAnTADOUT1 while master 0 of the TAUD module is in the down state.
0	1	The INT signal from slave 1 is output as TAPAnTADOUT1 while master 0 of the TAUD module is in the up state.
1	0	The INT signal from slave 1 of TAUD is output as TAPAnTADOUT1.
1	1	The INT and TAPAnTIVLY0 (trough interrupt signal 0) signals from slave 1 of TAUD are output as TAPAnTADOUT1.

20.4.3.2 Basic Operation

The following figure shows waveforms in control of A/D converter trigger output in triangle-wave PWM mode.

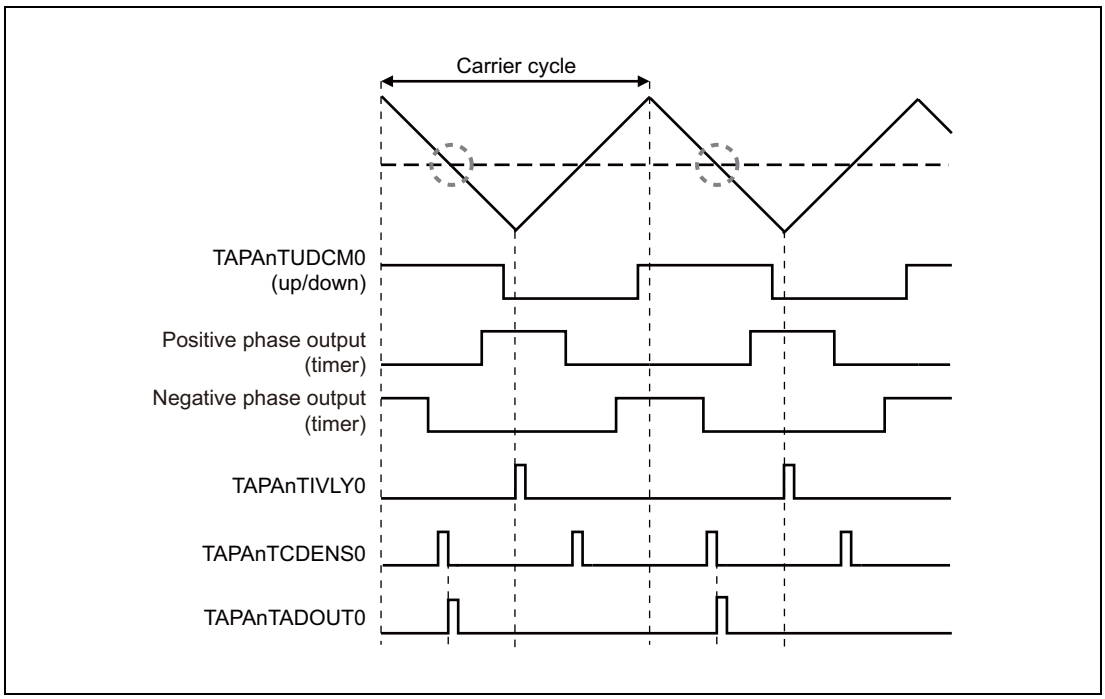


Figure 20.9 TAPAnATS = {0, 0}: INT Output while Master Channel is in the Down State

An INT signal from the slave while the master is in the down state is output as a trigger to start conversion by the A/D converter.

An INT signal from the slave while the master is in the up state is not output.

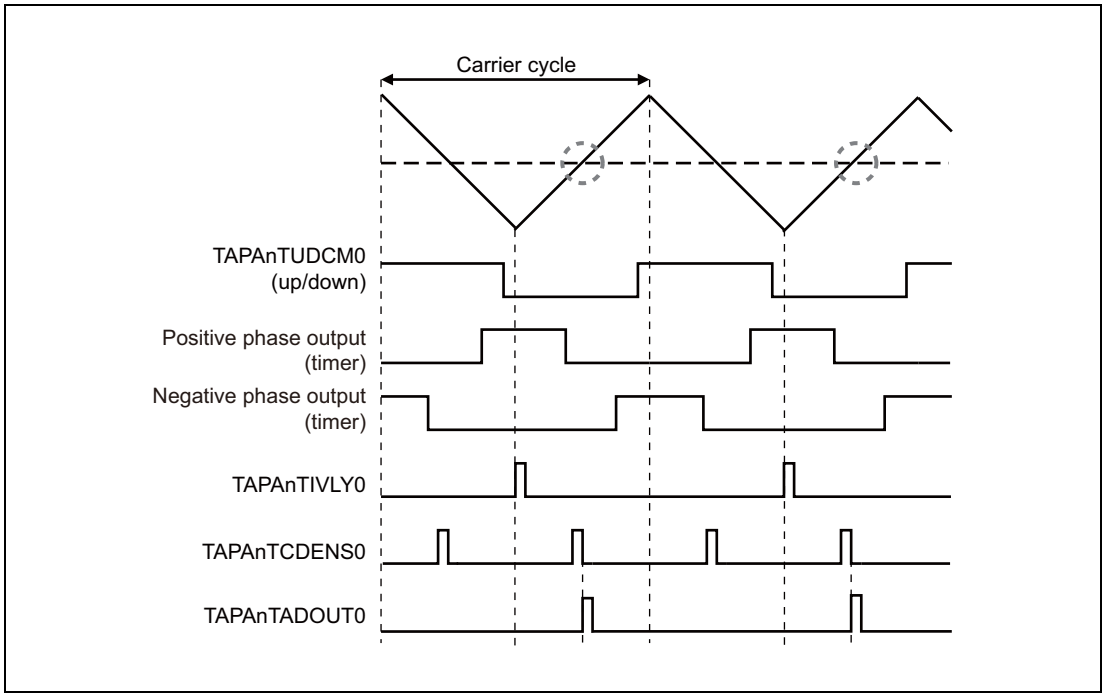


Figure 20.10 TAPAnATS = {0, 1}: INT Output while Master Channel is in the Up State

An INT signal from the slave while the master is in the up state is output as a trigger to start conversion by the A/D converter.

An INT signal from the slave while the master is in the down state is not output.

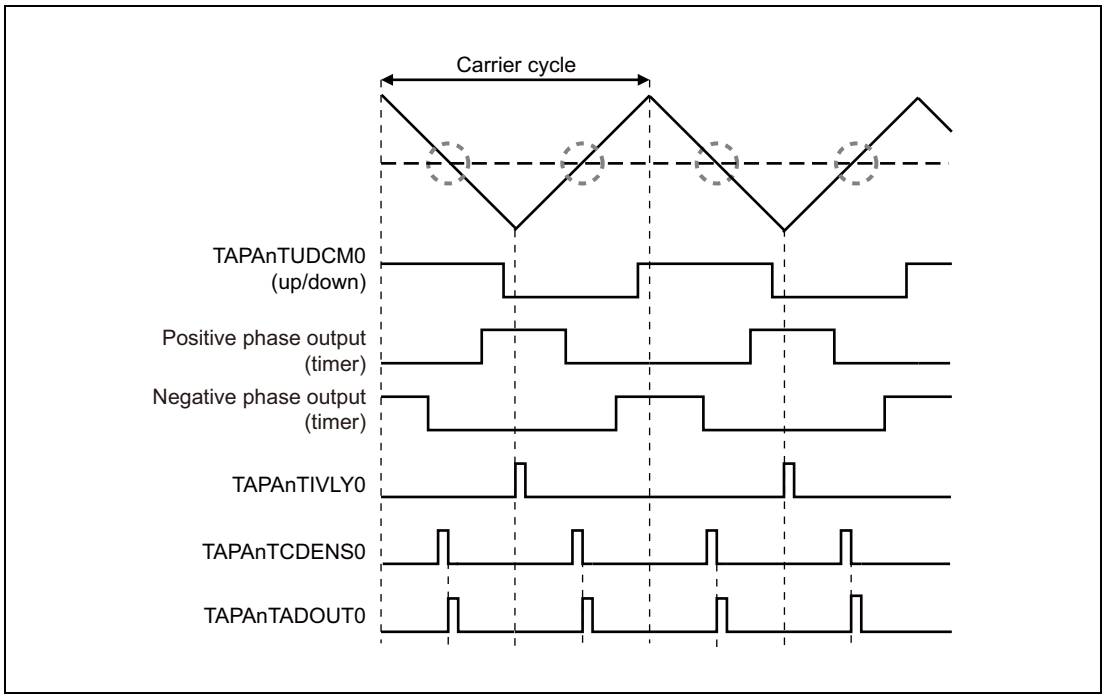


Figure 20.11 TAPAnATS = {1, 0}: INT Output while Master Channel is in either the Down or Up State

An INT signal from the slave at any time is output as a trigger to start conversion by the A/D converter.

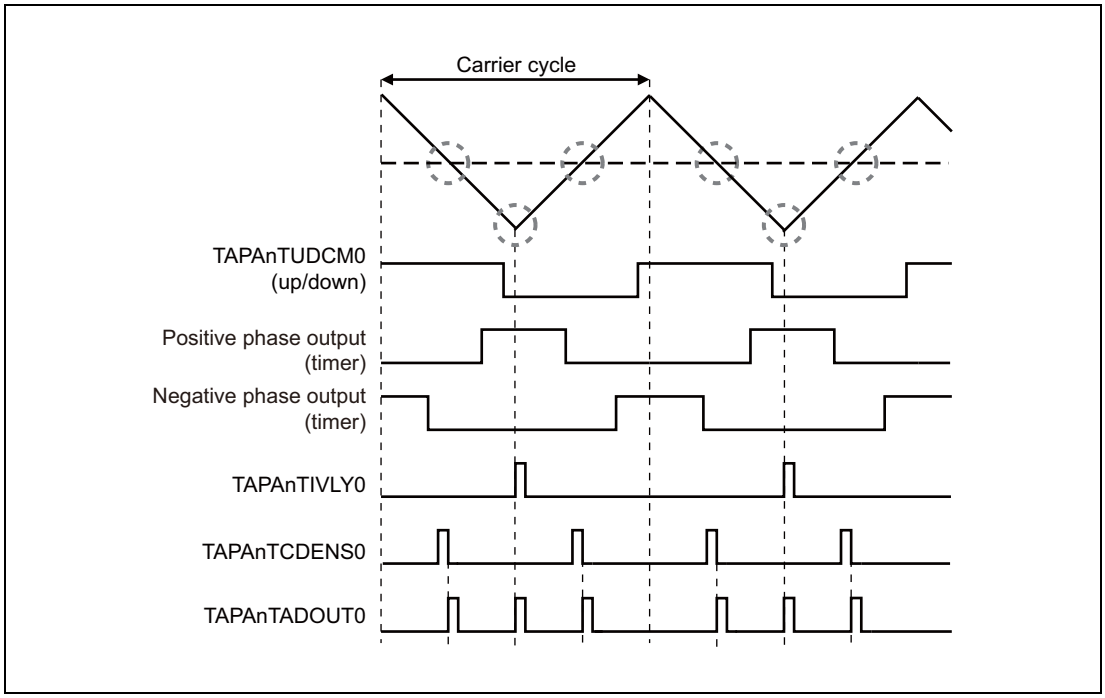


Figure 20.12 TAPAnATS = {1, 1}: Output of INT and Trough Interrupt while Master Channel is in either the Down or Up State

An INT signal or trough interrupt signal from the slave is output as a trigger to start conversion by the A/D converter.

20.4.3.3 Operating Procedure for Setting a Trigger to Start Conversion by the A/D Converter

The operating procedure for setting a trigger to start conversion by the A/D converter is as follows.

		Operation	States of TAUD and TAPA
Restart operation	Initial settings	Initialize TAUD. Determine the timer's operating mode.	TAUD and TAPA are stopped.
		Setting in the TAPAnCTL1 register. Set TAPAnATS[1:0] bits (for TAPAnTADOUT0). Set TAPAnATS[3:2] (for TAPAnTADOUT1).	
	Start operation	Start the TAUD module.	TAUD starts counting.
	During operation	TAUD runs in accord with the settings for the various functions.	The A/D converter conversion trigger selector outputs TAPAnTADOUT0 in accord with the settings of TAPAnATS[1:0] or TAPAnTADOUT1 in accord with the settings of TAPAnATS[3:2], based on the interrupt input (TAPAnTCDENS1 or TAPAnTCDENS0) and up/down input (TAPAnTUDCM1 or TAPAnTUDCM0) from the TAUD and the trough interrupt signal (TAPAnTIVLY1 or TAPAnTIVLY0) generated by TAPA.
	Stopping operation	Stopping the TAUD module.	TAUD stops counting.

Section 21 Timer Pattern Buffer (TPBA)

This section contains a generic description of the Timer Pattern Buffer (TPBA).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of TPBA.

21.1 Features of RH850/C1x TPBA

21.1.1 Units and Channels

This LSI has the following number of TPBA units.

TPBA Each TPBA unit has one channel interface.

Table 21.1 Number of Units

Product Name	RH850/C1H	RH850/C1M
Number of units	2	1
Name	TPBAn (n = 0, 1)	TPBAn (n = 0)

Table 21.2 Index

Index	Meaning
n	Throughout this section, the individual TPBA units are identified by the index “n” (n = 0, 1); for example, TPBAnCTL is the TPBAn control register.
m	The number of buffers are indicated by the index “m” (m = 00 to 63).

21.1.2 Register Base Address

TPBA base addresses are listed in the following table.

TPBA register addresses are given as offsets from the base addresses in general.

Table 21.3 Register Base Address

Base Address Name	Base Address
<TPBA0_base>	FFEA 0000_H
<TPBA1_base>	FFEA 1000_H*1

Note 1. This is not supported by RH850/C1M.

21.1.3 Clock Supply

Clock supply by and to TPBA is listed in the following table.

Table 21.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
TPBAn	PCLK	CLKC_HSB (unmodulated high-speed peripheral clock)

21.1.4 Interrupt Request

TPBA interrupt requests are listed in the following table.

Table 21.5 Interrupt Requests

Unit Interrupt Signal	Outline	Interrupt Number	DMA Trigger Number	DTS Trigger Number
TPBA0				
INTTPBA0IPRD	Cycle match detection interrupt	130	44	62
INTTPBA0IDTY	Duty match detection interrupt	131	45	63
INTTPBA0IPAT	Pattern count match detection interrupt	132	46	64
TPBA1				
INTTPBA1IPRD	Cycle match detection interrupt	133	47	65
INTTPBA1IDTY	Duty match detection interrupt	134	48	66
INTTPBA1IPAT	Pattern count match detection interrupt	135	49	67

21.1.5 Reset Sources

TPBA reset sources are listed in the following table.

Table 21.6 Reset Sources

Unit Name	Reset Source
TPBAn	All reset sources

21.1.6 External Input/Output Signals

External input/output signals of TPBA are listed in the following table.

Table 21.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
TPBA0		
TPBA0O	Timer output	TPBA0O
TPBA1		
TPBA1O	Timer output	TPBA1O

21.2 Overview

21.2.1 Functional Overview

TPBAn is a 16-bit PWM timer with the duty setting buffer.

- Count clock resolution: Min. 12.5 ns (count clock: 80 MHz)
- 16-bit counter
- 16-bit duty register
- 16-bit period setting register
- 7-bit address counter register
- 7-bit pattern number setting register
- Interrupt request signals
  - Period-matched detection interrupt
  - Duty-cycle-matched detection interrupt
  - Number-of-patterns matched detection interrupt
- Number of duty patterns
  - 64 patterns (16 bits) or 128 patterns (8 bits)
- Automatic duty generation according to the number of patterns
- Output control by software
- The count clock can be selected from PCLK, PCLK/2, PCLK/4, and PCLK/8 according to the prescaler set value.
- Synchronous start with another timer

21.2.2 Block Diagram

The following block diagram shows the main components of TPBA.

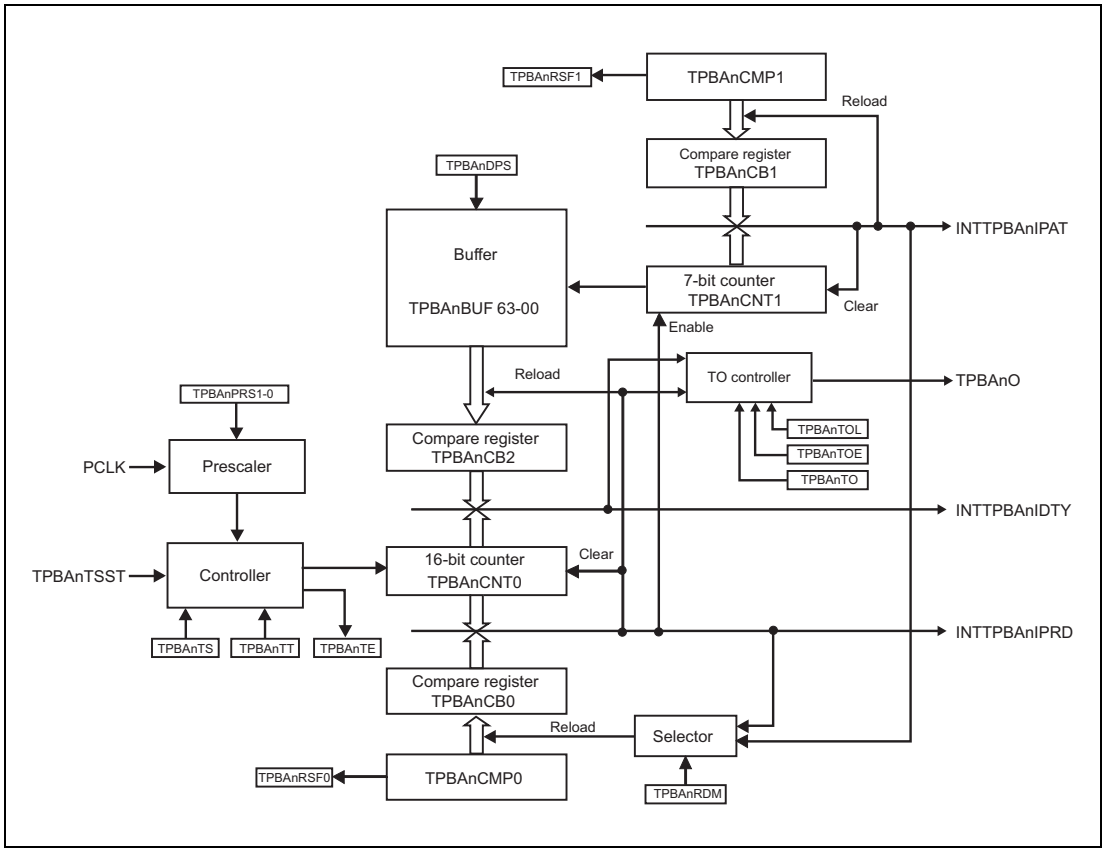


Figure 21.1 Block Diagram of TPBA

- TPBAnSST: Simultaneous start trigger (input from PIC1A)

21.3 Registers

21.3.1 TPBAn Registers Overview

TPBA registers are listed in the following table.

For information on <TPBAn_base>, see **Section 21.1.2, Register Base Address**.

Table 21.8 List of Registers

Module Name	Register Name	Symbol	Address
TPBAn	TPBAn control register	TPBAnCTL	<TPBAn_base> + 200_H
TPBAn	TPBAn reload data mode register	TPBAnRDM	<TPBAn_base> + 118_H
TPBAn	TPBAn reload status register	TPBAnRSF	<TPBAn_base> + 110_H
TPBAn	TPBAn reload data trigger register	TPBAnRDT	<TPBAn_base> + 114_H
TPBAn	TPBAn timer output enable register	TPBAnTOE	<TPBAn_base> + 120_H
TPBAn	TPBAn timer output register	TPBAnTO	<TPBAn_base> + $11C_H$
TPBAn	TPBAn timer output level register	TPBAnTOL	<TPBAn_base> + 124_H
TPBAn	TPBAn period setting register	TPBAnCMP0	<TPBAn_base> + 100_H
TPBAn	TPBAn duty setting register	TPBAnBUFm	<TPBAn_base> + m × 4_H
TPBAn	TPBAn pattern number setting register	TPBAnCMP1	<TPBAn_base> + 104_H
TPBAn	TPBAn timer counter register	TPBAnCNT0	<TPBAn_base> + 108_H
TPBAn	TPBAn address counter register	TPBAnCNT1	<TPBAn_base> + $10C_H$
TPBAn	TPBAn enable status register	TPBAnTE	<TPBAn_base> + 128_H
TPBAn	TPBAn start trigger register	TPBAnTS	<TPBAn_base> + $12C_H$
TPBAn	TPBAn stop trigger register	TPBAnTT	<TPBAn_base> + 130_H

21.3.2 TPBAnCTL — TPBAn Control Register

This register specifies the operation of the TPBAn.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 200_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	TPBAnPRS[1:0]		—	—	—	TPBAnDPS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Table 21.9 TPBAnCTL Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4, 5	TPBAnPRS [1:0]	Selects the count clock.
3 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnDPS	Selects the duty setting pattern mode. 0: 16 bits × 64 patterns mode 1: 8 bits × 128 patterns mode

TPBAnPRS1	TPBAnPRS0	Description
0	0	PCLK is selected.
0	1	PCLK/2 is selected.
1	0	PCLK/4 is selected.
1	1	PCLK/8 is selected.

CAUTION

This register should be set when the timer is stopped (TPBAnTE = 0). If this register is erroneously rewritten, set the register again after stopping the timer.

21.3.3 TPBAnRDM — TPBAn Reload Data Mode Register

This register controls the reload timing of the TPBAn period setting register and TPBAn timer output level register values.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 118_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnRDM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.10 TPBAnRDM Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnRDM0	Controls the reload timing of the TPBAn period setting register (TPBAnCMP0) and TPBAn timer output level register (TPBAnTOL) values. 0: Reloads the values synchronously with a number-of-patterns matched detection interrupt (INTTPBAnIPAT). 1: Reloads the values synchronously with a period-matched detection interrupt (INTTPBAnIPRD).

CAUTION

Although this register can be rewritten during operation, the rewritten value is reflected at any time. Accordingly, during operation, this register should be rewritten when the reload request flag (TPBAnRSF) is 0.

21.3.4 TPBAnRSF — TPBAn Reload Status Register

This register indicates whether or not reload requests from the corresponding registers have been generated.

Access: This register can be read in 8-bit units.

Address: <TPBAn_base> + 110_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TPBAnRSF1	TPBAnRSF0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.11 TPBAnRFS Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	TPBAnRSF1	Indicates whether or not a reload request from TPBAnCMP1 has been generated. 0: No reload request generated or reload completed. 1: Reload request has been generated. This bit is set to 1 when 1 is written to the TPBAnRDT1 bit in the TPBAnRDT register. This bit is cleared at the timing when reload is performed.
0	TPBAnRSF0	Indicates whether or not a reload request from TPBAnCMP0 and TPBAnTOL has been generated. 0: No reload request generated or reload completed. 1: Reload request has been generated. This bit is set to 1 when 1 is written to the TPBAnRDT0 bit in the TPBAnRDT register. This bit is cleared at the timing when reload is performed.

21.3.5 TPBAnRDT — TPBAn Reload Data Trigger Register

This register enables reload of the register values.

Access: This register can be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + 114_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TPBAnRDT1	TPBAnRDT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 21.12 TPBAnRDT Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When writing, write the value after reset.
1	TPBAnRDT1	Enables reload of the TPBAnCMP1 values. 0: Write access is ignored. 1: Reload is enabled (TPBAnRSF1 is set to 1). The values are updated simultaneously at the next reload timing (reload).
0	TPBAnRDT0	Enables reload of the TPBAnCMP0 and TPBAnTOL values. 0: Write access is ignored. 1: Reload is enabled (TPBAnRSF0 is set to 1). The values are updated simultaneously at the next reload timing (reload).

21.3.6 TPBAnTOE — TPBAn Timer Output Enable Register

This register enables or disables the timer output.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 120_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTOE0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.13 TPBAnTOE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTOE0	Enables or disables the timer output (TPBAnO). 0: Disables the timer output based on counter operation. 1: Enables the timer output based on counter operation. • When the timer output is disabled, the level specified in TPBAnTO is output from the TPBAnO pin, and can be controlled by software. • When the timer output is enabled, TPBAnTO is set or cleared by the timer operation, and a PWM signal is output. Write access is prohibited (ignored).

21.3.7 TPBAnTO — TPBAn Timer Output Register

This register controls or reads timer output level.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + $11C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTO0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.14 TPBAnTO Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTO0	Sets or indicates the output level of TPBAnO pin • When the timer output is disabled (TPBAnTOE.TPBAnTOE0 = 0) 0: Outputs low level. 1: Outputs high level. The output level can be controlled by rewriting this register during stopping of the timer output. • When the timer output is enabled (TPBAnTOE.TPBAnTOE0 = 1) 0: Low level is being output by the timer output. 1: High level is being output by the timer output. When the timer output is enabled, rewrite to this register is ignored.

21.3.8 TPBAnTOL — TPBAn Timer Output Level Register

This register controls the timer output level.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 124_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTOL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.15 TPBAnTOL Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TPBAnTOL0	Specifies the active level of the timer output. 0: High 1: Low. • Setting of this bit is enabled when the timer output is enabled (TPBAnTOE.TPBAnTOE0 = 1). • Setting of this bit is reflected when the timer output is started, and change of the output level is reflected at the next reload timing.

CAUTION

This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 21.4.2, Compare Register Rewrite Operation**.

21.3.9 TPBAnCMP0 — TPBAn Period Setting Register

This is a 16-bit compare register for setting the PWM period.

Access: This register can be read/written in 16-bit units.

Address: <TPBAn_base> + 100_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPBAnCMP0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.16 TPBAnCMP0 Register Setting

Operating Mode	PWM Period	Minimum Value (Period)	Maximum Value (Period)
8 bits	TPBAnCMP0 + 1	1	100_H
16 bits	TPBAnCMP0 + 1	1	10000_H

CAUTION

- The PWM period is (TPBAnCMP0 + 1) count clock periods. Accordingly, for PWM output with 100% duty cycle, the maximum settable value is $FFFE_H$ (FE_H).
- This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 21.4.2, Compare Register Rewrite Operation**.

21.3.10 TPBAnBUFm — TPBAn Duty Setting Register

This register is a 16 × 64 buffer register for duty setting.

Access: This register can be read/written in 16-bit units.

Address: <TPBAn_base> + 000_H to $0FC_H$

Value after reset: 0000_H

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

Table 21.17 TPBAnBUFm Register Contents

Bit Position	Bit Name	Function
15 to 0	TPBAnBUFm15 to TPBAnBUFm0	Sets the duty value. This register can set the duty value either in 16 bits × 64 patterns mode (TPBAnDPS = 0) or 8 bits × 128 patterns mode (TPBAnDPS = 1) by setting the TPBAnDPS bit. In either mode, this register is accessed in 16-bit units by the CPU. For details, see **Section 21.4.3, Duty Rewrite Operation**.

CAUTION

The value set to this register is transferred to the duty setting buffer register (TPBAnCB2) synchronously with a period-matched detection interrupt (INTTPBAnIPRD). Rewrite during timer operation is reflected at any time. For details, see **Section 21.4.3, Duty Rewrite Operation**.

- When duty setting register with 8 bits × 128 patterns is used, the duty is set in the range from 00_H to FF_H.

  The formula to output a waveform of duty 100% is: TPBAnBUFm = TPBAnCMP0 + 1 ≤ $00FF_H$.
  Therefore, when PWM output of duty 100% is required, the maximum value of TPBAnCMP0 is $00FE_H$. When TPBAnBUFm is greater than TPBAnCMP0 + 1, duty value exceeds 100%, but the output becomes 100% in total.
- When duty setting register with 16 bits × 64 patterns is used, the duty is set in the range from 0000H to $FFFF_H$.

  The formula to output a waveform of duty 100% is: TPBAnBUFm = TPBAnCMP0 + 1 ≤ $FFFF_H$.
  Therefore, when PWM output of duty 100% is required, the maximum value of TPBAnCMP0 is $FFFE_H$.
  When TPBAnBUFm is greater than TPBAnCMP0 + 1, the duty value exceeds 100%, but the output becomes 100% in total.

21.3.11 TPBAnCMP1 — TPBAn Pattern Number Setting Register

This register sets the number of PWM output patterns.

Access: This register can be read/written in 8-bit units.

Address: <TPBAn_base> + 104_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TPBAnCMP1						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.18 TPBAnCMP1 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 0	TPBAnCMP1 [6:0]	Sets the number of patterns within the following range. TPBAnDPS = 0: 0 to 63 TPBAnDPS = 1: 0 to 127

CAUTION

- This register is a register to be reloaded. Rewrite during timer operation is reflected at the next reload timing. For details on reload, see **Section 21.4.2, Compare Register Rewrite Operation**.
- If 64 or a greater number is set as the number of patterns when the duty setting pattern is in 16 bits × 64 patterns mode (TPBAnDPS = 0), the address pointer changes from 63 to 00, and the duty value is transferred from 00 again. A number-of-patterns matched detection interrupt signal (INTTPBAnIPAT) is output by the match of the specified number of patterns and the lower 7-bit values of TPBAnCNT1.

21.3.12 TPBAnCNT0 — TPBAn Timer Counter Register

This register is a timer counter register that generates PWM output.

Access: This register can be read in 16-bit units.

Address: <TPBAn_base> + 108_H

Value after reset: $FFFF_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPBAnCNT0															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

16-bit counter

This register is a counter register through which the 16-bit counter value can be read.

21.3.13 TPBAnCNT1 — TPBAn Address Counter Register

This register is a counter register that indicates the address pointer to the duty setting register.

Access: This register can only be read in 8-bit units.

Address: <TPBAn_base> + $10C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TPBAnCNT1						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

7-bit counter

This register is a counter register that indicates the address of the TPBAnBUFm register.

21.3.14 TPBAnTE — TPBAn Enable Status Register

This register indicates whether the timer counter is operating or stopped.

Access: This register can only be read in 8-bit units.

Address: <TPBAn_base> + 128_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTE0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.19 TPBAnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	TPBAnTE0	Indicates whether the timer counter is operating or stopped. 0: The timer counter is stopped. 1: The timer counter is operating. • The TPBAnTE0 bit is set to 1 when 1 is written to the TPBAnTS bit or when a synchronous start trigger is input. • The TPBAnTE0 bit is cleared to 0 when 1 is written to the TPBAnTT bit.

21.3.15 TPBAnTS — TPBAn Start Trigger Register

This register controls the timer counter start trigger.

Access: This register can only be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + $12C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.20 TPBAnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TPBAnTS0	This bit is a trigger bit that enables the timer counter. 0: Write access is ignored. 1: Starts counting (TPBAnTE = 1).

CAUTION

Write access to this register during counting (TPBAnTE = 1) is ignored.

21.3.16 TPBAnTT — TPBAn Stop Trigger Register

This register controls the timer counter stop trigger.

Access: This register can only be written in 8-bit units. It is always read as 0.

Address: <TPBAn_base> + 130_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TPBAnTT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 21.21 TPBAnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	TPBAnTT0	This bit is a trigger bit that disables the timer counter. 0: Write access is ignored. 1: Disables counting (TPBAnTE = 0).

21.4 Function

21.4.1 Basic Operation

21.4.1.1 Basic Operation of 16-Bit Counter (TPBAnCNT0)

Counting start

The 16-bit counter (TPBAnCNT0) starts counting from the value after reset $FFFF_H$.

Counter clear

The 16-bit counter is cleared by the match of the counter value and the buffer register (TPBAnCB0) set value of TPBAnCMP0.

Counter read during counting

The 16-bit counter value during counting can be read through TPBAnCNT0.

21.4.1.2 Basic Operation of 7-Bit Counter (TPBAnCNT1)

Counting start

The 7-bit counter (TPBAnCNT1) is initialized to 00_H and starts counting. Subsequently, the counter value is incremented synchronously with a period-matched detection interrupt (INTTPBAnIPRD).

Counter clear

The 7-bit counter is cleared by the match of the counter value and the buffer register (TPBAnCB1) set value of TPBAnCPM1.

Counter read during counting

The 7-bit counter value during counting can be read through TPBAnCNT1. The read value indicates TPBAnBUFm in which the duty value to be transferred next is stored.

21.4.2 Compare Register Rewrite Operation

The following registers are rewritten by reload.

- TPBAnCMP0
- TPBAnCMP1
- TPBAnTOL

Reload mode (simultaneous rewrite function)

Writing to TPBAnRDT enables reload of the registers corresponding to the set bits (sets the reload request flag (TPBAnRSF.TPBAnRSFk)), and the values of all the pertinent registers are updated simultaneously at the next reload timing (reload).

The reload timing of TPBAnCMP0 and TPBAnTOL is set by TPBAnRDM.

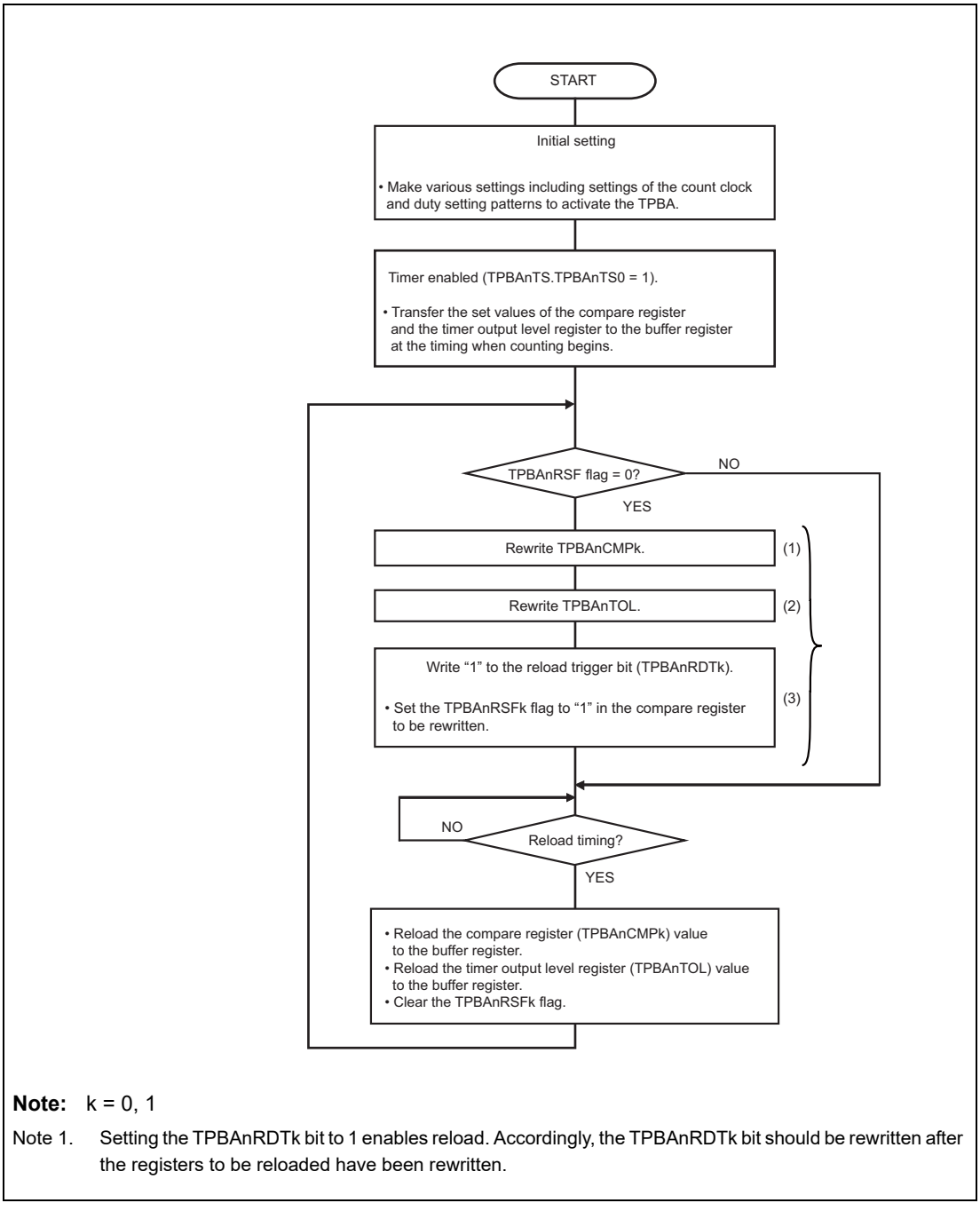
The reload timing of TPBAnCMP1 is the match timing (INTTPBAnIPAT) of the 7-bit counter (TPBAnCNT1) and the buffer register (TPBAnCB1) of TPBAnCMP1.

The registers to be reloaded should be rewritten when the reload request flag (TPBAnRFS.TPBAnRSFk) is 0.

Note: k = 0, 1

Setting Flow for Registers to Be Reloaded

The rewritten values of the registers to be reloaded (TPBAnCMP0, TPBAnCMP1, and TPBAnTOL) can be transferred to the respective buffer registers simultaneously at the reload timing.



Note: k = 0, 1

Note 1. Setting the TPBAnRDTk bit to 1 enables reload. Accordingly, the TPBAnRDTk bit should be rewritten after the registers to be reloaded have been rewritten.

Figure 21.2 Basic Operation Flow of Reload (Simultaneous Rewrite Function)

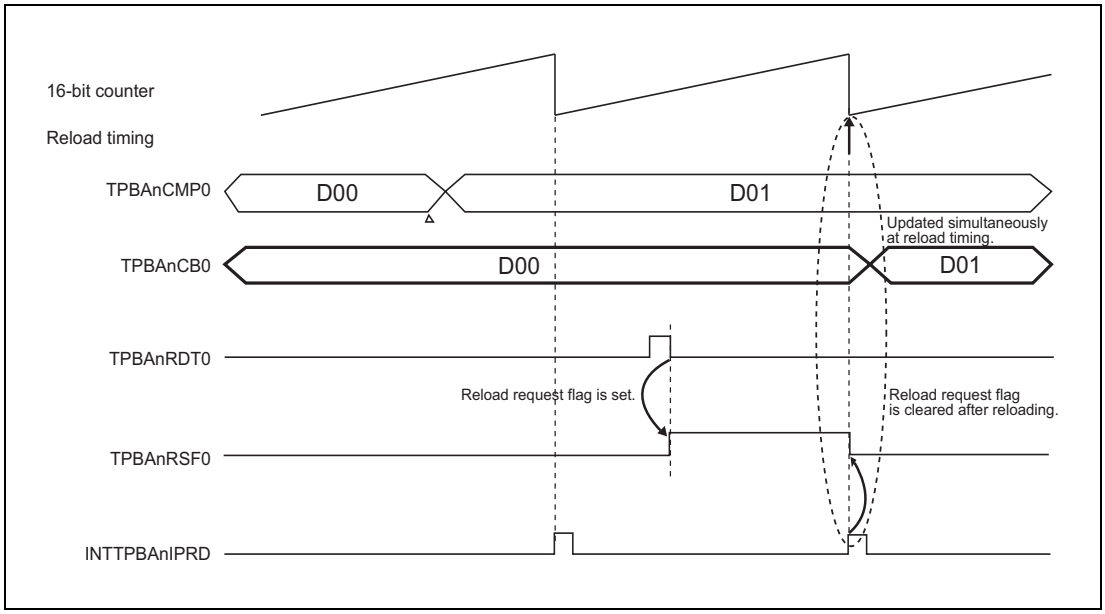


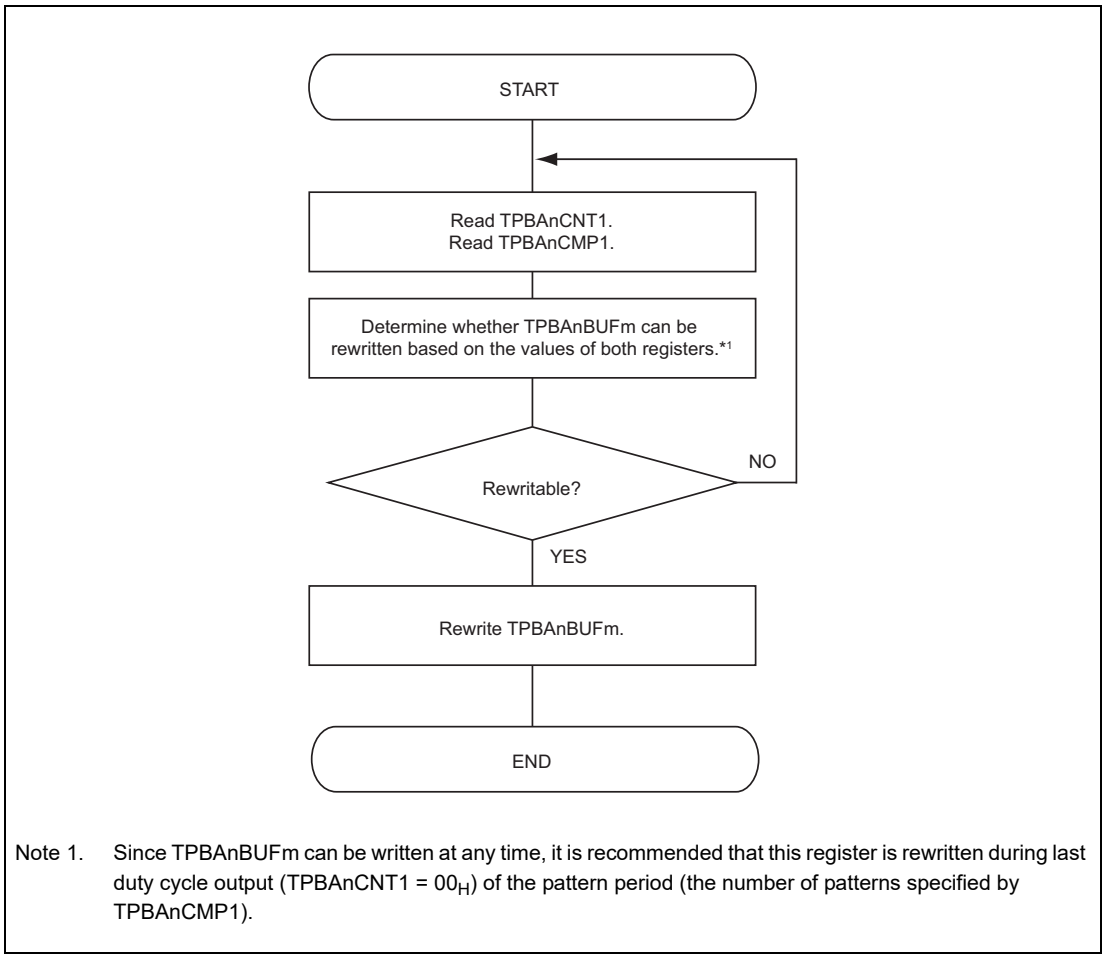
Figure 21.3 Simultaneous Rewrite Timing (TPBAnDPS = 0, TPBAnRDM = 0, and TPBAnTOL = 0)

21.4.3 Duty Rewrite Operation

TPBAnBUFm can be rewritten during operation.

The rewritten setting is reflected immediately.

21.4.3.1 TPBAnBUFm Setting Flow



Note 1. Since TPBAnBUFm can be written at any time, it is recommended that this register is rewritten during last duty cycle output (TPBAnCNT1 = 00_H) of the pattern period (the number of patterns specified by TPBAnCMP1).

Figure 21.4 Basic Rewrite Flow of TPBAnBUFm

21.4.3.2 Access to TPBAnBUFm

TPBAnBUFm is accessed in 16 bit units. The following shows the access in 16 bits × 64 patterns mode and the access in 8 bits × 128 patterns mode.

- In 16 bits × 64 patterns mode (TPBAnDPS = 0)
  This register is accessed by the CPU in units of one 16-bit pattern.

15 0	
Pattern 64	$00FC_H$
Pattern 63	$00F8_H$
:	:
Pattern 3	0008_H
Pattern 2	0004_H
Pattern 1	0000_H

- In 8 bits × 128 patterns mode (TPBAnDPS = 1)
  This register is accessed by the CPU in units of two 8-bit patterns.

15 8	7 0	
Pattern 128	Pattern 127	$00FC_H$
Pattern 126	Pattern 125	$00F8_H$
:	:	:
Pattern 6	Pattern 5	0008_H
Pattern 4	Pattern 3	0004_H
Pattern 2	Pattern 1	0000_H

21.4.3.3 Relationship between TPBAnCNT1 Read Value and TPBAnBUFm

The duty value of the currently output PWM waveform can be obtained by reading the TPBAnCNT1 count value during operation. TPBAnBUFm in which the currently output duty value is stored can be found by one of the following formulas.

TPBAnDPS Bit	Formula		
	TPBAnCNT1 ≠ 00_H		TPBAnCNT1 = 00_H
0: 16 bits × 64 patterns mode	TPBAnCNT1 - 01_H(1)		TPBAnCMP1(2)
1: 8 bits × 128 patterns mode	TPBAnCNT1 value is an odd number	TPBAnCNT1/2(3)	TPBAnCMP1/2(5)
	TPBAnCNT1 value is an even number	(TPBAnCNT1 / 2) - 01_H(4)	

(1) When TPBAnDPS = 0 and the TPBAnCNT1 ≠ 00_H
The applicable register is found by the formula TPBAnCNT1 – 01_H.
(Example) When TPBAnCNT1 = 08_H: $08_H - 01_H = 07_H$ → TPBA0BUF07

(2) When TPBAnDPS = 0 and the TPBAnCNT1 = 00_H
The applicable register is found by the TPBAnCMP1 value.
(Example) When TPBAnCMP1 = 08_H: TPBAnBUF08

(3) When TPBAnDPS = 1 and the TPBAnCNT1 = an odd number
The applicable register is found by the formula TPBAnCNT1 / 2
(Example) When TPBAnCNT1 = 07_H: $07_H / 02_H = 03_H$ → TPBAnBUF03 (lower 8 bits)

(4) When TPBAnDPS = 1 and the TPBAnCNT1 = an even number
The applicable register is found by the formula (TPBAnCNT1 / 2) – 01_H.
(Example) When TPBAnCNT1 = 08_H: $(08_H / 02_H) – 01_H = 03_H$ → TPBAnBUF03 (upper 8 bits)

(5) When TPBAnDPS = 1 and the TPBAnCNT1 = 00_H
The applicable register is found by the formula TPBAnCMP1 / 2.
(Example) When TPBAnCMP1 = 08_H: $08_H / 2 = 04_H$ → TPBAnBUF04 (lower 8 bits)

21.4.4 Basic Operation Example

Overview

A PWM signal is output from the TPBAnO pin according to the PWM period set in the TPBAnCMP0 register and duty cycle set in the TPBAnBUF00 to TPBAnBUF63 registers.

Prerequisites

- Select 16 bits × 64 patterns mode or 8 bits × 128 patterns mode by setting TPBAnDPS.
- Set the duty cycle to TPBAnBUF00 to TPBAnBUF63.
- Set the number of patterns to TPBAnCMP1.

Functional description

Set the PWM period, the number of patterns, duty cycle, and level to be output. Set TPBAnTS.TPBSnTS0 = 1 (or input a synchronous start trigger) to start incrementing the timer counter value.

The TPBAnO output is set to the active level at the same time the counting begins. TPBAnCNT1 is incremented, and points to the address of the buffer in which the subsequent duty value is stored.

The output is set to the inactive level by the match of the 16-bit counter and the TPBAnBUFm buffer register (TPBAnCB2).

The duty value is then transferred from TPBAnBUFm to the buffer register (TPBAnCB2) by the match of the 16-bit counter and the TPBAnCMP0 buffer register (TPBAnCB0). Then, TPBAnCNT1 is incremented, and a period-matched detection interrupt (INTTPBAnIPRD) is generated. The TPBAnO output is set to the active level after one count clock.

During counting, a duty-cycle-matched detection interrupt (INTTPBAnIDTY) is generated by the match of the 16-bit counter and the buffer register (TPBAnCB2) of TPBAnBUFm.

A number-of-patterns matched detection interrupt (INTTPBAnIPAT) is generated by the match of the 7-bit counter and the TPBAnCMP1 buffer register (TPBAnCB1).

21.4.4.1 List of Operations

Table 21.22 16-Bit Counter Function

Operation		Setting Condition
16-bit counter	Start	Writing 1 to TPBAnTS or set 1 to simultaneous start trigger.
	Clear	Compare match of TPBAnCMP0 buffer register and 16-bit counter
	Stop	Writing 1 to TPBAnTT

Table 21.23 7-Bit Counter Function

Operation		Setting Condition
7-bit counter	Start	Writing 1 to TPBAnTS or set 1 to simultaneous start trigger.
	Clear	Compare match of TPBAnCMP1 buffer register and 7-bit counter
	Stop	Writing 1 to TPBAnTT

Table 21.24 Functions of Compare Registers and Buffer Registers

Register (Data)	Buffer Register	Rewrite Method	Rewrite during Operation	Function
TPBAnCMP0	TPBAnCB0	Reload	Possible	Setting period
TPBAnCMP1	TPBAnCB1	Reload	Possible	Setting number of patterns
TPBAnBUFm	TPBAnCB2	Rewrite at any time	Possible	Setting duty
TPBAnTOL	TPBAnTOLB	Reload	Possible	Setting output level

Buffer Registers

The registers that specify period, the number of patterns, duty, and timer output level consist of data registers that a user can directly set and buffer registers that a user cannot directly set.

Table 21.25 Timer Output Function

Pin	Function
TPBAnO	• When output is enabled (TPBAnTOE = 01_H) PWM output by compare match of the TPBAnBUFm buffer register (TPBAnCB2) and the 16-bit counter • When output is disabled (TPBAnTOE = 00_H) TPBAnTO set value

Table 21.26 Interrupt Requests

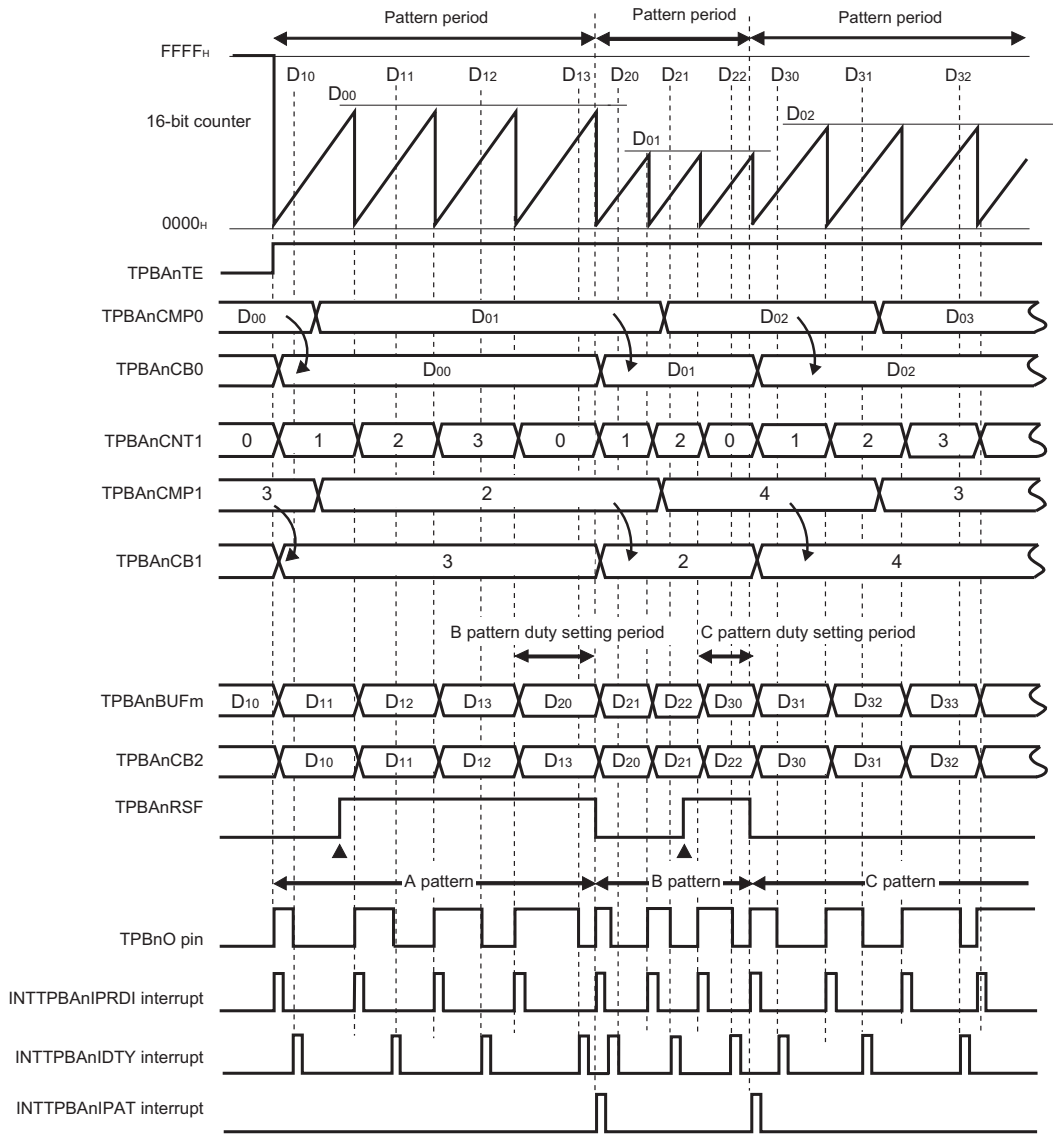
Interrupt	Function
INTTPBAnIPRD	Period-matched detection interrupt
INTTPBAnIDTY	Duty-cycle-matched detection interrupt
INTTPBAnIPAT	Number-of-patterns matched detection interrupt

Table 21.27 Compare Match Timing

Compare Match	Timing
TPBAnCMP0	When the 16-bit counter changes from TPBAnCMP0 to 0000_H.
TPBAnCMP1	When the 7-bit counter changes from TPBAnCMP1 to 01_H.
TPBAnBUFm	When the 16-bit counter matches with the buffer register (TPBAnCB2).

Table 21.28 Example of Setting Each Timer Output Condition

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TPBAnO	PWM output	(TPBAnCMP0 + 1) × count clock	Outputs an inactive level throughout one period (duty cycle 0%).	TPBAnBUFm = 0000_H
			Outputs an active level of one count clock in one period.	TPBAnBUFm = 0001_H
			Outputs an inactive level of one count clock in one period	TPBAnBUFm = TPBAnCMP0
			Outputs an active level throughout one period (duty cycle 100%).	TPBAnBUFm ≥ TPBAnCMP0 + 1



Note 1. PWM duty cycle = (TPBAnBUFm) / (TPBAnCMP0 set value + 1)
PWM period = (TPBAnCMP0 set value + 1) × (count clock period)

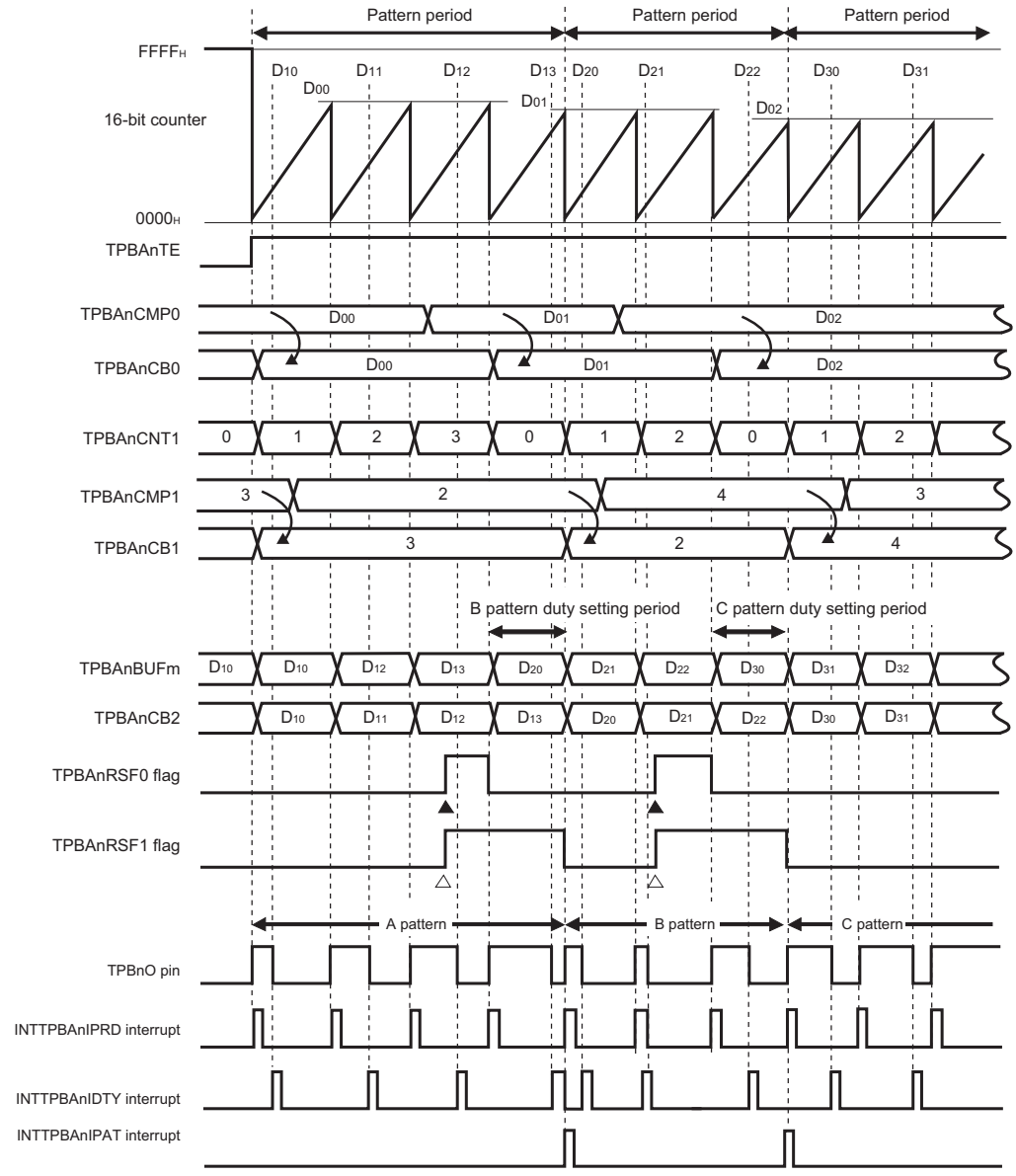
Note 2. ▲: Writing 1 to TPBAnRDT

Figure 21.5 Example of Basic Timing (1/2)

CAUTION

TPBAnO outputs active level 1 count clock after output of INTTPBAnIPRD and outputs inactive level at INTTPBAnIDTY output timing.

When a number-of-patterns matched detection interrupt is used as a trigger of the TPBAnCMP0 and TPBAnTOL reload timing (TPBAnIRDM.TPBAnRDM0 = 0 and TPBAnTOL = 0)



Note 1. PWM duty cycle = (TPBAnBUFm) / (TPBAnCMP0 set value + 1)
PWM period = (TPBAnCMP0 set value + 1) × (count clock period)
Note 2. ▲: Writing 1 to TPBAnRDT
Note 3. Δ: Writing 1 to TPBAnRDT1

Figure 21.5 Example of Basic Timing (2/2)

CAUTION

TPBAnO outputs active level 1 count clock after output of INTTPBAnIPRD and outputs inactive level at INTTPBAnIDTY output timing.

When a period-matched detection interrupt is used as a trigger of the TPBAnCMP0 and TPBAnTOL reload timing (TPBAnIRDM.TPBAnRDM0 = 1 and TPBAnTOL = 0)

Section 22 Encoder Timer (ENCA)

This section contains a generic description of the encoder timer (ENCA).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc.
The remainder of the section describes the functions and registers of the ENCA.

22.1 Features of RH850/C1x ENCA

22.1.1 Number of Units

This microcontroller has the following number of ENCA units.

Table 22.1 Number of Units

Product	RH850/C1x
Number of Units	2
Name	ENCAn (n = 0, 1)

Table 22.2 Index

Index	Meaning
n	Throughout this section, the individual ENCA units are identified by the index “n” (n = 0, 1); for example, ENCAnCTL is the ENCAn control register.

22.1.2 Register Base Address

ENCA base addresses are listed in the following table.

ENCA register addresses are given as offsets from the base addresses in general.

Table 22.3 Register Base Address

Base Address Name	Base Address
<ENCA0_base>	FFE8 0000_H
<ENCA1_base>	FFE8 1000_H

22.1.3 Clock Supply

The ENCA clock supply is shown in following table.

Table 22.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
ENCAn	PCLK	CLKC_HSB (unmodulated high-speed peripheral clock)

22.1.4 Interrupts and DMA

ENCA interrupt requests are listed in the following table.

Table 22.5 Interrupt Requests

Interrupt Name	Outline	Interrupt Number	DMAC Trigger Number	DTS Trigger Number
ENCA0				
INTENCA0IOV	Overflow interrupt	118	0	54
INTENCA0IUD	Underflow interrupt	120	2	56
INTENCA0I0	Compare match 0 or capture 0 interrupt	116	50	48
INTENCA0I1	Compare match 1 or capture 1 interrupt	119	1	55
INTENCA0IEC	Interrupt to indicate clearing due to clearing input from the encoder	121	3	57
ENCA1				
INTENCA1IOV	Overflow interrupt	122	4	58
INTENCA1IUD	Underflow interrupt	124	6	60
INTENCA1I0	Compare match 0 or capture 0 interrupt	117	51	49
INTENCA1I1	Compare match 1 or capture 1 interrupt	123	5	59
INTENCA1IEC	Interrupt to indicate clearing due to clearing input from the encoder	125	7	61

22.1.5 Reset Sources

ENCA reset sources are listed in the following table. ENCA is initialized by these reset sources.

Table 22.6 Reset Sources

Unit Name	Reset Source
ENCAn	All reset sources

22.1.6 External Input/Output Signals

External input/output signals of ENCA are listed in the following table.

Table 22.7 External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal
ENCA0		
ENCA0I0	ENCA0 capture trigger input 0	ENCA0TIN0
ENCA0I1	ENCA0 capture trigger input 1 *1	ENCA0TIN1
ENCA0E0	ENCA0 encoder input 0 *1	ENCA0E0
ENCA0E1	ENCA0 encoder input 1 *1	ENCA0E1
ENCA0EC	ENCA0 encoder clear input*1	ENCA0EC
ENCA1		
ENCA1I0	ENCA1 capture trigger input 0	ENCA1TIN0
ENCA1I1	ENCA1 capture trigger input 1 *1	ENCA1TIN1
ENCA1E0	ENCA1 encoder input 0 *1	ENCA1E0
ENCA1E1	ENCA1 encoder input 1 *1	ENCA1E1
ENCA1EC	ENCA1 encoder clear input *1	ENCA1EC

Note 1. These signals are input via the PIC.

22.2 Overview

22.2.1 Functional Overview

- Generation of the counter control signal from the encoder input signal, and counter operation in synchronization with PCLK.
- Capture function for capturing the counter value with an external trigger signal
- Compare function for compare match judgment with the counter value
- Two capture compare registers that can be set separately for capture operation and for compare operation
- Interrupt mask function for masking the interrupt request signal output as a result of compare match judgment during compare operation
- Function for loading the value of the capture compare register to the counter upon underflow occurrence
- Encoder input signal can be applied to the timer counter clearing condition
- Edge or level for clearing the encoder input signal of the timer counter clearing condition can be selected
- Detection of counter overflow and underflow and output of error flags and error interrupts
- Five interrupts: two capture compare interrupts, one counter clear interrupt, one overflow interrupt, and one underflow interrupt.

22.2.2 Block Diagram

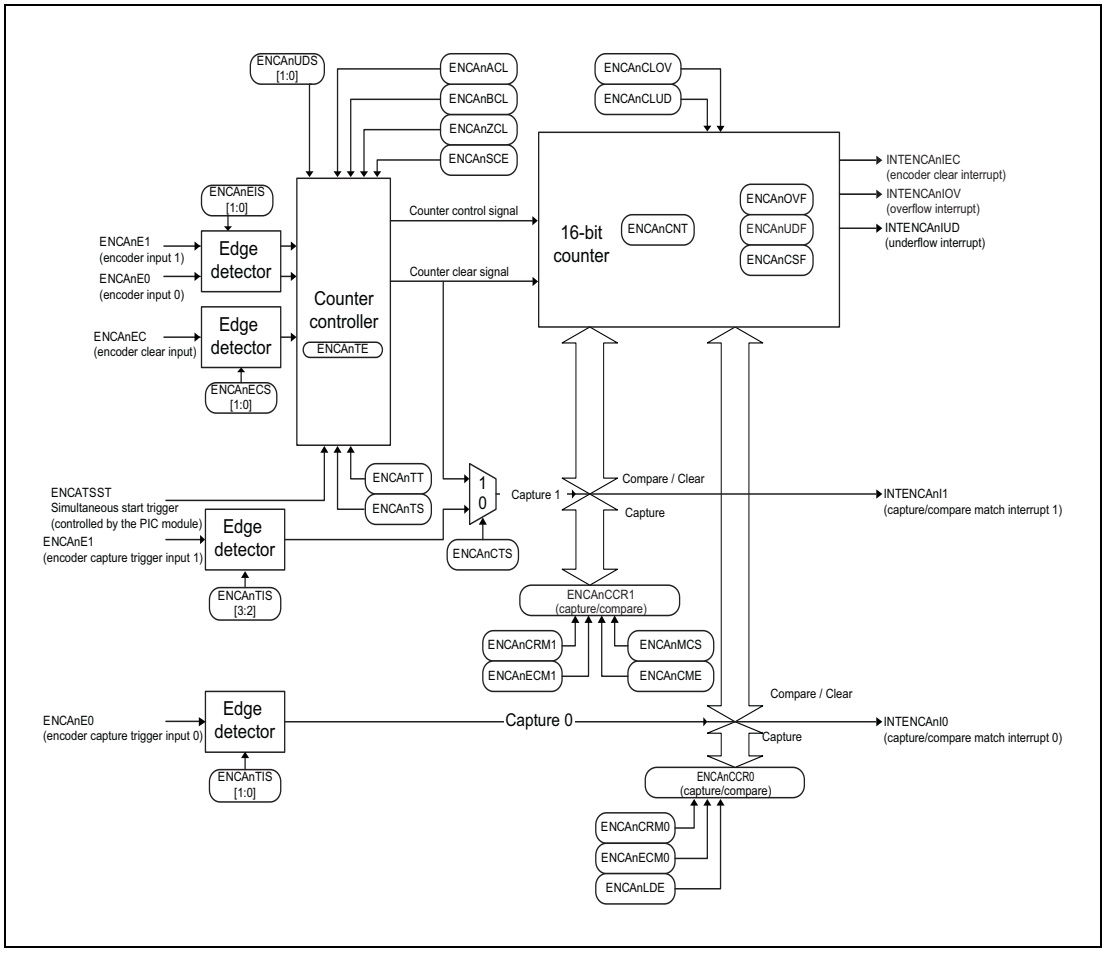


Figure 22.1 Block Diagram of ENCA

22.3 Registers

22.3.1 List of Registers

ENCA registers are listed in the following table.

For details about <ENCAn_base>, see **Section 22.1.2, Register Base Address**.

Table 22.8 List of Registers

Module Name	Register Name	Symbol	Address
ENCAn	ENCAn capture compare register 0	ENCAnCCR0	<ENCAn_base>
ENCAn	ENCAn capture compare register 1	ENCAnCCR1	<ENCAn_base> + 04_H
ENCAn	ENCAn counter register	ENCAnCNT	<ENCAn_base> + 08_H
ENCAn	ENCAn status flag register	ENCAnFLG	<ENCAn_base> + $0C_H$
ENCAn	ENCAn status flag clear register	ENCAnFGC	<ENCAn_base> + 10_H
ENCAn	ENCAn timer enable status register	ENCAnTE	<ENCAn_base> + 14_H
ENCAn	ENCAn timer start trigger register	ENCAnTS	<ENCAn_base> + 18_H
ENCAn	ENCAn timer stop trigger register	ENCAnTT	<ENCAn_base> + $1C_H$
ENCAn	ENCAn I/O control register 0	ENCAnIOC0	<ENCAn_base> + 20_H
ENCAn	ENCAn control register	ENCAnCTL	<ENCAn_base> + 40_H
ENCAn	ENCAn I/O control register 1	ENCAnIOC1	<ENCAn_base> + 44_H

22.3.2 ENCAnCTL — ENCAn Control Register

This register is used to configure various operation settings for ENCAn.

Access: This register can be read/written in 16-bit units.
Writing to this register during operation is prohibited.

Address: <ENCAn_base> + 40_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAn CME	ENCAn MCS	—	—	—	—	ENCAn CRM1	ENCAn CRM0	ENCAn CTS	—	—	ENCAn LDE	ENCAn ECM1	ENCAn ECM0	ENCAnUDS [1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Table 22.9 ENCAnCTL Register Contents (1/2)

Bit Position	Bit Name	Function
15	ENCAnCME	Encoder Clear Mask Enable This bit is used to enable/disable masking of compare match interrupt detection when the compare function is used. 0: Disables the compare match interrupt (INTENCAnl1) mask function for the ENCAnCCR1 register 1: Enables the compare match interrupt (INTENCAnl1) mask function for the ENCAnCCR1 register. This bit is valid only when ENCAnCRM1 = 0. When this bit is set to 1, setting ENCAnECM1 to 1 is prohibited.
14	ENCAnMCS	Encoder Mask Clear Select This bit is used to select the trigger for cancelling masking of compare match interrupt detection when the compare function is used. This bit is valid only when ENCAnCRM1 = 0. 0: Masking of compare match interrupt detection is cancelled when the ENCAnCCR1 register is written. 1: Masking of compare match interrupt detection is cancelled when one of the following three operations is performed. - Timer counter clearing by encoder clear input - Timer counter clearing upon compare match between ENCAnCNT and ENCAnCCR0 when ENCAnECM0 = 1 - Loading from ENCAnCCR0 to timer counter upon underflow detection when ENCAnLDE = 1
13 to 10	Reserved	When writing, write the value after reset.
9	ENCAnCRM1	ENCAnCCR1 Register Mode 0: ENCAnCCR1 used as compare register. 1: ENCAnCCR1 used as capture register.
8	ENCAnCRM0	ENCAnCCR0 Register Mode 0: ENCAnCCR0 used as compare register. 1: ENCAnCCR0 used as capture register.
7	ENCAnCTS	ENCAnCCR1 Capture Trigger Select This is a trigger selection bit for the capture operation to the ENCAnCCR1 register. This bit is valid only when ENCAnCRM1 = 1. 0: Uses ENCAnI1 of capture trigger 1 signal as the trigger for capturing to the ENCAnCCR1 register. 1: Uses the counter clear signal selected with ENCAnSCE as the trigger for capturing to the ENCAnCCR1 register.
6, 5	Reserved	When writing, write the value after reset.

Table 22.9 ENCAnCTL Register Contents (2/2)

Bit Position	Bit Name	Function
4	ENCAnLDE	ENCAn Counter Load Enable This bit is used to enable/disable setting value loading to the counter upon underflow occurrence. This bit is valid only when ENCAnCRM0 = 0. When ENCAnCRM0 = 1, loading of the ENCAnCCR0 register setting value to the counter upon occurrence of an underflow is not performed, regardless of the value of this bit. 0: Disable loading of ENCAnCCR0 register setting value to counter upon occurrence of a counter underflow. 1: Enable loading of ENCAnCCR0 register setting value to counter upon occurrence of a counter underflow.
3	ENCAnECM1	Encoder Clear Mode 1 This bit is used to set the counter clear operation upon match between the counter value and ENCAnCCR1 setting value. This bit is valid only when ENCAnCRM1 = 0. 0: Does not clear the counter to 0000_H upon match of timer counter value and ENCAnCCR1 setting value. 1: Clears the counter to 0000_H upon match of timer counter value and ENCAnCCR1 setting value if the next counting operation is down-counting.
2	ENCAnECM0	Encoder Clear Mode 0 This bit is used to set the counter clear operation upon match between the counter value and ENCAnCCR0 setting value. This bit is valid only when ENCAnCRM0 = 0. 0: Does not clear the counter to 0000_H upon match of timer counter value and ENCAnCCR0 setting value. 1: Clears the counter to 0000_H upon match of timer counter value and ENCAnCCR0 setting value if the next counting operation is up-counting.
1, 0	ENCAnUDS [1:0]	UPDOWN Count Selection 1 and 0 These bits are the counter up/down control bits using ENCAnE0 and ENCAnE1. 00: Upon detection of effective edge of ENCAnE0, - down-count when ENCAnE1 = H, - up-count when ENCAnE1 = L 01: Upon detection of effective edge of ENCAnE0, up-count, Upon detection of effective edge of ENCAnE1, down-count 10: At rising edge of ENCAnE0, down-count At falling edge of ENCAnE0, up-count However, counting is performed only when ENCAnE1 = L. 11: Detection of both edges of ENCAnE0, ENCAnE1. Judgment of counter operation combining both detected edge and level.

22.3.3 ENCAnIOC0 — ENCAn I/O Control Register 0

This register is used to select the input edge of capture triggers 0 and 1 (ENCAnI0, ENCAnI1).

Access: This register can be read/written in 8-bit units.

Address: <ENCAn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ENCAnTIS[3:2]		ENCAnTIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 22.10 ENCAnIOC0 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When writing, write the value after reset.
3, 2	ENCAnTIS[3:2]	Input Edge Selection for Capture Trigger 1 These bits are valid only when the ENCAnCTL register's ENCAnCRM1 = 1 and these bits are valid only when the ENCAnCTL register's ENCAnCRM1 = 1 and ENCAnCTS = 0. All other settings of ENCAnCRM1 and ENCAnCTS are invalid. 00: No edge detection 01: Rising edge detection 10: Falling edge detection 11: Both edges detection
1, 0	ENCAnTIS[1:0]	Input Edge Selection for Capture Trigger 0 These bits are valid only when ENCAnCTL.ENCAnCRM0 = 1. 00: No edge detection 01: Rising edge detection 10: Falling edge detection 11: Both edges detection

22.3.4 ENCAnIOC1 — ENCAn I/O Control Register 1

This register is used to perform the clear condition setting and edge selection upon encoder input.

Access: This register can be read/written in 8-bit units.
Writing to this register during operation is prohibited.

Address: <ENCAn_base> + 44_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ENCAnSCE	ENCAnZCL	ENCAnBCL	ENCAnACL	ENCAnECS[1:0]		ENCAnEIS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.11 ENCAnIOC1 Register Contents

Bit Position	Bit Name	Function
7	ENCAnSCE	Encoder Special-Clear Enable This is an encoder special clear enable bit. When setting this bit to 1, set both ENCAnUDS1 and ENCAnUDS0 to 10_B or 11_B. The operation is not guaranteed if this bit is set to 1 with ENCAnUDS1 and ENCAnUDS0 set to 00_B or 01_B. 0: Clears the counter upon detection of ENCAnEC effective edge (set with ENCAnECS1 and ENCAnECS0). 1: Clears the counter upon detection of input level condition of ENCAnEC, ENCAnE1 and ENCAnE0 (set with ENCAnZCL bit, ENCAnBCL bit, and ENCAnACL bit).
6	ENCAnZCL	Input-Z Clear Condition Selection This bit is used to set the condition for clearing by encoder clear input (ENCAnEC) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level
5	ENCAnBCL	Input-B Clear Condition Selection This bit is used to set the condition for clearing by encoder input 1 (ENCAnE1) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level
4	ENCAnACL	Input-A Clear Condition Selection This bit is used to set the condition for clearing by encoder input 0 (ENCAnE0) when using the encoder special clear function. This bit is valid only when ENCAnSCE = 1; it is invalid when ENCAnSCE = 0. 0: Clear condition: Low level 1: Clear condition: High level
3, 2	ENCAnECS[1:0]	Encoder Clear Input Edge Selection 1 and 0 These are the encoder clear input edge selection bits. These bits are valid only when ENCAnSCE = 0; they are invalid when ENCAnSCE = 1. 00: No edge detection 01: Rising edge detection 10: Falling edge detection 11: Both edges detection
1, 0	ENCAnEIS[1:0]	Encoder Edge Input Selection 1 and 0 These are the encoder input edge selection bits. These bits are valid when ENCAnUDS1 and ENCAnUDS0 = 00_B or 01_B, and are invalid when ENCAnUDS1 and ENCAnUDS0 = 10_B or 11_B. 00: No edge detection 01: Rising edge detection 10: Falling edge detection 11: Both edges detection

22.3.5 ENCAnFLG — ENCAn Status Flag Register

This register holds the status flags of the timer counter of ENCAn.

Access: This register can only be read in 8-bit units.

Address: <ENCAn_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENCAnCSF	ENCAnUDF	ENCAnOVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 22.12 ENCAnFLG Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read.
2	ENCAnCSF	Counter Status Flag This bit reflects the current timer counter operation. 0: Timer counter in up-count status 1: Timer counter in down-count status
1	ENCAnUDF	Underflow Flag This bit reflects the occurrence of an underflow during the timer counter operation. This bit is cleared at the start of counting. 0: This flag is cleared upon any of the following events: – 1 is written to ENCAnFGC.ENCAnCLUD – The flag is cleared to 0 by setting the ENCAnTS bit to 1 when ENCAnTE = 0 or by setting the simultaneous start trigger input (ENCAnTSST signal) to “High”. 1: This flag is set to 1 upon occurrence of an underflow during the encoder timer counting.
0	ENCAnOVF	Overflow Flag This bit reflects the occurrence of an overflow during the timer counter operation. This bit is cleared at the start of counting. 0: This flag is cleared upon any of the following events: – 1 is written to ENCAnFGC.ENCAnCLOV – The flag is cleared to 0 by setting the ENCAnTS bit to 1 when ENCAnTE = 0 or by setting the simultaneous start trigger input (ENCAnTSST signal) to “High”. 1: This flag is set to 1 upon occurrence of an overflow during the encoder timer counting.

22.3.6 ENCAnFGC — ENCAn Status Flag Clear Register

This register is used to clear the timer counter status flags of ENCAnFLG.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H.

Address: <ENCAn_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ENCAnCLUD	ENCAnCLOV
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 22.13 ENCAnFGC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When writing, write the value after reset.
1	ENCAnCLUD	Underflow Flag Clear This bit clears the underflow flag. 0: Writing is ignored. 1: Clears ENCAnUDF of the ENCAnFLG register (clears underflow detection).
0	ENCAnCLOV	Overflow Flag Clear This bit clears the overflow flag. 0: Writing is ignored. 1: Clears ENCAnOVF of the ENCAnFLG register (clears overflow detection).

22.3.7 ENCAnCCR0 — ENCAn Capture Compare Register 0

This register is a 16-bit capture compare register 0.

Access: This register can be read/written in 16-bit units.
When used as a capture register, this register is only readable. Writing to this register is ignored.
When used as a compare register, this register is readable/writable.

Address: <ENCAn_base> + 00_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAnCCR0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.14 ENCAnCCR0 Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCCR0 [15:0]	Capture Compare Register 0 Upon occurrence of an underflow, the setting value of this register may be loaded to the counter according to the ENCAnCTL.ENCAnLDE setting. See the description of the ENCAnLDE bit in ENCA control register ENCAnCTL for details. • If ENCAnCTL.ENCAnCRM0 = 0: ENCAnCCR0 is compare register. Set the value to be compared with the timer counter value. • If ENCAnCTL.ENCAnCRM0 = 1: ENCAnCCR0 is capture register. The captured timer counter value is stored.

22.3.8 ENCAnCCR1 — ENCAn Capture Compare Register 1

This register is a 16-bit capture compare register 1.

Access: This register can be read/written in 16-bit units.
When used as a capture register, this register is only readable. Writing to this register is ignored.
When used as a compare register, this register is readable/writable.

Address: <ENCAn_base> + 04_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ENCAnCCR1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.15 ENCAnCCR1 Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCCR1 [15:0]	Capture Compare Register 1 During capture operation, the trigger for capturing to this register differs according to the ENCAnCTL.ENCAnCTS setting. See the description of the ENCAnCTS bit in ENCA control register ENCAnCTL for details. • If ENCAnCTL.ENCAnCRM1 = 0: ENCAnCCR1 is compare register. Set the value to be compared with the timer counter value. • If ENCAnCTL.ENCAnCRM1 = 1: ENCAnCCR1 is capture register. The captured timer counter value is stored.

22.3.9 ENCAnCNT — ENCAn Counter Register

This register is the 16-bit timer counter register.

Access: This register can be read/written in 16-bit units.
This register can be written only when the operation is stopped.

Address: <ENCAn_base> + 08_H

Value after reset: 0000_H

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

Table 22.16 ENCAnCNT Register Contents

Bit Position	Bit Name	Function
15 to 0	ENCAnCNT [15:0]	Counter Register • ENCAnTE.ENCAnTE status: 0 (initial setting): Counting stops. An arbitrary value can be set to timer counter. • ENCAnTE.ENCAnTE status: 0 → 1 (operation start)): Counting starts. Counting up or down is started with the set arbitrary value. • ENCAnTE.ENCAnTE status: 1 (operating): Counting Counting up or down is performed. • ENCAnTE.ENCAnTE status: 1 → 0 (stopped): Counting stops. The counter value immediately before the operation was stopped is held, and counting is stopped.

22.3.10 ENCAnTE — ENCAn Timer Enable Status Register

This register indicates the operating status of ENCAn.

Access: This register can only be read in 8-bit units.

Address: <ENCAn_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 22.17 ENCAnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	ENCAnTE	Timer Status Enable This is a status bit that indicates the operation enabled/stopped status of ENCAn. This bit is cleared to 0 when 1 is written to ENCAnTT.ENCAnTT. This bit is set to 1 when 1 is written to ENCAnTS.ENCAnTS or when the input signal of ENCAnTSST is set to the high level. 0: Operation stopped status 1: Operation enabled status

22.3.11 ENCAnTS — ENCAn Timer Start Trigger Register

This register provides the trigger bit for setting the ENCAn to the operation enabled state.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H.This register can be written only when ENCAnTE.ENCAnTE is 0.

Address: <ENCAn_base> + 18_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 22.18 ENCAnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	ENCAnTS	Timer Start Trigger This is the trigger bit that sets the ENCAn to the operation enabled state. 0: Writing is ignored. 1: The ENCAn is set to the operation enabled state by setting ENCAnTE.ENCAnTE = 1.

22.3.12 ENCAnTT — ENCAn Timer Stop Trigger Register

This register provides the trigger bit for setting the ENCAn to the operation stopped state.

Access: This register can only be written in 8-bit units.
This register is always read as 00_H.

Address: <ENCAn_base> + $1C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ENCAnTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 22.19 ENCAnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	ENCAnTT	Timer Stop Trigger This is the trigger bit that sets the ENCAn to the operation stopped state. 0: Writing is ignored. 1: Clears ENCAnTE.ENCAnTE to 0 to set the ENCAn to the counter operation stopped state.

22.4 Functions

The ENCAn operates the timer counter with counter up/down control and clear control by encoder inputs. The ENCAnCCR0 and ENCAnCCR1 registers can be used as dedicated compare registers or as dedicated capture registers.

22.4.1 Timer Counter Operation

The timer counter operations of the ENCAn are described below.

The figure below shows the operation phases. See the corresponding section with the section number for detailed descriptions on each operation.

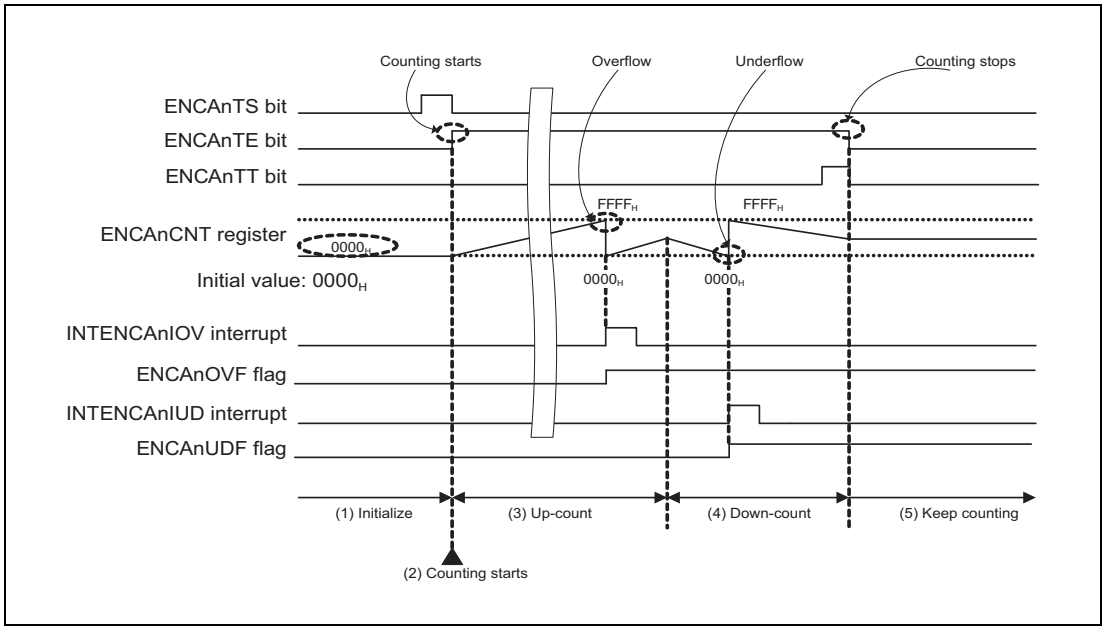


Figure 22.2 Timer Counter Initial Value Setting/Start/Stop

(1) Timer counter initial value setting

The initial value of the ENCAn counter register (ENCAnCNT) can be set in the counter operation stopped status (ENCAnTE = 0).

(2) Timer counter startup

By writing 1 to the timer start trigger bit (ENCAnTS), the timer status enable bit (ENCAnTE) is set to 1, the counter operation is enabled, and counting operation is performed upon detection of the effective edge of the encoder input.

(3) Overflow operation

An overflow occurs when up-counting is performed when the counter value is $FFFF_H$. If the counter value changes from $FFFF_H$ to 0000_H, an overflow interrupt (INTENCAnIOV) is generated, and the overflow flag (ENCAnOVF) is set to 1. The overflow flag (ENCAnOVF) is cleared to 0 when 1 is set to the overflow flag clear bit (ENCAnCLOV). For details about the operation, see **Section 22.6.6, Overflow Occurrence and Overflow Flag Clear Operation**.

(4) Underflow operation

An underflow occurs when down-counting is performed when the counter value is 0000_H. If the counter value changes from 0000_H to $FFFF_H$, an underflow interrupt (INTENCAnIUD) is generated, and the underflow flag (ENCAnUDF) is set to 1. The underflow flag (ENCAnUDF) is cleared to 0 when 1 is set to the underflow flag clear bit (ENCAnCLUD). For details about the operation, see **Section 22.6.7, Underflow Occurrence and Underflow Flag Clear Operation**.

(5) Timer counter stop

By writing 1 to the timer stop trigger bit (ENCAnTT), the timer status enable bit (ENCAnTE) is cleared to 0, and counting is stopped. At this time, the timer counter is not reset to 0000_H and holds the value before counting stops.

22.4.2 Up/Down Control of Timer Counter

Up/down control is performed by judging the phase of the encoder inputs (ENCAnE0, ENCAnE1) according to the settings of the ENCAnUDS1 and ENCAnUDS0 bits.

22.4.2.1 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B

Table 22.20 When ENCAnUDS1 and ENCAnUDS0 Bits = 00_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counting Operation
0	0	Rising edge	High level	Down
		Falling edge		
		Both edges		
		Rising edge	Low level	Up
		Falling edge		
		Both edges		

The effective edge of the signal on the ENCAnE0 pin is specified by setting the ENCAnEIS1 and ENCAnEIS0 bits.

The timer starts counting up/down according to the condition of the edges and levels of the ENCAnE0 and ENCAnE1 pins.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 00_B.

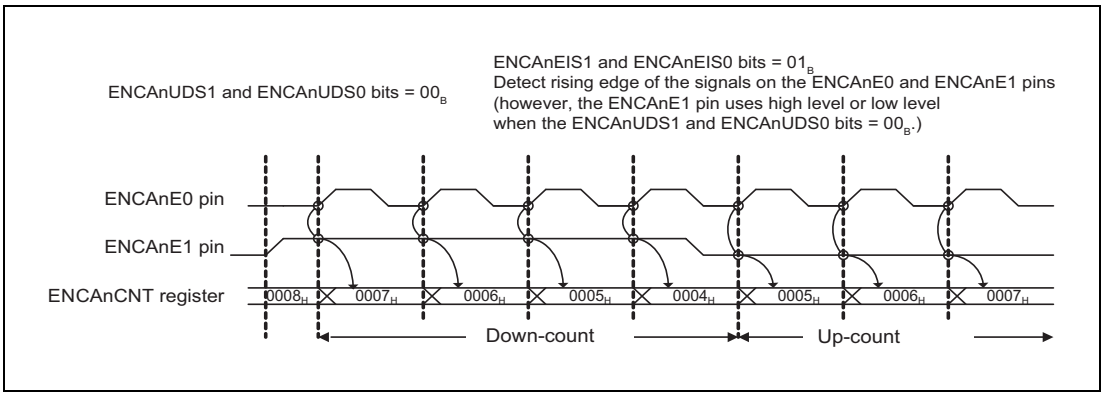


Figure 22.3 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B

22.4.2.2 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B

Table 22.21 When ENCAnUDS1 and ENCAnUDS0 Bits = 01_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counting operation
0	1	Low level	Rising edge	Down
			Falling edge	
			Both edges	
		High level	Rising edge	
			Falling edge	
			Both edges	
		Rising edge	Low level	Up
		Falling edge		
		Both edges		
		Rising edge	High level	
		Falling edge		
		Both edges		
		Simultaneous input		Hold

The effective edges of the signals on the ENCAnE0 and ENCAnE1 pins are specified by setting the ENCAnEIS1 and ENCAnEIS0 bits.

The timer starts counting up/down according to the condition of the edges and levels of the ENCAnE0 and ENCAnE1 pins. When effective edges coincide, the counter keeps counting.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 01_B.

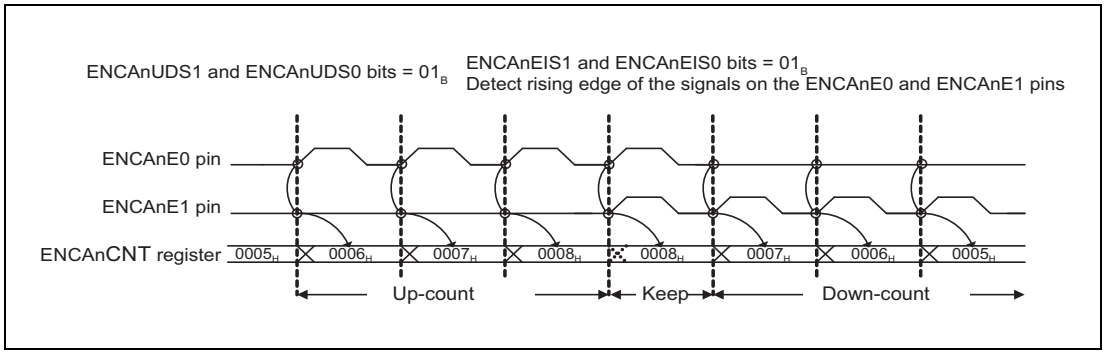


Figure 22.4 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B

22.4.2.3 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B

Table 22.22 When ENCAnUDS1 and ENCAnUDS0 Bits = 10_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counting Operation
1	0	Rising edge	Low level	Down
		Rising edge	Falling edge	
		Falling edge	Low level	Up
		Falling edge	Falling edge	
		Low level	Rising edge	Hold
		Rising edge	Rising edge	
		High level	Rising edge	
		Falling edge	Rising edge	
		Low level	Falling edge	
		Rising edge	High level	
		High level	Falling edge	
		Falling edge	High level	

Specifying effective edges of the signals on the ENCAnE0 and ENCAnE1 pins (by setting the ENCAnEIS1 and ENCAnEIS0 bits) is invalid.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 10_B.

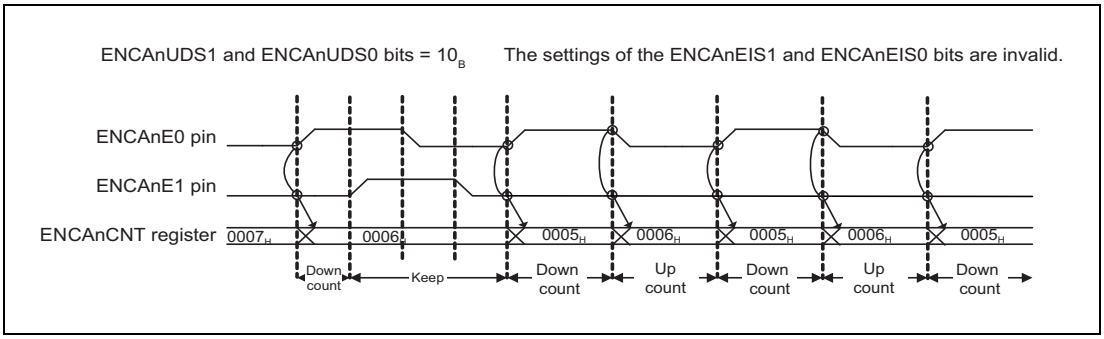


Figure 22.5 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B

22.4.2.4 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B

Table 22.23 When ENCAnUDS1 and ENCAnUDS0 Bits = 11_B

ENCAnUDS1	ENCAnUDS0	Operation Description		
		ENCAnE0 Pin	ENCAnE1 Pin	Counter Operation
1	1	Low level	Falling edge	Down
		Rising edge	Low level	
		High level	Rising edge	
		Falling edge	High level	
		Rising edge	High level	Up
		High level	Falling edge	
		Falling edge	Low level	
		Low level	Rising edge	
		Simultaneous input		Hold

Specifying effective edges of the signals on the ENCAnE0 and ENCAnE1 pins (by setting the ENCAnEIS1 and ENCAnEIS0 bits) is invalid.

The counter value is held when the effective edges of the signals on the ENCAnE0 and ENCAnE1 pins coincide.

The following timing chart shows the counter operation when the ENCAnUDS1 and ENCAnUDS0 bits = 11_B.

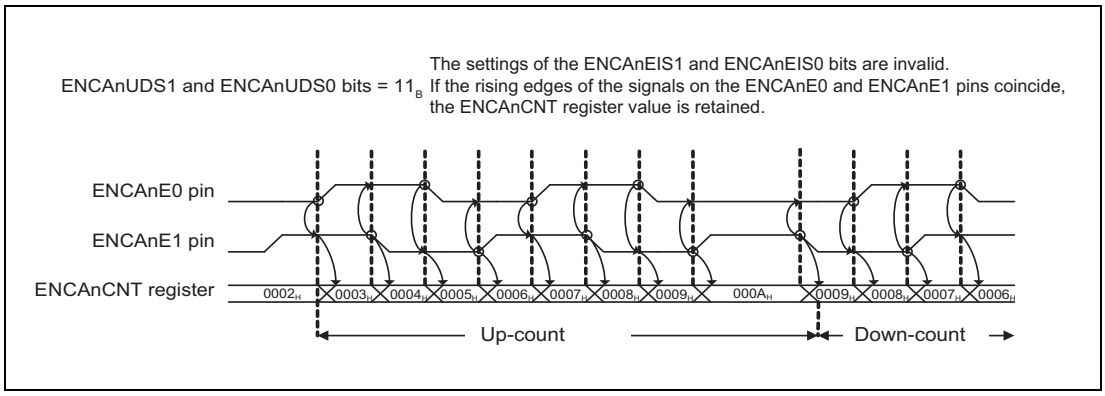


Figure 22.6 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B

22.4.3 Timer Counter Clear Control by Encoder Input

The timer counter is cleared to 0000_H by the encoder clearing input signal (signal on ENCAnEC).

Two types of clearing methods can be selected by controlling the ENCAnSCE, ENCAnZCL, ENCAnBCL, ENCAnACL, ENCAnECS1, and ENCAnECS0 bits of the ENCAnIOC1 register.

Table 22.24 Timer Counter Clear Control by Encoder Input

Clearing Method	ENCAnSCE	ENCAnZCL	ENCAnBCL	ENCAnACL	ENCAnECS1, ENCAnECS0
See **22.4.3.1**	0	Invalid	Invalid	Invalid	Valid
See **22.4.3.2**	1	Valid	Valid	Valid	Invalid

22.4.3.1 Clearing Method when ENCAnSCE = 0

- Upon detection of the effective edge of ENCAnEC, the timer counter is cleared to 0000_H in synchronization with the operation clock.
- The effective edge of ENCAnEC is specified by the setting of the ENCAnECS1 and ENCAnECS0 bits.
- The settings of the ENCAnZCL, ENCAnBCL, and ENCAnACL bits are invalid.
- An encoder clear interrupt request signal (INTENCAnIEC) is output simultaneously with timer counter clearing.

For details about clear operation when ENCAnSCE = 0, see the timing chart in **Section 22.6.24, Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0**.

22.4.3.2 Clearing Method when ENCAnSCE = 1

- When the clear levels of the ENCAnEC, ENCAnE1, ENCAnE0 inputs are detected, the timer counter is cleared to 0000_H in synchronization with the operating clock.
- Specify the clear levels of the ENCAnEC, ENCAnE1, ENCAnE0 inputs by setting the ENCAnZCL, ENCAnBCL, and ENCAnACL bits.
- The settings of the ENCAnECS1 and ENCAnECS0 bits are invalid.
- An encoder clear interrupt request signal (INTENCAnIEC) is output simultaneously with timer counter clearing.

For details about the clear operation when ENCAnSCE = 1, see the timing charts from **Section 22.6.23.2, When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count** to **Section 22.6.23.5, When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count**.

The clearing conditions of the timer counter according to the ENCAnZCL, ENCAnBCL, and ENCAnACL settings are listed in the table below.

Table 22.25 Clearing Conditions of the Timer Counter

Counter Clearing Condition Setting			Encoder Pin Input Level		
ENCAnZCL	**ENCAnBCL**	**ENCAnACL**	**ENCAnEC**	**ENCAnE1**	**ENCAnE0**
0	0	0	Low	Low	Low
0	0	1	Low	Low	High
0	1	0	Low	High	Low
0	1	1	Low	High	High
1	0	0	High	Low	Low
1	0	1	High	Low	High
1	1	0	High	High	Low
1	1	1	High	High	High

22.4.4 Functions of ENCAnCCR0

22.4.4.1 Compare Function

- When ENCAnCRM0 = 0, the ENCAnCCR0 register functions as a dedicated compare register.
- Upon compare match between the value of the timer counter and the ENCAnCCR0 setting value, a compare 0 match interrupt (INTENCAnI0) is output.
- When ENCAnECM0 = 1, the timer counter is cleared to 0000_H in synchronization with the operating clock upon compare match if the next counting operation is up-count.

Table 22.26 Compare Function of ENCAnCCR0

ENCAnCCR0 function	Compare match clear control	Next counting operation	Timer counter clearing upon compare match with ENCAnCCR0
ENCAnCRM0	ENCAnECM0		
0 (Compare)	0	Up-count	Does not clear (continues counter operation).
		Down-count	
	1	Up-count	Clears timer counter to 0000_H.
		Down-count	Does not clear (continues counter operation).

When ENCAnLDE = 1

- Upon occurrence of an underflow, the setting value of the ENCAnCCR0 register is loaded to the timer counter.
- An underflow interrupt (INTENCAnIUD) is output.

NOTE

For details about the timing chart when ENCAnLDE = 1, see the description from **Section 22.6.13, Using the ENCAnLDE Function Immediately after Startup** to **Section 22.6.17, Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input**.

22.4.4.2 Capture Function

- When ENCAnCRM0 = 1, the ENCAnCCR0 register functions as a dedicated capture register.
- Upon effective edge detection of the capture trigger input 0 (ENCAnI0), the value of the timer counter is stored into ENCAnCCR0.
- A capture 0 interrupt (INTENCAnI0) is output during capture operation.

NOTE

For details about capture operation for ENCAnCCR0, see the timing charts in **Section 22.6.19, Capture Operation between Counter Clocks (ENCAnCCR0)** and **Section 22.6.22, Encoder Operation when Compare Match Clear Control is Disabled**.

22.4.5 Functions of ENCAnCCR1

22.4.5.1 Compare Function

- When ENCAnCRM1 = 0, the ENCAnCCR1 register functions as a dedicated compare register.
- Upon compare match between the value of the timer counter and the ENCAnCCR1 setting value, a compare 1 match interrupt (INTENCAnI1) is output.
- When ENCAnECM1 = 1, the timer counter is cleared to 0000_H in synchronization with the operating clock upon compare match if the next counting operation is down-count.

Table 22.27 Compare Function of ENCAnCCR1

ENCAnCCR1 Function	Compare Match Clear Control	Next Counting Operation	Timer Counter Clearing upon Compare Match with ENCAnCCR1
ENCAnCRM1	ENCAnECM1		
0 (Compare)	0	Up-count	Does not clear (continues counting).
		Down-count	
	1	Up-count	Does not clear (continues counting).
		Down-count	Clears timer count to 0000_H.

Compare match interrupt mask function

- When ENCAnCME = 1, the compare 1 match interrupt mask function is enabled. In this state, the compare 1 match interrupt is output upon the first match of the value of the timer counter and the ENCAnCCR1 setting value, and interrupts are then masked for the second and subsequent compare matches.
- When ENCAnCME = 1 and ENCAnMCS = 0, a compare 1 match interrupt is output once upon the first compare match by writing to the ENCAnCCR1 register (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- When ENCAnCME = 1 and ENCAnMCS = 1, a compare 1 match interrupt is output once upon the first compare match by a timer counter clear operation by the encoder clearing input signal or by a timer counter clear operation upon match between the ENCAnCCR0 register value and the timer counter value (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- When ENCAnCME = 1, ENCAnMCS = 1 and ENCAnLDE = 1, a compare 1 match interrupt is output once upon the first compare match by a loading operation of the ENCAnCCR0 register to the timer counter upon underflow detection (interrupts are masked for the second and subsequent matches until the cancel trigger occurs again).
- Setting ENCAnECM1 to 1 is prohibited when enabling the compare 1 match interrupt mask function.

Table 22.28 Compare Match Interrupt Mask Function

ENCAnCCR1 Function	Compare 1 Match Interrupt Mask	Interrupt Mask Cancel Trigger	Compare 1 Match Interrupt Output upon Compare Match with ENCAnCCR1
ENCAnCRM1	ENCAnCME	ENCAnMCS	
0 (Compare)	0 (Mask function disabled)	— (Setting invalid)	Outputs compare 1 match interrupt upon each compare match.
	1 (Mask function enabled)	0 (Write operation to ENCAnCCR1)	Outputs compare 1 match interrupt once upon the first compare match. (Interrupts are masked for the second and subsequent matches until the cancel trigger occurs.)
		1 (Timer counter clear operation) (Loading from ENCAnCCR0 to timer counter upon underflow when ENCAnLDE = 1)	

22.4.5.2 Capture Function

When ENCAnCRM1 = 1, the ENCAnCCR1 register functions as a dedicated capture register.

NOTE

For details about capture operation to ENCAnCCR1, see the timing chart in **Section 22.6.18, Capture Operation between Counter Clocks (ENCAnCCR1)**.

The operations for each of the ENCAnCTS settings are shown in the table below.

Table 22.29 Capture Function of ENCAnCTS

ENCAnCCR1 Function	Capture Trigger Selection	Capture Trigger Signal	Timer Counter Clearing	Interrupt Occurrence
ENCAnCRM1	ENCAnCTS			
1 (Capture)	0	Capture trigger 1 input (ENCAnI1)	Does not clear timer counter.	(1) Capture 1 interrupt (INTENCAnI1)
	1	Encoder clear input (set with ENCAnSCE)	Clears timer counter.	(1) Capture 1 interrupt (INTENCAnI1) (2) Encoder clear interrupt (INTENCAnIEC)

NOTE

For details about the timing chart when ENCAnCTS = 0 or ENCAnCTS = 1, see the following:

Section 22.6.8, Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin), **Section 22.6.9, Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)**, **Section 22.6.10, Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)**, **Section 22.6.16, Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input (ENCAnEC pin)** and **Section 22.6.17, Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input**.

22.4.5.3 Timer Counter Clearing upon Compare Register Match

Clearing of the timer counter upon compare match between the value of the timer counter and the setting of ENCAnCCR0 or ENCAnCCR1, according to the settings of the ENCAnECM1 and ENCAnECM0 bits in ENCAnCTL, is detailed in the following table.

Table 22.30 Timer Counter Clearing Operation upon Compare Register Match

ENCAnECM1 and ENCAnECM0	Next Counting Operation	Timer Counter Clearing upon Compare Match with ENCAnCCR1	Timer Counter Clearing upon Compare Match with ENCAnCCR0
00	Up-count	Does not clear (continues counting).	Does not clear (continues counting).
	Down-count	Does not clear (continues counting).	Does not clear (continues counting).
01	Up-count	Does not clear (continues counting).	Clears timer counter to 0000_H.
	Down-count	Does not clear (continues counting).	Does not clear (continues counting).
10	Up-count	Does not clear (continues counting).	Does not clear (continues counting).
	Down-count	Clears timer counter to 0000_H.	Does not clear (continues counting).
11	Up-count	Does not clear (continues counting).	Clears timer counter to 0000_H.
	Down-count	Clears timer counter to 0000_H.	Does not clear (continues counting).

22.4.6 Starting and Stopping the Timer Counter

22.4.6.1 Starting the Timers

This product has two encoder timers, for which independent or synchronized operation is selectable.

In the case of independent operation, operation is started by setting the ENCAnTS bits in the respective ENCAnTS registers to 1.

Synchronous operation and simultaneous start with other timers are possible by setting the PIC. For details, see **Section 23.2.3.1, Simultaneous Start Trigger Function**.

22.4.6.2 Stopping the Timers

Setting the ENCAnTT bit in the ENCAnTT register of a given encoder timer to 1 causes the ENCAnTE bit in the ENCAnTE register to be set to 0.

Writing to the ENCAnTT bits in the ENCAnTT registers of the individual encoder timers will lead to each encoder timer stopping with different timing, creating a possible margin of error in the held counter values. Therefore, when resuming operation after it has been stopped, the value in the counter must be re-set or corrected as in the examples below.

Example 1: Remaking the Counter Setting Prior to Restarting Operation

Condition:	Input on the ENCAnEC pin while two encoder timers are operating with common ENCAnE0 and ENCAnE1 pins.
Restart procedure:	Execute simultaneous restarting by setting the encoder timers to the same value. Setting the timers to the same value eliminates errors in the counted value due to operation being stopped and allows restarting of the operation.

Example 2: Correcting the Counter Setting Prior to Restarting Operation

Condition:	Input on the ENCAnEC pin while two encoder timers are operating with separate ENCAnE0 and ENCAnE1 pins.
Restart procedure:	Execute simultaneous restarting by calculating the difference between the values of the counters in the respective encoder timers and compensate for the difference in setting the counters for simultaneous restarting. Since the information of the CPU includes the information on the differences between the values of the counters of the encoder timers, operation can be restarted by correcting the error in the counter value that arose when operation was stopped, i.e. by calculating the difference and setting corrected values.

22.4.6.3 Example of Connection when Two ENCAn Units are Used

To simultaneously operate the counters of the two ENCAn units, make the same settings in the ENCAnUDS[1:0] bits in the ENCAnIOC1 register and the ENCAnCTL register.

When using ENCAnCCR0 as the comparison register for the two ENCAn units, set the same value in the ENCAnCCR0, ENCAnECM0, and ENCAnLDE registers of the two ENCAn units.

When using ENCAnCCR1 as the comparison register for the two ENCAn units, set the same value in the ENCAnECM1 registers of the two ENCAn units.

Since the values are separate if they are not the same, synchronized operation will not be possible.

An example where two units of ENCAn are installed is shown below.

This setting example assumes the use of the ENCAnCCR0 register as a comparison register, and the ENCAnCCR1 register as a capture register.

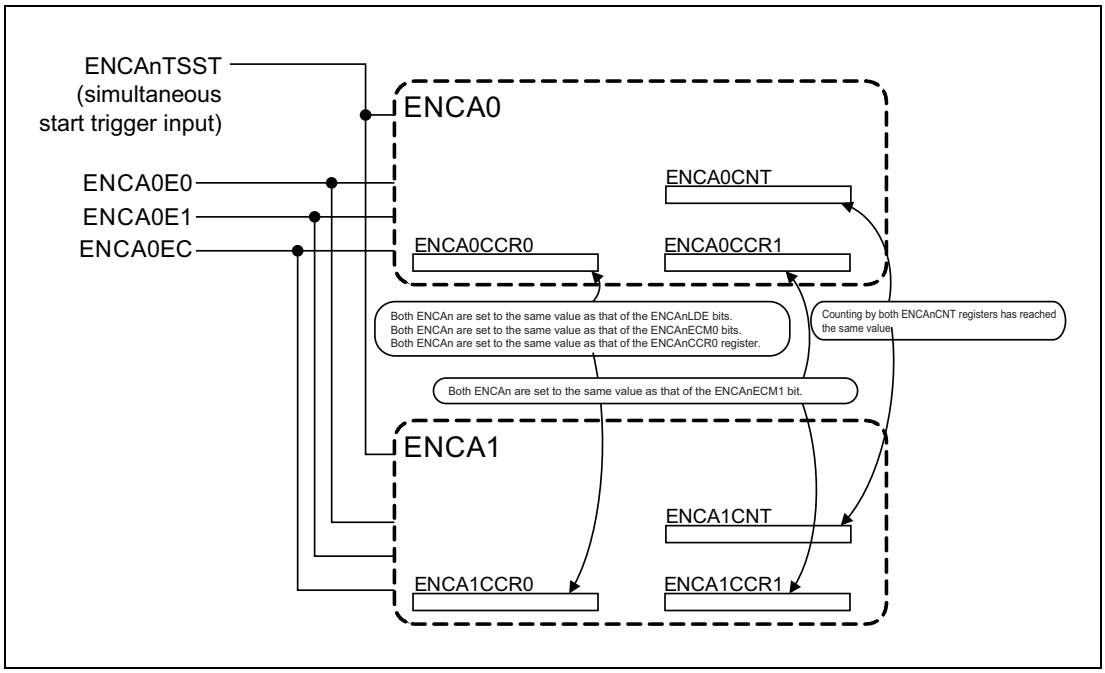


Figure 22.7 Example of Connection when Two ENCAn Units are Used

22.5 Procedure

22.5.1 ENCAn Setting Procedure

The setting procedure for ENCAn is described below.

Table 22.31 ENCAn Setting Procedure

	Action	Setting Status
Initial Setting	Release from the reset state	Power-on status, operation stopped status. (Writing to each register is enabled)
ENCAn Initial Settings	Perform the following initial settings. • Setting for counter • Setting for counter clear • Setting for ENCAnCCR0 register • Setting for ENCAnCCR1 register	This is the counter operation stopped status. The value of the ENCAnTE bit indicating the operating status is 0.
	Make initial settings for the counter. • Set any 16-bit value to ENCAnCNT register. (When, after setting this register, the ENCAnTS bit is set to 1, the counter operation starts from the set counter value.)	The value is set as the initial value of the counter register.
Operation Start	Perform the counter operation start setting. • Set the ENCAnTS bit to 1.	This is the counter operation start status. The value of the ENCAnTE bit indicating the operating status is 1, and the counter clock is supplied to the internal circuit.
Operating	Only those registers whose setting can be changed during operation can be rewritten. • ENCAnCCR0 register setting • ENCAnCCR1 register setting • ENCAnIOC0 register setting	The counter operation set with the initial setting is performed, and counting up or down proceeds according to the ENCAnE0 and ENCAnE1 pins.
Operation Stop	Perform the counter operation stop setting during operation. • Set the ENCAnTT bit to 1.	This is the counter operation stopped status. The value of the ENCAnTE bit indicating the operating status is 0.
ENCAn stop	Reset	The setting registers are initialized.

22.5.1.1 Initial Setting Procedure for the Counter

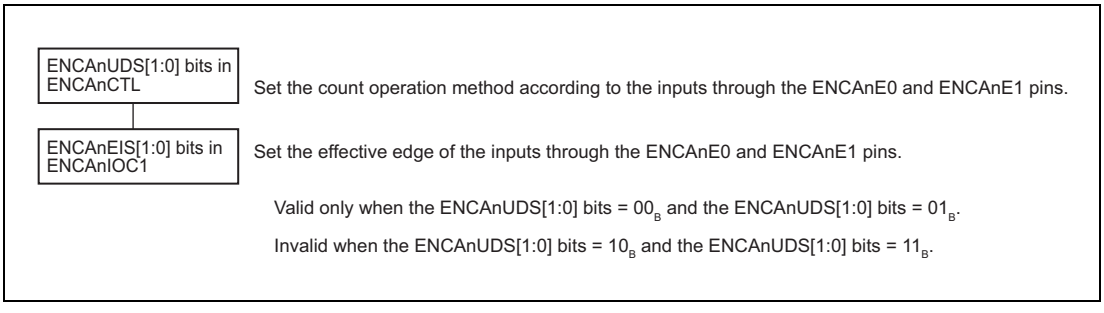


Figure 22.8 Initial Setting Procedure for the Counter

22.5.1.2 Initial Setting Procedure for Counter Clearing

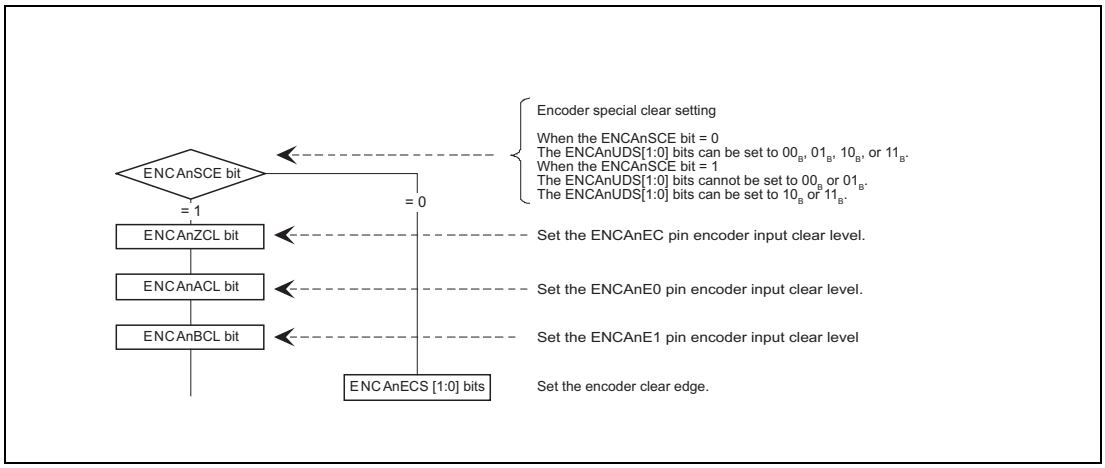


Figure 22.9 Initial Setting Procedure for Counter Clearing

22.5.1.3 Setting Procedure for ENCAnCCR0 Register

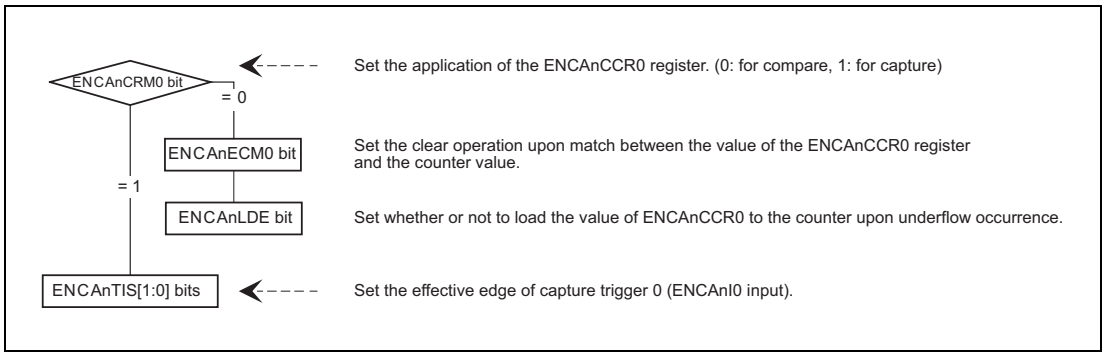


Figure 22.10 Setting Procedure for ENCAnCCR0 Register

22.5.1.4 Setting Procedure for ENCAnCCR1 Register

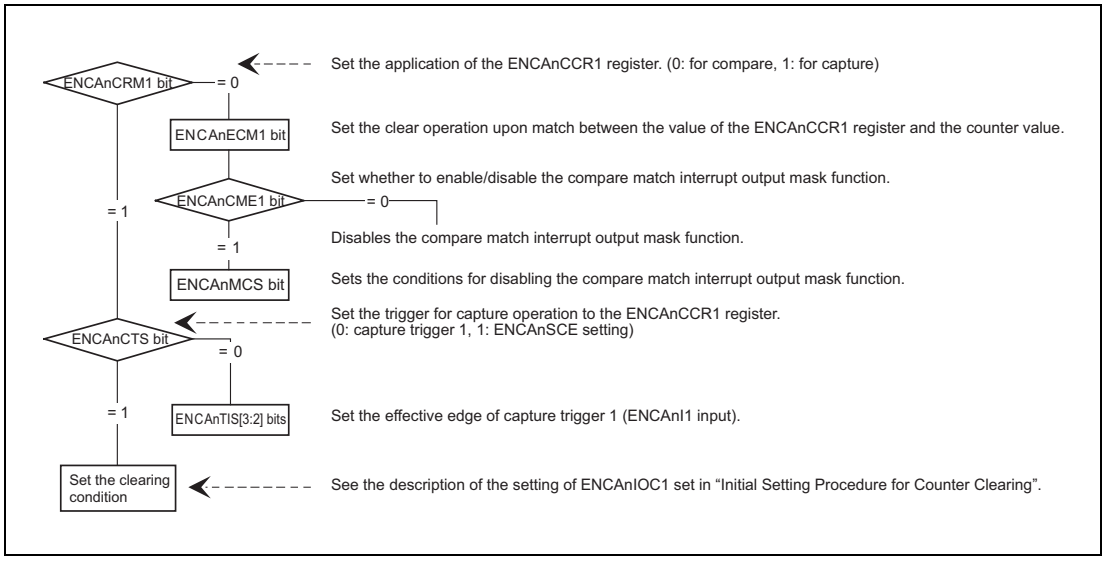


Figure 22.11 Setting Procedure for ENCAnCCR1 Register

22.6 Timing Charts for Encoder Operations

22.6.1 Timing of Basic Encoder Operation 1 (Encoder Comparison Mode 1)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 01_B:
  If the next counting after a match between the values in the counter and the ENCAnCCR0 register is up-counting, the counter is cleared.
- ENCAnCTL.ENCAnLDE = 1:
  When the counter underflows, it is loaded with the value from the ENCAnCCR0 register.

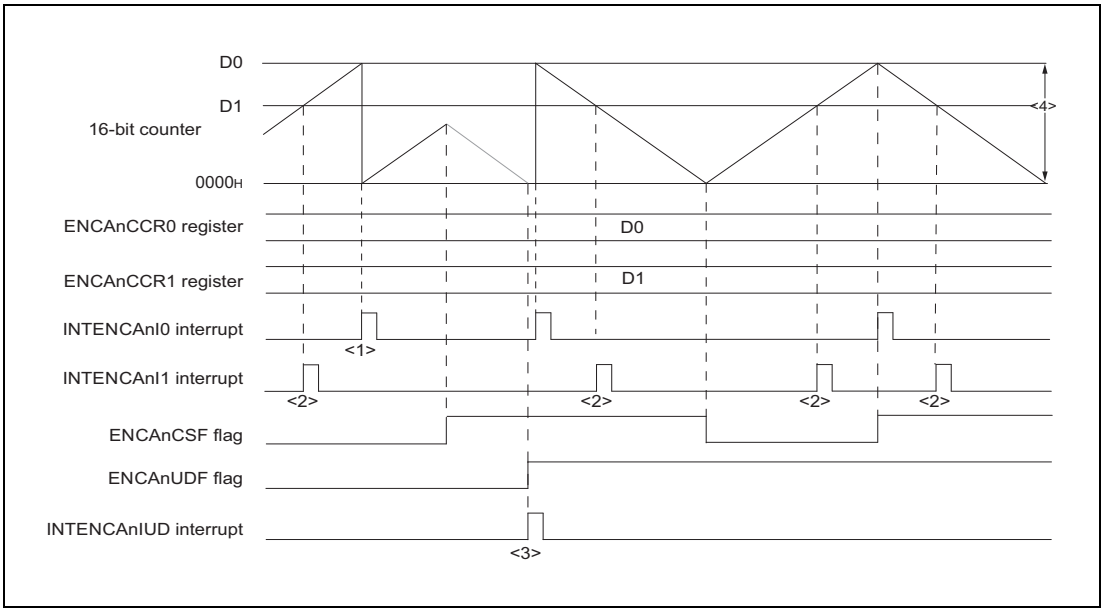


Figure 22.12 Timing of Basic Encoder Operation 1 (Encoder Comparison Mode 1)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1) match.
   Counter clearing due to a match with ENCAnCCR1 does not proceed because ENCAnECM1 = 0.
3. An underflow interrupt (INTENCAnIUD) is generated when the counter underflows.
   ENCAnLDE = 1, so the counter is loaded with the value from the ENCAnCCR0 register (D0) when the counter underflows.
4. ENCAnLDE = 1 and ENCAnECM[1:0] = 01_B, so counting is from 0000_H to the setting of the ENCAnCCR0 register.

22.6.2 Timing of Basic Encoder Operation 2 (Encoder Comparison Mode 2)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 00_B:
  The counter is not cleared on a match between its value and that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 0:
  The counter is not loaded with the value from the ENCAnCCR0 register.

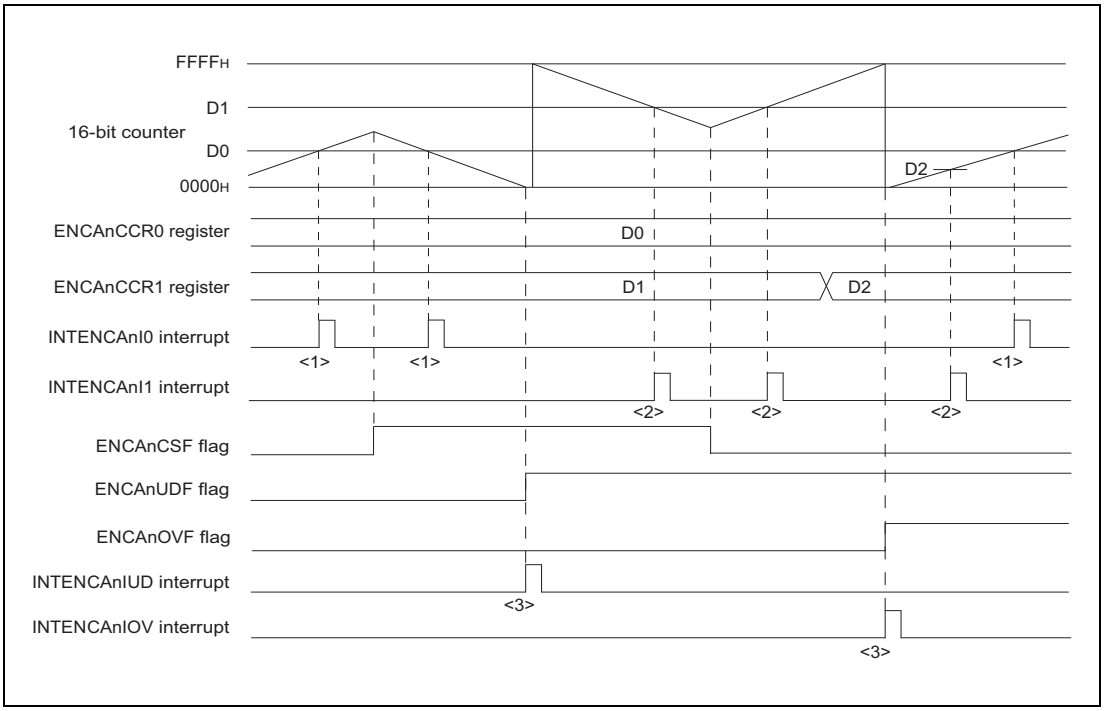


Figure 22.13 Timing of Basic Encoder Operation 2 (Encoder Comparison Mode 2)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   Counter clearing due to matching with ENCAnCCR0 does not proceed because ENCAnECM0 = 0.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1, D2) match.
   Counter clearing due to matching with ENCAnCCR1 does not proceed because ENCAnECM1 = 0.
3. Overflow interrupts (INTENCAnIOV) or underflow interrupts (INTENCAnIUD) are generated in response to an overflow or underflow of the counter.

22.6.3 Timing of Basic Encoder Operation 3 (Encoder Comparison Mode 3)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 00_B:
  Comparison is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 11_B:
  If the next counting after a match between the values in the counter and the ENCAnCCR0 register is up-counting, the counter is cleared.
  If the next counting after a match between the values in the counter and the ENCAnCCR1 register is down-counting, the counter is cleared.
- ENCAnCTL.ENCAnLDE = 0
  The counter is not loaded with the value from the ENCAnCCR0 register.

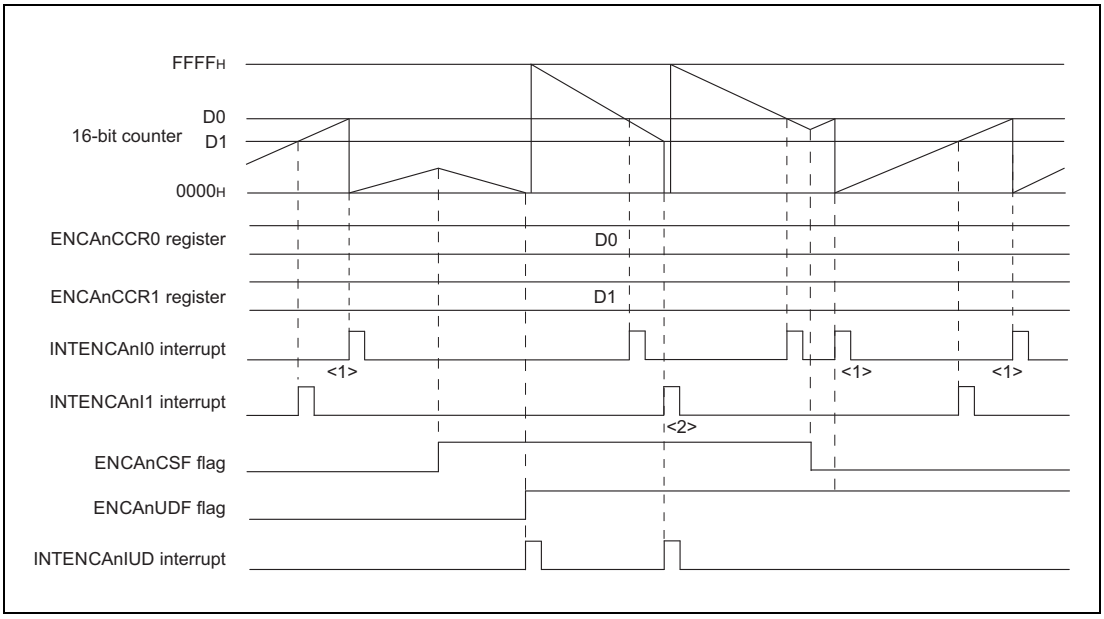


Figure 22.14 Timing of Basic Encoder Operation 3 (Encoder Comparison Mode 3)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. A compare match interrupt (INTENCAnI1) is generated when the counter value and the ENCAnCCR1 register setting (D1) match.
   If the next counting is down-counting, the counter is cleared to 0000_H because ENCAnECM1 = 1.

22.6.4 Timing of Basic Encoder Operation 4 (Encoder Capture Mode)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 11_B:
  Capture is selected as the function of the ENCAnCCR0 and ENCAnCCR1 registers.
- ENCAnCTL.ENCAnECM[1:0] = 00_B:
  The counter is not cleared on a match between its value and that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 0:
  The setting from the ENCAnCCR0 register is not loaded to the counter.
- ENCAnIOC1.ENCAnSCE = 0, ENCAnECS[1:0] = 00_B:
  Input on the ENCAnEC pin does not lead to edge detection.
- ENCAnIOC0.ENCAnTIS[3:2] = 01_B:
  Selects detection of rising edges of the signal on the ENCAnI1 pin.
- ENCAnIOC0.ENCAnTIS[1:0] = 01_B:
  Selects detection of rising edges of the signal on the ENCAnI0 pin.

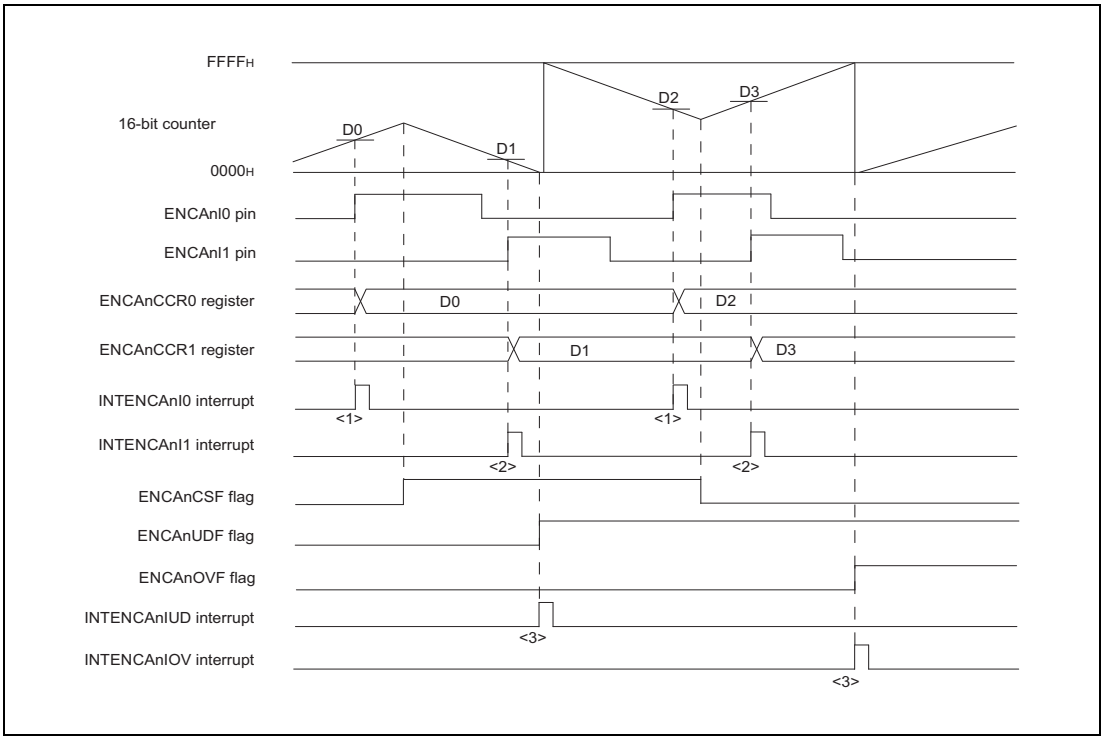


Figure 22.15 Timing of Basic Encoder Operation 4 (Encoder Capture Mode)

1. Detection of a rising edge on the ENCAnI0 pin leads to storage of the counter value (D0, D2) in the capture register (ENCAnCCR0) and the generation of a capture interrupt (INTENCAnI0).
2. Detection of a rising edge on the ENCAnI1 pin leads to storage of the counter value (D1, D3) in the capture register (ENCAnCCR1) and the generation of a capture interrupt (INTENCAnI1).
3. Overflow interrupts (INTENCAnIOV) or underflow interrupts (INTENCAnIUD) are generated in response to an overflow or underflow of the counter.

22.6.5 Timing of Basic Encoder Operation 5 (Encoder Capture and Comparison Mode)

<Setting conditions>

- ENCAnCTL.ENCAnCRM[1:0] = 10_B:
  Select the comparison function for the ENCAnCCR0 register and the capture function for the ENCAnCCR1 register.
- ENCAnCTL.ENCAnECM[1:0] = 01_B:
  The counter is cleared when its value matches that of the ENCAnCCR0 register.
- ENCAnCTL.ENCAnLDE = 1:
  When the counter underflows, it is loaded with the value from the ENCAnCCR0 register.
- ENCAnIOC1.ENCAnSCE = 0, ENCAnECS[1:0] = 00_B
- ENCAnIOC0.ENCAnTIS[3:2] = 11_B
  Selects detection of both edges of the signal on the ENCAnI1 pin.

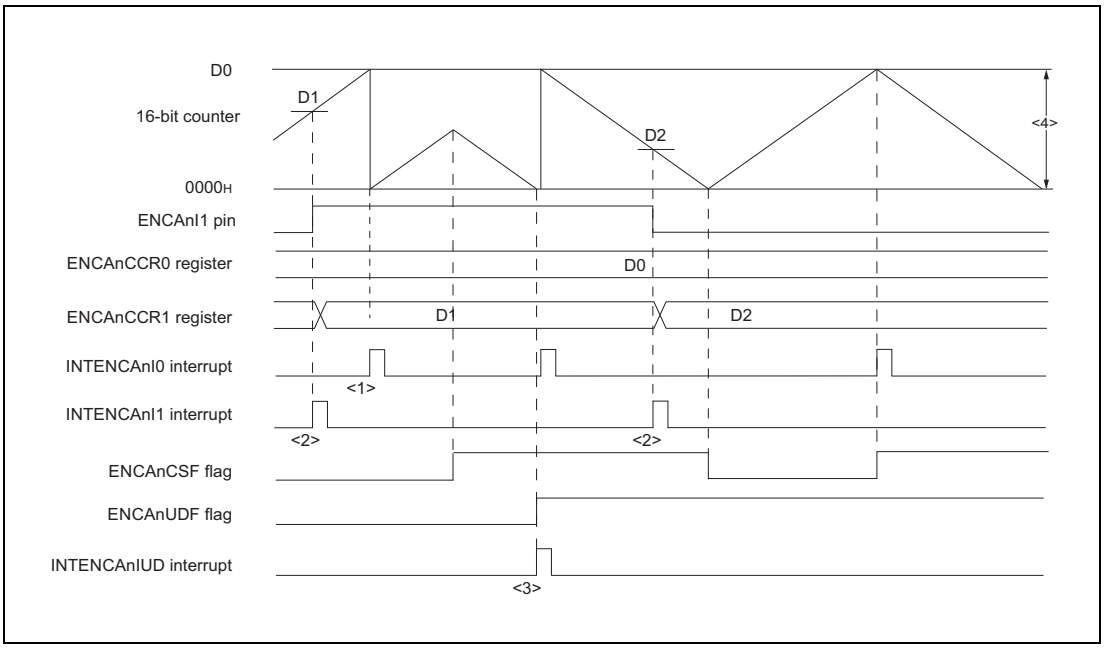


Figure 22.16 Timing of Basic Encoder Operation 5 (Encoder Capture and Comparison Mode)

1. A compare match interrupt (INTENCAnI0) is generated when the counter value and the ENCAnCCR0 register setting (D0) match.
   If the next counting is up-counting, the counter is cleared to 0000_H because ENCAnECM0 = 1.
2. Detection of both edges on the ENCAnI1 pin leads to storage of the counter value (D1) in the capture register (ENCAnCCR1) and the generation of a capture interrupt (INTENCAnI1).
3. An underflow interrupt (INTENCAnIUD) is generated when the counter underflows.
   ENCAnLDE = 1, so the counter is loaded with the value from the ENCAnCCR0 register (D0) when the counter underflows.
4. ENCAnLDE = 1 and ENCAnECM[1:0] = 01_B, so counting is from 0000_H to the setting of the ENCAnCCR0 register.

22.6.6 Overflow Occurrence and Overflow Flag Clear Operation

When up-counting is performed while the counter value is $FFFF_H$, an overflow occurs. If an overflow occurs, an overflow interrupt (INTENCAnIOV) is output and the overflow flag (ENCAnOVF) is set to 1. When the overflow flag clear bit (ENCAnCLOV) is set to 1, the overflow flag (ENCAnOVF) is cleared to 0.

The overflow occurrence and overflow flag clear operation are described as follows.

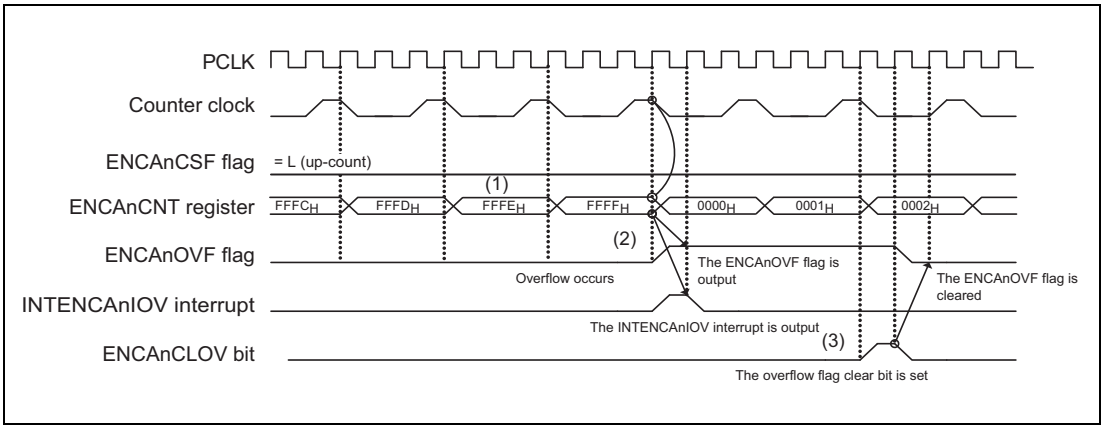


Figure 22.17 Overflow Occurrence and Setting for Clearing of the Overflow Flag

(1) The counter value is counted up from $FFFE_H$ to $FFFF_H$.

(2) When the counter value changes from $FFFF_H$ to 0000_H, an overflow occurs. At the same time, an overflow interrupt is output, and the overflow flag is set to 1.

(3) An overflow flag is cleared to 0 by setting 1 to the ENCAnCLOV bit in the ENCAnFGC register according to the clearing procedure. In addition, an overflow flag is also cleared by setting the ENCAnTS bit in the ENCAnTS register to 1 while ENCAnTE.ENCAnTE is 0, or by setting an input signal of the ENCAnTSST (simultaneous start trigger input) to high.

22.6.7 Underflow Occurrence and Underflow Flag Clear Operation

When down-counting is performed while the counter value is 0000_H, an underflow occurs. If an underflow occurs, an underflow interrupt (INTENCAnIUD) is output and the underflow flag (ENCAnUDF) is set to 1. When the underflow flag clear bit (ENCAnCLUD) is set to 1, the underflow flag (ENCAnUDF) is cleared to 0.

The underflow occurrence and underflow flag clear operation are described as follows.

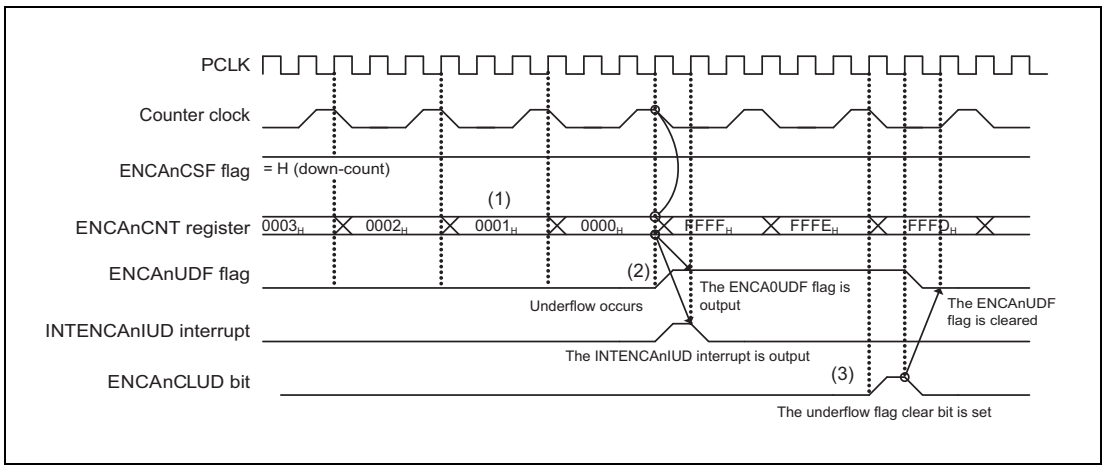


Figure 22.18 Underflow Occurrence and Setting for Clearing of the Underflow Flag

(1) The counter value is counted down from 0001_H to 0000_H.

(2) When the counter value changes from 0000_H to $FFFF_H$, an underflow occurs. At the same time, an underflow interrupt is output, and the underflow flag is set to 1.

(3) An underflow flag is cleared to 0 by setting 1 to the ENCAnCLUD bit in the ENCAnFGC register according to the clearing procedure. In addition, an underflow flag is also cleared by setting the ENCAnTS bit in the ENCAnTS register to 1 while the ENCAnTE bit in the ENCAnTE register is 0, or by setting an input signal of the ENCAnTSST (simultaneous start trigger input) to high.

22.6.8 Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin)

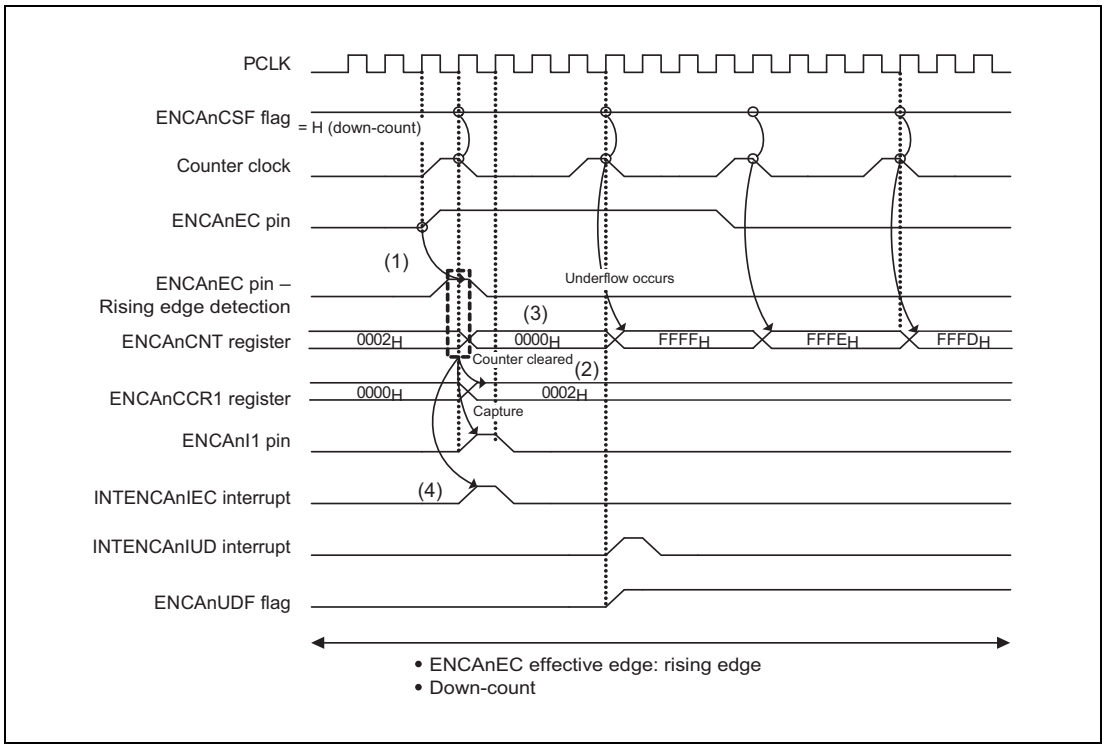


Figure 22.19 Timing Chart of Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin)

[Setting Conditions]

- ENCAnCTL.ENCAnCRM1 = 1
  (ENCAnCCR1 register is selected as capture)
- ENCAnCTL.ENCAnCTS = 1
  (ENCAnEC pin input is selected as a capture trigger input)
- ENCAnIOC1.ENCAnECS1 and ENCAnECS0 bits = 01_B
  (Selected as a rising edge detection of the ENCAnEC pin input)

(1) Capture operation is performed at the rising edge of the ENCAnEC pin input trigger.

(2) The counter value (0002_H) is captured at the rising edge of the ENCAnEC pin input, and it is stored in the ENCAnCCR1 register.

(3) Clear operation is performed by the input through the ENCAnEC pin, and the counter value is reset to 0000_H.

(4) At the same time, by the input through the ENCAnEC pin, an encoder clear interrupt (INTENCAnIEC) and capture interrupt 1 (INTENCAnI1) are output.

22.6.9 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

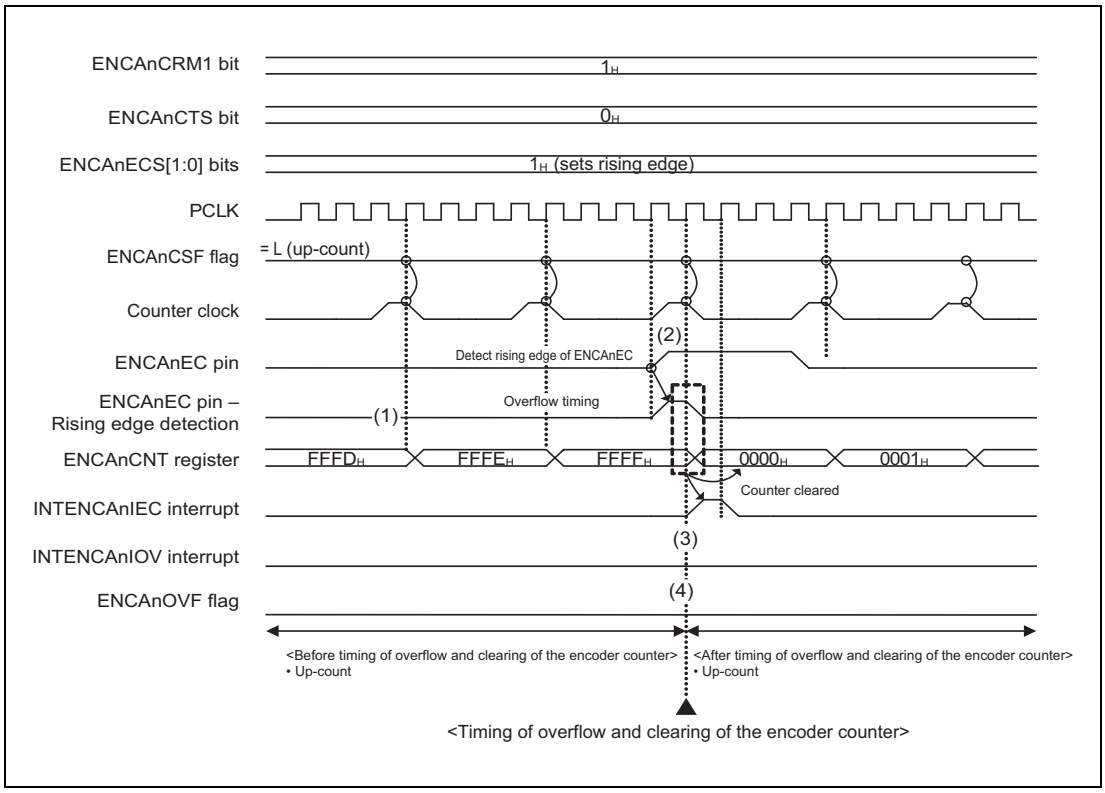


Figure 22.20 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

(1) Up-counting from $FFFD_H$ is continuously performed.

(2) When an overflow occurs if the counter value is $FFFF_H$, and the rising edge of ENCAnEC is detected simultaneously, clear operation by the encoder clear input is performed. The counter value is cleared to 0000_H.

(3) When the counter value is cleared by the encoder clear input, a clear interrupt (INTENCAnIEC) by the encoder clear input is output simultaneously. Because a clear operation by the encoder clear input is performed simultaneously with the overflow occurrence, an overflow interrupt is not output (An overflow does not occur. Clear operation is performed by the encoder clear input).

(4) Because an overflow does not occur as is the case with step 3, the overflow flag is not set.

22.6.10 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

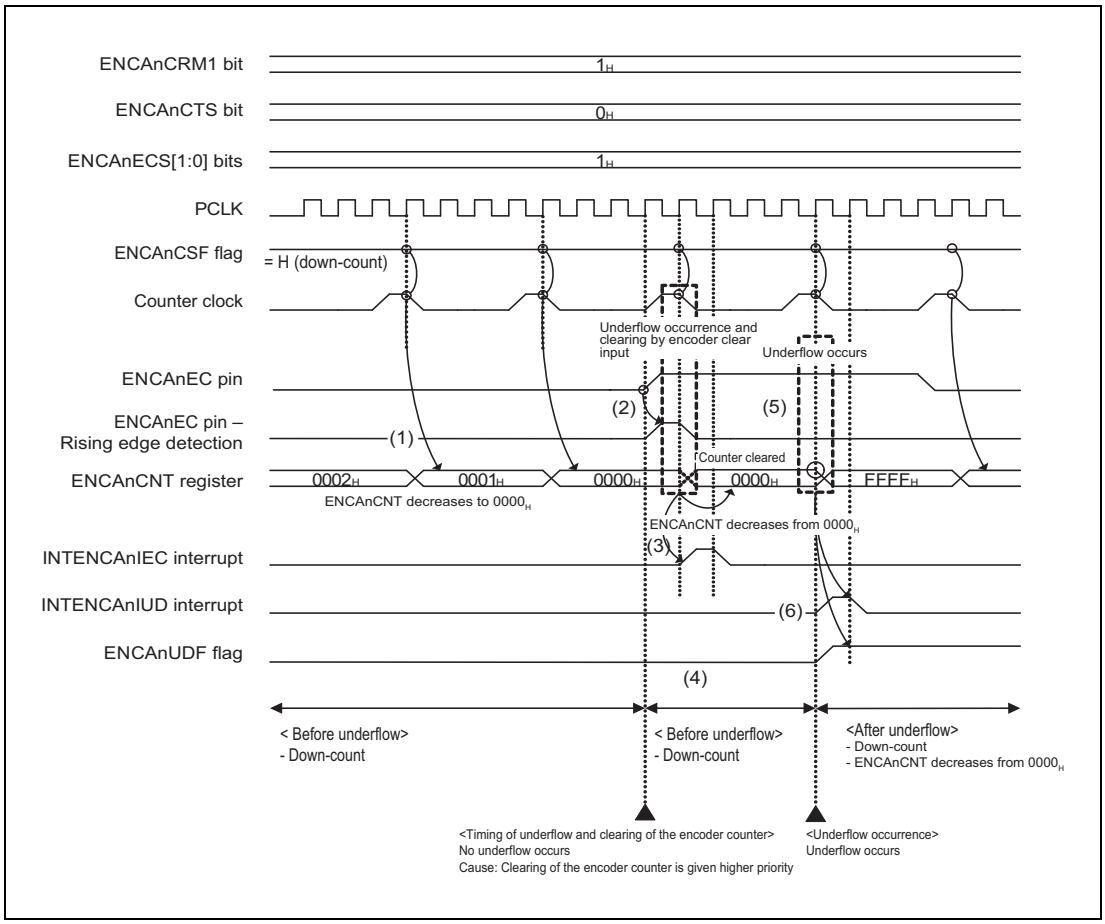


Figure 22.21 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin)

(1) Down-counting from 0002_H is continuously performed.

(2) When an underflow occurs if the counter value is 0000_H, and the rising edge of ENCAnEC is detected simultaneously, clear operation by the encoder clear input is performed. Even if the next clock signal is input during clear operation, the counter value remains at 0000_H.

(3) When the counter value is cleared by the encoder clear input, an encoder clear interrupt (INTENCAnIEC) by phase Z is output simultaneously. Because a clear operation by the encoder clear input is performed simultaneously with the underflow occurrence, an underflow interrupt is not output (An underflow does not occur. Clear operation is performed by the encoder clear input).

(4) Because an underflow does not occur as is the case with step 3, the underflow flag is not set.

(5) When down-counting is further performed after the counter value changes to 0000_H by clear operation by the encoder clear input, the counter value changes from 0000_H to $FFFF_H$, and an underflow occurs.

(6) When an underflow occurs, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

22.6.11 Overflow Operation Immediately after Startup

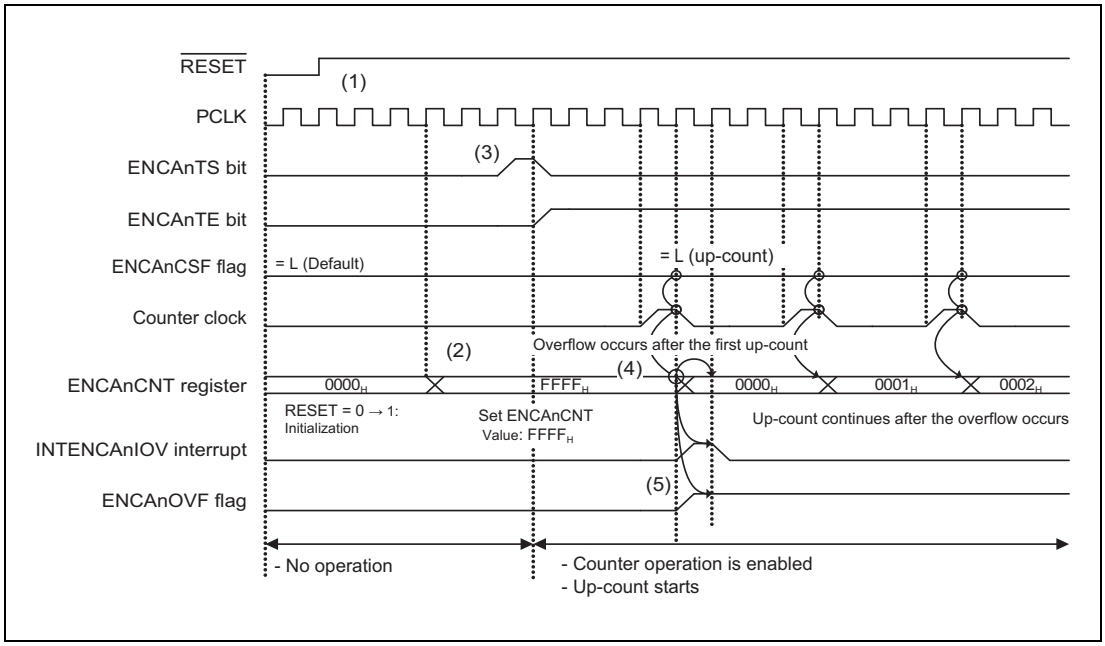


Figure 22.22 Overflow Operation Immediately after Startup

(1) When the $\overline{\text{RESET}}$ value changes from 0 to 1, the status is changes from “reset” to “reset release”.

(2) The timer counter is set to $FFFF_H$ as the initial value.

(3) ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.

(4) When up-counting is performed from $FFFF_H$ which is the initially set counter value, the counter value changes from $FFFF_H$ to 0000_H, and an overflow occurs immediately after operation starts.

(5) At the same time, by an overflow occurrence immediately after operation starts, an overflow interrupt (INTENCAnIOV) is output, and the overflow flag (ENCAnOVF) is set.

22.6.12 Underflow Operation Immediately after Startup

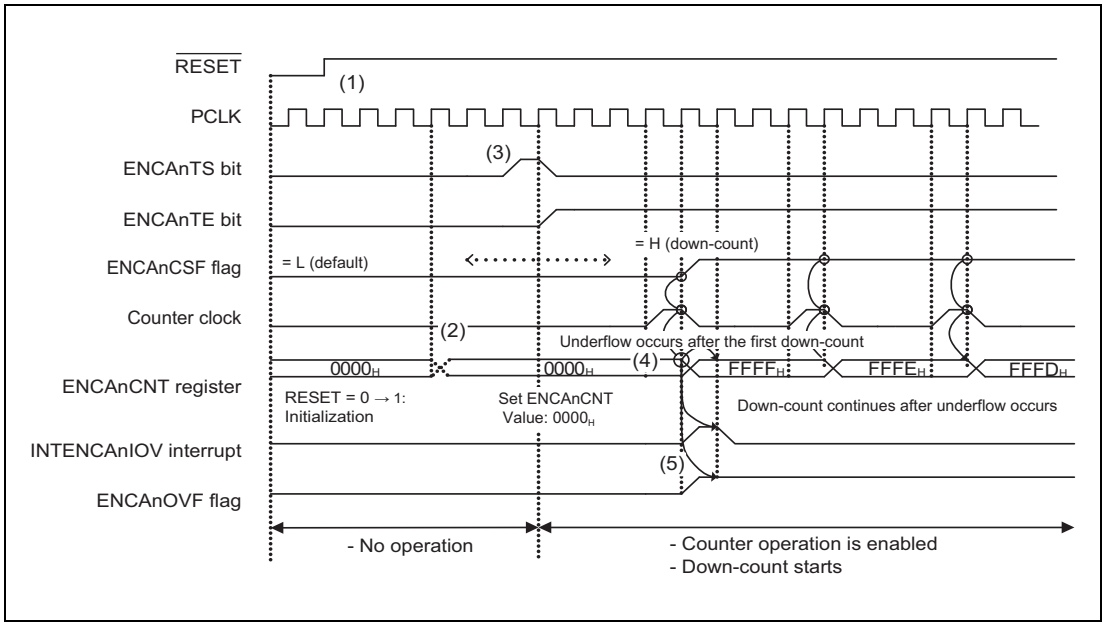


Figure 22.23 Underflow Operation Immediately after Startup

(1) When the $\overline{\text{RESET}}$ value changes from 0 to 1, the status is changes from “reset” to “reset release”.

(2) The timer counter is set to 0000_H as the initial value.

(3) ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.

(4) When down-counting is performed from 0000_H which is the initially set counter value, the counter value changes from 0000_H to $FFFF_H$, and an underflow occurs immediately after operation starts.

(5) At the same time, by an underflow occurrence immediately after operation starts, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

22.6.13 Using the ENCAnLDE Function Immediately after Startup

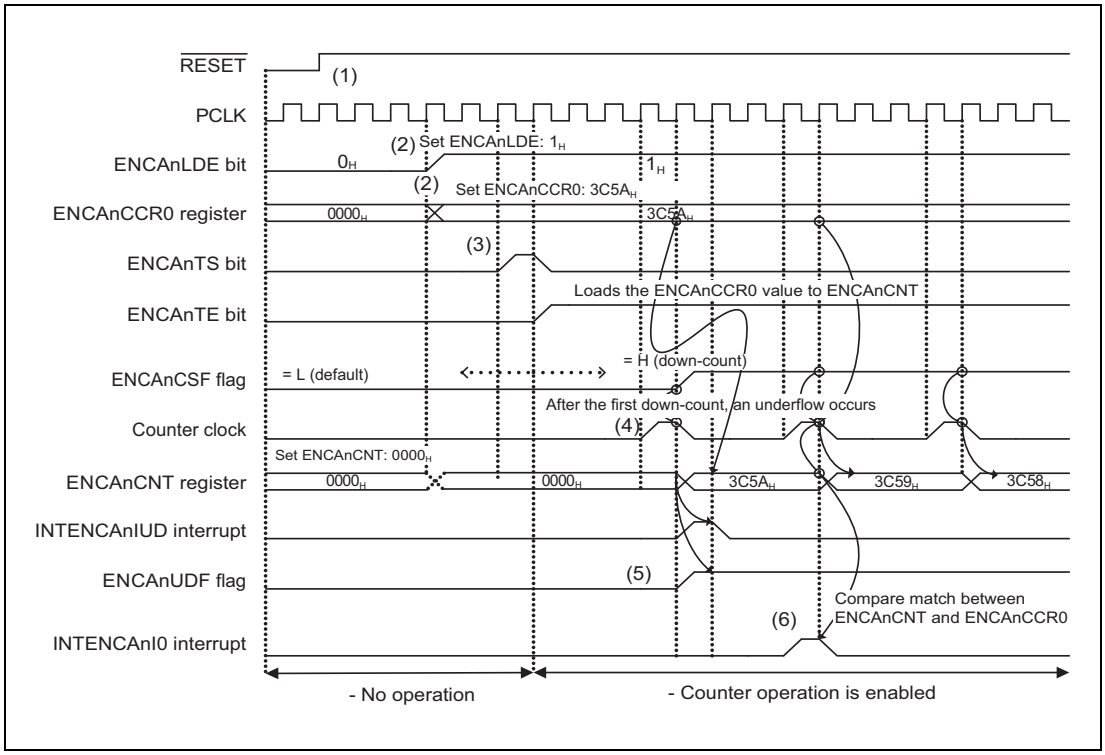


Figure 22.24 Using the ENCAnLDE Function Immediately after Startup

(1) When the $\overline{\text{RESET}}$ value changes from 0 to 1, the status is changes from “reset” to “reset release”.

(2) The load enable bit (ENCAnLDE) is set to 1, capture compare register 0 (ENCAnCCR0) is set to $3C5A_H$, and the timer counter is set to the initial value 0000_H.

(3) ENCAnTS is set to 1, and operation starts. ENCAnTE changes to 1, which indicates that operation is enabled.

(4) When down-counting is performed from 0000_H which is the initially set counter value, an underflow occurs immediately after operation starts. Because ENCAnLDE is set to 1, the ENCAnCCR0 value, $3C5A_H$, is loaded to the timer counter (INTENCAnI0 is not output during loading).

(5) At the same time, by an underflow occurrence immediately after operation starts, an underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set (after an underflow occurs, down-counting from the loaded value ($3C5A_H$) continues).

(6) After the ENCAnCCR0 value is loaded to ENCAnCNT, a match with ENCAnCCR0 is detected, and INTENCAnI0 is output.

22.6.14 ENCAnLDE Function (loading counter value)

(1) <When ENCAnLDE = 0>

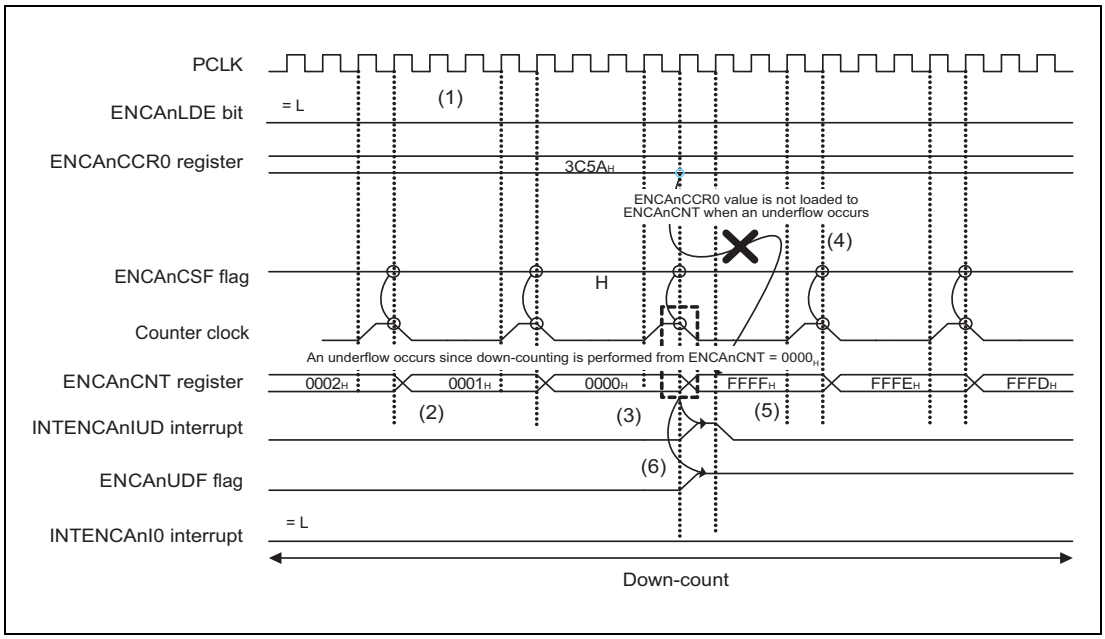


Figure 22.25 ENCAnLDE Function (when ENCAnLDE = 0)

(1) ENCAnLDE is set to 0 (even if an underflow occurs, the ENCAnCCR0 value is not loaded).

(2) Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$

(3) When down-counting is further performed after the counter value changes to 0000_H, an underflow occurs.

(4) Because ENCAnLDE is set to 0, the setting value of the ENCAnCCR0 register is not loaded to the counter even if an underflow occurs.

(5) Operation changes to underflow operation (counter value: $0000_H \rightarrow FFFF_H$).

(6) An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

(2) <When ENCAnLDE = 1>

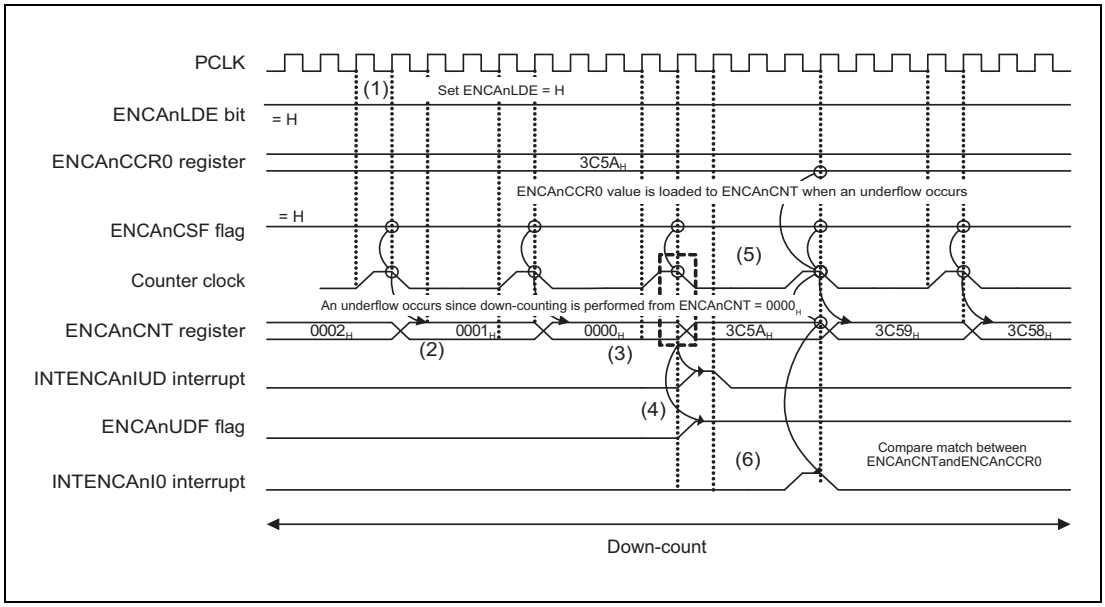


Figure 22.26 ENCAnLDE Function (when ENCAnLDE = 1)

(1) ENCAnLDE is set to 1 (if an underflow occurs, the ENCAnCCR0 value is loaded to the counter).

(2) Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$

(3) When down-counting is further performed after the counter value changes to 0000_H, an underflow occurs.

(4) An underflow interrupt is output, and the underflow flag is set.

(5) Because ENCAnLDE is set to 1, the setting value of the ENCAnCCR0 register is loaded to the counter if an underflow occurs. ENCAnCNT is set to $3C5A_H$.

(6) After the ENCAnCCR0 value is set to ENCAnCNT, if the ENCAnCNT value matches with the ENCAnCCR0 value on a counter clock, a compare match interrupt (INTENCAnI0) is output.

22.6.15 Conflict between ENCAnLDE Function (loading counter value) and Rewriting of ENCAnCCR0 Register

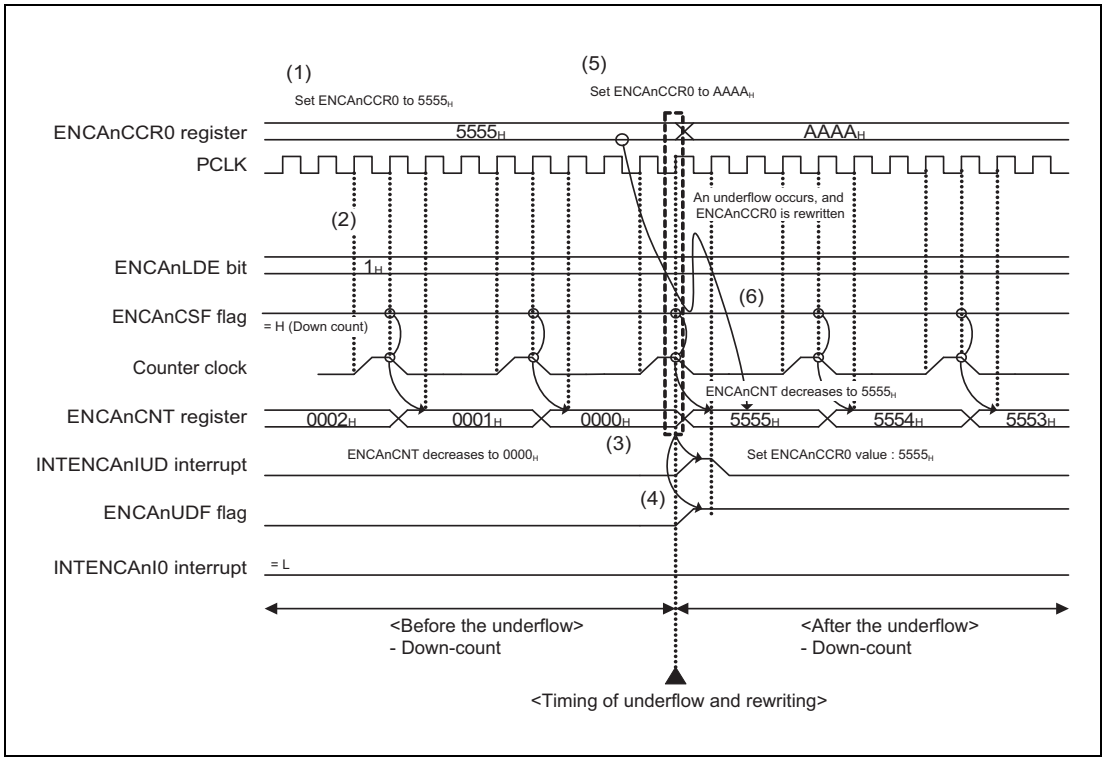


Figure 22.27 Conflict between ENCAnLDE Function and Rewriting of ENCAnCCR0 Register

(1) The ENCAnCCR0 register is currently set to 5555_H.

(2) ENCAnLDE is currently set to 1.

(3) Down-counting is performed ($0002_H \rightarrow 0001_H \rightarrow 0000_H$), and an underflow occurs.

(4) An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

(5) When an underflow occurs, the ENCAnCCR0 register value is changed from 5555_H to $AAAA_H$.

(6) Additionally, when an underflow occurs, the ENCAnCCR0 value before the rewrite was performed (5555_H) is set in ENCAnCNT.

22.6.16 Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input (ENCAnEC pin)

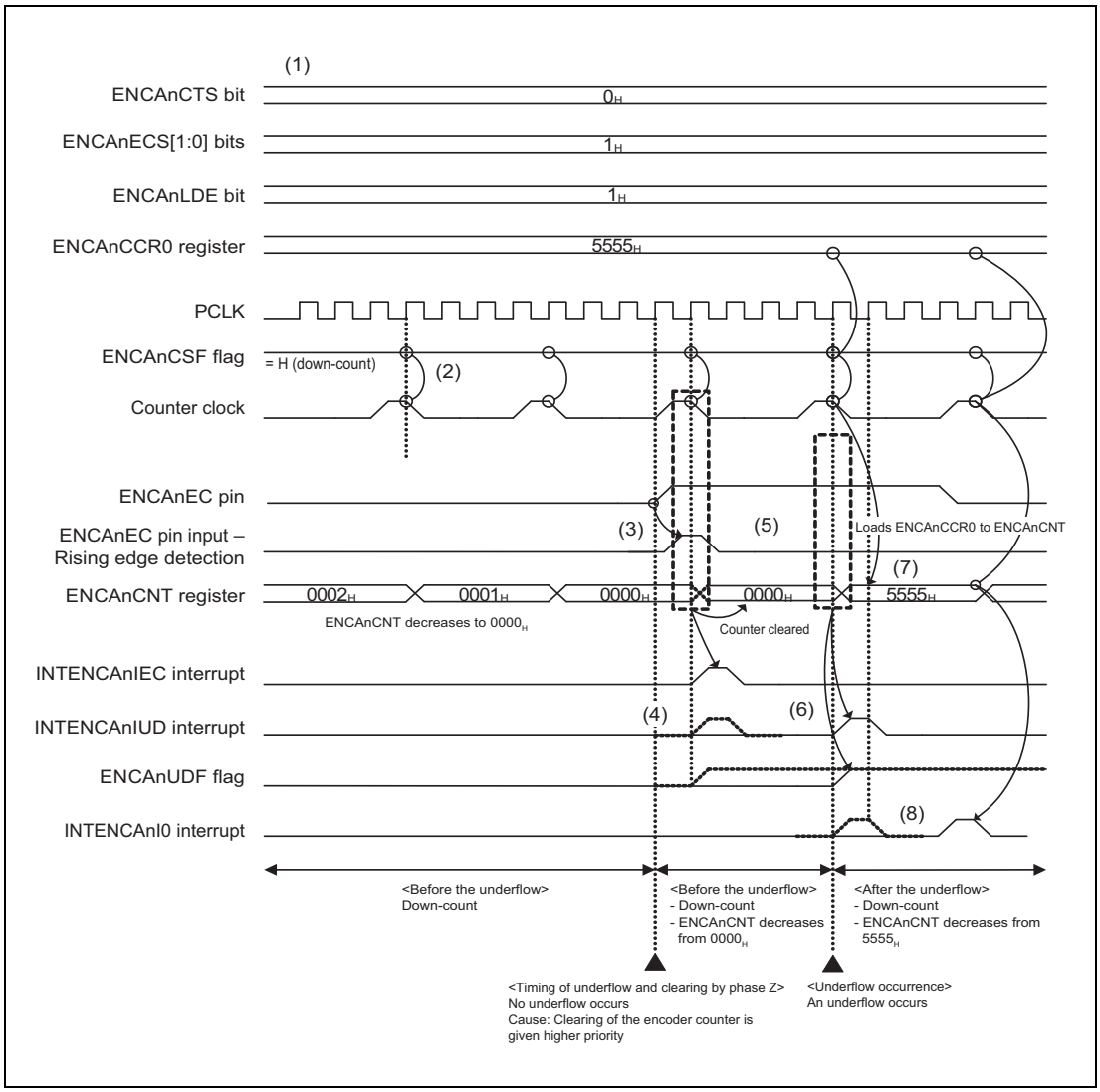


Figure 22.28 Conflict between ENCAnLDE Function and Clear Operation by Encoder Clear Input

(1) The values are set as follows: ENCAnCTS = 0, ENCAnECS[1:0] = 01_B, ENCAnLDE = 1, and ENCAnCCR0 = 5555_H.

(2) Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$

(3) When the counter value becomes 0000_H, the rising edge of the ENCAnEC pin is detected, and clear operation by the encoder clear input is performed.

(4) Because a count clear is performed when the counter value reaches 0000_H, a counter clear interrupt (INTENCAnIEC) by the encoder clear input is output. An underflow does not occur because down-counting is not performed when the counter value is 0000_H. Therefore, an underflow interrupt (INTENCAnIUD) is not output, and the underflow flag (ENCAnUDF) is not set.

(5) After the counter value is cleared to 0000_H by clear operation by the encoder clear input, down-counting is performed and an underflow occurs.

(6) An underflow interrupt (INTENCAnIUD) is output, and the underflow flag (ENCAnUDF) is set.

(7) Because ENCAnLDE = 1, if an underflow occurs, the ENCAnCCR0 value is loaded to ENCAnCNT.

(8) After the ENCAnCCR0 value is set to ENCAnCNT, a compare match is detected according to the counter clock. If the ENCAnCNT value matches with the ENCAnCCR0 value, a compare match interrupt (INTENCAnI0) is output.

22.6.17 Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input

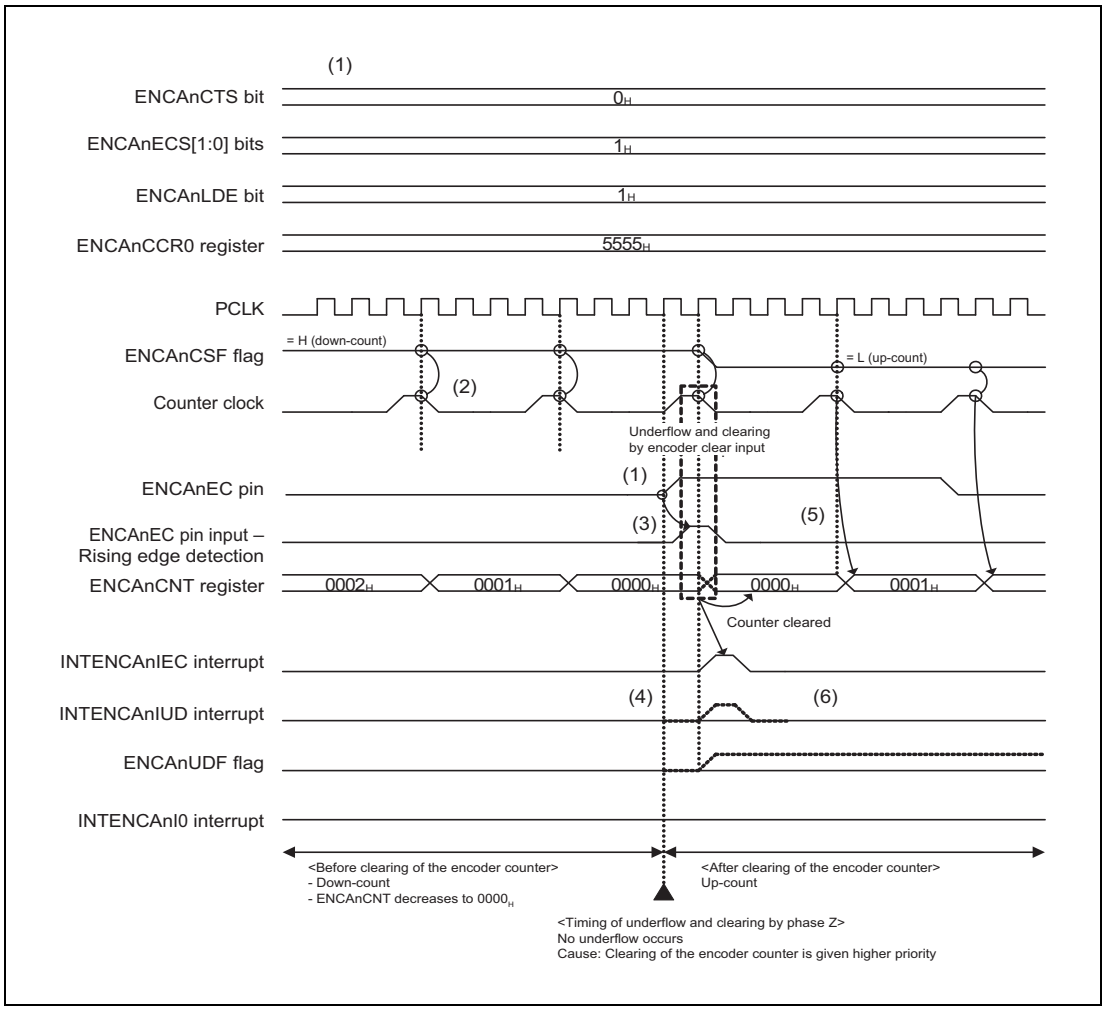


Figure 22.29 Up-counting after Conflict between ENCAnLDE Function and Clearing of the Encoder Counter

(1) The values are set as follows: ENCAnCTS = 0, ENCAnECS[1:0] = 01_B, ENCAnLDE = 1, and ENCAnCCR0 = 5555_H.

(2) Down-counting is performed: $0002_H \rightarrow 0001_H \rightarrow 0000_H$

(3) When the counter value becomes 0000_H, the rising edge of ENCAnEC is detected, and clear operation by the encoder clear input is performed.

(4) Because a count clear is performed when the counter value reaches 0000_H, a counter clear interrupt (INTENCAnIEC) by the encoder clear input is output. An underflow does not occur because down-counting is not performed when the counter value is 0000_H. Therefore, an underflow interrupt (INTENCAnIUD) is not output, and the underflow flag (ENCAnUDF) is not set.

(5) After the counter value is cleared to 0000_H by clear operation by the encoder clear input, up-counting is performed.

(6) An underflow interrupt (INTENCAnIUD) is not output, and the underflow flag (ENCAnUDF) is not set.

22.6.18 Capture Operation between Counter Clocks (ENCAnCCR1)

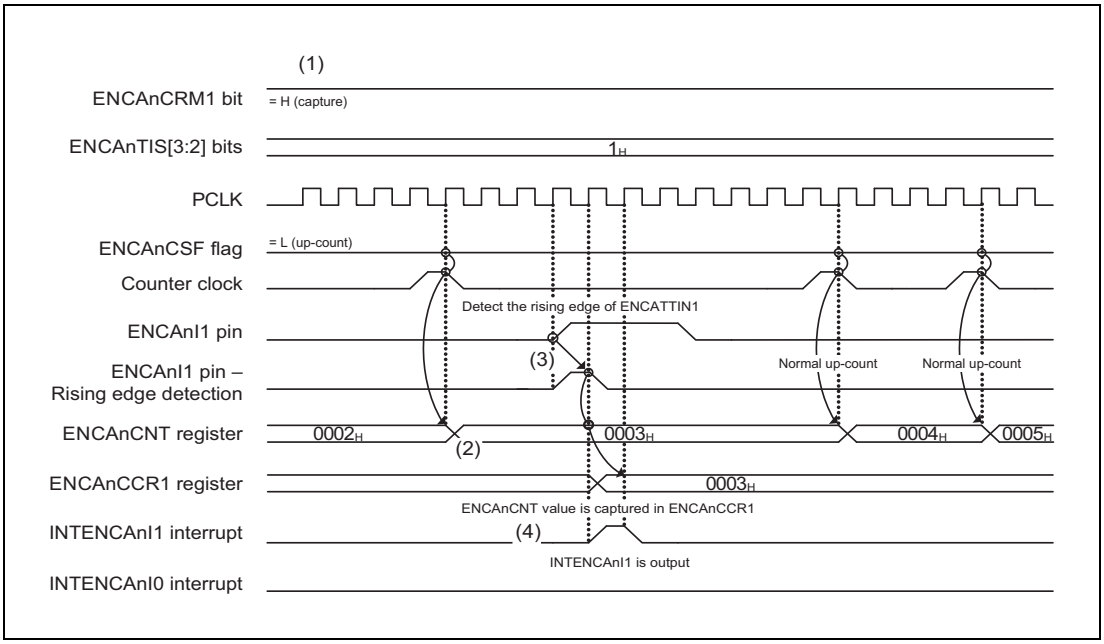


Figure 22.30 Capture Operation between Counter Clocks (ENCAnCCR1)

(1) The values are set as follows: ENCAnCRM1 = 1 and ENCAnTIS[3:2] = 01_B.

(2) Up-counting is performed.

(3) The rising edge of the ENCAnI1 input is detected, and the counter value is captured in ENCAnCCR1.

(4) An interrupt (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register is output.

22.6.19 Capture Operation between Counter Clocks (ENCAnCCR0)

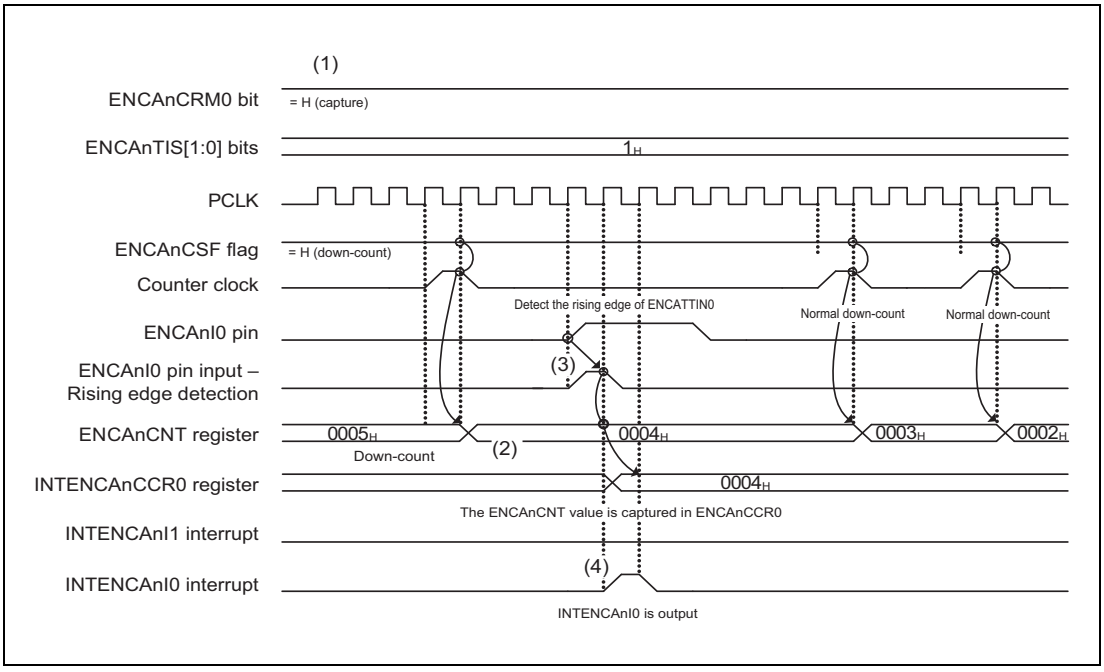


Figure 22.31 Capture Operation between Counter Clocks (ENCAnCCR0)

(1) The values are set as follows: ENCAnCRM0 = 1 and ENCAnTIS[1:0] = 01_B.

(2) Down-counting is performed.

(3) The rising edge of the ENCAnI0 input is detected, and the counter value is captured in ENCAnCCR0.

(4) An interrupt (INTENCAnI0) corresponding to the capture to the ENCAnCCR0 register is output.

22.6.20 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0

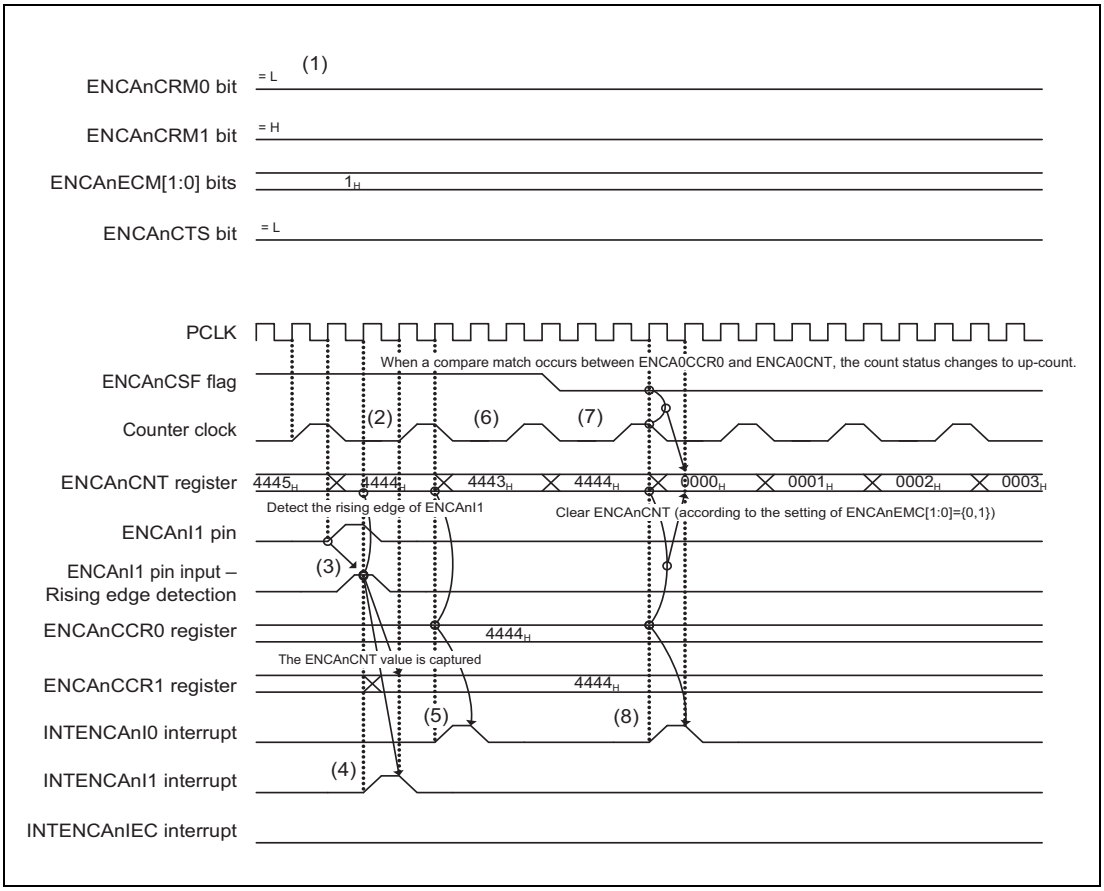


Figure 22.32 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0

(1) The values are set as follows: ENCAnCCR0 = 4444_H, ENCAnCRM0 = 0, ENCAnCRM1 = 1, ENCAnECM[1:0] = 01_B, and ENCAnCTS = 0.

(2) Down-counting is performed.

(3) The rising edge of the ENCAnI1 input is detected, and the ENCAnCNT value (4444_H) is captured in the ENCAnCCR1 register.

(4) An interrupt signal (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register is output.

(5) When a compare match occurs between ENCAnCNT (counted down from 4445_H to 4444_H) and ENCAnCCR0 (4444_H), a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.

(6) The counter operation changes to up-counting.

(7) When ENCAnCNT is counted up from 4443_H to 4444_H, a compare match with ENCAnCCR0 occurs again. Because the counter operation is up-counting when the compare match occurs, the counter value is cleared according to the setting of ENCAnECM[1:0] (01_B), and the ENCAnCNT value changes to 0000_H.

(8) When ENCAnCNT changes to 4444_H, a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.

22.6.21 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1

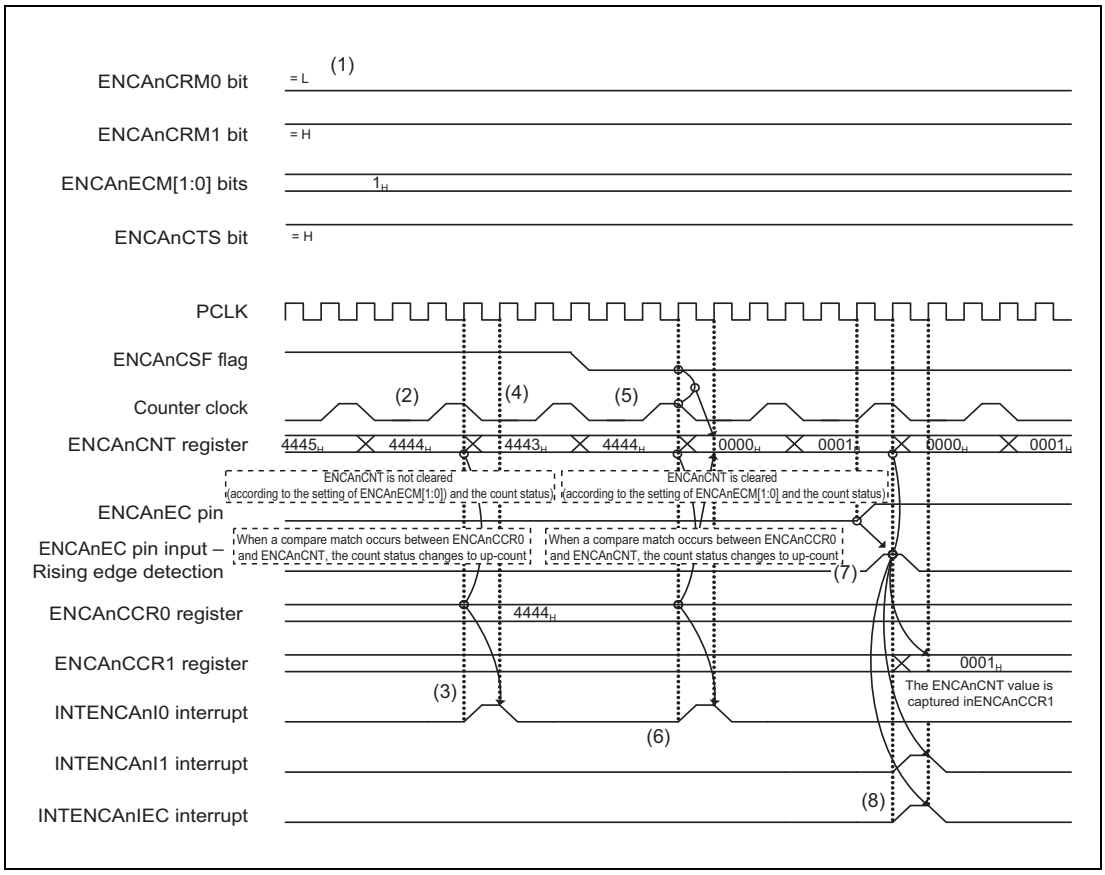


Figure 22.33 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1

(1) The values are set as follows: ENCAnCCR0 = 4444_H, ENCAnCRM0 = 0, ENCAnCRM1 = 1, ENCAnECM[1:0] = 01_B, and ENCAnCTS = 1.

(2) Down-counting is performed.

(3) When a compare match occurs between ENCAnCNT (counted down from 4445_H to 4444_H) and ENCAnCCR0 (4444_H), a compare/capture interrupt 0 (INTENCAnI0) is output.

(4) The counter operation changes to up-counting.

(5) When ENCAnCNT is counted up from 4443_H to 4444_H, a compare match with ENCAnCCR0 occurs again. Because the counter operation is up-counting when the compare match occurs, the counter value is cleared according to the setting of ENCAnECM[1:0] (01_B), and the ENCAnCNT value changes to 0000_H.

(6) When ENCAnCNT changes to 4444_H, a compare match interrupt (INTENCAnI0) with ENCAnCCR0 is output.

(7) After the counter value is cleared, up-counting is performed, and the counter value changes to 0001_H. At this point, the ENCAnCNT value (0001_H) is captured in ENCAnCCR1 by detecting the rising edge of the ENCAnEC signal, and the counter is cleared to 0000_H.

(8) An interrupt (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register and a clear interrupt (INTENCAnIEC) by ENCAnEC are output.

22.6.22 Encoder Operation when Compare Match Clear Control is Disabled

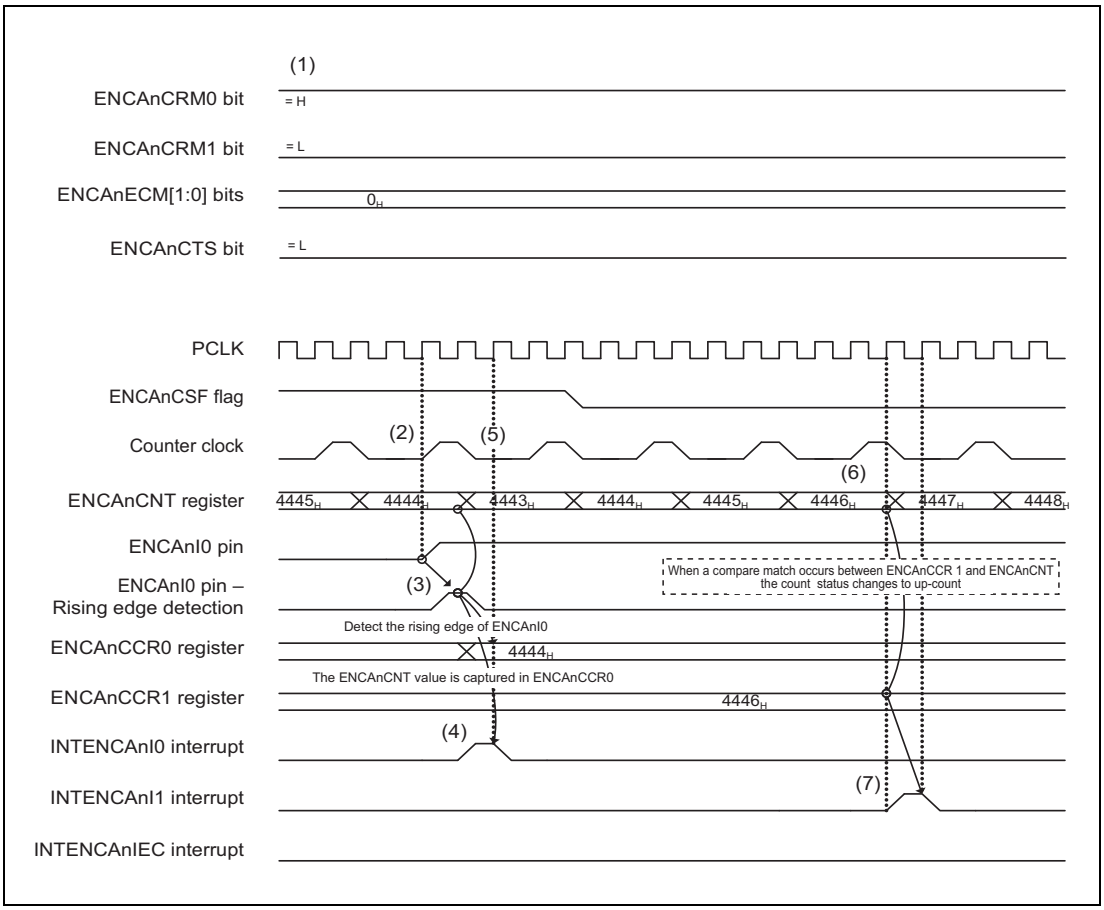


Figure 22.34 Encoder Operation when Compare Match Clear Control is Disabled

(1) The values are set as follows: ENCAnCCR1 = 4446_H, ENCAnCRM0 = 1, ENCAnCRM1 = 0, ENCAnECM[1:0] = 00_B, and ENCAnCTS = 0.

(2) Down-counting is performed.

(3) When the rising edge of ENCAnI0 is detected, the ENCAnCNT value (4444_H) is captured in ENCAnCCR0.

(4) An interrupt signal (INTENCAnI0) corresponding to the capture to the ENCAnCCR0 register is output.

(5) The counter operation changes to up-count.

(6) When ENCAnCNT changes to 4446_H, a compare match with ENCAnCCR1 is detected.

(7) A compare match interrupt (INTENCAnI1) with ENCAnCCR1 is output.

22.6.23 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1

22.6.23.1 Accompanying Capture Operation

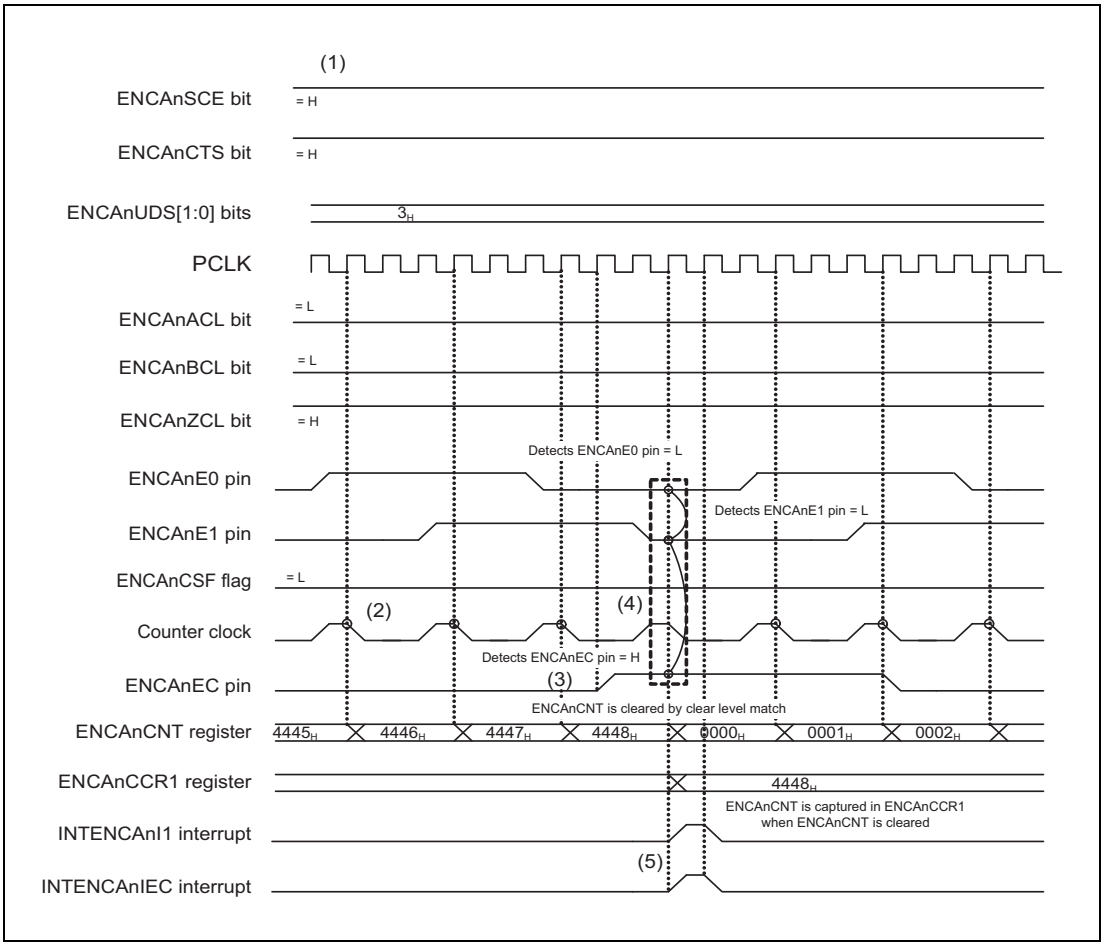


Figure 22.35 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1

(1) The values are set as follows: ENCAnSCE = 1, ENCAnCTS = 1, ENCAnUDS[1:0] = 11_B, ENCAnACL = 0, ENCAnBCL = 0, and ENCAnZCL = 1.

(2) Up-counting is performed.

(3) The counter value is not cleared upon the rising edge of ENCAnEC.

(4) When ENCAnE0, ENCAnE1, and ENCAnEC reach the set clear level, the counter value is cleared. The counter value is captured in ENCAnCCR1 at the time of the clearing.

(5) At the time of the clearing, an interrupt (INTENCAnI1) corresponding to the capture to the ENCAnCCR1 register and a clear interrupt (INTENCAnIEC) by ENCAnEC are output.

22.6.23.2 When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

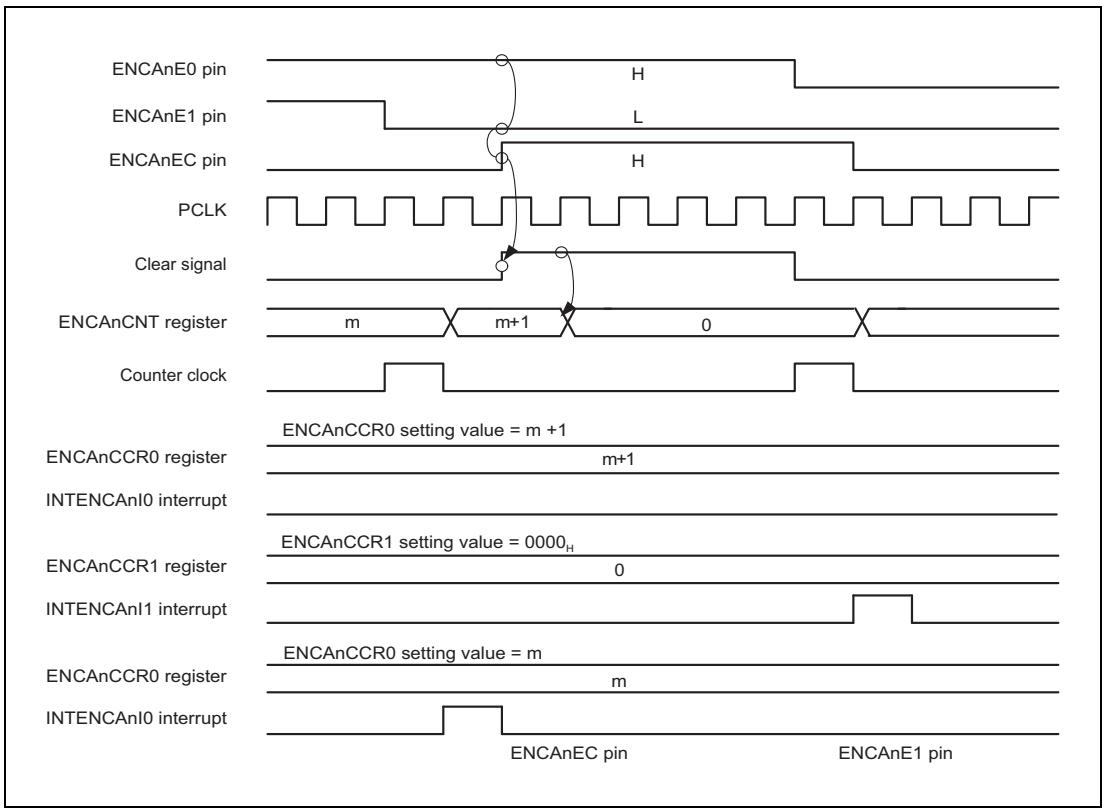


Figure 22.36 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count

22.6.23.3 When the Timing of the ENCAnEC Input is the Same as that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

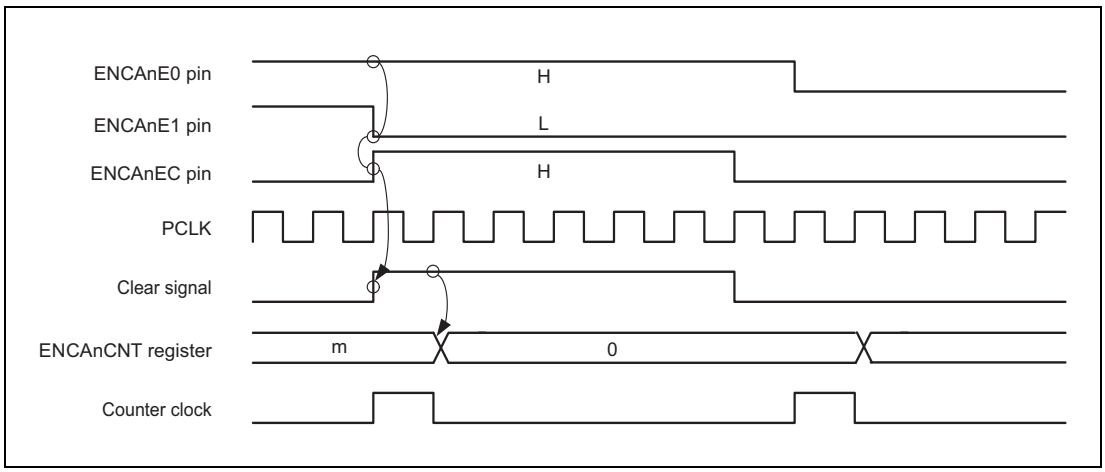


Figure 22.37 Clearing for when the Timing of the ENCAnEC Input is the Same as that of the ENCAnE1 Input during Up-count

22.6.23.4 When the Timing of the ENCAnEC Input is Earlier than that of the ENCAnE1 Input during Up-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

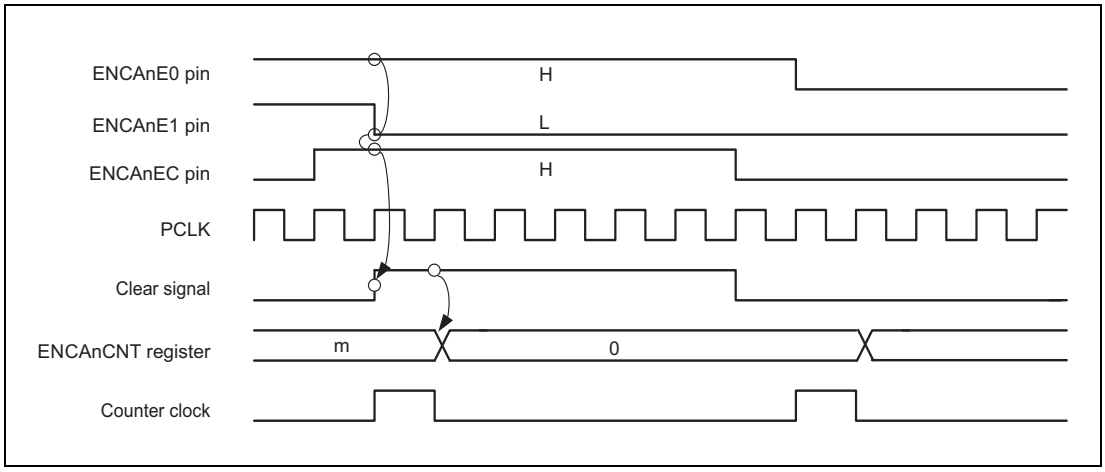


Figure 22.38 Clearing for when the Timing of the ENCAnEC Input is Earlier than that of the ENCAnE1 Input during Up-count

22.6.23.5 When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count

(When ENCAnACL = 1, ENCAnBCL = 0, ENCAnZCL = 1, and ENCAnUDS[1:0] = 11_B)

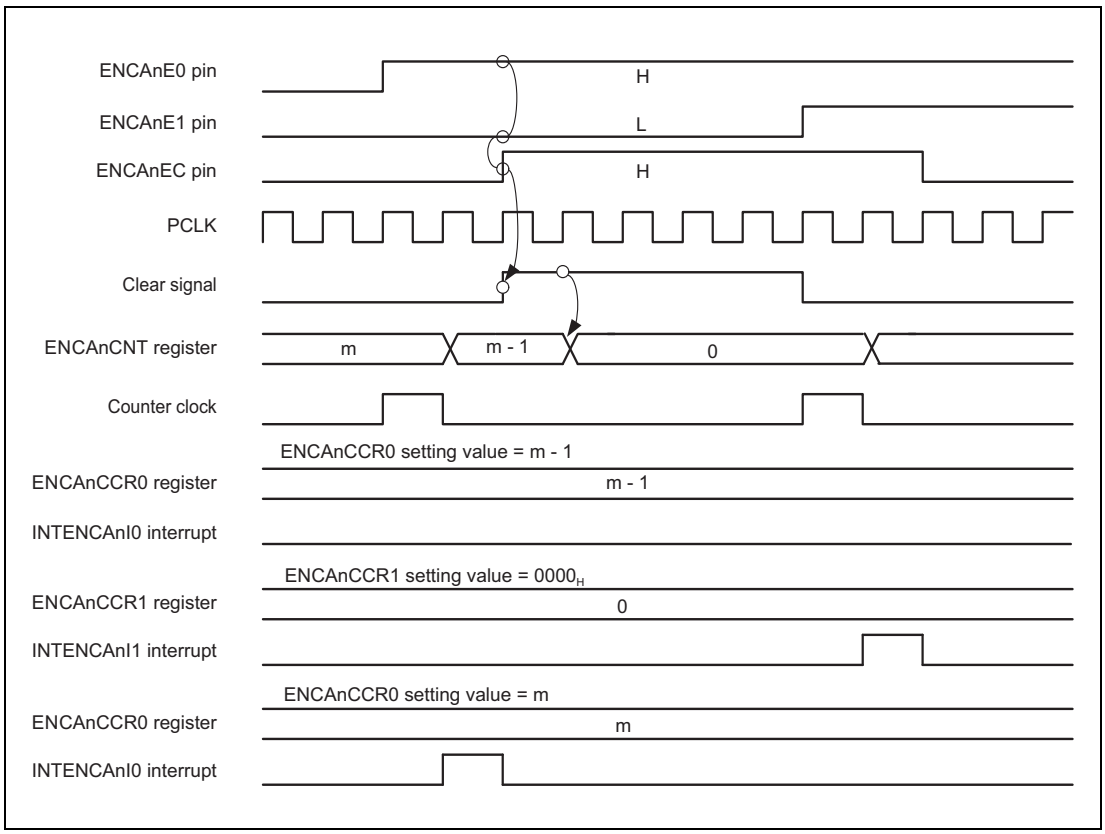


Figure 22.39 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count

22.6.24 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0

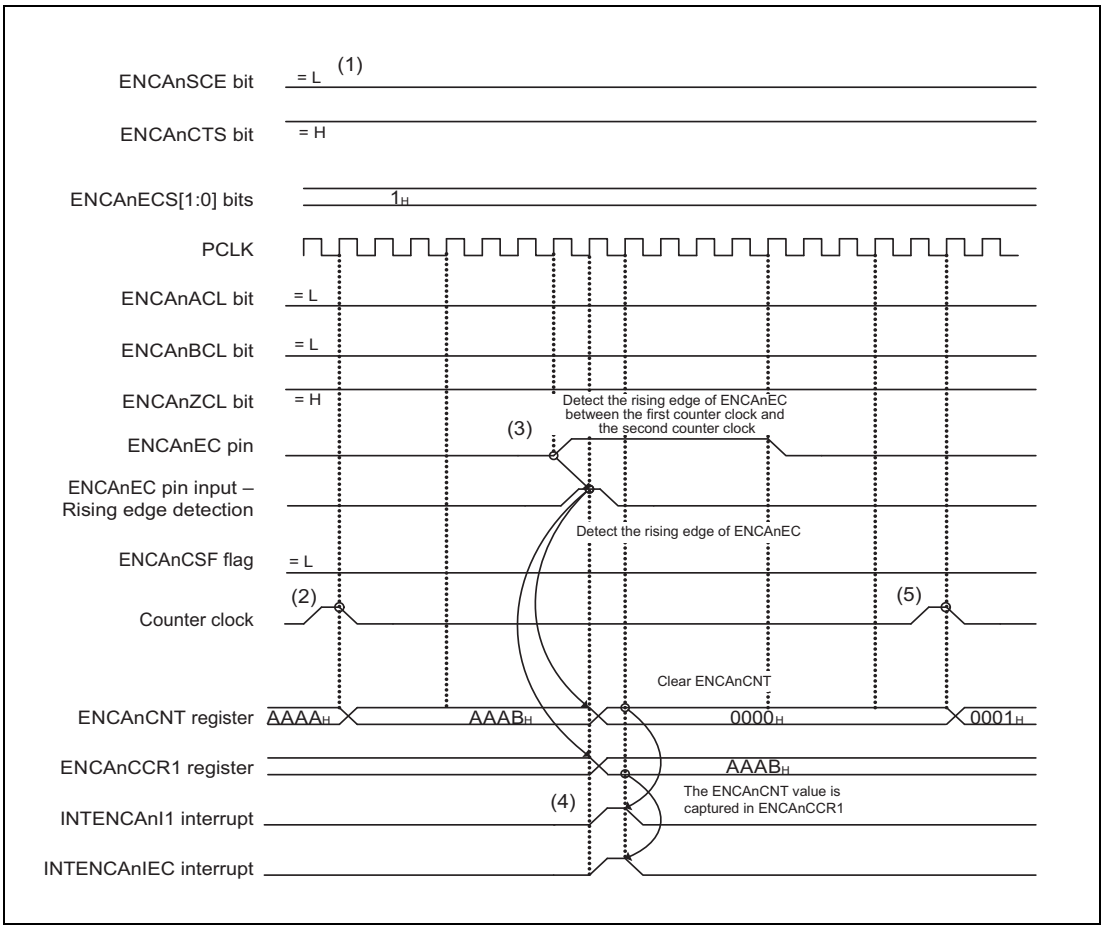


Figure 22.40 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0

(1) The values are set as follows: ENCAnSCE = 0, ENCAnCTS = 1, and ENCAnECS[1:0] = 01_B.

(2) Up-counting is performed.

(3) The rising edge of the ENCAnEC input is detected, and the ENCAnCNT value ($AAAB_H$) is captured in the ENCAnCCR1 register. Concurrently, clear operation by ENCAnEC is performed, and ENCAnCNT is cleared to 0000_H.

(4) A capture interrupt 1 (INTENCAnI1) to the ENCAnCCR1 register and an encoder clear interrupt (INTENCAnIEC) by ENCAnEC are output.

(5) After the counter value is cleared, up-counting is performed, and the counter value changes to 0001_H.

Section 23 Peripheral Interconnection (PIC)

23.1 Features of RH850/C1x PIC

This section contains a generic description of the peripheral interconnection (PIC). The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the PIC (PIC1A, PIC2B).

23.1.1 Number of Units

This microcontroller has the following number of units of the PIC.

Table 23.1 Number of Units (PIC1A)

Product	RH850/C1x
Number of Units	1
Name	PIC1A

Table 23.2 Number of Units (PIC2B)

Product	RH850/C1x
Number of Units	1
Name	PIC2B

Table 23.3 Index

Index	Meaning
n	The individual unit of each timer and A/D converter is identified by the index “n”.
m	The individual channel of each timer and A/D converter is identified by the index “m”.
x	The scan group number of A/D converter is indicated by the index “x”.
i	The variable used for descriptions is indicated by the index “i”.

23.1.2 Register Base Address

PIC base addresses are listed in the following table.

PIC register addresses are given as offsets from the base addresses in general.

Table 23.4 Register Base Address

Base Address Name	Base Address
<PIC1A_base>	FFDD 0000_H
<PIC2B_base>	FFDD 1000_H

23.1.3 Clock Supply

PIC clocks are listed in the following table.

Table 23.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
PIC1A	PCLK	CLKC_HSB (Unmodulated high-speed peripheral clock)
PIC2B	PCLK	CLKC_HSB (Unmodulated high-speed peripheral clock)

23.1.4 Reset Sources

PIC reset sources are listed in the following table. PIC is initialized by these reset sources.

Table 23.6 Reset Sources

Unit Name	Reset Source
PIC1A	Reset by all reset sources.
PIC2B	Reset by all reset sources.

23.1.5 External Input/Output Signals

External input/output signals of PIC are listed in the following table.

Table 23.7 PIC1A External Input/Output Signals

Unit Signal Name	Description	Alternative Port Pin Signal Name
ENCAnI1	ENCAn capture trigger input 1	ENCAnTIN1
ENCAnE0	ENCAn encoder input (count pulse 0)	ENCAnE0
ENCAnE1	ENCAn encoder input (count pulse 1)	ENCAnE1
ENCAnEC	ENCAn encoder input (clear pulse)	ENCAnEC
TAUDnTINm	TAUDn channel m	TAUDnIm
ESOn	Hi-Z control	TAPAnESO
TSG3nO1-6	TSG3n channel output 1 to 6	TSG3nO1-6
TOPnU	Motor control output U phase	TAPAnUP
TOPnUB	Motor control output UB phase	TAPAnUN
TOPnV	Motor control output V phase	TAPAnVP
TOPnVB	Motor control output VB phase	TAPAnVN
TOPnW	Motor control output W phase	TAPAnWP
TOPnWB	Motor control output WB phase	TAPAnWN

Table 23.8 PIC2B External Input/Output Signals

Unit Signal Name	Outline	Alternative Port Pin Signal Name
ADTRGnZ	ADCCn trigger	ADCCnTRG

23.2 Peripheral Interconnection 1 (PIC1A)

23.2.1 Overview

23.2.1.1 Functional Overview

The peripheral interconnection 1 (PIC1A) handles synchronous operation using multiple timers and by connecting the timer internal I/O signals between the timers.

CAUTIONS

- **The signal names used in the following descriptions are abbreviations. The actual signal names corresponding to each abbreviation are as follows:**
  INTm → TAUDnTINTm
  TINm → TAUDnTTINm
  TOUTm → TAUDnTTOUTm
  CDRm → TAUDnCDRm
  CNTm → TAUDnCNTm
- **The functions of the ENCA internal signals used in the following descriptions are as follows:**
  ENCATnEQ0: Internal signals to be output one PCLK cycle before INTENCAnI0 interrupt signals
  ENCATnEQ1: Internal signals to be output one PCLK cycle before INTENCAnI1 interrupt signals
  ENCATnIEC: Internal signals of INTENCAnIEC interrupt signals

The PIC1A has the following functions.

- Simultaneous start trigger function
- PWM output function with dead time
- High accuracy triangle wave PWM output function with dead time
- Delay pulse output function with dead time
- Trigger and pulse width measurement function
- Encoder capture trigger select function
- Two-phase encoder control function (control method 1)
- Two-phase encoder control function (control method 2)
- Two-phase encoder control function (control method 3)
- Three-phase pulse input control function
- Three-phase encoder control function
- ENCA input select function
- TAUD input select function
- Switch function between TSG output and low/high level output
- Hi-Z control function

23.2.2 Registers

23.2.2.1 List of Registers

The registers are listed in the following table.

The bits can be accessed only in 32-bit units. Access in 16-bits or 8-bits is operated as 32-bit access.

See **Section 23.1.2, Register Base Address** for <PIC1A_base>.

Table 23.9 Registers

Register	Symbol	Address
Simultaneous start trigger control register	PIC1ASST	<PIC1A_base> + 04_H
Simultaneous start control register 0	PIC1ASSER0	<PIC1A_base> + 10_H
Simultaneous start control register 1	PIC1ASSER1	<PIC1A_base> + 14_H
Simultaneous start control register 2	PIC1ASSER2	<PIC1A_base> + 18_H
Simultaneous start control register 3	PIC1ASSER3	<PIC1A_base> + $1C_H$
RS flip-flop circuit initialization register 00	PIC1AINI00	<PIC1A_base> + 20_H
DT initialization register 01	PIC1AINI01	<PIC1A_base> + 24_H
RS flip-flop circuit initialization register 10	PIC1AINI10	<PIC1A_base> + $2C_H$
DT initialization register 11	PIC1AINI11	<PIC1A_base> + 30_H
TSG30 output low/high level select register	PIC1ALHSEL0	<PIC1A_base> + 60_H
TSG30 output control register	PIC1ATSGOUTCTR0	<PIC1A_base> + 64_H
TSG31 output low/high level select register	PIC1ALHSEL1	<PIC1A_base> + 68_H
TSG31 output control register	PIC1ATSGOUTCTR1	<PIC1A_base> + $6C_H$
Hall sensor input select register	PIC1ATSGHALLSEL	<PIC1A_base> + 74_H
TAUD0 input select register	PIC1ATAUD0SEL	<PIC1A_base> + 78_H
TAUD1 input select register	PIC1ATAUD1SEL	<PIC1A_base> + $7C_H$
Hi-Z control register 0	PIC1AHIZCEN0	<PIC1A_base> + 80_H
Hi-Z control register 1	PIC1AHIZCEN1	<PIC1A_base> + 84_H
Hi-Z control register 2	PIC1AHIZCEN2	<PIC1A_base> + 88_H
Hi-Z control register 3	PIC1AHIZCEN3	<PIC1A_base> + $8C_H$
ENCATIN 1 input select register 400	PIC1AENCSEL400	<PIC1A_base> + $B8_H$
ENCATIN 1 input select register 410	PIC1AENCSEL410	<PIC1A_base> + BC_H
Timer input/output control register 200	PIC1AREG200	<PIC1A_base> + $C0_H$
Timer input/output control register 201	PIC1AREG201	<PIC1A_base> + $C4_H$
Timer input/output control register 202	PIC1AREG202	<PIC1A_base> + $C8_H$
Timer input/output control register 203	PIC1AREG203	<PIC1A_base> + CC_H
Timer input/output control register 210	PIC1AREG210	<PIC1A_base> + $D4_H$
Timer input/output control register 211	PIC1AREG211	<PIC1A_base> + $D8_H$
Timer input/output control register 212	PIC1AREG212	<PIC1A_base> + DC_H
Timer input/output control register 213	PIC1AREG213	<PIC1A_base> + $E0_H$
Timer input/output control register 30	PIC1AREG30	<PIC1A_base> + $E8_H$
Timer input/output control register 31	PIC1AREG31	<PIC1A_base> + EC_H
Timer input/output control register 50	PIC1AREG50	<PIC1A_base> + $F8_H$
Timer input/output control register 51	PIC1AREG51	<PIC1A_base> + FC_H

Combinations of registers used for each function are listed in the following table

Table 23.10 Registers Used by Each Function

Section Number	Function Name	PIC1ASST	PIC1ASSER				PIC1AINI				PIC1ALHSEL0	PIC1ATSGOUTCTR0	PIC1ALHSEL1	PIC1ATSGOUTCTR1	PIC1AHALLSEL	PIC1ATSGHALLSEL	PIC1ATAUD0SEL	PIC1ATAUD1SEL	PIC1AHIZCEN				PIC1AENCSEL4		PIC0REG											
			0	1	2	3	00	01	10	11	—	—	—	—	—	—	—	—	0	1	2	3	00	10	200	201	202	203	210	211	212	213	30	31	50	51
23.2.3.1	Simultaneous start trigger function	√	√	√	√	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
23.2.3.2	PWM output function with dead time	—	—	—	—	—	√	—	√	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	—	√	√	√	?	√	√	√	—	—	—	—
23.2.3.3	High accuracy triangle wave PWM output function with dead time	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	√	√	√	√	√	√	√	√	—	—	—	—
23.2.3.4	Delay pulse output function with dead time	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	—	√	√	√	?	√	√	√	—	—	—	—
23.2.3.5	Trigger and pulse width measurement function	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—
23.2.3.6	Encoder capture trigger select function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	—	—	√	—	—	—
23.2.3.7	Two-phase encoder control function (control method 1)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
23.2.3.8	Two-phase encoder control function (control method 2)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
23.2.3.9	Two-phase encoder control function (control method 3)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
23.2.3.10	Three-phase pulse input control function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	—	—	—	—	—	—	√	—	—	—	√	—	—	—	—	—	√	√
23.2.3.11	Three-phase encoder control function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	√	√
23.2.3.12	ENCA input select function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—
23.2.3.13	TAUD input select function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	—	—	—	—	—	√	—	√	—	√	—	√	—	—	√	—	—
23.2.3.14	Switch function between TSG output and low/high level output	—	—	—	—	—	—	—	—	—	√	√	√	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
23.2.3.15	Hi-Z control function	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	—	—	—	—	—	—	—	—	—	—	—	—	—	—

23.2.2.2 PIC1ASST — Simultaneous Start Trigger Control Register

The PIC1ASST register is an 8-bit register that selects the simultaneous start trigger.

Address: FFDD 0004_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1ASYNCTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 23.11 PIC1ASST Register Contents

Bit Position	Bit Name	Function
7 to 1	–	Reserved These bits are read as 0. The write value should be 0.
0	PIC1ASYNCTRG	Generates a start trigger for the timer whose simultaneous start is enabled. 0: Disabled 1: Simultaneous start trigger (the pulse of the width of 1PCLK is output.)

Note: PIC1ASYNCTRG reads 0 when read.

23.2.2.3 PIC1ASSER0 — Simultaneous Start Control Register 0

The PIC1ASSER0 register enables a start trigger for each channel of TAUD0.

Address: FFDD 0010$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1ASSER015	PIC1ASSER014	PIC1ASSER013	PIC1ASSER012	PIC1ASSER011	PIC1ASSER010	PIC1ASSER009	PIC1ASSER008	PIC1ASSER007	PIC1ASSER006	PIC1ASSER005	PIC1ASSER004	PIC1ASSER003	PIC1ASSER002	PIC1ASSER001	PIC1ASSER000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.12 PIC1ASSER0 Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC1ASSER015 to PIC1ASSER000	Enable or disable a simultaneous start trigger for CHm in the TAUD0 timer. 0: Disabled 1: Enabled

23.2.2.4 PIC1ASSER1 — Simultaneous Start Control Register 1

The PIC1ASSER1 register enables a start trigger for each channel of TAUD1.

Address: FFDD 0014$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1ASSER115	PIC1ASSER114	PIC1ASSER113	PIC1ASSER112	PIC1ASSER111	PIC1ASSER110	PIC1ASSER109	PIC1ASSER108	PIC1ASSER107	PIC1ASSER106	PIC1ASSER105	PIC1ASSER104	PIC1ASSER103	PIC1ASSER102	PIC1ASSER101	PIC1ASSER100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.13 PIC1ASSER1 Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC1ASSER115 to PIC1ASSER100	Enable or disable a simultaneous start trigger for CHm in the TAUD1 timer. 0: Disabled 1: Enabled

23.2.2.5 PIC1ASSER2 — Simultaneous Start Control Register 2

The PIC1ASSER2 register enables a start trigger of TAUJ0, TSG3n, TPBAn, and ENCAn.

Address: FFDD 0018_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PIC1ASSER213	PIC1ASSER212	PIC1ASSER211	PIC1ASSER210	PIC1ASSER209	PIC1ASSER208	—	—	—	—	PIC1ASSER203	PIC1ASSER202	PIC1ASSER201	PIC1ASSER200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 23.14 PIC1ASSER2 Register Contents

Bit Position	Bit Name	Function
15, 14	—	Reserved These bits are read as 0. The write value should be 0.
13	PIC1ASSER213	Enables or disables a simultaneous start trigger for the ENCA1. 0: Disabled 1: Enabled
12	PIC1ASSER212	Enables or disables a simultaneous start trigger for the ENCA0. 0: Disabled 1: Enabled
11	PIC1ASSER211	Enables or disables a simultaneous start trigger for the TPBA1. 0: Disabled 1: Enabled
10	PIC1ASSER210	Enables or disables a simultaneous start trigger for the TPBA0. 0: Disabled 1: Enabled
9	PIC1ASSER209	Enables or disables a simultaneous start trigger for the TSG31. 0: Disabled 1: Enabled
8	PIC1ASSER208	Enables or disables a simultaneous start trigger for the TSG30. 0: Disabled 1: Enabled
7 to 4	—	Reserved These bits are read as 0. The write value should be 0.
3	PIC1ASSER203	Enables or disables a simultaneous start trigger for CH03 of TAUJ0. 0: Disabled 1: Enabled
2	PIC1ASSER202	Enables or disables a simultaneous start trigger for CH02 of TAUJ0. 0: Disabled 1: Enabled
1	PIC1ASSER201	Enables or disables a simultaneous start trigger for the CH01 of TAUJ0. 0: Disabled 1: Enabled
0	PIC1ASSER200	Enables or disables a simultaneous start trigger for the CH00 of TAUJ0. 0: Disabled 1: Enabled

23.2.2.6 PIC1ASSER3 — Simultaneous Start Control Register 3

The PIC1ASSER3 register enables a start trigger for each channel of OSTMn.

Address: FFDD 001C$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PIC1ASSER302	PIC1ASSER301	PIC1ASSER300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 23.15 PIC1ASSER3 Register Contents

Bit Position	Bit Name	Function
15 to 3	—	Reserved These bits are read as 0. The write value should be 0.
2	PIC1ASSER302	Enables or disables a simultaneous start trigger for OSTM2. 0: Disabled 1: Enabled
1	PIC1ASSER301	Enables or disables a simultaneous start trigger for OSTM1. 0: Disabled 1: Enabled
0	PIC1ASSER300	Enables or disables a simultaneous start trigger for OSTM0. 0: Disabled 1: Enabled

23.2.2.7 PIC1AINIn0 — Flip-flop Circuit Initialization Register n0

The PIC1AINIn0 register enables initialization of the RS flip-flop circuits 4 to 2 (RSn4 to 2).

Address: FFDD 0020$_H$ (n = 0), FFDD 002C$_H$ (n = 1)

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	PIC1AINIn04	PIC1AINIn03	PIC1AINIn02	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	W	W	W	R	R

Table 23.16 PIC1AINIn0 Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are read as 0. The write value should be 0.
4 to 2	PIC1AINIn0[4:2]	Enable or disable initialization of the RS flip-flop circuits 4 to 2 (RSn4 to RSn2) used for the PWM output function with dead time. These bits read 0 when read. 0: Disabled 1: Initialized
1, 0	—	Reserved These bits are read as 0. The write value should be 0.

23.2.2.8 PIC1AINIn1 — DT Initialization Register n1

The PIC1AINIn1 register enables initialization of the latch & toggle (DT) circuit.

Address: FFDD 0024_H (n = 0), FFDD 0030_H (n = 1)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1AINIn12	PIC1AINIn11	PIC1AINIn10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	W	W	W

Table 23.17 PIC1AINIn1 Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Reserved These bits are read as 0. The write value should be 0.
2 to 0	PIC1AINIn1[2:0]	Enable or disable initialization of the DT circuit to be used for the trigger and pulse width measurement function. These bits read 0 when read. 0: Disabled 1: Initialized

23.2.2.9 PIC1ALHSEL0 — TSG30 Output Low/High Level Select Register

The PIC1ALHSEL0 register selects low/high level output of the TSG30 output.

Address: FFDD 0060_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	PIC1ALHSEL06	PIC1ALHSEL05	PIC1ALHSEL04	PIC1ALHSEL03	PIC1ALHSEL02	PIC1ALHSEL01	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 23.18 PIC1ALHSEL0 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved This bit is read as 0. The write value should be 0.
6 to 1	PIC1ALHSEL0m	These bits are applied to TSG30 output [6:1] and PIC1ALHSEL[6:1]. 0: Low level output. 1: High level output.
0	—	Reserved This bit is read as 0. The write value should be 0.

23.2.2.10 PIC1ATSGOUTCTR0 — TSG30 Output Control Register

The PIC1ATSGOUTCTR0 register selects the output type of TSG30 output signal.

Address: FFDD 0064$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	PIC1ASEL06	PIC1ASEL05	PIC1ASEL04	PIC1ASEL03	PIC1ASEL02	PIC1ASEL01	PIC1ASEL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.19 PIC1ATSGOUTCTR0 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved This bit is read as 0. The write value should be 0.
6 to 1	PIC1ASEL0m	Select the output signal from either TSG30 output or low/high level output. 0: TSG30 output 1: Low/high level output
0	PIC1ASEL00	Switches on/off of the function to output low/high level. 0: Off (only TSG30 output is available)*1 1: On (TSG30 output and low/high level output can be switched)*1

Note 1. When low/high level output function of TSG30 output is turned on, a delay of one cycle of the clock (CLKC_HSB) is generated from the output when the function is turned off.

23.2.2.11 PIC1ALHSEL1 — TSG31 Output Low/High Level Select Register

The PIC1ALHSEL1 register selects low/high level output of the TSG31 output.

Address: FFDD 0068$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	PIC1ALHSEL16	PIC1ALHSEL15	PIC1ALHSEL14	PIC1ALHSEL13	PIC1ALHSEL12	PIC1ALHSEL11	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 23.20 PIC1ALHSEL1 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved This bit is read as 0. The write value should be 0.
6 to 1	PIC1ALHSEL1m	These bits are applied to TSG31 output [6:1] and PIC1ALHSEL[6:1]. 0: Low level output 1: High level output
0	—	Reserved This bit is read as 0. The write value should be 0.

23.2.2.12 PIC1ATSGOUTCTR1 — TSG31 Output Control Register

The PIC1ATSGOUTCTR1 register selects the output type of TSG31 output.

Address: FFDD 006C$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	PIC1ASEL16	PIC1ASEL15	PIC1ASEL14	PIC1ASEL13	PIC1ASEL12	PIC1ASEL11	PIC1ASEL10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.21 PIC1ATSGOUTCTR1 Register Contents

Bit Position	Bit Name	Function
7	—	Reserved This bit is read as 0. The write value should be 0.
6 to 1	PIC1ASEL1m	Select the output signal between TSG31 output and low/high-level output. 0: TSG31 output 1: Low/high level output
0	PIC1ASEL10	Switches on/off the function to output low/high level. 0: Off (only TSG31 output is available)*1 1: On (TSG31 output and low/high level output can be switched)*1

Note 1. When low/high level output function of TSG31 output is turned on, a delay of one cycle of the clock (CLKC_HSB) is generated from the output when the function is turned off.

23.2.2.13 PIC1ATSGHALLSEL — Hall Sensor Input Select Register

The PIC1ATSGHALLSEL register sets the pin conditions to input the external hall sensor signal.

Address: FFDD 0074$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC1A TSG1HALLSEL	PIC1A TSG0HALLSEL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 23.22 PIC1ATSGHALLSEL Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are read as 0. The write value should be 0.
1	PIC1ATSG1HALLSEL	Sets pin condition to input the external hall sensor signal.*1 0: Separate input 1: Alternative input with ENCA
0	PIC1ATSG0HALLSEL	Sets pin condition to input the external hall sensor signal. *1 0: Separate input 1: Alternative input with ENCA

Note 1. For the products of C1x series, set this register to 1 because the external hall sensor input pins are also used as ENCA input pins. Set the bit 0 of the PIC1AREG50 register and the PIC1AREG51 register as the following tables.

PIC1ATSG1HALLSEL	PIC1AREG5100	Function
1	1	Selects input pin ENCA1E0, ENCA1E1, and ENCA1EC.
Other than above		Setting prohibited.

PIC1ATSG0HALLSEL	PIC1AREG5000	Function
1	0	Selects input pin ENCA0E0, ENCA0E1, and ENCA0EC.
Other than above		Setting prohibited.

23.2.2.14 PIC1ATAUD0SEL — TAUD0 Input Select Register

The PIC1ATAUD0SEL register is a 32-bit register that selects TAUDTIN input signals.

Address: FFDD 0078_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1ATAUD0IN143	PIC1ATAUD0IN142	PIC1ATAUD0IN141	PIC1ATAUD0IN140	PIC1ATAUD0IN123	PIC1ATAUD0IN122	PIC1ATAUD0IN121	PIC1ATAUD0IN120	PIC1ATAUD0IN103	PIC1ATAUD0IN102	PIC1ATAUD0IN101	PIC1ATAUD0IN100	PIC1ATAUD0IN83	PIC1ATAUD0IN82	PIC1ATAUD0IN81	PIC1ATAUD0IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1ATAUD0IN63	PIC1ATAUD0IN62	PIC1ATAUD0IN61	PIC1ATAUD0IN60	PIC1ATAUD0IN43	PIC1ATAUD0IN42	PIC1ATAUD0IN41	PIC1ATAUD0IN40	PIC1ATAUD0IN23	PIC1ATAUD0IN22	PIC1ATAUD0IN21	PIC1ATAUD0IN20	PIC1ATAUD0IN03	PIC1ATAUD0IN02	PIC1ATAUD0IN01	PIC1ATAUD0IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.23 PIC1ATAUD0SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1ATAUD0INm3	Select the signal to output to TAUD0TIN (m + 1) output pin. 00: TAUD0TIN (m + 1) is selected. 01: TAUD0TIN (m) is selected. 10: TAUD1TIN (m + 1) is selected. 11: TAUD1TIN (m) is selected.
2m + 2	PIC1ATAUD0INm2	
2m + 1	PIC1ATAUD0INm1	Select the signal to output to TAUD0TIN (m) output pin. 00: TAUD0TIN (m) is selected. 01: TAUD0TIN (m + 1) is selected. 10: TAUD1TIN (m) is selected. 11: TAUD1TIN (m + 1) is selected.
2m	PIC1ATAUD0INm0	

Note: m is an even channel number of TAUD0 (CHm_even)

23.2.2.15 PIC1ATAUD1SEL — TAUD1 Input Select Register

The PIC1ATAUD1SEL register is a 32-bit register that selects TAUDTIN input signals.

Address: FFDD 007C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC1ATAUD1IN143	PIC1ATAUD1IN142	PIC1ATAUD1IN141	PIC1ATAUD1IN140	PIC1ATAUD1IN123	PIC1ATAUD1IN122	PIC1ATAUD1IN121	PIC1ATAUD1IN120	PIC1ATAUD1IN103	PIC1ATAUD1IN102	PIC1ATAUD1IN101	PIC1ATAUD1IN100	PIC1ATAUD1IN83	PIC1ATAUD1IN82	PIC1ATAUD1IN81	PIC1ATAUD1IN80
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1ATAUD1IN63	PIC1ATAUD1IN62	PIC1ATAUD1IN61	PIC1ATAUD1IN60	PIC1ATAUD1IN43	PIC1ATAUD1IN42	PIC1ATAUD1IN41	PIC1ATAUD1IN40	PIC1ATAUD1IN23	PIC1ATAUD1IN22	PIC1ATAUD1IN21	PIC1ATAUD1IN20	PIC1ATAUD1IN03	PIC1ATAUD1IN02	PIC1ATAUD1IN01	PIC1ATAUD1IN00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.24 PIC1ATAUD1SEL Register Contents

Bit Position	Bit Name	Function
2m + 3	PIC1ATAUD1INm3	Select the signal to output to TAUD1TIN (m + 1) output pin 00: TAUD1TIN (m + 1) is selected. 01: TAUD1TIN (m) is selected. 10: TAUD0TIN (m + 1) is selected. 11: TAUD0TIN (m) is selected.
2m + 2	PIC1ATAUD1INm2	
2m + 1	PIC1ATAUD1INm1	Select the signal to output to TAUD1TIN (m) output pin 00: TAUD1TIN (m) is selected. 01: TAUD1TIN (m + 1) is selected. 10: TAUD0TIN (m) is selected. 11: TAUD0TIN (m + 1) is selected.
2m	PIC1ATAUD1INm0	

Note: m is an even channel number of TAUD1 (CHm_even)

23.2.2.16 PIC1AHIZCEN0 — Hi-Z Control Register 0

The PIC1AHIZCEN0 register selects Hi-Z control input signals of TAUD0.

Address: FFDD 0080_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1AHIZCEN05	—	—	—	—	PIC1AHIZCEN00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R/W

Table 23.25 PIC1AHIZCEN0 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are read as 0. The write value should be 0.
5	PIC1AHIZCEN05	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4 to 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AHIZCEN00	Enables or disables Hi-Z control by ESO0 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting U/V/W outputs or UB/VB/WB outputs of TAUD0.

Set TAPA0CTL0.TAPA0DCN = 0 and TAPA0CTL0.TAPA0DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

23.2.2.17 PIC1AHIZCEN1 — Hi-Z Control Register 1

The PIC1AHIZCEN1 register selects Hi-Z control input signals of TAUD1.

Address: FFDD 0084$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1AHIZCEN 15	—	—	—	—	PIC1AHIZCEN 10
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R/W

Table 23.26 PIC1AHIZCEN1 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are read as 0. The write value should be 0.
5	PIC1AHIZCEN 15	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4 to 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AHIZCEN 10	Enables or disables Hi-Z control by ESO1 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting U/V/W outputs or UB/VB/WB outputs of TAUD1.

Set TAPA1CTL0.TAPA1DCN = 0 and TAPA1CTL0.TAPA1DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

23.2.2.18 PIC1AHIZCEN2 — Hi-Z Control Register 2

The PIC1AHIZCEN2 register selects Hi-Z control input signals of TSG30.

Address: FFDD 0088$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1AHIZCEN 25	—	PIC1AHIZCEN 23	—	—	PIC1AHIZCEN 20
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R	R	R/W

Table 23.27 PIC1AHIZCEN2 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are read as 0. The write value should be 0.
5	PIC1AHIZCEN 25	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4	—	Reserved This bit is read as 0. The write value should be 0.
3	PIC1AHIZCEN 23	Enables or disables Hi-Z control by INTTSG30IER interrupt signal. 0: Disabled 1: Enabled
2, 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AHIZCEN 20	Enables or disables Hi-Z control by ESO2 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting TSG30 output.

Set TAPA2CTL0.TAPA2DCN = 0 and TAPA2CTL0.TAPA2DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

23.2.2.19 PIC1AHIZCEN3 — Hi-Z Control Register 3

The PIC1AHIZCEN3 register selects Hi-Z control input signals of TSG31.

Address: FFDD 008C$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1AHIZCEN35	PIC1AHIZCEN34	—	—	—	PIC1AHIZCEN30
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Table 23.28 PIC1AHIZCEN3 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are read as 0. The write value should be 0.
5	PIC1AHIZCEN35	Enables or disables Hi-Z control by ERROROUTZ signal. 0: Disabled 1: Enabled
4	PIC1AHIZCEN34	Enables or disables Hi-Z control by INTTSG31IER interrupt signal. 0: Disabled 1: Enabled
3 to 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AHIZCEN30	Enables or disables Hi-Z control by ESO3 pin input. 0: Disabled 1: Enabled

CAUTION

Set this register before starting TSG31 output.

Set TAPA3CTL0.TAPA3DCN = 0 and TAPA3CTL0.TAPA3DCP = 1 when performing Hi-Z control by ERROROUTZ signal.

23.2.2.20 PIC1AENCSEL400 — ENCATIN1 Input Select Register 400

The PIC1AENCSEL400 register is used for encoder capture trigger function.

Address: FFDD 00B8$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	PIC1AENCSEL 4007	—	—	—	PIC1AENCSEL 4003	PIC1AENCSEL 4002	PIC1AENCSEL 4001	PIC1AENCSEL 4000
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Table 23.29 PIC1AENCSEL400 Register Contents

Bit Position	Bit Name	Function
7	PIC1AENCSEL 4007	Enables or disables output of INTTAUD0Im signal selected by PIC1AENCSEL400[3:0]. 0: Disabled 1: Enabled
6 to 4	—	Reserved These bits are read as 0. The write value should be 0.
3 to 0	PIC1AENCSEL 400 [3:0]	Select TAUD0TINTm to be used as a capture trigger signal for ENCA0 and ENCA1. 0: INTTAUD0I0 is selected. 1: INTTAUD0I1 is selected. 2: INTTAUD0I2 is selected. 3: INTTAUD0I3 is selected. 4: INTTAUD0I4 is selected. 5: INTTAUD0I5 is selected. 6: INTTAUD0I6 is selected. 7: INTTAUD0I7 is selected. 8: INTTAUD0I8 is selected. 9: INTTAUD0I9 is selected. 10: INTTAUD0I10 is selected. 11: INTTAUD0I11 is selected. 12: INTTAUD0I12 is selected. 13: INTTAUD0I13 is selected. 14: INTTAUD0I14 is selected. 15: INTTAUD0I15 is selected.

23.2.2.21 PIC1AENCSEL410 — ENCATIN1 Input Select Register 410

The PIC1AENCSEL410 register is used for encoder capture trigger function.

Address: FFDD 00BC$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	PIC1AENCSEL 4107	—	—	—	PIC1AENCSEL 4103	PIC1AENCSEL 4102	PIC1AENCSEL 4101	PIC1AENCSEL 4100
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Table 23.30 PIC1AENCSEL410 Register Contents

Bit Position	Bit Name	Function
7	PIC1AENCSEL 4107	Enables or disables output of INTTAUD1Im signal selected by PIC1AENCSEL410[3:0]. 0: Disabled 1: Enabled
6 to 4	—	Reserved These bits are read as 0. The write value should be 0.
3 to 0	PIC1AENCSEL 410 [3:0]	Selects TAUD1TINTm to be used as a capture trigger signal for ENCA0 and ENCA1. 0: INTTAUD1I0 is selected. 1: INTTAUD1I1 is selected. 2: INTTAUD1I2 is selected. 3: INTTAUD1I3 is selected. 4: INTTAUD1I4 is selected. 5: INTTAUD1I5 is selected. 6: INTTAUD1I6 is selected. 7: INTTAUD1I7 is selected. 8: INTTAUD1I8 is selected. 9: INTTAUD1I9 is selected. 10: INTTAUD1I10 is selected. 11: INTTAUD1I11 is selected. 12: INTTAUD1I12 is selected. 13: INTTAUD1I13 is selected. 14: INTTAUD1I14 is selected. 15: INTTAUD1I15 is selected.

23.2.2.22 PIC1AREG200 — Timer Input/Output Control Register 200

The PIC1AREG200 register selects TAUD0 input signals.

Address: FFDD 00C0$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PIC1AREG 20025	PIC1AREG 20024	—	—	—	—	—	PIC1AREG 20018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIC1AREG 20011	PIC1AREG 20010	PIC1AREG 20009	PIC1AREG 20008	—	—	—	—	PIC1AREG 20003	PIC1AREG 20002	PIC1AREG 20001	PIC1AREG 20000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 23.31 PIC1AREG200 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are read as 0. The write value should be 0.
25, 24	PIC1AREG 20025, PIC1AREG 20024	Select TAUD channel to use for TAPA0TSIM0 and TAPA0TUDCM0. 00: No channel selected. 01: TAUD0 channel 0 is selected. 10: TAUD0 channel 2 is selected. 11: TAUD0 channel 8 is selected.
23 to 19	—	Reserved These bits are read as 0. The write value should be 0.
18	PIC1AREG 20018	Selects the signal for input as TAUD0TIN10, TAUD0TIN12, and TAUD0TIN14 signals of TAUD0. 1: TOUT of TAUD0 CH02. Settings other than above are prohibited.*1
17 to 12	—	Reserved These bits are read as 0. The write value should be 0.
11, 10	PIC1AREG 20011, PIC1AREG 20010	Select the signal for input as TAUD0TIN6 and TAUD0TIN7 signals of TAUD0. 10: TS0PTE signal of TSG30. Settings other than above are prohibited.*1
9, 8	PIC1AREG 20009, PIC1AREG 20008	Select the signal for input as TAUD0TIN4 and TAUD0TIN5 signals of TAUD0. 10: TS0PTE signal of TSG30. Settings other than above are prohibited.*1
7 to 4	—	Reserved These bits are read as 0. The write value should be 0.
3	PIC1AREG 20003	Selects the signal for input as TAUD0TIN7 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1AREG20011 and PIC1AREG20010 bits (TS0PTE signal).
2	PIC1AREG 20002	Selects the signal for input as TAUD0TIN6 signal of TAUD0. 0: TIN pin input 1: Input signal selected by PIC1AREG20011 and PIC1AREG20010 bits (TS0PTE signal).
1	PIC1AREG 20001	Selects the signal for input as TAUD0TIN5 signal of TAUD0. 0: TIN pin input. 1: Input signal selected by PIC1AREG20009 and PIC1AREG20008 bits (TS0PTE signal).

Table 23.31 PIC1AREG200 Register Contents (2/2)

Bit Position	Bit Name	Function
0	PIC1AREG 20000	Selects the signal for input as TAUD0TIN4 signal of TAUD0. 0: TIN pin input. 1: Input signal selected by PIC1AREG20009 and PIC1AREG20008 bits (TS0PTE signal).

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

23.2.2.23 PIC1AREG210 — Timer Input/Output Control Register 210

The PIC1AREG210 register selects TAUD1 input signals.

Address: FFDD 00D4$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PIC1AREG21025	PIC1AREG21024	—	—	—	—	—	PIC1AREG21018	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIC1AREG21011	PIC1AREG21010	PIC1AREG21009	PIC1AREG21008	—	—	—	—	PIC1AREG21003	PIC1AREG21002	PIC1AREG21001	PIC1AREG21000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 23.32 PIC1AREG210 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are read as 0. The write value should be 0.
25, 24	PIC1AREG21025, PIC1AREG21024	Select the TAUD channel to use for TAPA1TSIM0 and TAPA1TUDCM0. 00: No channel selected. 01: TAUD1 channel 0 is selected. 10: TAUD1 channel 2 is selected. 11: TAUD1 channel 8 is selected.
23 to 19	—	Reserved These bits are read as 0. The write value should be 0.
18	PIC1AREG21018	Selects the signal for input as TAUD1TIN10, TAUD1TIN12, and TAUD1TIN14 signals of TAUD1. 1: TOUT of TAUD1 CH02 Settings other than above are prohibited.*1
17 to 12	—	Reserved These bits are read as 0. The write value should be 0.
11, 10	PIC1AREG21011, PIC1AREG21010	Select the signal for input as TAUD1TIN6 and TAUD1TIN7 signals of TAUD1. 10: TS0PTE signal of TSG31 Settings other than above are prohibited.*1
9, 8	PIC1AREG21009, PIC1AREG21008	Select the signal for input as TAUD1TIN4 and TAUD1TIN5 signals of TAUD1. 10: TS0PTE signal of TSG31 Settings other than above are prohibited.*1
7 to 4	—	Reserved These bits are read as 0. The write value should be 0.
3	PIC1AREG21003	Selects the signal for input as TAUD1TIN7 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1AREG21011 and PIC1AREG21010 bits (TS0PTE signal).
2	PIC1AREG21002	Selects the signal for input as TAUD1TIN6 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1AREG21011 and PIC1AREG21010 bits (TS0PTE signal).
1	PIC1AREG21001	Selects the signal for input as TAUD1TIN5 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1AREG21009 and PIC1AREG21008 bits (TS0PTE signal).

Table 23.32 PIC1AREG210 Register Contents (2/2)

Bit Position	Bit Name	Function
0	PIC1AREG21000	Selects the signal for input as TAUD1TIN4 signal of TAUD1. 0: TIN pin input 1: Input signal selected by PIC1AREG21009 and PIC1AREG21008 bits (TS0PTE signal).

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

23.2.2.24 PIC1AREG2n1 — Timer Input/Output Control Register 2n1

The PIC1AREG2n1 register selects combination circuit PFN0xx logical operation.

Address: FFDD 00C4$_H$ (n = 0), FFDD 00D8$_H$ (n = 1)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PIC1AREG2n127	PIC1AREG2n126	PIC1AREG2n125	PIC1AREG2n124	PIC1AREG2n123	PIC1AREG2n122	PIC1AREG2n121	PIC1AREG2n120	PIC1AREG2n119	PIC1AREG2n118	PIC1AREG2n117	PIC1AREG2n116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 23.33 PIC1AREG2n1 Register Contents

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are read as 0. The write value should be 0.
27, 26	PIC1AREG2n127, PIC1AREG2n126	Select PFN045 WO2 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
25, 24	PIC1AREG2n125, PIC1AREG2n124	Select PFN045 WO1 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
23, 22	PIC1AREG2n123, PIC1AREG2n122	Select PFC023 VO2 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
21, 20	PIC1AREG2n121, PIC1AREG2n120	Select PFN023 VO1 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
19, 18	PIC1AREG2n119, PIC1AREG2n118	Select PFN001 UO2 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
17, 16	PIC1AREG2n117, PIC1AREG2n116	Select PFN001 UO1 output.*1 10: Combination circuit output. 11: Inverted combination circuit output. Settings other than above are prohibited.*2
15 to 0	—	Reserved These bits are read as 0. The write value should be 0.

Note 1. The register value for some functions needs to be set depending on the value of TAUD. For the setting values, see **Section 23.2.3, Function**.

Note 2. Set any appropriate value because the value after reset is “setting prohibited”.

Block diagram of PFN001 is shown in the following figure.

PFN023 and PFN045 are operated under the same logic with different input signals and registers.

For connection of PFN0xx with peripheral circuits, see **Figure 23.11**.

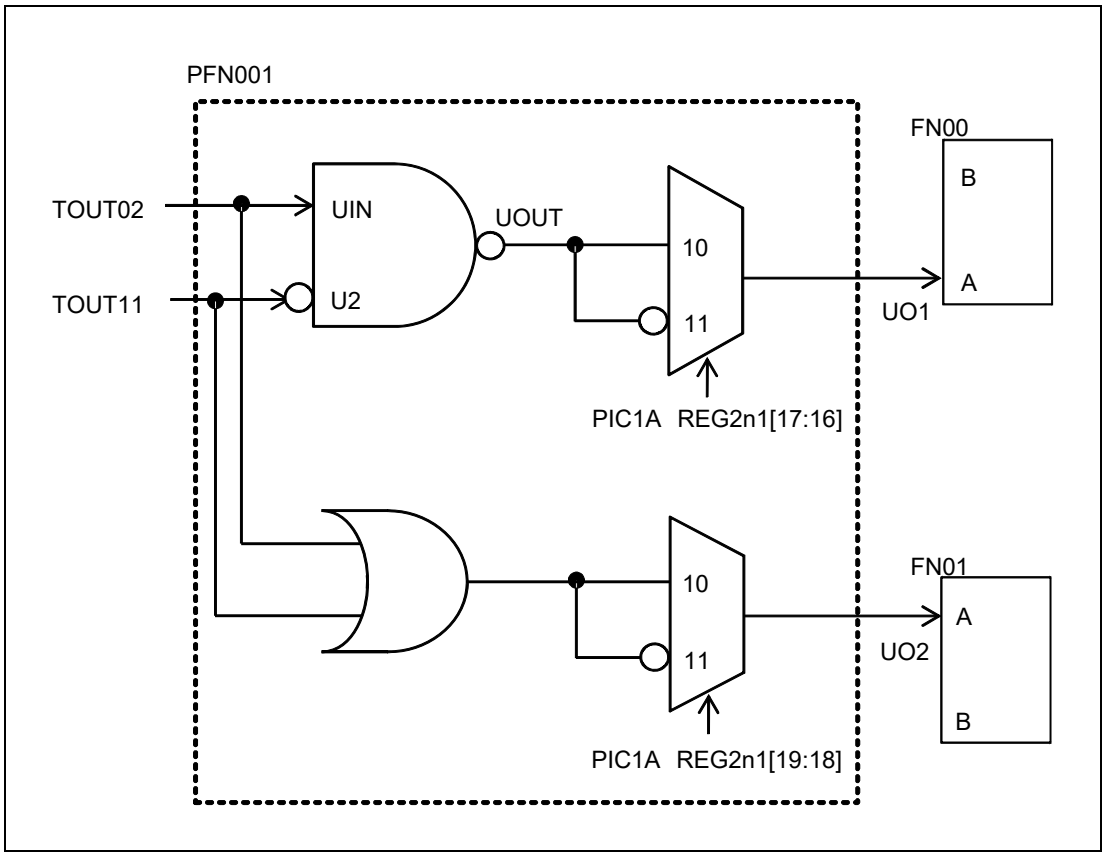


Figure 23.1 Block Diagram of PFN001

23.2.2.25 PIC1AREG2n2 — Timer Input/Output Control Register 2n2

PIC1AREG2n2 selects input signals of TAUDn CHm.

Address: FFDD 00C8$_H$ (n = 0), FFDD 00DC$_H$ (n = 1)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PIC1AREG2n227	PIC1AREG2n226	PIC1AREG2n225	PIC1AREG2n224	PIC1AREG2n223	PIC1AREG2n222	PIC1AREG2n221	PIC1AREG2n220	PIC1AREG2n219	PIC1AREG2n218	PIC1AREG2n217	PIC1AREG2n216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PIC1AREG2n204	PIC1AREG2n203	PIC1AREG2n202	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Table 23.34 PIC1AREG2n2 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are read as 0. The write value should be 0.
27, 26	PIC1AREG2n227, PIC1AREG2n226	Select INT input signal of TAUDnCH15. 00: TIN pin input. 10: Signal selected by PIC1AREG2n204 bit (TOUT of TAUDn CH09). Settings other than above are prohibited.
25, 24	PIC1AREG2n225, PIC1AREG2n224	Select INT input signal of TAUDnCH14. 00: TIN pin input 10: Signal selected by PIC1AREG2n018 register bit (TOUT of TAUDn CH02). Settings other than above are prohibited.
23, 22	PIC1AREG2n223, PIC1AREG2n222	Select INT input signal of TAUDnCH13. 00: TIN pin input 10: Signal selected by PIC1AREG2n203 bit (TOUT of TAUDn CH07). Settings other than above are prohibited.
21, 20	PIC1AREG2n221, PIC1AREG2n220	Select INT input signal of TAUDnCH12. 00: TIN pin input 10: Signal selected by PIC1AREG2n018 register bit (TOUT of TAUDn CH02). Settings other than above are prohibited.
19, 18	PIC1AREG2n219, PIC1AREG2n218	Select INT input signal of TAUDnCH11. 00: TIN pin input 10: Signal selected by PIC1AREG2n202 bit (TOUT of TAUDn CH05). Settings other than above are prohibited.
17, 16	PIC1AREG2n217, PIC1AREG2n216	Select INT input signal of TAUDnCH10. 00: TIN pin input 10: Signal selected by PIC1AREG2n018 register bit (TOUT of TAUDn CH02). Settings other than above are prohibited
15 to 5	—	Reserved These bits are read as 0. The write value should be 0.
4	PIC1AREG2n204	Selects the signal to be supplied to TIN of TAUDn CH15. 0: TOUT of TAUDnCH09. 1: Set/clear output by TAUDnINT08 and TAUDnINT09.
3	PIC1AREG2n203	Selects the signal to be supplied to TIN of TAUDn CH13. 0: TOUT of TAUDnCH07. 1: Set/clear output by TAUDnINT06 and TAUDnINT07.
2	PIC1AREG2n202	Selects the signal to be supplied to TIN of TAUDn CH11. 0: TOUT of TAUDnCH05. 1: Set/clear output by TAUDnINT04 and TAUDnINT05.

Table 23.34 PIC1AREG2n2 Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	—	Reserved These bits are read as 0. The write value should be 0.

23.2.2.26 PIC1AREG2n3 — Timer Input/Output Control Register 2n3

PIC1AREG2n3 selects logical operation for the combination circuit FN0i.

Address: FFDD 00CC$_H$ (n = 0), FFDD 00E0$_H$(n = 1)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1AREG2n322	PIC1AREG2n321	PIC1AREG2n320	—	PIC1AREG2n318	PIC1AREG2n317	PIC1AREG2n316
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PIC1AREG2n314	PIC1AREG2n313	PIC1AREG2n312	—	PIC1AREG2n310	PIC1AREG2n309	PIC1AREG2n308	—	PIC1AREG2n306	PIC1AREG2n305	PIC1AREG2n304	—	PIC1AREG2n302	PIC1AREG2n301	PIC1AREG2n300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 23.35 PIC1AREG2n3 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are read as 0. The write value should be 0.
22 to 20	PIC1AREG2n322, PIC1AREG2n321, PIC1AREG2n320	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.
19	—	Reserved This bit is read as 0. The write value should be 0.
18 to 16	PIC1AREG2n318, PIC1AREG2n317, PIC1AREG2n316	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.
15	—	Reserved This bit is read as 0. The write value should be 0.
14 to 12	PIC1AREG2n314, PIC1AREG2n313, PIC1AREG2n312	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.
11	—	Reserved This bit is read as 0. The write value should be 0.
10 to 8	PIC1AREG2n310, PIC1AREG2n309, PIC1AREG2n308	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.
7	—	Reserved This bit is read as 0. The write value should be 0.

Table 23.35 PIC1AREG2n3 Register Contents (2/2)

Bit Position	Bit Name	Function
6 to 4	PIC1AREG2n306, PIC1AREG2n305, PIC1AREG2n304	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.
3	—	Reserved This bit is read as 0. The write value should be 0.
2 to 0	PIC1AREG2n302, PIC1AREG2n301, PIC1AREG2n300	Select a logical operation to be performed on input signals A and B.*1 000: A 100: A and B 101: A or B Settings other than above are prohibited.

Note 1. The register value for some functions needs to be set depending on the value of TAUD. For the setting values, see **Section 23.2.3, Function**.

Block diagram of FN00 is shown in the following figure.

FN01 to FN05 are operated under the same logical with different input signals and registers.

For connection of FN0i with peripheral circuits, see **Figure 23.11**.

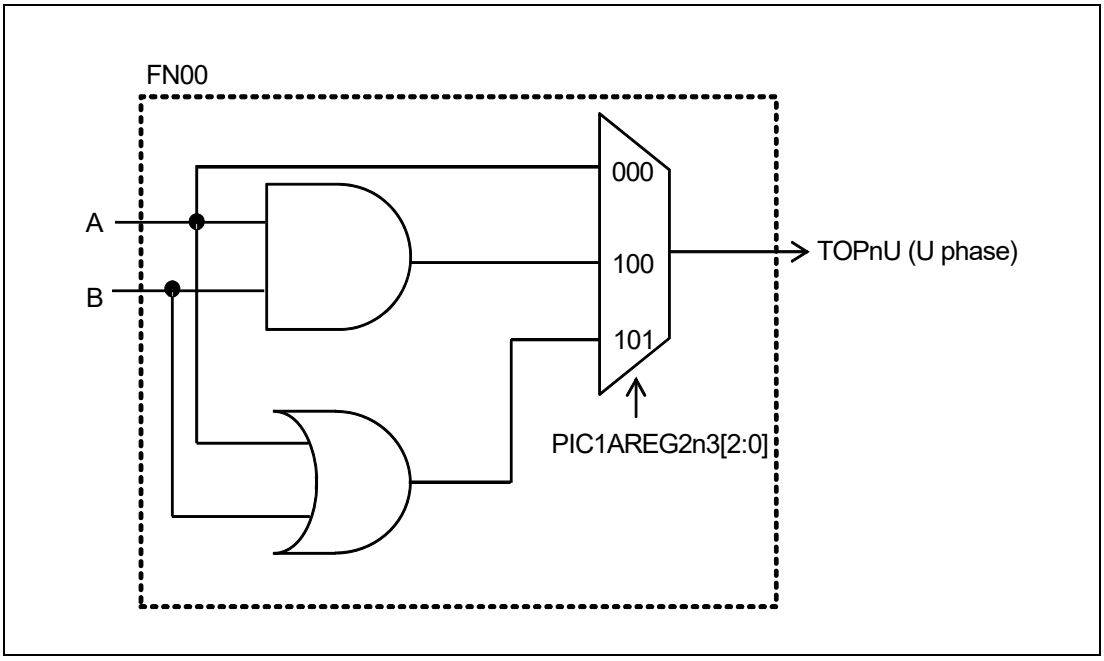


Figure 23.2 Block Diagram of FN00

23.2.2.27 PIC1AREG30 — Timer Input/Output Control Register 30

PIC1AREG30 selects ENCAn input signals.

Address: FFDD 00E8$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1AREG3022	PIC1AREG3021	PIC1AREG3020	PIC1AREG3019	PIC1AREG3018	PIC1AREG3017	PIC1AREG3016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1AREG3015	PIC1AREG3014	PIC1AREG3013	PIC1AREG3012	PIC1AREG3011	PIC1AREG3010	PIC1AREG3009	PIC1AREG3008	PIC1AREG3007	PIC1AREG3006	PIC1AREG3005	PIC1AREG3004	PIC1AREG3003	PIC1AREG3002	PIC1AREG3001	PIC1AREG3000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.36 PIC1AREG30 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are read as 0. The write value should be 0.
22	PIC1AREG3022	Selects the input pins (ENCA0E0, ENCA0E1, ENCA0EC) of ENCA0 timer. 0: The signal selected by PIC1AREG3000 (ENCA0E0), PIC1AREG3001 (ENCA0E1), PIC1AREG3017, and PIC1AREG3016 (ENCA0EC). 1: The signal selected by PIC1AREG3020 and PIC1AREG3019.
21	PIC1AREG3021	Selects the signal to be supplied to PIC1AREG3012 to PIC1AREG3014. 0: ENCA1I1 (signal 1 of the ENCA1 external pin) 1: The signal selected by the PIC1AENCSEL4107 bit of the PIC1AENCSEL410 register.
20, 19	PIC1AREG3020, PIC1AREG3019	Select the input pins (ENCA1E0, ENCA1E1, ENCA1EC) of ENCA1 timer. 00: The ENCA1E0, ENCA1E1, ENCA1EC input pins of ENCA1 timer. 01: The RDC1BOUT, RDC1AOUT, RDC1ZOUT input pins of RDC1. 10: The RDC0BOUT, RDC0AOUT, RDC0ZOUT input pins of RDC0. Settings other than above are prohibited.
18	PIC1AREG3018	Selects the signal to be supplied to the PIC1AREG3002 to PIC1AREG3004. 0: ENCA0I1 (signal 1 of the ENCA0 external pin) 1: The signal selected by the PIC1AENCSEL4007 bit of the PIC1AENCSEL400 register.
17, 16	PIC1AREG3017, PIC1AREG3016	Select the input pins (ENCA0E0, ENCA0E1, ENCA0EC) of ENCA0 timer. 00: The ENCA0E0, ENCA0E1, ENCA0EC input pins of ENCA0 timer. 01: The RDC0BOUT, RDC0AOUT, RDC0ZOUT input pins of RDC0. 10: The RDC1BOUT, RDC1AOUT, RDC1ZOUT input pins of RDC1. Settings other than above are prohibited.
15 to 12	PIC1AREG30 [15:12]	Select the signal for input as the ENCAT1TIN1 signal. 0: The signal selected by PIC1AREG3021. 1: The signal selected by PIC1AREG3018. 2: ADCC0TRG4 3: ADCC0TRG3 4: ADCC0TRG2 5: ADCC0TRG1 6: ADCC0TRG0 7: ADCC1TRG4 8: ADCC1TRG3 9: ADCC1TRG2 10: ADCC1TRG1 11: ADCC1TRG0 Settings other than above are prohibited.

Table 23.36 PIC1AREG30 Register Contents (2/2)

Bit Position	Bit Name	Function
11, 10	PIC1AREG3011, PIC1AREG3010	Select the ENCAEC pin input of timer ENCA1. 00: The signal selected by the PIC1AREG3019 and PIC1AREG3020 bits. 10: The signal selected by the PIC1AREG3016 and PIC1AREG3017 bits. 11: ENCA0EQ1 signal (ENCA0 timer) Settings other than above are prohibited.
9, 8	PIC1AREG 3009, PIC1AREG 3008	Select the ENCA1E1 pin input of timer ENCA1. 00: Signal selected by the PIC1AREG3019 and PIC1AREG3020 bits. 01: The signal selected by the PIC1AREG3016 and PIC1AREG3017 bits. 10: TS1PUD signal of TSG31. Settings other than above are prohibited.
7, 6	PIC1AREG 3007, PIC1AREG 3006	Select the ENCA1E0 pin input of timer ENCA1. 00: The signal selected by the PIC1AREG3019 and PIC1AREG3020 bits. 01: The signal selected by the PIC1AREG3016 and PIC1AREG3017 bits. 10: TS1PEC signal of TSG31. Settings other than above are prohibited.
5 to 2	PIC1AREG30 [05:02]	Select the signal for input as the ENCAT0TIN1 signal. 0: The signal selected by the PIC1AREG3018 bit. 1: The signal selected by the PIC1AREG3021 bit. 2: ADCC0TRG4 3: ADCC0TRG3 4: ADCC0TRG2 5: ADCC0TRG1 6: ADCC0TRG0 7: ADCC1TRG4 8: ADCC1TRG3 9: ADCC1TRG2 10: ADCC1TRG1 11: ADCC1TRG0 Settings other than above are prohibited.
1	PIC1AREG3001	Selects the signal to input to the ENCA0E1 internal signal. 0: Signal selected by PIC1AREG3017 and PIC1AREG3016. 1: TS0PUD signal of TSG30.
0	PIC1AREG3000	Selects the signal to input to the ENCA0E0 internal input pin. 0: Signal selected by PIC1AREG3017 and PIC1AREG3016. 1: TS0PEC signal of TSG30.

23.2.2.28 PIC1AREG31 — Timer Input/Output Control Register 31

The PIC1AREG31 register selects TAUDn and TAUJ0 input signals.

Address: FFDD 00EC$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PIC1AREG3122	PIC1AREG3121	PIC1AREG3120	PIC1AREG3119	PIC1AREG3118	PIC1AREG3117	PIC1AREG3116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC1AREG3115	—	PIC1AREG3113	PIC1AREG3112	PIC1AREG3111	PIC1AREG3110	PIC1AREG3109	PIC1AREG3108	PIC1AREG3107	PIC1AREG3106	—	PIC1AREG3104	PIC1AREG3103	—	PIC1AREG3101	PIC1AREG3100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R/W	R/W

Table 23.37 PIC1AREG31 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are read as 0. The write value should be 0.
22, 21	PIC1AREG3122, PIC1AREG3121	Select TIN input signal of TAUD1 CH02. 00: TIN pin input 01: DT output signal of ENCAT1EQ0 Settings other than above are prohibited.
20	PIC1AREG3120	Selects TIN input signal of TAUD1 CH01. 0: TIN pin input 1: The signal selected by PIC1AREG3115 to PIC1AREG3117.
19, 18	PIC1AREG3119, PIC1AREG3118	Select TIN input signal of TAUD1 CH00. 00: The signal selected by PIC1AREG3115 to PIC1AREG3117. 10: DT output signal of ENCAT1EQ0 Settings other than above are prohibited.
17 to 15	PIC1AREG3117, PIC1AREG3116, PIC1AREG3115	Select TIN input signal of TAUD1 CH00 and CH01. 000: TIN pin input. 001: DT output signal of ENCAT1EQ1 Settings other than above are prohibited.
14	—	Reserved This bit is read as 0. The write value should be 0.
13, 12	PIC1AREG3113, PIC1AREG3112	Select TIN input signal of TAUD0 CH02. 00: TIN pin input. 10: DT output signal of ENCAT0EQ0 Settings other than above are prohibited.
11	PIC1AREG3111	Selects TIN input signal of TAUD0 CH01. 0: TIN pin input. 1: The signal selected by PIC1AREG3106 to PIC1AREG3108.
10, 9	PIC1AREG3110, PIC1AREG3109	Select TIN input signal of TAUD0 CH00. 00: The signal selected by PIC1AREG3106 to PIC1AREG3108. 10: DT output signal of ENCAT0EQ0 Settings other than above are prohibited.
8 to 6	PIC1AREG3108, PIC1AREG3107, PIC1AREG3106	Select TIN input signal of TAUD0 CH00 and CH01. 000: TIN pin input. 001: DT output signal of ENCAT0EQ1 Settings other than above are prohibited.

Table 23.37 PIC1AREG31 Register Contents (2/2)

Bit Position	Bit Name	Function
5	—	Reserved This bit is read as 0. The write value should be 0.
4	PIC1AREG3104	Selects TIN input signal of TAUJ0 CH03. 0: TIN pin input. 1: DT output signal of ENCAT1IEC
3	PIC1AREG3103	Selects TIN input signal of TAUJ0 CH02. 0: TIN pin input. 1: DT output signal of ENCAT1IEC
2	—	Reserved This bit is read as 0. The write value should be 0.
1	PIC1AREG3101	Selects TIN input signal of TAUJ0 CH01. 0: TIN pin input. 1: DT output signal of ENCAT0IEC
0	PIC1AREG3100	Selects TIN input signal of TAUJ0 CH00. 0: TIN pin input. 1: DT output signal of ENCAT0IEC

23.2.2.29 PIC1AREG50 — Timer Input/Output Control Register 50

The PIC1AREG50 register selects TSG30 input signals.

Address: FFDD 00F8$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1AREG5010	—	PIC1AREG5008	PIC1AREG5007	PIC1AREG5006	PIC1AREG5005	—	—	—	—	PIC1AREG5000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Table 23.38 PIC1AREG50 Register Contents

Bit Position	Bit Name	Function
15 to 11	—	Reserved These bits are read as 0. The write value should be 0.
10	PIC1AREG5010	Selects the signal for input as the TSG30TSTOPC0 signal of the TSG30 timer. 0: INTENCA1I1 input of the ENCA1 timer 1: Setting prohibited.
9	—	Reserved This bit is read as 0. The write value should be 0.
8	PIC1AREG5008	Selects the signal for input as the TSG30TSTOPC0 signal of the TSG30 timer. 0: INTENCA0I1 input of the ENCA0 timer 1: Setting prohibited.
7	PIC1AREG5007	Selects the signal for input as the TS0OPCI1 signal of the TSG30 timer. 0: INTTAUD0I7 input of TAUD0 1: Setting prohibited.
6, 5	PIC1AREG5006, PIC1AREG5005	Select the signal for input as the TSG30TSTOPC0 (TS0OPCI0) signal of the TSG30 timer. 01: The signal selected by the PIC1AREG5008 bit 10: The signal selected by the PIC1AREG5010 bit 11: Select input of the INTTAUD0I5 signal of TAUD0. Settings other than above are prohibited.*1
4 to 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AREG5000	Switches the ENCA signal and external hall sensor signal. For the note on selection of the signals, see Note 1 of **Section 23.2.2.13, PIC1ATSGHALLSEL — Hall Sensor Input Select Register**. 0: Select the pin input ENCA0E0, ENCA0E1, ENCA0EC. 1: Setting prohibited.

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

23.2.2.30 PIC1AREG51 — Timer Input/Output Control Register 51

The PIC1AREG51 register selects TSG31 input signals.

Address: FFDD 00FC$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PIC1AREG5110	—	PIC1AREG5108	PIC1AREG5107	PIC1AREG5106	PIC1AREG5105	—	—	—	—	PIC1AREG5100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Table 23.39 PIC1AREG51 Register Contents

Bit Position	Bit Name	Function
15 to 11	—	Reserved These bits are read as 0. The write value should be 0.
10	PIC1AREG5110	Selects the signal for input as the TSG31TSTOPC0 signal from the TSG31 timer. 0: INTENCA1I1 input signal of the ENCA1 timer 1: Setting prohibited.
9	—	Reserved This bit is read as 0. The write value should be 0.
8	PIC1AREG5108	Selects the signal for input as the TSG31TSTOPC0 signal from TSG31 timer. 0: INTENCA0I1 input signal of the ENCA0 timer 1: Setting prohibited.
7	PIC1AREG5107	Selects the signal for input as the TS1OPCI1 signal from TSG31 timer. 0: The INTTAUD1I7 signal input of the TAUD1 timer. 1: Setting prohibited.
6, 5	PIC1AREG 5106, PIC1AREG 5105	Select the signal for input as the TSG31TSTOC0 (TS1OPCI0) signal from TSG31 timer. 01: The signal selected by the PIC1AREG5108 bit. 10: The signal selected by the PIC1AREG5110 bit. 11: Select input of the INTTAUD1I5 signal of TAUD1. Settings other than above are prohibited.*1
4 to 1	—	Reserved These bits are read as 0. The write value should be 0.
0	PIC1AREG5100	Switches the ENCA signal and external hall sensor signal. For the note on selection of the signals, see Note 1 of **Section 23.2.2.13, PIC1ATSGHALLSEL — Hall Sensor Input Select Register**. 1: Select the pin input ENCA1E0, ENCA1E1, ENCA1EC. Settings other than above are prohibited.*1

Note 1. Set any appropriate value because the value after reset is “setting prohibited”.

23.2.3 Function

23.2.3.1 Simultaneous Start Trigger Function

(1) Overview

The function allows any combination of timers (TAUDn, TAUJ0, TSG3n, TPBAn, OSTMn, and ENCAn) to be started simultaneously.

(2) Configuration

The timers which support simultaneous start trigger function are listed as follows.

- TAUDn
- TAUJ0
- TSG3n
- TPBAn
- OSTMn
- ENCAn

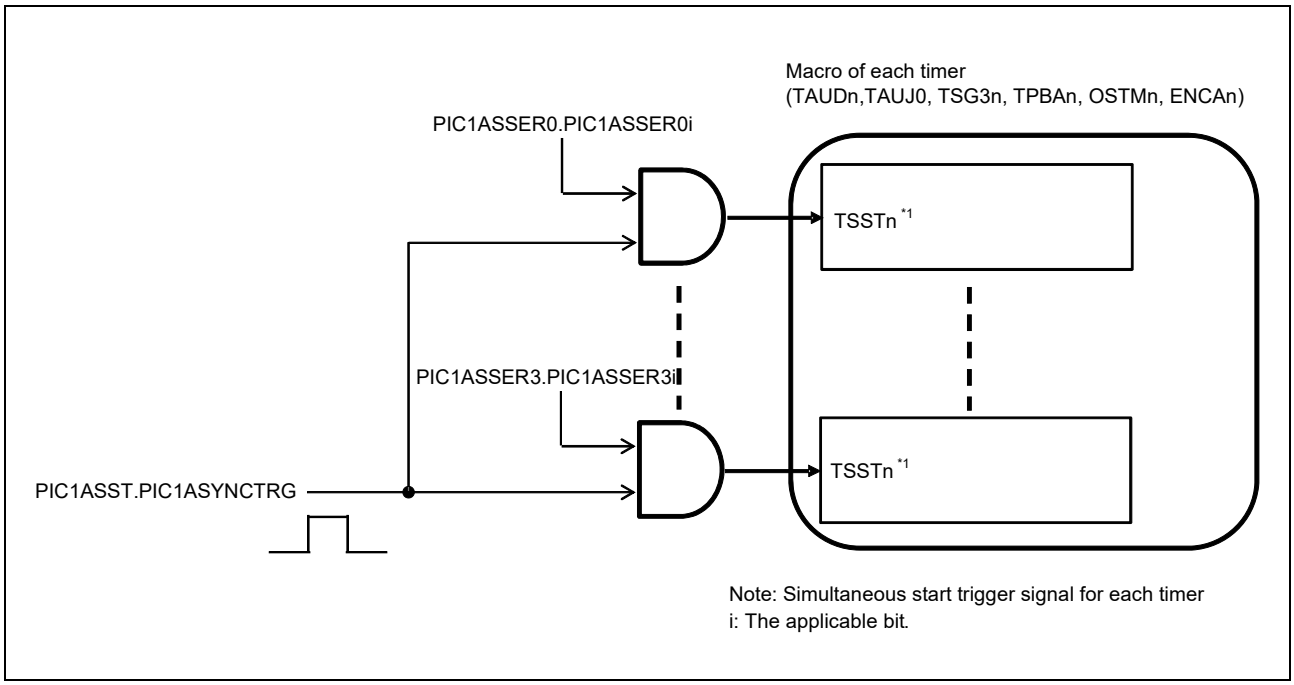


Figure 23.3 Block Diagram of Simultaneous Start Trigger Function

Set 1 to the PIC1ASYNCTRG bit of the simultaneous start trigger control register (PIC1ASST) after unmasking the target timers. The timer operations start by active signal input to the start trigger of each timer.

(3) Registers

The PIC1A registers set by this function is listed as follows. For setting value of the registers, see Sections **23.2.2.2** to **23.2.2.6**.

- PIC1A registers to be set
  PIC1ASST
  PIC1ASSER0
  PIC1ASSER1
  PIC1ASSER2
  PIC1ASSER3

(4) Function

The function allows any combination of timers (TAUDn, TAUJ0, TSG3n, TPBAn, OSTMn, and ENCAn) to be started simultaneously.

(5) Flow Chart

The following figure shows the setting flow of this function.

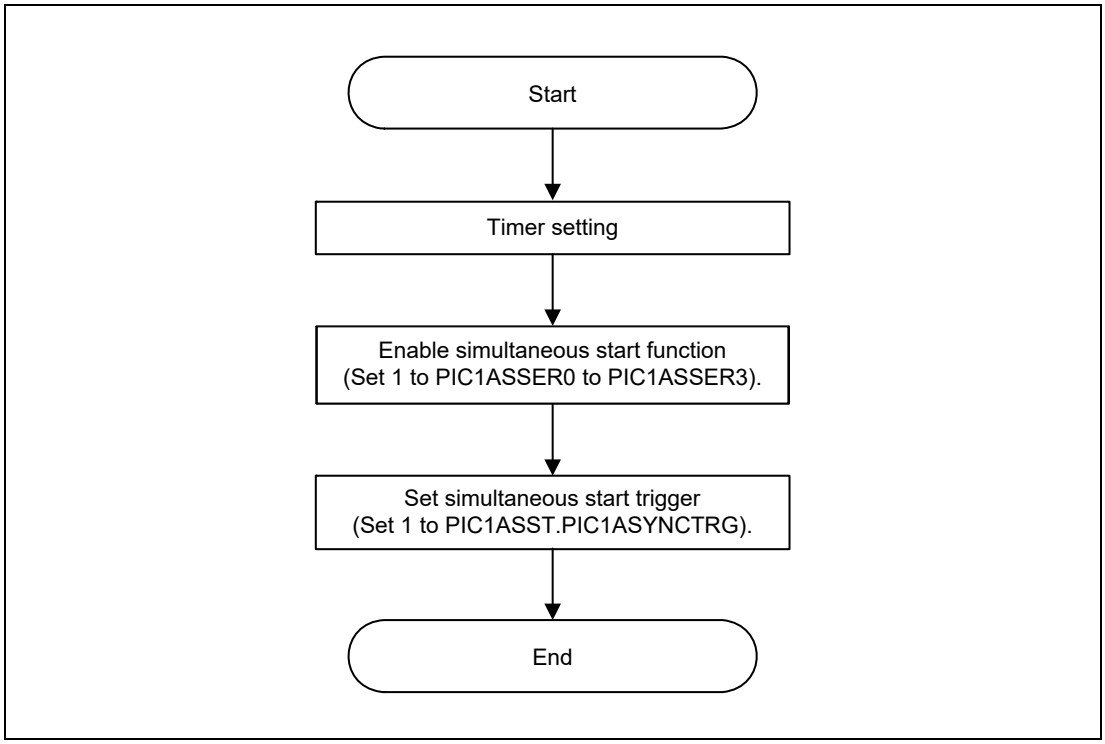


Figure 23.4 Setting Flow

Setting procedures are described as follows.

- Timer setting
  Set the timers to start simultaneously.
- Enable simultaneous start function
  Set 1 to the applicable bits of the PIC1ASSER0, PIC1ASSER1, PIC1ASSER2, and PIC1ASSER3 registers to enable simultaneous start of timers.
- Set simultaneous start trigger
  Setting 1 to the PIC1ASYNCTRG bit of the simultaneous start trigger control register (PIC1ASST) simultaneously starts the timers.

23.2.3.2 PWM Output Function with Dead Time

(1) Overview

This function generates and outputs PWM waveforms with the dead time from one phase to three phases using TAUDn.

The PWM output function of TAUD sets only the clear timing in a period under the duty ratio specification. On the other hand, this function can specify the set timing in addition to the clear timing to output more flexible PWM waveforms with the dead time.

The following table lists the number of channels used.

PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/ VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel.

TAUDn Channel	U phase / UB phase	V phase / VB phase	W phase / WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	—	—	—	Not used.
CH4	√	—	—	Duty (U phase setting)
CH5	√	—	—	Duty (U phase clearing)
CH6	—	√	—	Duty (V phase setting)
CH7	—	√	—	Duty (V phase clearing)
CH8	—	—	√	Duty (W phase setting)
CH9	—	—	√	Duty (W phase clearing)
CH10	√	—	—	U phase output (TOUT10)
CH11	√	—	—	UB phase output (TOUT11)
CH12	—	√	—	V phase output (TOUT12)
CH13	—	√	—	VB phase output (TOUT13)
CH14	—	—	√	W phase output (TOUT14)
CH15	—	—	√	WB phase output (TOUT15)

Note: √: Used, —: Not used

(2) Configuration

The PWM output function with the dead time is realized by using the PWM output function/one-phase output PWM function of TAUDn and PIC1A in combination. The following figure shows the block diagram of the PWM output function with the dead time.

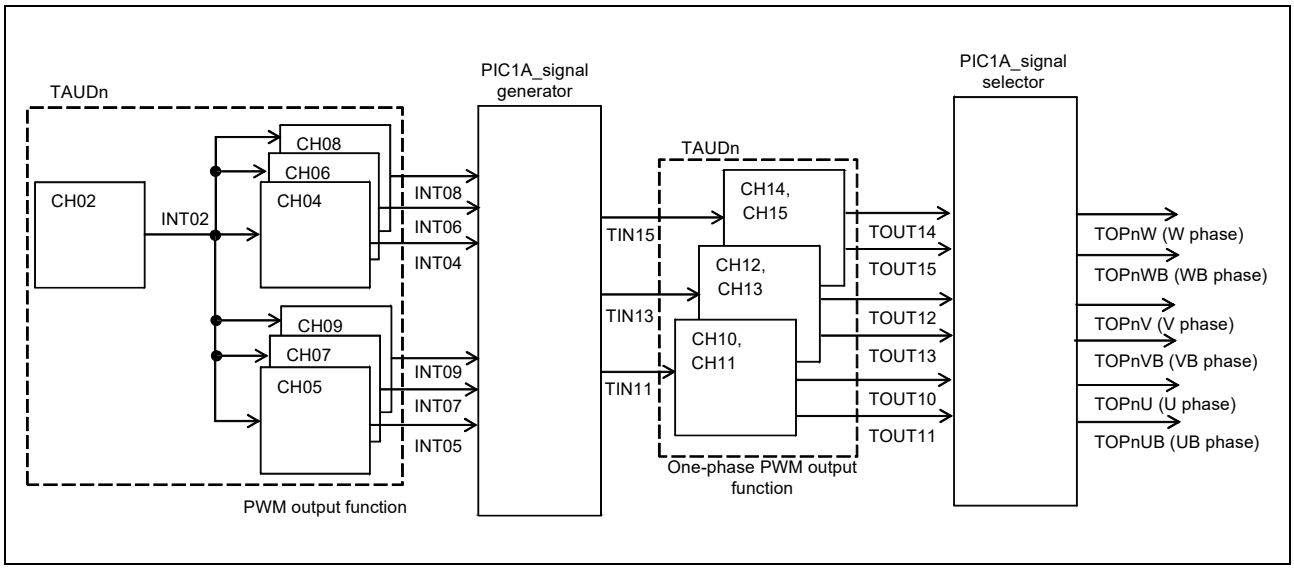


Figure 23.5 Block Diagram of PWM Output Function with Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [TAUDn] PWM output function
  CH02, CH04, and CH05 are used in combination. Setting the period, U phase set value, and U phase clear value to CDR02, CDR04, and CDR05, respectively generates the set and clear signals (INT04 and INT05).
- [PIC1A_signal generator] RS flip-flop circuit (RSn2)
  Selecting the INT04 and INT05 inputs allows TIN11 (PWM signal) to be generated.
- [TAUDn] One-phase PWM output function
  CH10 and CH11 are used in combination. Setting the dead time value to CDR11, and inserting dead time into the PWM signal to be input to TIN11 allows TOUT10 (U phase PWM signal) and TOUT11 (UB phase PWM signal) to be output.
- [PIC1A_signal selector]
  TOUT10 and TOUT11 inputs are selected and output to TOPnU and TOPnUB pins, respectively.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

The block diagram of PIC1A is shown in the following figure.

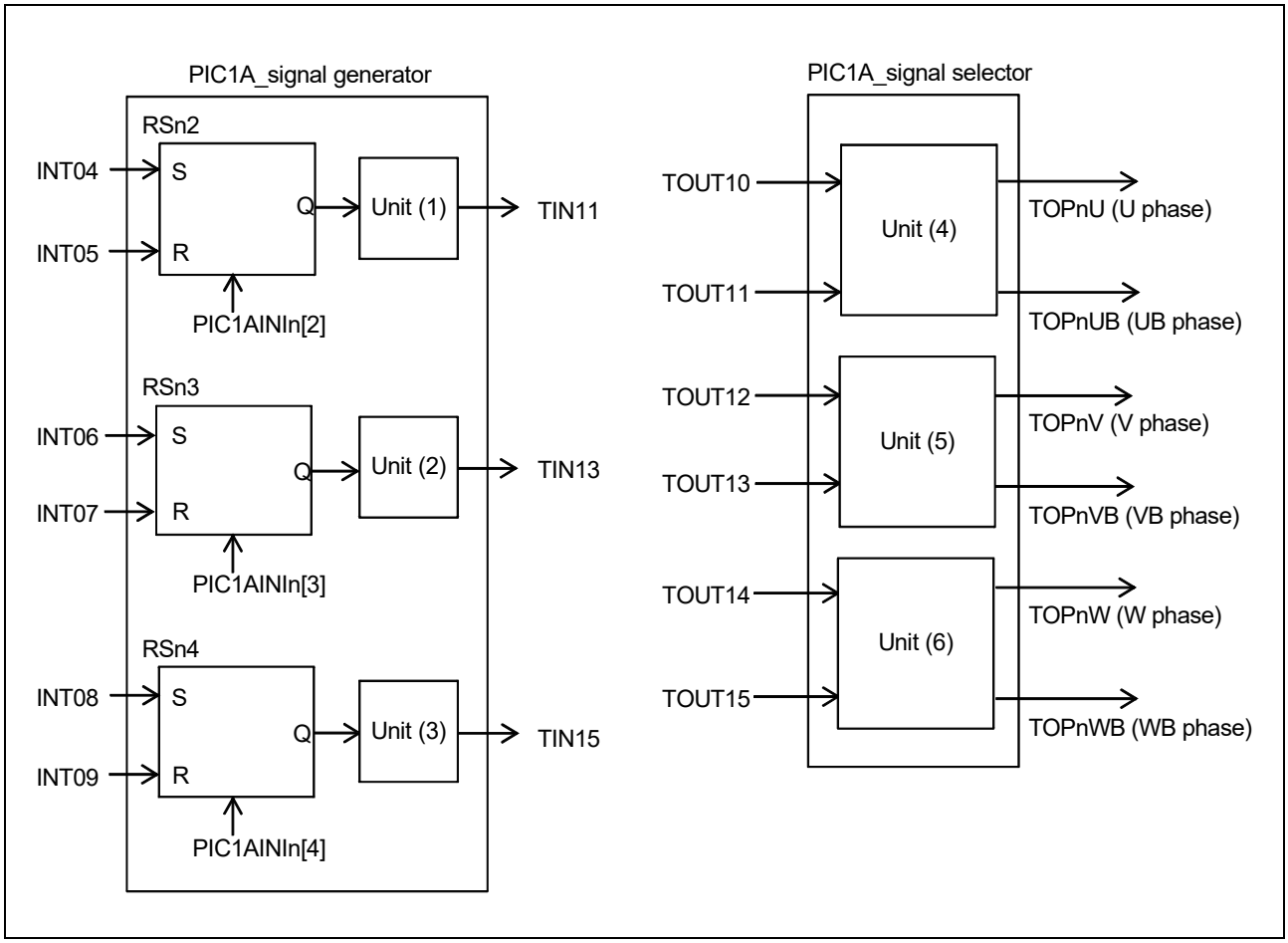


Figure 23.6 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

U phase/ UB phase

The values to output the signal Q from RSn2 as TIN11. (**Figure 23.6**, unit (1))

PIC1AREG2n2[19:18] = 10_B

PIC1AREG2n2[2] = 1_B

PIC1ATAUDnSEL[23:22] = 00_B

The values to output TOUT10 and TOUT11 as TOPnU and TOPnUB, respectively (**Figure 23.6**, unit (4))

PIC1AREG2n1[19:16] = 0000_B

PIC1AREG2n3[2:0] = 000_B

PIC1AREG2n3[6:4] = 000_B

V phase/ VB phase

The values to output the signal Q from RSn3 as TIN13. (**Figure 23.6**, unit (2))

PIC1AREG2n2[23:22] = 10_B

PIC1AREG2n2[3] = 1_B

PIC1ATAUDnSEL[27:26] = 00_B

The values to output TOUT12 and TOUT13 as TOPnV and TOPnVB, respectively (**Figure 23.6**, unit (5))

PIC1AREG2n1[23:20] = 0000_B

PIC1AREG2n3[10:8] = 000_B

PIC1AREG2n3[14:12] = 000_B

W phase/ WB phase

The values to output the signal Q from RSn4 as TIN15. (**Figure 23.6**, unit (3))

PIC1AREG2n2[27:26] = 10_B

PIC1AREG2n2[4] = 1_B

PIC1ATAUDnSEL[31:30] = 00_B

The values to output TOUT14 and TOUT15 as TOPnW and TOPnWB, respectively (**Figure 23.6**, unit (6))

PIC1AREG2n1[27:24] = 0000_B

PIC1AREG2n3 [18:16] = 000_B

PIC1AREG2n3[22:20] = 000_B

Enables initialization of RSn2 to RSn4

The values to enable initialization of RSn2 to RSn4

PIC1AINIn0 [4] = 1_B (initialized)

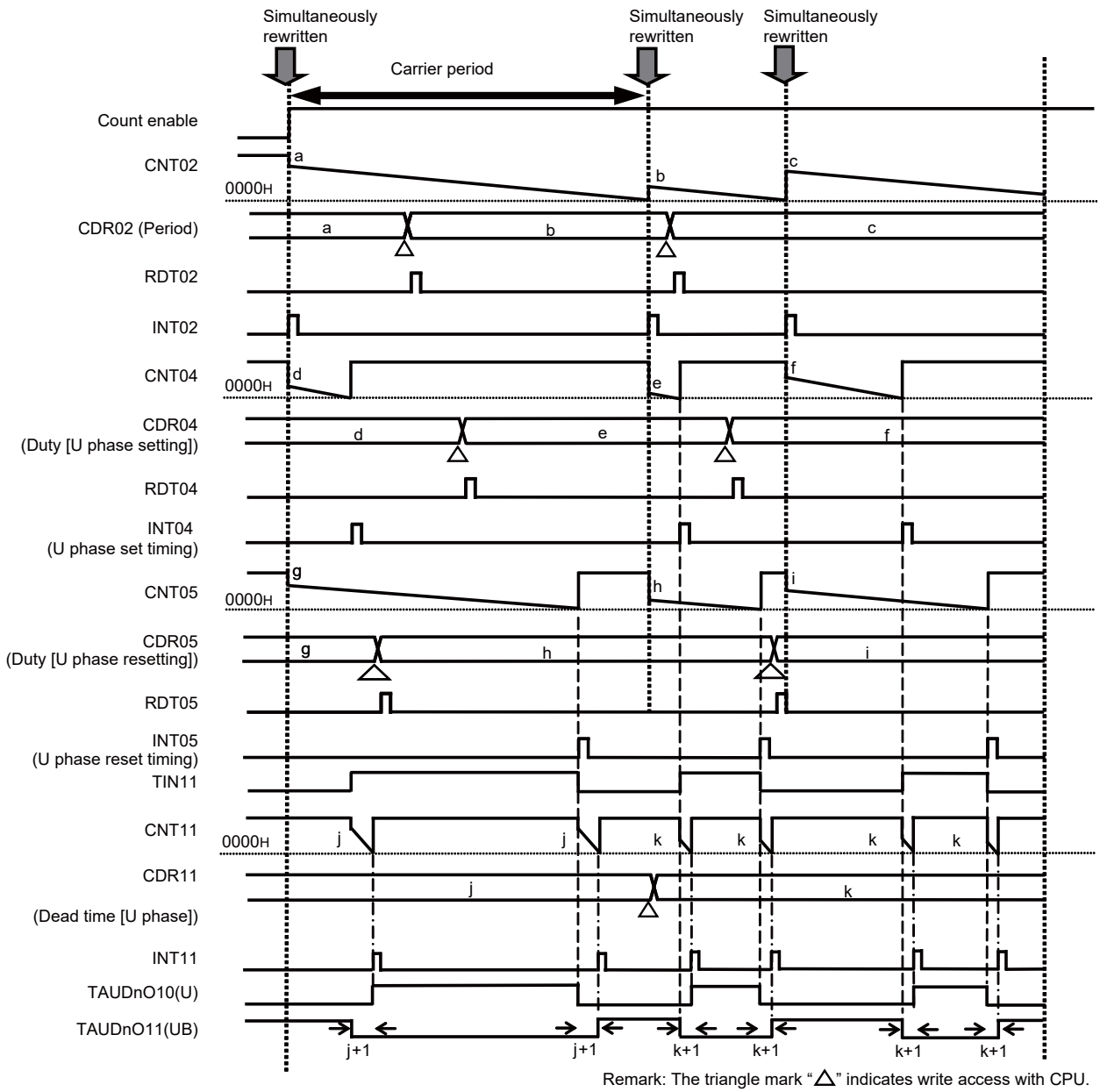
PIC1AINIn0 [3] = 1_B (initialized)

PIC1AINIn0 [2] = 1_B (initialized)

(4) Function

Detail of the function is described using the one-phase PWM output (U phase/UB phase) with dead time as an example.

The following figure shows the timing diagram.



Note: For the detail of TAUD function including operation timing, see the section for TAUD function.

Figure 23.7 Timing Diagram of One-Phase PWM Output with Dead Time (U phase/ UB phase)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.*1

(2) For CH04 and CH05, CH02 underflow allows the set values in CDR04 and CDR05 to be reloaded to CNT04 and CNT05, respectively.

(3) CH04 underflow allows the INT04 to be generated and TIN11 to become high level. CH05 underflow allows the INT05 to be generated and TIN11 to become low level, and the PWM waveform to be generated.

(4) Both edges of TIN11 allows the set values to be reloaded to CNT11.*2

(5) CH11 underflow allows the INT11to be generated and TAUDnO10 to become high level. CH05 underflow allows the INT05 to be generated and TAUDnO10 to become low level. The PWM waveform of U phase is generated and output to TOPnU.

(6) The rising edge of TIN11 allows TAUDnO11 to become low level. CH11 underflow allows the INT11to be generated and TAUDnO11 to become high level. The PWM waveform of UB phase is generated and output to TOPnUB.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

Note 1. Select the count clock signal of the same clock for TAUDn.

Note 2. Set the effective edge to be detected by TIN11 of TAUDn as both edges (rising edge and falling edge) for this function.

Detail of the function when setting the longer clear timing than the carrier period is described using V phase/VB phase as an example.

The following figure shows the timing diagram.

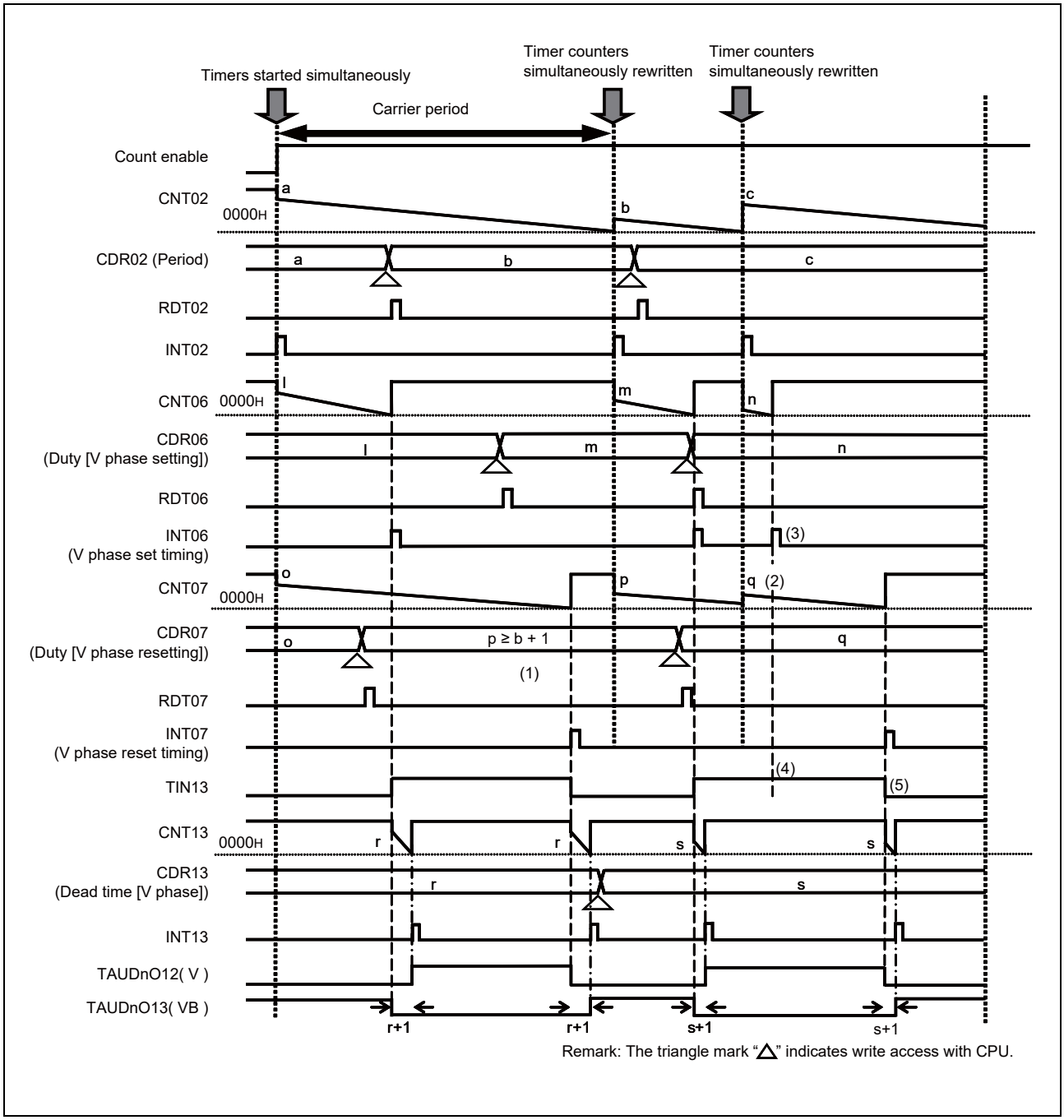


Figure 23.8 Clear Timing Value > Carrier Period Value (V phase/VB phase)

Setting the longer clear time than the carrier period allows the waveform output to be extended over the carrier period.

An operation example of one-phase PWM output (V phase/VB phase) is shown as follows. The operation flow from timer operation start to one-phase PWM output by one-phase PWM output function, refer to the description for one-phase PWM output with dead time (U phase/UB phase).

When the value set in CH07 is longer than the value set in CH02 (**Figure 23.8** (1)), underflow of the carrier period timer is generated before generation of V phase clearing timing signal (INT07), and the value is reloaded (**Figure 23.8** (2)).

This causes generation of the V phase set timing signal (INT06) to precede, thus preventing generation of the V phase clear timing signal (INT07) (**Figure 23.8** (3)).

Then, the PWM waveform is not influenced because V phase set timing signal is ignored in the PIC circuit (**Figure 23.8** (4)). Therefore, the PWM waveform is output extended over the carrier period (**Figure 23.8** (5))

The following figure shows the timing diagram of three-phase PWM output with the dead time.

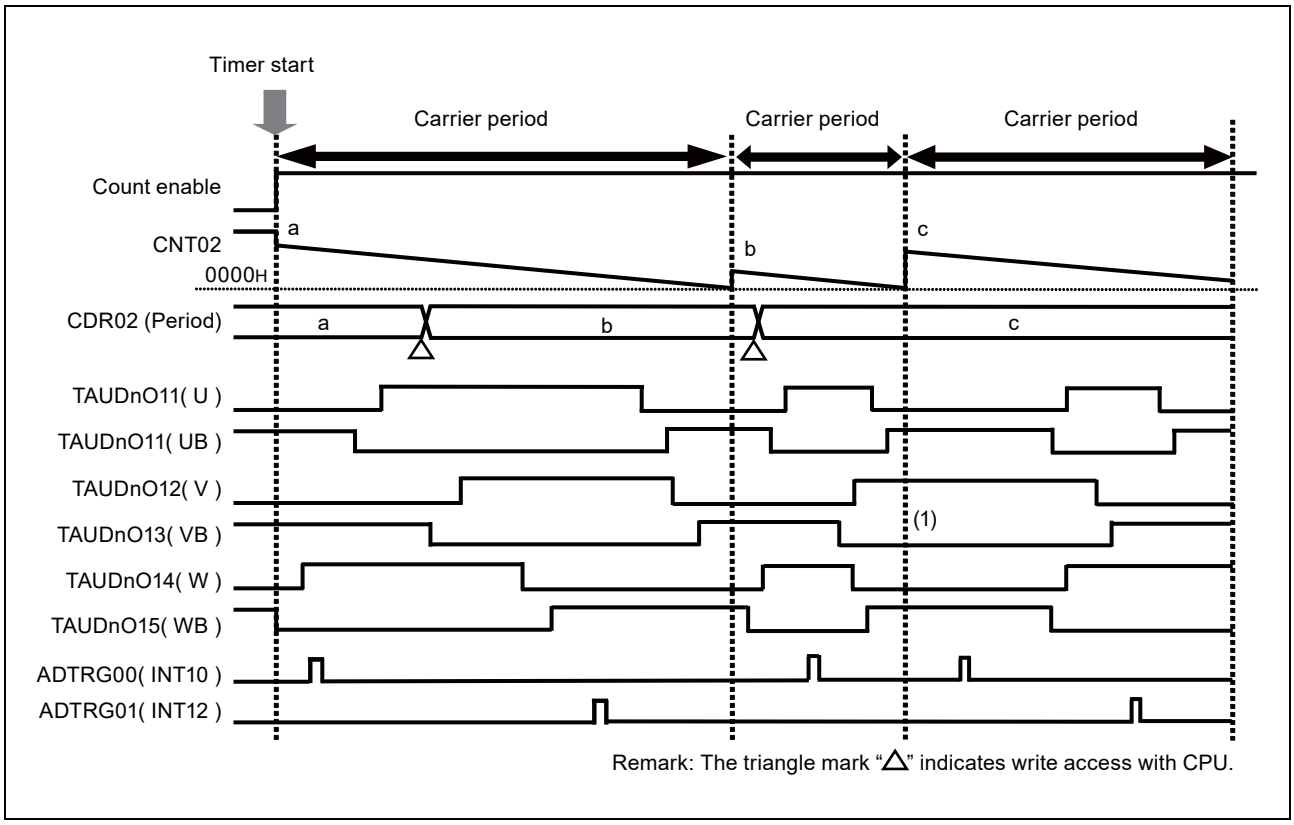
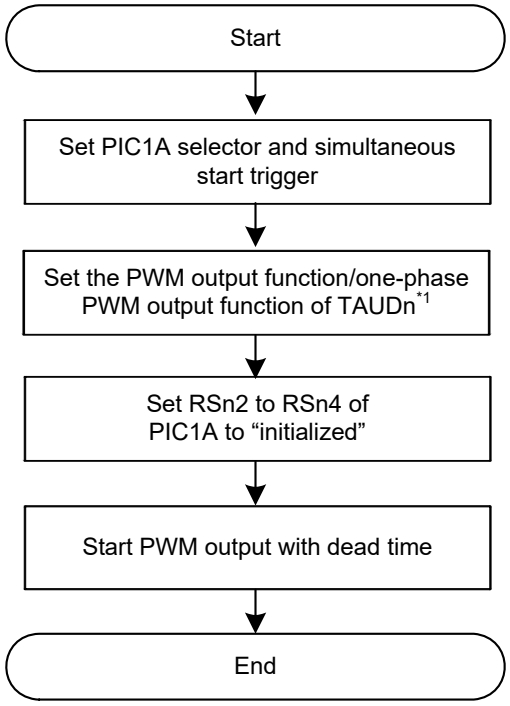


Figure 23.9 Timing Diagram of Three-Phase PWM Output with Dead Time

(5) Flow Chart

The following flow chart shows the PMW output function with the dead time.



Note 1. For the setting of PWM output function of TAUDn, see **Section 17.4.12.1, PWM Output Function**. For the setting of one-phase PWM output function, see **Section 17.4.12.11, One-Phase PWM Output Function**.

23.2.3.3 High Accuracy Triangle Wave PWM Output Function with Dead Time

(1) Overview

This function generates triangle wave PWM output with dead time from one-phase to three-phases. Compared to the triangle wave PWM output function with dead time of TAUDn, this function enables a control of the variable dead time range, where duty cycle is close to 100% and 0%.

The TAUDn channels used in this function are listed in the following table.

High Accuracy Triangle Wave PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel of CH3 to 9.

CHm is used as the master channel of CHm+1 (m = 10, 12, 14).

TAUDn Channel	U phase / UB phase	V phase / VB phase	W phase / WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	—	—	—	Not used.
CH4	√	—	—	Triangle PWM output with dead time (U phase/UB phase)
CH5	√	—	—	
CH6	—	√	—	Triangle PWM output with dead time (V phase/VB phase)
CH7	—	√	—	
CH8	—	—	√	Triangle PWM output with dead time (W phase/WB phase)
CH9	—	—	√	
CH10	√	—	—	Reduced dead time pulse (U phase/UB phase)
CH11	√	—	—	
CH12	—	√	—	Reduced dead time pulse (V phase/VB phase)
CH13	—	√	—	
CH14	—	—	√	Reduced dead time pulse (W phase/WB phase)
CH15	—	—	√	

Note: √: Used; —: Not used

(2) Configuration

The high accuracy triangle wave PWM output function with the dead time is realized by using the triangle wave PWM output function/one-shot pulse output function of TAUDn and PIC1A in combination. The following figure shows the block diagram of the high accuracy triangle wave PWM output function with the dead time.

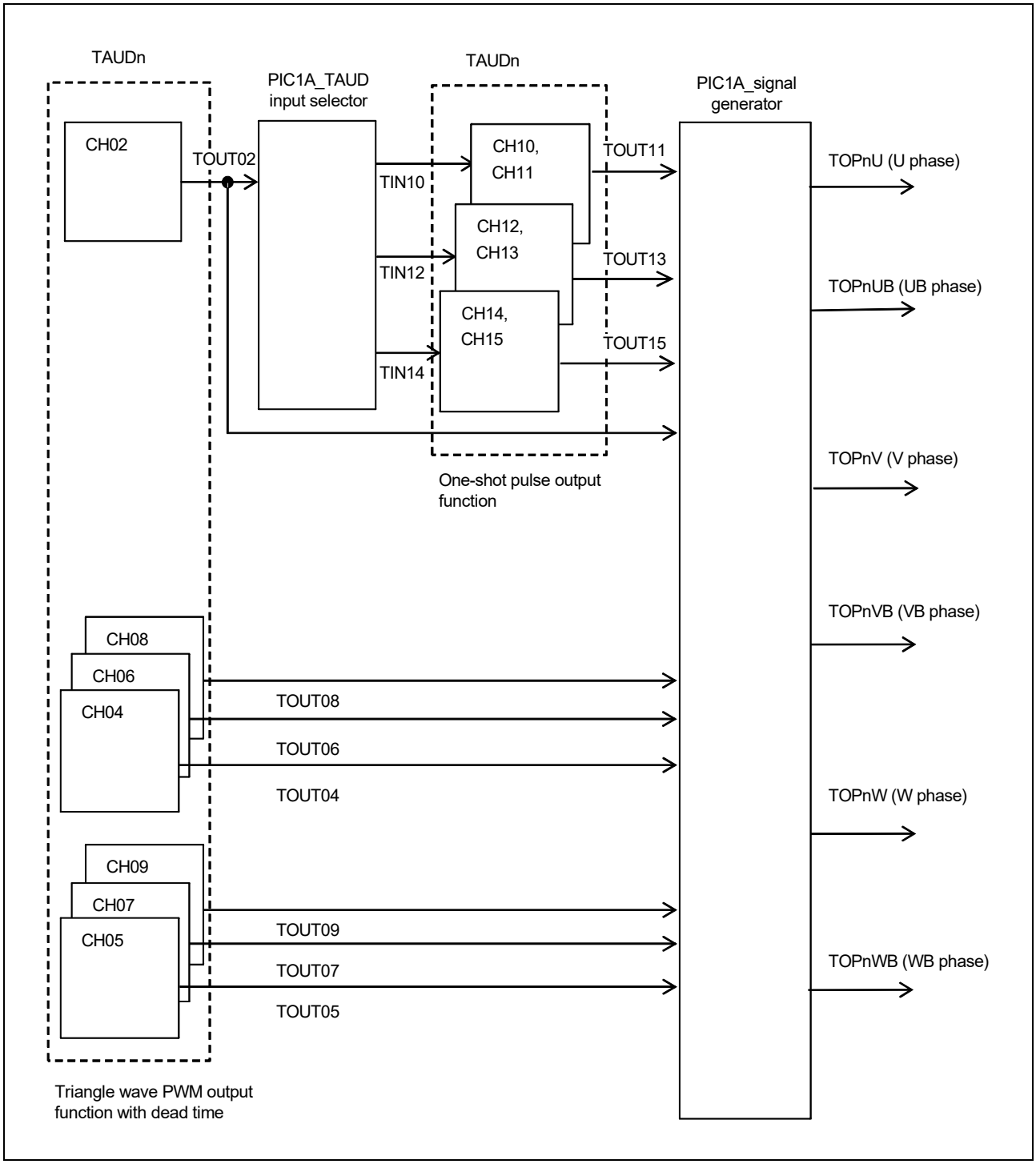


Figure 23.10 Block Diagram of High Accuracy Triangle Wave Three-phase PWM Output With Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [PIC1A_TAUD input selector]
  TOUT02 is selected and output to TIN10.
- [TAUDn] One-shot pulse output function
  CH10 and CH11 are used in combination. Setting the delay value and the pulse width to CDR10 and CDR11, respectively allows the one-shot pulse output signal (TOUT11) to be generated.
- [TAUDn] Triangle wave PWM output function with dead time
  CH02, CH04, and CH05 are used in combination. Setting the period, duty, and dead time to CDR02, CDR04, and CDR05, respectively generates the triangle wave PWM output signal with dead time (TOUT04 and TOUT05).
- [PIC1A_signal generator]
  The reduced dead time pulses (UO1 and UO2) are generated at PFN001 from the one-shot pulse output signal.
  UO1 and UO2 are synthesized with TOUT04 and TOUT05 at FN00 and FN01, respectively, added dead time variable range pulse, and TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal) are generated.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

The following figure shows the block diagram of PIC1A.

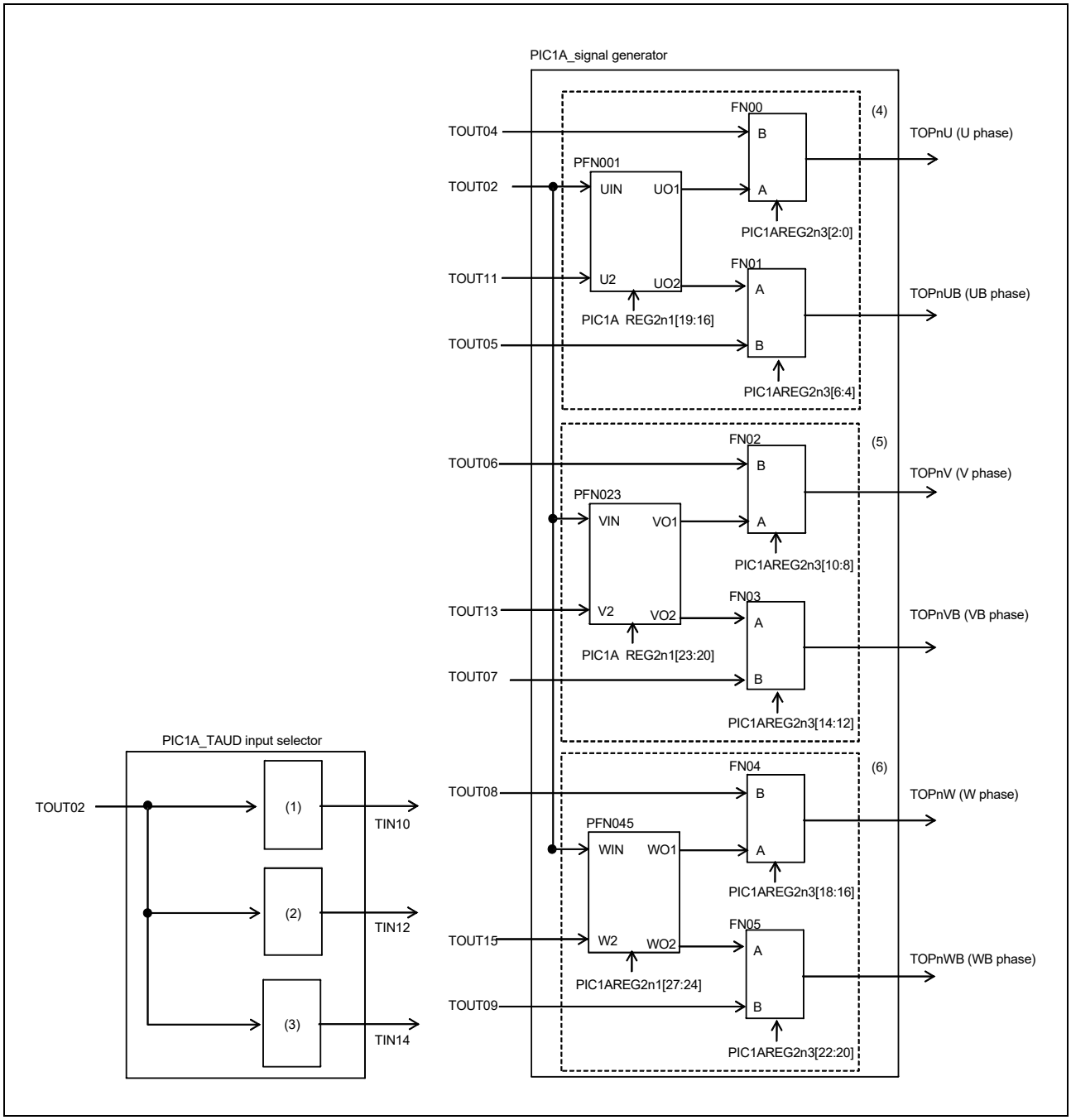


Figure 23.11 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

(1) PIC1A_TAUD input selector (U phase/ UB phase)
The values to output TOUT02 as TIN10 (common to active high/low)
PIC1AREG2n0[18] = 1_B
PIC1AREG2n2[17:16] = 10_B
PIC1ATAUDnSEL[21:20] = 00_B

(2) PIC1A_TAUD input selector (V phase/ VB phase)
The values to output TOUT02 as TIN12 (common to active high/low)
PIC1AREG2n0[18] = 1_B
PIC1AREG2n2[21:20] = 10_B
PIC1ATAUDnSEL[25:24] = 00_B

(3) PIC1A_TAUD input selector (W phase/ WB phase)
The values to output TOUT02 as TIN14 (common to active high/low)
PIC1AREG2n0[18] = 1_B
PIC1AREG2n2[25:24] = 10_B
PIC1ATAUDnSEL[29:28] = 00_B

(4) PIC1A_ signal generator (U phase/ UB phase)
The values to output one-phase PWM (active high/low) from TAUDnO10 and TAUDnO11
PIC1AREG2n1[19:16] = 1010_B (active high), 1111_B (active low)
PIC1AREG2n3 [06:04] = 100_B (active high), 101_B (active low)
PIC1AREG2n3[02:00] = 100_B (active high), 101_B (active low)

(5) PIC1A_signal generator (V phase/ VB phase)
The values to output one-phase PWM (active high/low) from TAUDnO12 and TAUDnO13
PIC1AREG2n1[23:20] = 1010_B (active high), 1111_B (active low)
PIC1AREG2n3 [14:12] = 100_B (active high), 101_B (active low)
PIC1AREG2n3[10:08] = 100_B (active high), 101_B (active low)

(6) PIC1A_signal generator (W phase/ WB phase)
The values to output one-phase PWM (active high/low) from TAUDnO14 and TAUDnO15
PIC1AREG2n1[27:24] = 1010_B (active high), 1111_B (active low)
PIC1AREG2n3 [22:20] = 100_B (active high), 101_B (active low)
PIC1AREG2n3[18:16] = 100_B (active high), 101_B (active low)

(4) Function

Detail of the function is described using U phase/UB phase as an example. The function with V phase/VB phase and W phase/WB phase are operated under the same logic as that of U phase/UB phase with different input signals and register settings.

- U phase combination circuit (PFN001)
  This circuit generates reduced dead time pulses*1 (FN00A and FN01A) that are used to insert the dead time pulse generated by the one-shot pulse output function into the triangle wave PWM generated by the triangle wave PWM output function with the dead time. For the block diagram, see **Figure 23.1, Block Diagram of PFN001**.

Note 1. Reduced dead time pulses are the pseudo pulses that are inserted into the PWM output provided by the triangle wave PWM output function with dead time of TAUDn and that are modeled on the dead time pulses that are generated in the range where duty cycle is close to 100% or 0% in PWM output in HT-PWM mode of TSG3n.

- Logical operation circuits (FN0i (i = 0, 1))
  This circuit synthesizes the triangle wave PWM output (TOUT04 and TOUT05) from the triangle wave PWM output function with the dead time and the outputs from the combinational circuit PFN001 (UO0 and UO1) to generate PWM with reduced dead time pulses inserted. The synthesizing logic of this circuit is selected with PIC1AREG2n3k (k = 00 to 02, 04 to 06). For the block diagram, see **Figure 23.2, Block Diagram of FN00**.

Detail of the function is described using high accuracy triangle wave PWM output function (U phase/UB phase) as an example.

The following figure shows the timing diagram when U phase duty cycle = 0% and UB phase duty cycle = 100% at active high.

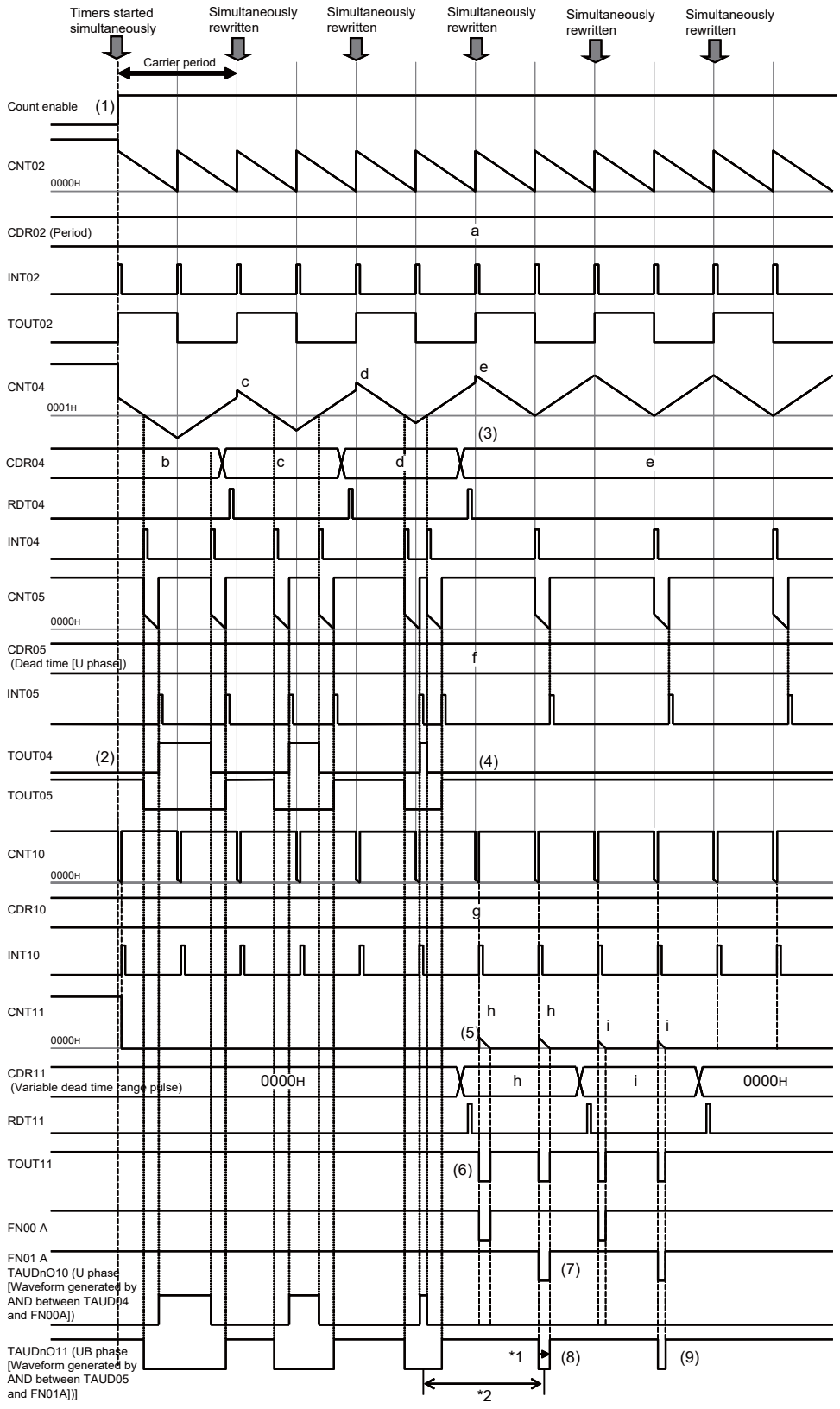


Figure 23.12 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 0%, UB phase duty cycle = 100%) Output with Dead Time (Active High)

(1) With the simultaneous timer start function, the timers to be used are started simultaneously.

(2) TOUT04 and TOUT05 are generated by the triangle wave PWM output function with dead time.

(3) A 0% duty cycle value is set in CDR04 for U phase output.

(4) The setting described in step (3) sets the TOUT04 output to the inactive level and the TOUT05 output to the active level.

(5) To generate the reduced dead time pulse, a value for the reduced dead time pulse width is set in CDR11 when a 0% duty cycle value for U phase output is set in step (3).

(6) CH10 count starts with the effective edge of TOUT02 and INT10 is generated when the counter underflows. CH11 count starts by generation of INT10 and the reduced dead time pulse (TOUT11) of the width set in CDR11 is output.

(7) The reduced dead time pulse (UO1 and UO2) are generated from TOUT02 and TOUT11 in PFN001.

(8) UO1 and UO2 are synthesized with TOUT04 and TOUT05 in FN00 and FN01, and are output as TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal).

CAUTION

Since the reduced dead time pulses are sawtooth waves, they expand and contract on one side, unlike the triangle wave pulses, which expand and contract on both sides. Since one-side expansion and contraction applies to the reduced dead time pulses, a one-phase PWM output period in the reduced dead time range is longer by half the width of an inserted reduced dead time pulse.

The following figure shows the timing diagram when U phase duty cycle = 100% and UB phase duty cycle = 0% at active high.

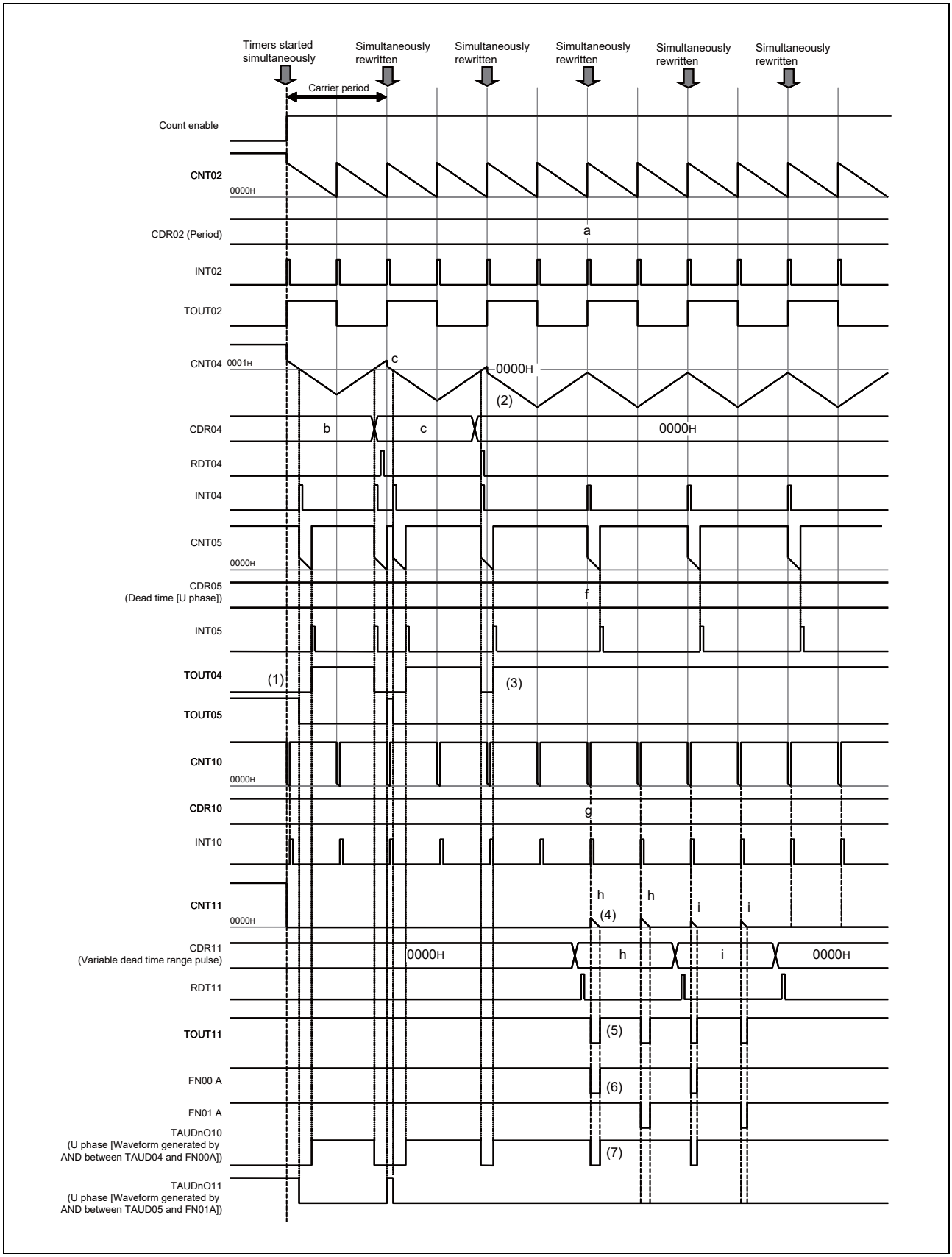


Figure 23.13 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active High)

(1) With the simultaneous timer start function, the timers to be used are started simultaneously.

(2) TOUT04 and TOUT05 are generated by the triangle wave PWM output function with dead time.

(3) A 100% duty cycle value (CDR 04 = 0000_H) is set in CDR04 for U phase output.

(4) TOUT04 outputs the active level and TOUT05 outputs the inactive level.

(5) A value for the reduced dead time pulse width is set in CDR11 one period after setting 100% duty cycle for U phase output.

(6) CH10 count starts with the effective edge of TOUT02 and INT10 is generated when the counter underflows. CH11 count starts by generation of INT10 and the reduced dead time pulse (TOUT11) of the width set in CDR11 is output.

(7) The reduced dead time pulse (UO1 and UO2) are generated from TOUT02 and TOUT11 in PFN001.

(8) UO1 and UO2 are synthesized with TOUT04 and TOU05 in FN00 and FN01, and are output as TOPnU (U phase PWM signal) and TOPnUB (UB phase PWM signal).

CAUTION

As shown in Figure 23.14, if a value is set for the variable dead time range pulse width in CDR11 at the same time as a 100% duty cycle value is set for U phase output in CDR04, the variable dead time range pulse affects the last PWM output from TOUT04 (indicated by Figure 23.14, (1)) by the amount of time indicated by Figure 23.14, (2), which is due to the functional specification. To eliminate this influence, CDR11 must be set one period after setting CDR04.

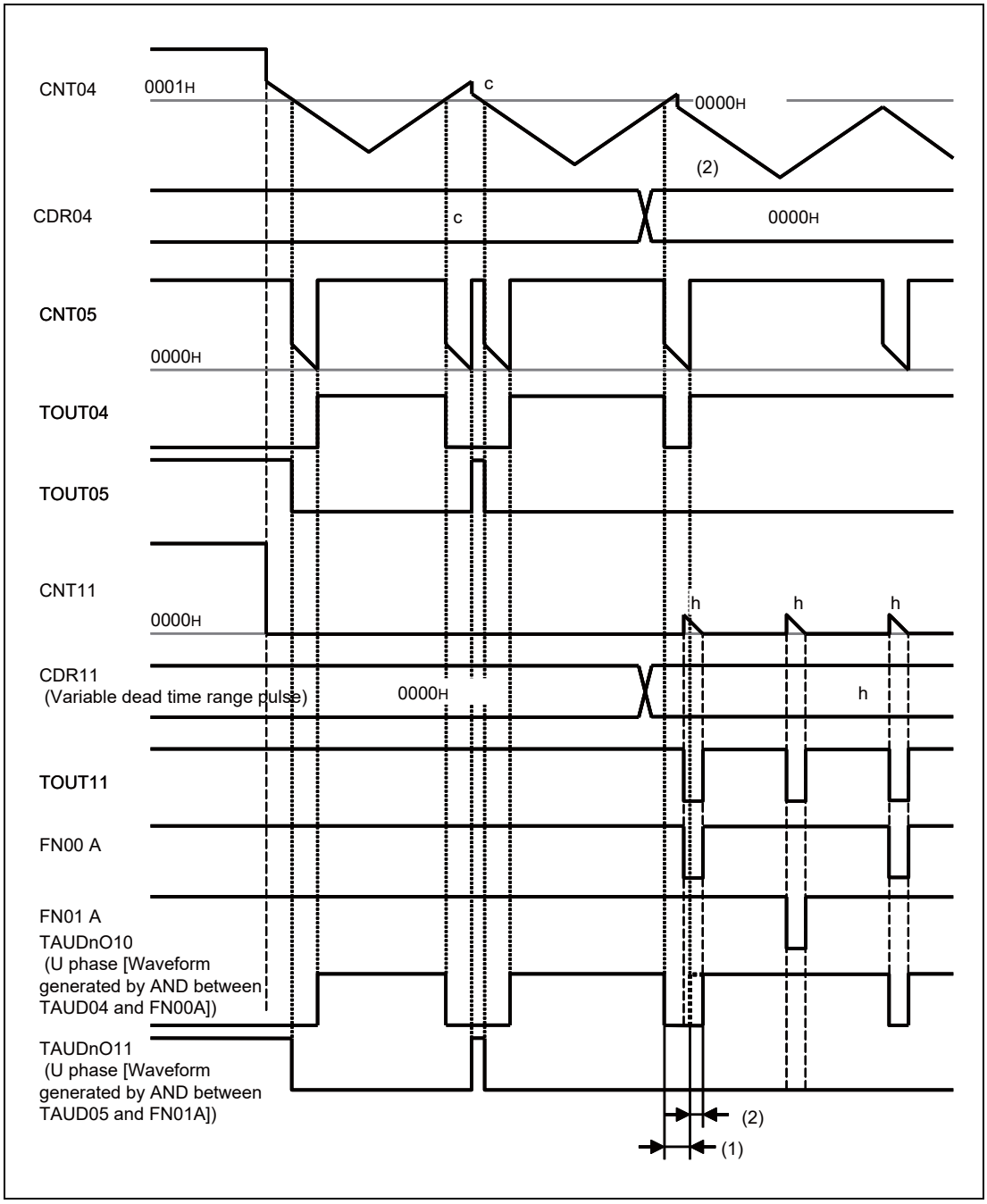


Figure 23.14 Timing Diagram of an Example of Reduced Dead Time Pulse Giving Influence on Triangle Wave PWM Output with Dead Time

The following figure shows the timing diagram when U phase duty cycle = 100% and UB phase duty cycle = 0% at active low.

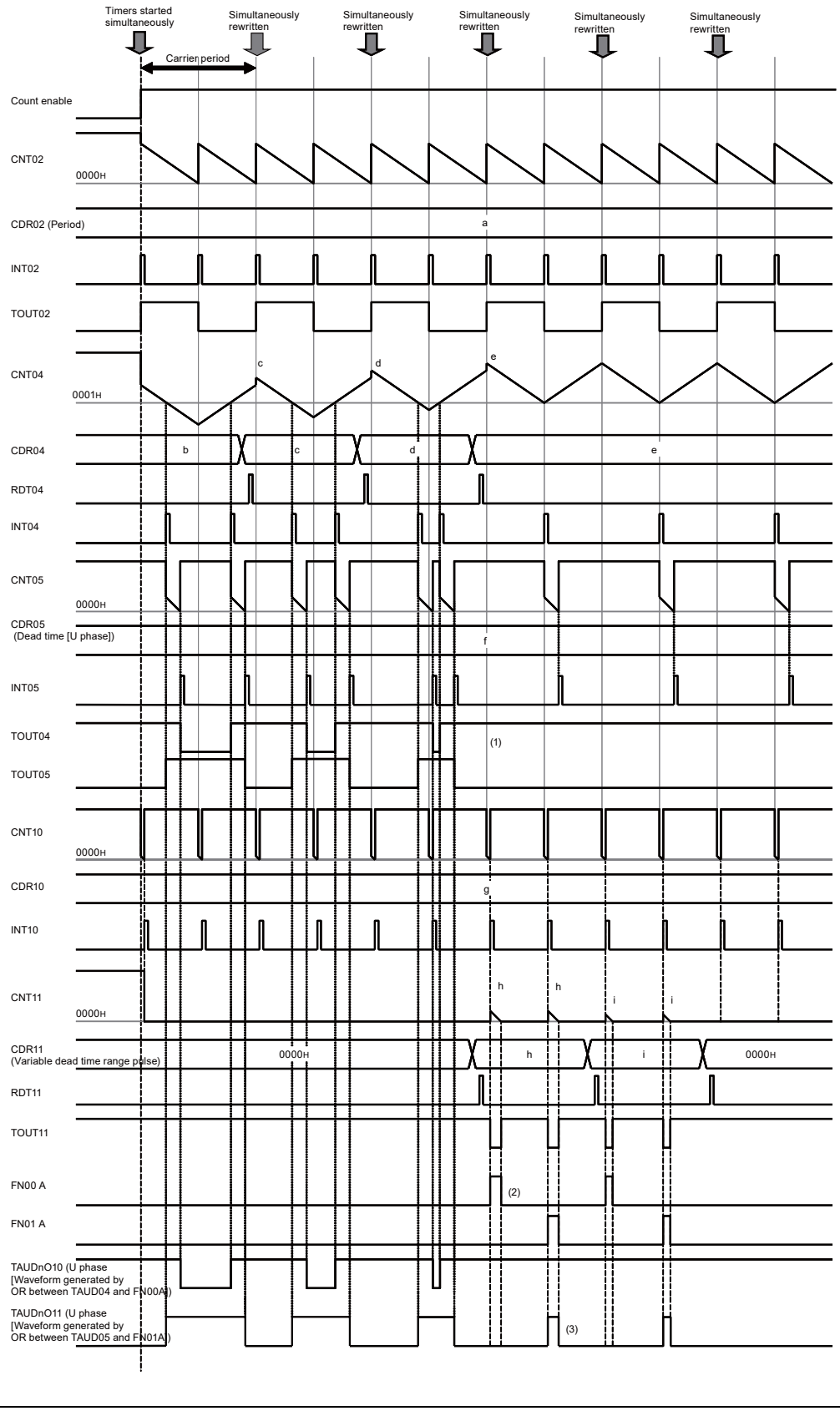


Figure 23.15 Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active Low)

The operation flow from timer operation start to triangle wave PWM output with dead time is same as **Figure 23.13, Timing Diagram of High Accuracy Triangle Wave PWM (U phase duty cycle = 100%, UB phase duty cycle = 0%) Output with Dead Time (Active High)**, with the difference of the PWM output signal from TOUT04 and TOUT05 are active low.

CAUTION

Set each CDR for the one-shot pulse output function so that the following condition is satisfied.

CDR05 ≥ (CDR10 + CDR11)

If the condition above is not satisfied, the output waveform may be influenced. To minimize the influence, satisfy the condition above, and also fix the value of CDR11 to 0000_H until the reduced dead time pulse is required.

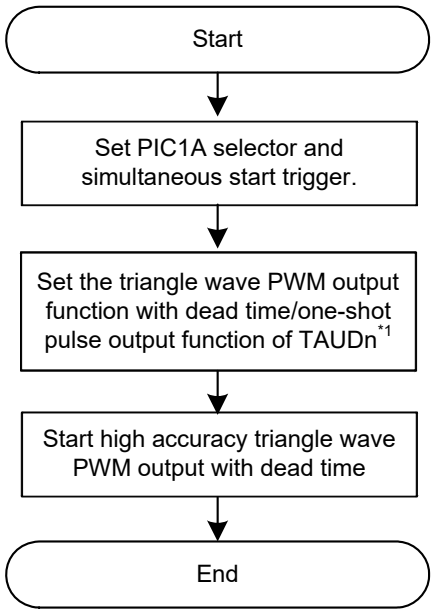
Set the both edges to be detected of TIN10 (TOUT02) as effective, and set TAUDnTOL11 to 1 (active low).

Select the count clock signal (CK0 to 3) of the same clock for TAUDn.

After high accuracy triangle wave PWM output with dead time is started, do not set the value of variable dead time pulse width at the same time as setting a 100% duty cycle value for U phase, V phase, and W phase.

(5) Flow Chart

The flow chart of this function is shown in the following figure.



Note 1. For the setting of triangle wave PWM output function with dead time of TAUDn, see **Section 17.4.12.8, Triangle PWM Output Function with Dead Time**, for the setting of one-shot pulse output function, see **Section 17.4.12.2, One-Shot Pulse Output Function**.

23.2.3.4 Delay Pulse Output Function with Dead Time

(1) Overview

This function allows generation of PWM output with the dead time, that is, delayed as specified from the period timing using TAUDn.

With this function, PWM output can be reset in the next period unlike with the function described in **23.2.3.2, PWM Output Function with Dead Time**.

The following table lists the number of channels used.

PWM Output with Dead Time	Number of TAUDn Channels
One-phase PWM output (U phase/UB phase)	5 channels (master channel: 1, slave channel: 4)
Two-phase PWM output (U phase/UB phase, V phase/VB phase)	9 channels (master channel: 1, slave channel: 8)
Three-phase PWM output (U phase/UB phase, V phase/VB phase, W phase/WB phase)	13 channels (master channel: 1, slave channel: 12)

Note: Above are examples of combination.

Usages of each channel of TAUDn are listed in the following table. CH2 is used as the master channel of CH3 to 9.

TAUDn Channel	U phase / UB phase	V phase / VB phase	W phase / WB phase	Usage
CH0	—	—	—	Not used.
CH1	—	—	—	Not used.
CH2	√	√	√	Carrier period (common to each phase)
CH3	√	√	√	Reserved
CH4	√	—	—	Delay pulse input (U phase/UB phase)
CH5	√	—	—	
CH6	—	√	—	Delay pulse input (V phase/VB phase)
CH7	—	√	—	
CH8	—	—	√	Delay pulse input (W phase/WB phase)
CH9	—	—	√	
CH10	√	—	—	U phase output (TOUT10)
CH11	√	—	—	UB phase output (TOUT11)
CH12	—	√	—	V phase output (TOUT12)
CH13	—	√	—	VB phase output (TOUT13)
CH14	—	—	√	W phase output (TOUT14)
CH15	—	—	√	WB phase output (TOUT15)

Note: √: Used, —: Not used

(2) Configuration

The delay pulse output function with the dead time is realized by using the delay pulse output function/ one-phase PWM output function of TAUDn and PIC1A in combination. The following figure shows the block diagram of the delay pulse output function with the dead time.

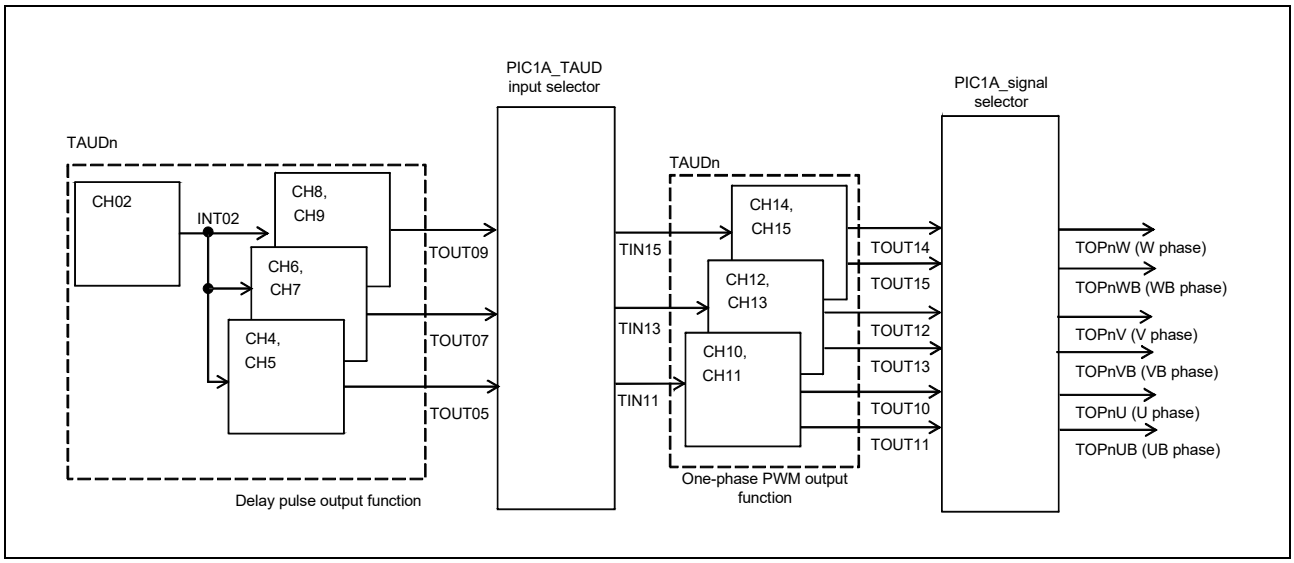


Figure 23.16 Block Diagram of Delay Pulse Output Function with Dead Time

The configuration of this function is described as follows using the PWM output of U phase/UB phase as an example.

- [TAUDn] Delay pulse output function
  CH02, CH04, and CH05 are used in combination. Setting the period, delay value, and the pulse width to CDR02, CDR04, and CDR05, respectively generates the delay pulse output signal (TOUT05).
- [PIC1A_TAUD input selector]
  TOUT05 is selected for output as TIN11.
- [TAUDn] One-phase PWM output function
  CH10 and CH11 are used in combination. Setting the dead time value to CDR11, and inserting dead time into the PWM signal to be input to TIN11 allows TOUT10 (U phase PWM signal) and TOUT11 (UB phase PWM signal) to be output.
- [PIC1A_signal selector]
  TOUT10 and TOUT11 inputs are selected and output to TOPnU and TOPnUB pins, respectively.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

(3) Registers

Block diagram of PIC1A is shown in the following figure.

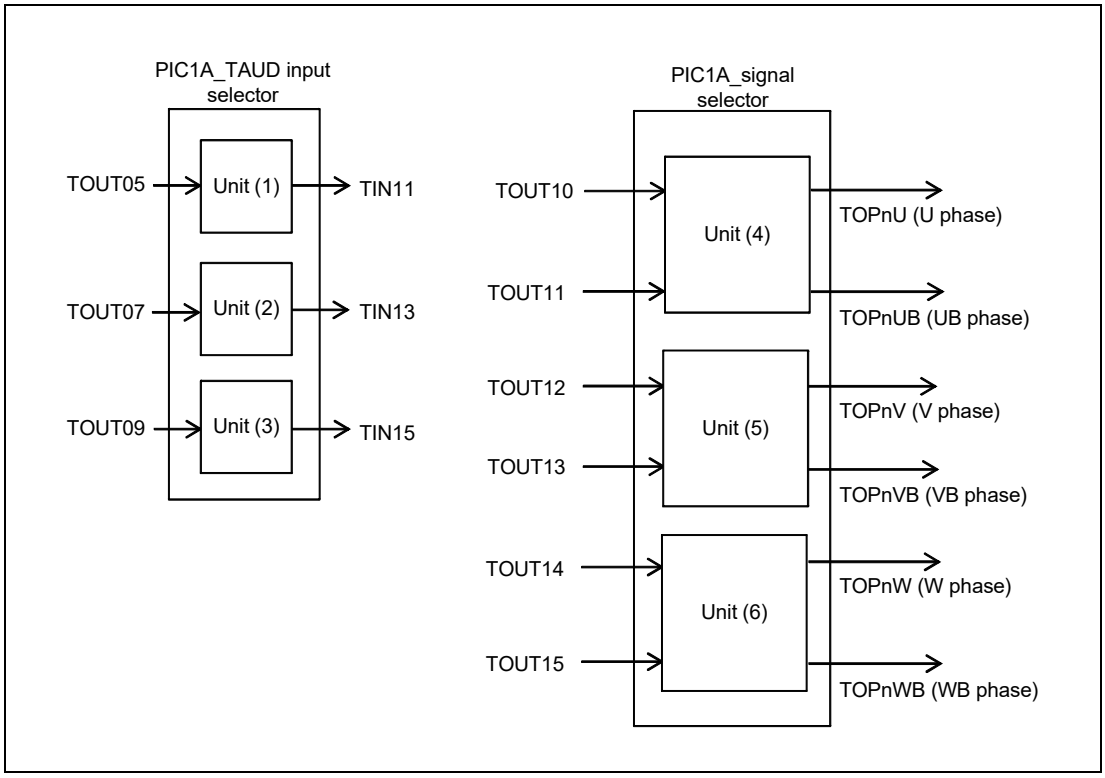


Figure 23.17 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

U phase/ UB phase

The values to output TOUT05 as TIN11 (**Figure 23.17**, unit (1))

PIC1AREG2n2[19:18] = 10_B

PIC1AREG2n2[2] = 0_B

PIC1ATAUDnSEL[23:22] = 00_B

The values to output the TOUT10 and TOUT11 as TOPnU and TOPnUB, respectively (**Figure 23.17**, unit (4))

PIC1AREG2n1[19:16] = 0000_B

PIC1AREG2n3[2:0] = 000_B

PIC1AREG2n3[6:4] = 000_B

V phase/ VB phase

The values to output TOUT07 as TIN13 (**Figure 23.17**, unit (2))

PIC1AREG2n2[23:22] = 10_B

PIC1AREG2n2[3] = 0_B

PIC1ATAUDnSEL[27:26] = 00_B

The values to output TOUT12 and TOUT13 as TOPnV and TOPnVB, respectively (**Figure 23.17**, unit (5))

PIC1AREG2n1[23:20] = 0000_B

PIC1AREG2n3[10:8] = 000_B

PIC1AREG2n3[14:12] = 000_B

W phase/ WB phase

The values to output TOUT09 as TIN15 (**Figure 23.17**, unit (3))

PIC1AREG2n2[27:26] = 10_B

PIC1AREG2n2[4] = 0_B

PIC1ATAUDnSEL[31:30] = 00_B

The values to output TOUT14 and TOUT15 as TOPnW and TOPnWB, respectively (**Figure 23.17**, unit (6))

PIC1AREG2n1[27:24] = 0000_B

PIC1AREG2n3 [18:16] = 000_B

PIC1AREG2n3[22:20] = 000_B

(4) Function

Detail of the function is described using the delay pulse output (U phase/UB phase) as an example.

The following figure shows the timing diagram.

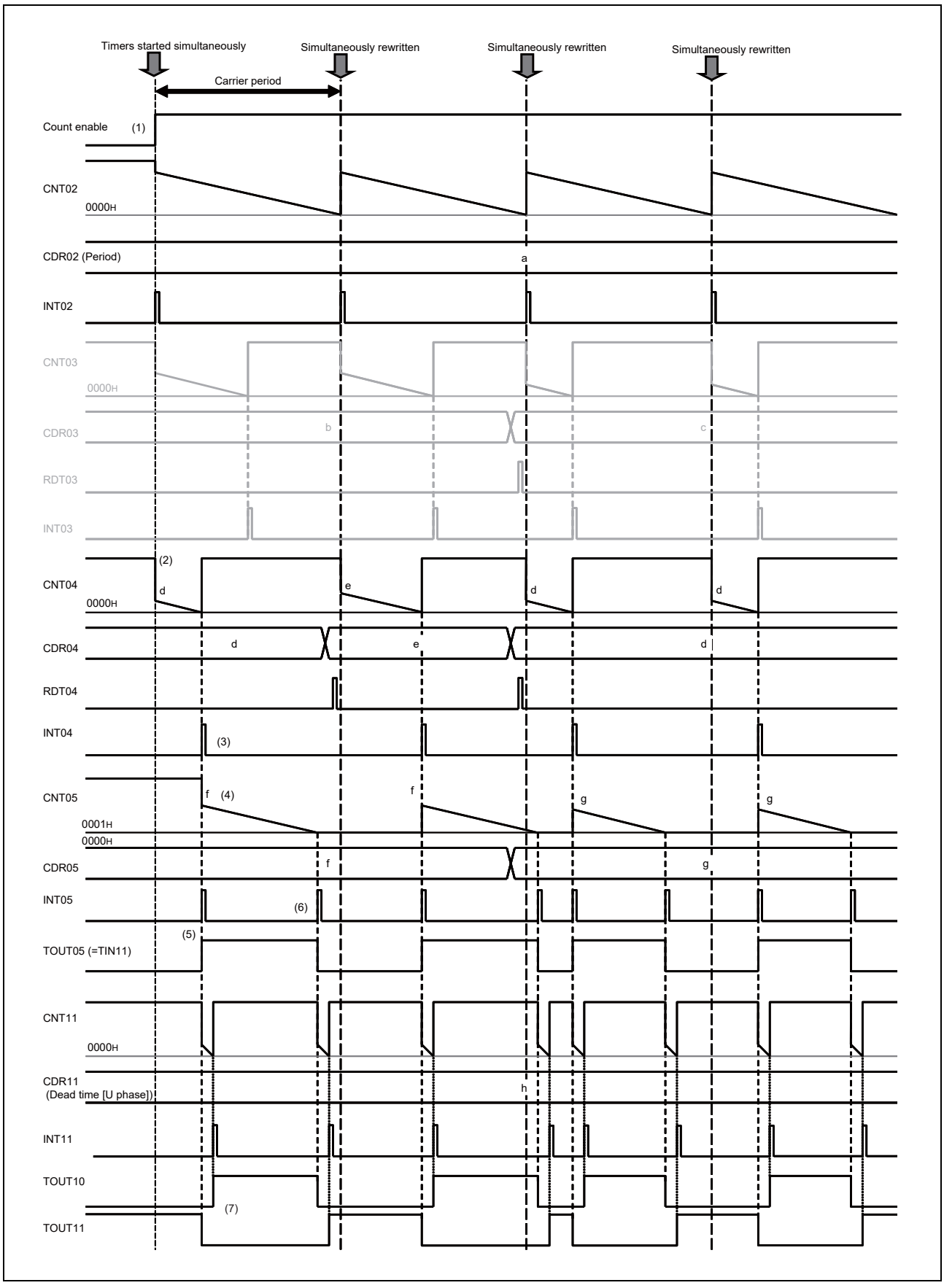


Figure 23.18 Timing Diagram of Delay Pulse Output with Dead Time (U phase/ UB phase)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

(2) For CH04, CH02 underflow allows the set value in CDR04 to be reloaded to CNT04.

(3) CH04 underflow allows the delay timing signal (INT04) to be generated.

(4) INT04 generation allows the set value to be reloaded from CDR05 to CNT05, thus starting the count operation of CH05.

(5) Here, INT05 is generated and the TOUT05 output level changes to the active level.

(6) Upon CH05 underflow, INT05 is generated again and the TOUT05 output level changes to the inactive level. TOUT05 is supplied to TIN11.

(7) The U phase PWM signal (TOUT10) and UB phase PWM signal (TOUT11) with the dead time is generated and output according to the TIN11 edges detected, and output to TOPnU and TOPnUB.

Similar configuration is applied to V phase/VB phase and W phase/WB phase.

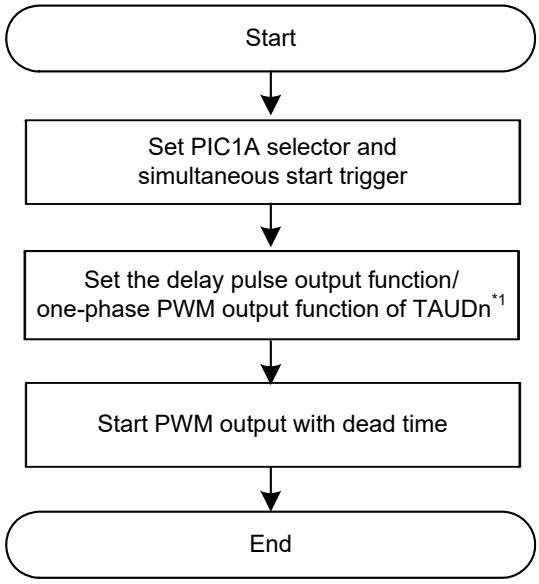
CAUTION

Do not set the delay value which extends over carrier period.

Select the count clock signal of the same clock for TAUDn.

(5) Flow Chart

The following flow chart shows the PMW output function with the dead time.



Note 1. For the setting of delay pulse output function of TAUDn, see **Section 17.4.12.3, Trigger Start PWM Output Function**, for the setting of one-phase PWM output function, see **Section 17.4.12.11, One-Phase PWM Output Function**.

23.2.3.5 Trigger and Pulse Width Measurement Function

(1) Overview

This function allows measurement of trigger periods by inputting the trigger signal output from ENCAn to TAUJ0 and TAUDn.

The following table lists the ENCAn interrupt trigger signals to be measured by each timer and channel.

Timer	Channel	Interrupt Signal to be Measured
TAUJ0	CH0	ENCAT0IEC
	CH1	ENCAT0IEC
	CH2	ENCAT1IEC
	CH3	ENCAT1IEC
TAUD0	CH0	ENCAT0EQ0 or ENCAT0EQ1
	CH1	ENCAT0EQ1
	CH2	ENCAT0EQ0
TAUD1	CH0	ENCAT1EQ0 or ENCAT1EQ1
	CH1	ENCAT1EQ1
	CH2	ENCAT1EQ0

(2) Configuration

The trigger and pulse width measurement function is realized by using the TAUJ0 and TINm input pulse of TAUDn, and PIC1A in combination. The following figure shows the block diagram of the trigger and pulse width measurement function.

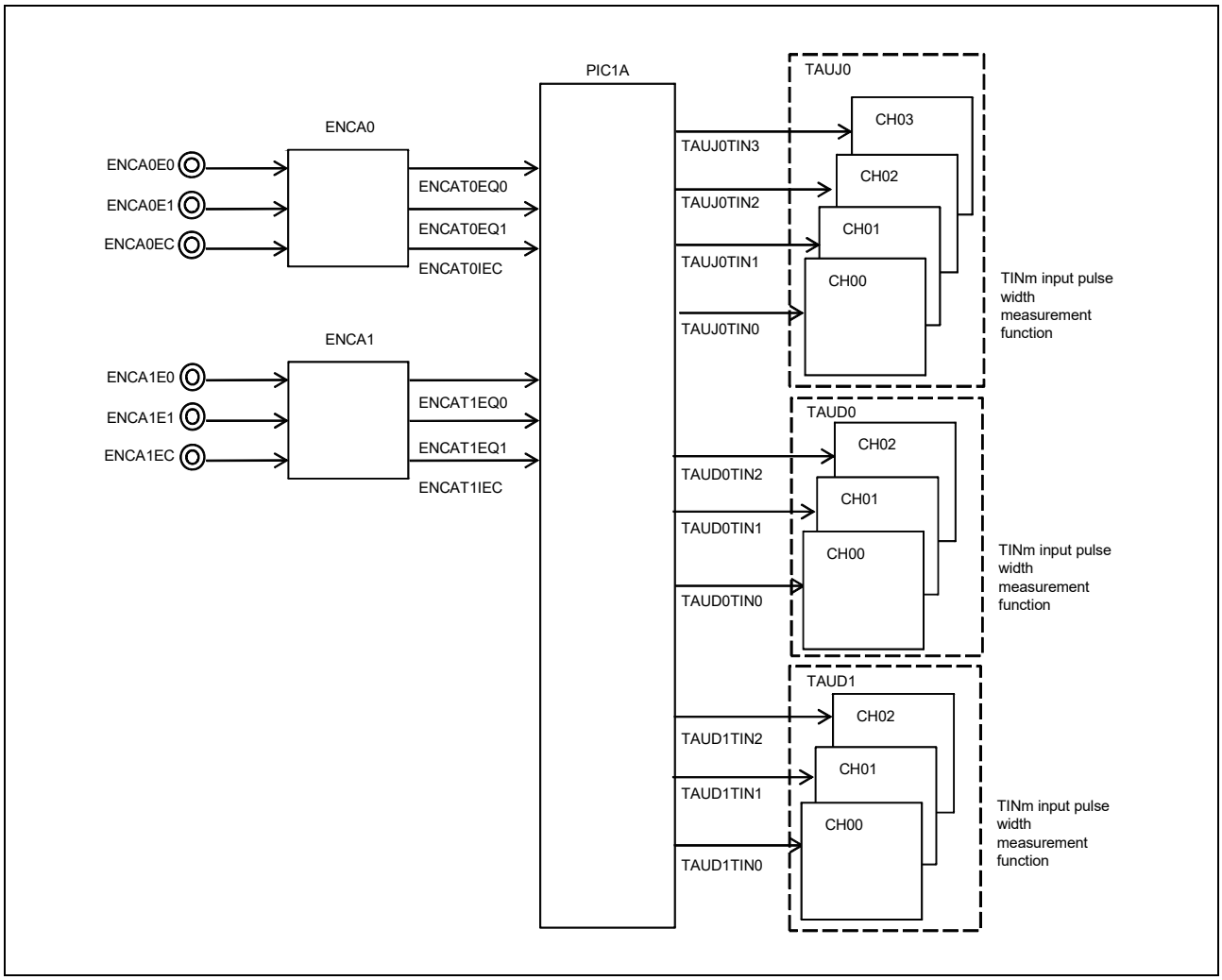


Figure 23.19 Block Diagram of Trigger and Pulse Width Measurement Function

The configuration of this function is described as follows using TAUJ CH0 as an example.

- [ENCA0]
  ENCAT0IEC interrupt signal is generated each time ENCA0 timer counter is cleared by input from ENCA0EC pin.
- [PIC1A] Latch and toggle output (DT) circuit
  ENCA0IEC interrupt trigger signal selected by the DT circuit is converted into a level-sensitive toggle signal and output to TAUJTIN0.
- [TAUJ0] TINm input pulse width measurement function
  TAUJ0 CH0 is used. TAUJ0CNT0 is captured each time input signal is toggled. The counter is cleared and restarted.

Similar configuration is applied for trigger and pulse width measurement of TAUD0 and TAUD1.

(3) Registers

Block diagram of PIC1A is shown in the following figure.

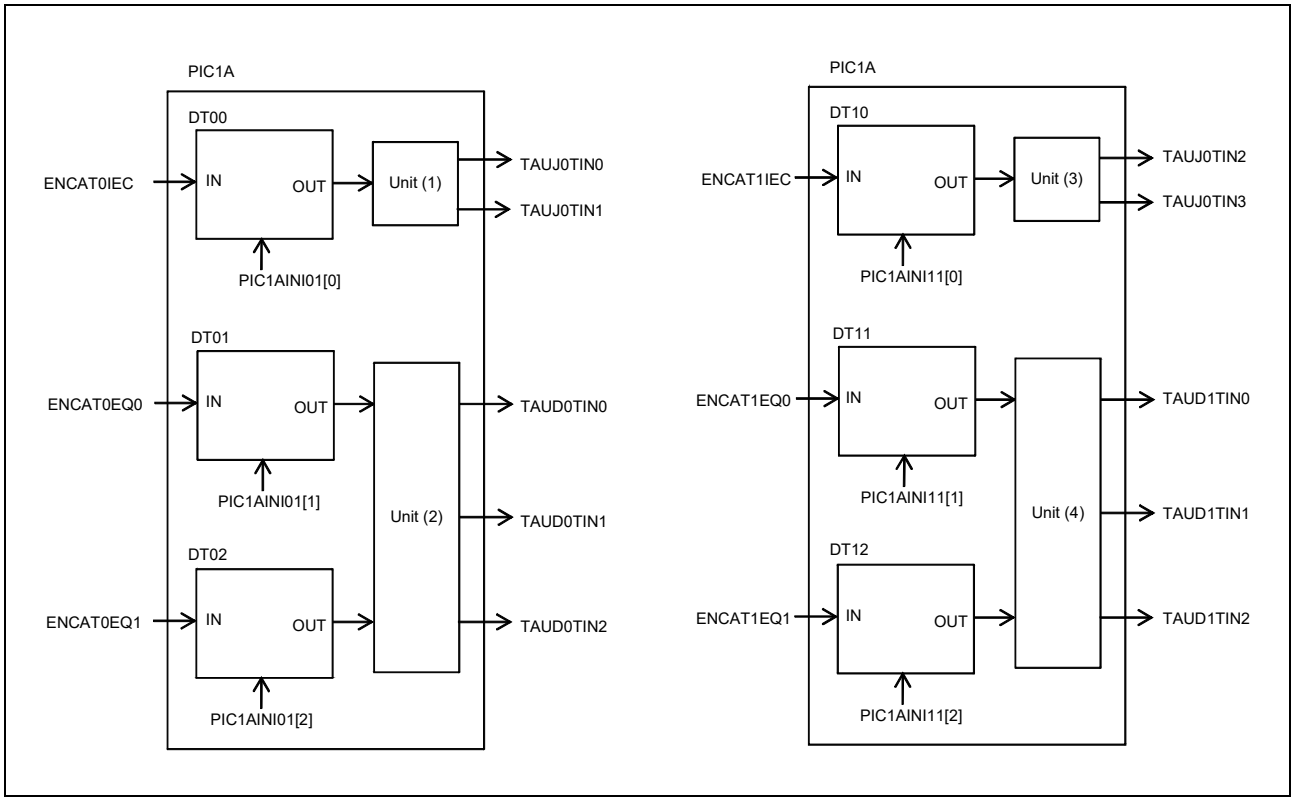


Figure 23.20 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

ENCA0

- ENCAT0EC trigger and pulse width measurement
  The register values to select the timer which performs ENCAT0EC trigger and pulse width measurement. (**Figure 23.20**, unit (1))

Register Value			
PIC1AREG31			
1	**0**	**TAUJ0.TIN00**	**TAUJ0.TIN01**
0	0	Not selected	
0	1	ENCAT0IEC	—
1	0	—	ENCAT0IEC
1	1	ENCAT0IEC	ENCAT0IEC

Note: Write 0 (value after reset) to PIC1AREG30[22,17:16]

- ENCAT0EQ0 and ENCAT0EQ1 trigger and pulse width measurement
  The register values to select the timer which performs ENCAT0EQ0 and ENCAT0EQ1 trigger and pulse width measurement. (**Figure 23.20**, unit (2))

Register Value PIC1AREG31										
13	12	11	10	9	8	7	6	TAUD0.TIN00	TAUD0.TIN01	TAUD0.TIN02
0	0	0	0	0	0	0	0	Not selected		
0	0	1	1	0	0	0	1	ENCAT0EQ0	ENCAT0EQ1	—
0	1	0	0	0	0	0	1	ENCAT0EQ1	—	ENCAT0EQ0
0	1	1	0	0	0	0	1	ENCAT0EQ1	ENCAT0EQ1	ENCAT0EQ0
0	1	1	1	0	0	0	1	ENCAT0EQ0	ENCAT0EQ1	ENCAT0EQ0

Note: Do not set the values other than the settings listed above for this function. Write 0 (value after reset) to PIC1ATAUD0SEL[5:0] and PIC1AREG30[22,17:16,1:0].

- Enables initialization of the DT02 to DT00 circuits
  The register values to enable initialization of the DT02 to DT00 circuits.
  PIC1AINI01[2:0] = 111_B (initialized)

ENCA1

ENCAT1EC trigger and pulse width measurement

The register values to select the timer which performs ENCAT1EC trigger and pulse width measurement. (**Figure 23.20**, unit (3))

Register Value PIC1AREG31		TAUJ0.TIN02	TAUJ0.TIN03
4	3		
0	0	Not selected	
0	1	ENCAT1IEC	—
1	0	—	ENCAT1IEC
1	1	ENCAT1IEC	ENCAT1IEC

Note: Write 0 (value after reset) to PIC1AREG30[20:19,11:10]

- ENCAT1EQ0 and ENCAT1EQ1 trigger and pulse width measurement
  The register values to select the timer which performs ENCAT1EQ0 and ENCAT1EQ1 trigger and pulse width measurement. (**Figure 23.20**, unit (4))

Register Value PIC1AREG31								TAUD1.TIN00	TAUD1.TIN01	TAUD1.TIN02
22	21	20	19	18	17	16	15			
0	0	0	0	0	0	0	0	Not selected		
0	0	1	1	0	0	0	1	ENCAT1EQ0	ENCAT1EQ1	—
0	1	0	0	0	0	0	1	ENCAT1EQ1	—	ENCAT1EQ0
0	1	1	0	0	0	0	1	ENCAT1EQ1	ENCAT1EQ1	ENCAT1EQ0
0	1	1	1	0	0	0	1	ENCAT1EQ0	ENCAT1EQ1	ENCAT1EQ0

Note: Do not set the values other than the settings listed above for this function. Write 0 (value after reset) to PIC1ATAUD1SEL[5:0] and PIC1AREG30[20:19,9:6].

- Enable initialization of DT12 to DT10 circuits
  The register values to enable initialization of the DT12 to DT10 circuits.
  PIC1AINI11[2:0] = 111_B (initialized)

(4) Function

Detail of the function is described here.

The following figure shows the timing diagram.

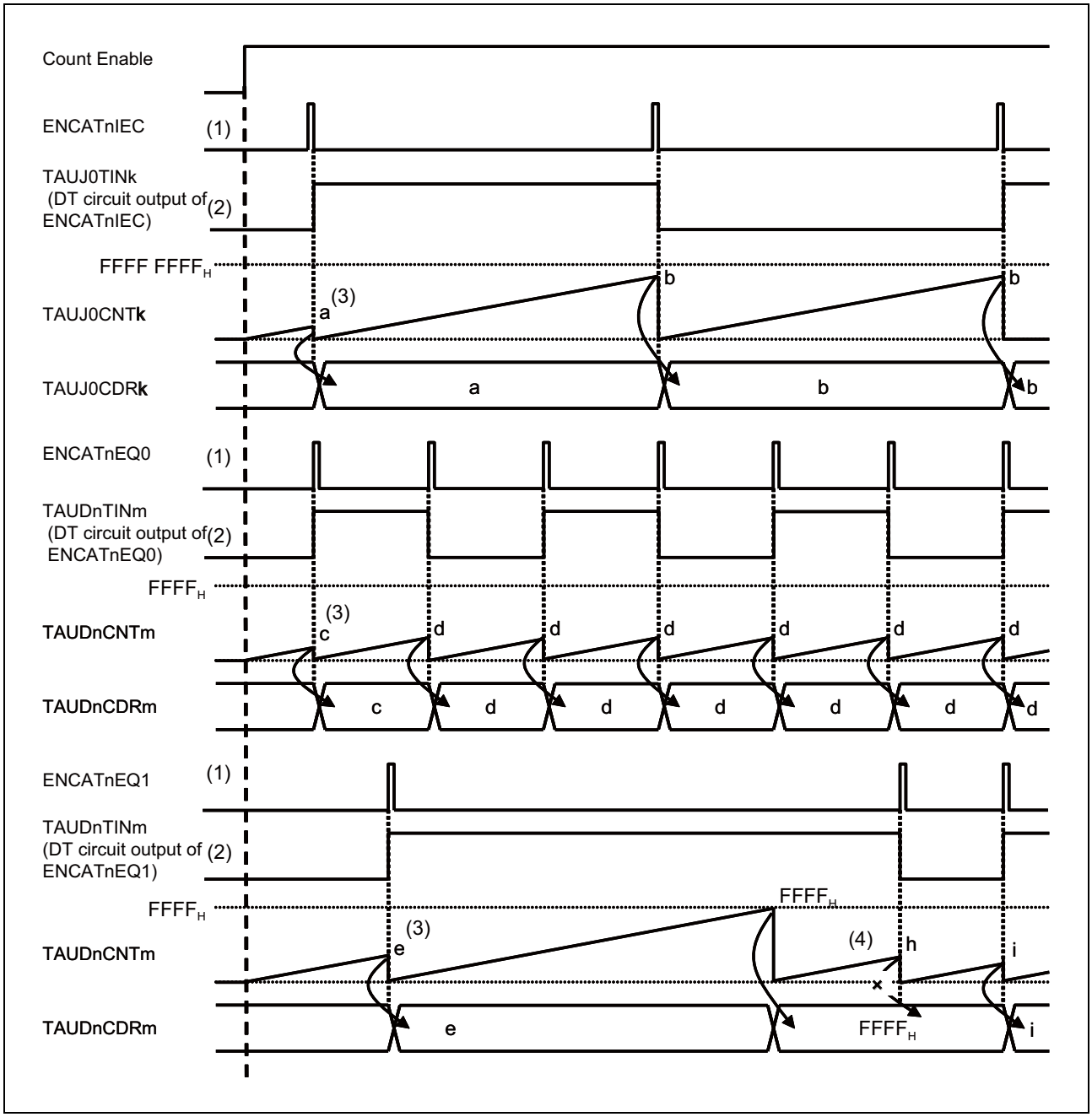


Figure 23.21 Timing Diagram of Trigger and Pulse Width Measurement

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

(2) The interrupt trigger signal output from ENCAn is converted to the level-sensitive toggle signal by the DT circuit and is output to TINm of TAUJ0 and TINm of TAUDn.

(3) The CNTm value is captured into CDRm on the TINm toggle timing and cleared.

(4) When an overflow occurs, the greatest count value ($FFFF_H$ for TAUDn and FFFF $FFFF_H$ for TAUJ0) is captured and the counter is cleared at the same time. The count value is not captured on the first trigger after the overflow. (When TAUDnCMORm.TAUDnCOS[1] = 1_B)

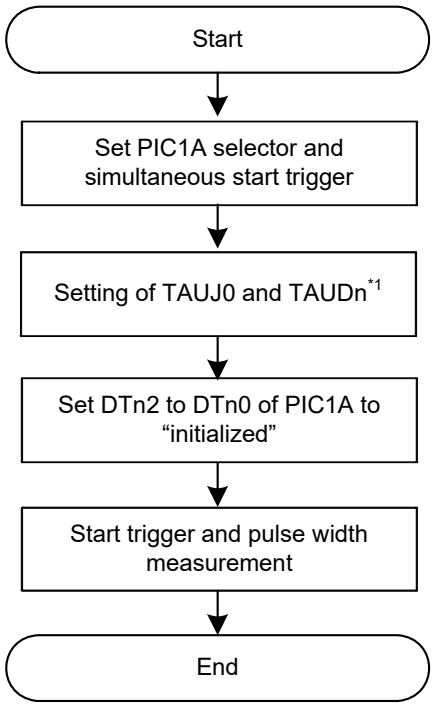
CAUTION

Operation at an overflow varies depending on the setting of TAUJ and TAUD. For the detail of the TAUJ setting, see Section 18.4.9.3, TAUJTTINm input pulse interval measurement function, for the detail of the TAUD setting, see Section 17.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function. In this function, set the effective edge to be detected by TAUJ0 and TINm of TAUDn as both (rising edge and falling edge).

(5) Flow Chart

The following figure shows the setting flow of this function.

This flow chart can be set at both during ENCAn operation and while waiting for simultaneous start trigger.



Note 1. For the setting of TAUDn, see **Section 17.4.9.7, TAUDTTINm Input Pulse Interval Measurement Function**, for the setting of TAUJ0, see **Section 18.4.9.3, TAUJTTINm input pulse interval measurement function**.

The ENCAn registers to use this function are as follows.

ENCAnCTL[15:0] = $xx00\_0000\_x00x\_xxxx_B$

ENCAnIOC0[7:0] = 0000_0000_B

ENCAnIOC1[7:0] = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of ENCA.

23.2.3.6 Encoder Capture Trigger Select Function

(1) Overview

The function selects any of the ADCCnTRGm (ADCCn conversion start trigger m), TAUDnTINTm (TAUDn-CHm interrupt signal), and ENCAnI1 (signal 1 of the ENCAn external pin input).

(2) Configuration

The encoder capture trigger select function is realized by using ADCCnTRGm (ADCCn conversion start trigger m), TAUDnTINTm (TAUDn CHm interrupt signal), ENCAnI1 (ENCAn external pin input 1 signal), and PIC1A.

Block diagram of this function is shown in the following figure.

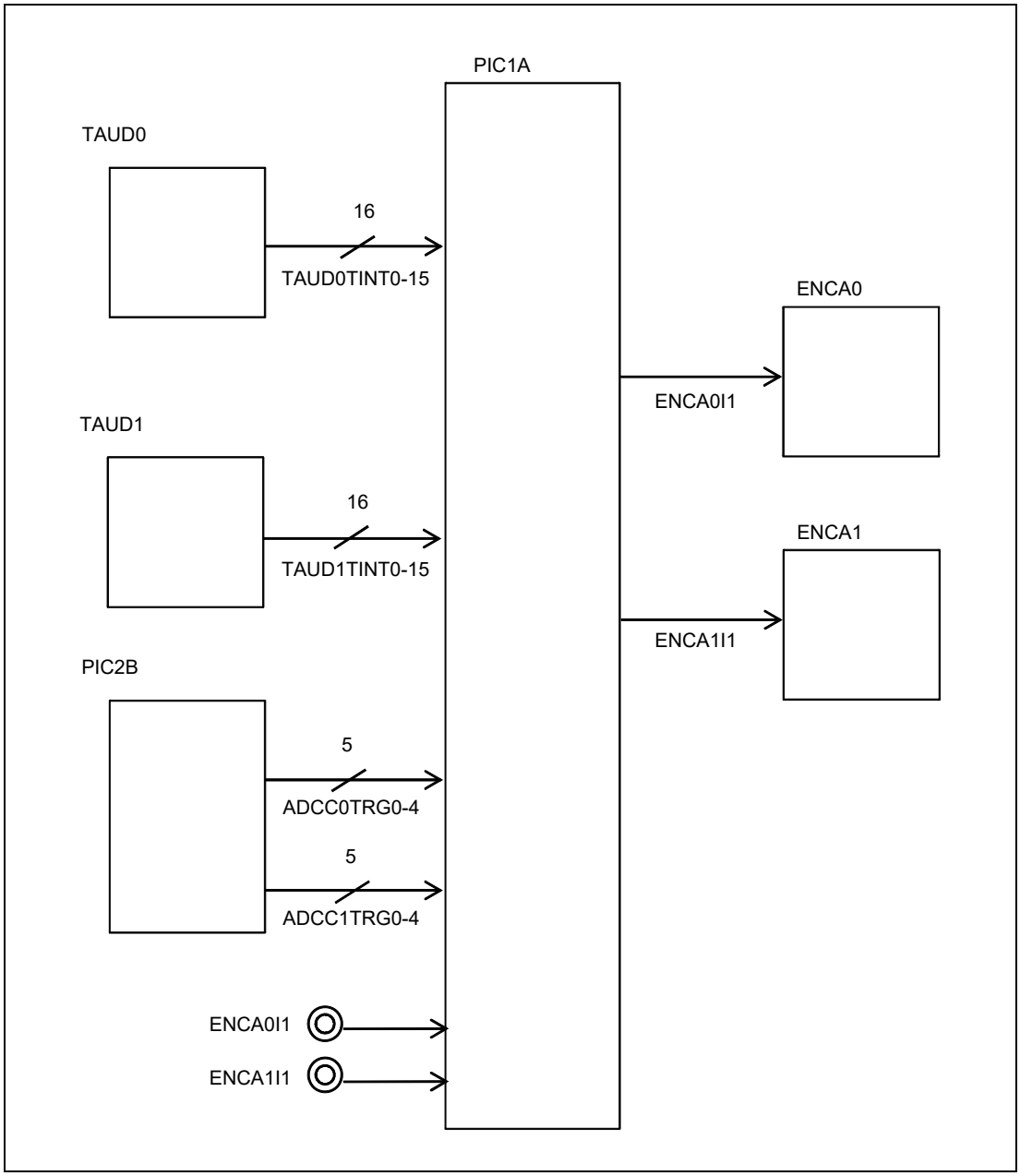


Figure 23.22 Block Diagram of Encoder Capture Trigger Select Function

An example of selecting CH0 of TAUD0 as a capture trigger input of ENCA0 is described as follows.

PIC1AENCSEL400[7] = 1_B

PIC1AENCSEL400[3:0] = 0000_B

PIC1AREG30[18] = 1_B

PIC1AREG30[5:2] = 0000_B

(3) Registers

Block diagram of PIC1A is shown in the following figure.

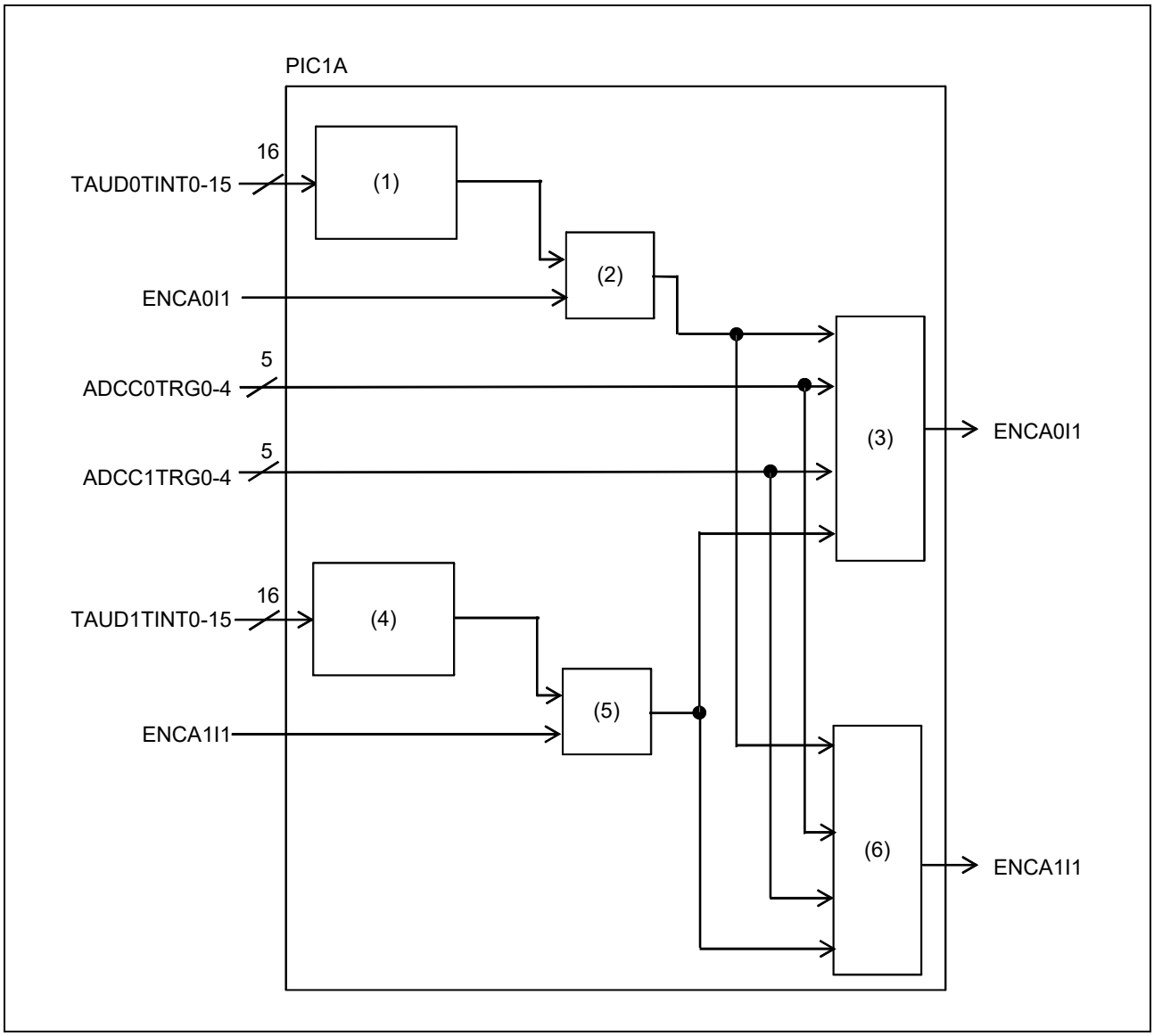


Figure 23.23 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

ENCA0

- (1) TAUDnTINTm selection
  The register values to select TAUDnTINTm. Set 1 to the PIC1AENCSEL400[7] to select TAUD0TINTm.

Register Value					Output of (1)
PIC1AENCSEL400					
7	3	2	1	0	
1	0	0	0	0	INTTAUD0I0
1	0	0	0	1	INTTAUD0I1
1	0	0	1	0	INTTAUD0I2
1	0	0	1	1	INTTAUD0I3
1	0	1	0	0	INTTAUD0I4
1	0	1	0	1	INTTAUD0I5
1	0	1	1	0	INTTAUD0I6
1	0	1	1	1	INTTAUD0I7
1	1	0	0	0	INTTAUD0I8
1	1	0	0	1	INTTAUD0I9
1	1	0	1	0	INTTAUD0I10
1	1	0	1	1	INTTAUD0I11
1	1	1	0	0	INTTAUD0I12
1	1	1	0	1	INTTAUD0I13
1	1	1	1	0	INTTAUD0I14
1	1	1	1	1	INTTAUD0I15

- (2) TAUD0TINTm and ENCA0I1 pins selection
  The register values to select either the output (1) or ENCA0I1.

Register Value	Output (2)
PIC1AREG30	
18	
1	Output (1)
0	ENCA0I1

- (3) ENCA0I1 selection
  The register values to select any of the output (2), output (5), ADCC0TRG0 to 4, and ADCC1TRG0 to 4.

Register Value				
PIC1AREG30				
5	4	3	2	ENCA0I1
0	0	0	0	Output (2)
0	0	0	1	Output (5)
0	0	1	0	ADCC0TRG4
0	0	1	1	ADCC0TRG3
0	1	0	0	ADCC0TRG2
0	1	0	1	ADCC0TRG1
0	1	1	0	ADCC0TRG0
0	1	1	1	ADCC1TRG4
1	0	0	0	ADCC1TRG3
1	0	0	1	ADCC1TRG2
1	0	1	0	ADCC1TRG1
1	0	1	1	ADCC1TRG0

Note: Do not set the values other than the settings listed above for this function.

ENCA1

- (4) TAUD1TINTm selection
  The register values to select TAUD1TINTm. Set 1 to the PIC1AENCSEL410[7] to select TAUD1TINTm.

Register Value					
PIC1AENCSEL410					
7	3	2	1	0	Output (4)
1	0	0	0	0	INTTAUD1I0
1	0	0	0	1	INTTAUD1I1
1	0	0	1	0	INTTAUD1I2
1	0	0	1	1	INTTAUD1I3
1	0	1	0	0	INTTAUD1I4
1	0	1	0	1	INTTAUD1I5
1	0	1	1	0	INTTAUD1I6
1	0	1	1	1	INTTAUD1I7
1	1	0	0	0	INTTAUD1I8
1	1	0	0	1	INTTAUD1I9
1	1	0	1	0	INTTAUD1I10
1	1	0	1	1	INTTAUD1I11
1	1	1	0	0	INTTAUD1I12
1	1	1	0	1	INTTAUD1I13
1	1	1	1	0	INTTAUD1I14
1	1	1	1	1	INTTAUD1I15

- (5) TAUD1TINTm and ENCA1I1 pins selection
  The register values to select either the output (4) or ENCA1I1.

Register Value	
PIC1AREG30	
21	Output (5)
1	Output (4)
0	ENCA1I1

- (6) ENCA1I1 selection
  The register values to select any of the output (2), output (5), ADCC0TRG0 to 4, and ADCC1TRG0 to 4.

Register Value				
PIC1AREG30				
15	14	13	12	ENCA1I1
0	0	0	0	Output (5)
0	0	0	1	Output (2)
0	0	1	0	ADCC0TRG4
0	0	1	1	ADCC0TRG3
0	1	0	0	ADCC0TRG2
0	1	0	1	ADCC0TRG1
0	1	1	0	ADCC0TRG0
0	1	1	1	ADCC1TRG4
1	0	0	0	ADCC1TRG3
1	0	0	1	ADCC1TRG2
1	0	1	0	ADCC1TRG1
1	0	1	1	ADCC1TRG0

Note: Do not set the values other than the settings listed above for this function.

(4) Function

Detail of the function is described with an example of selecting TAUDnTINTm as a capture trigger signal.

The following figure shows the timing diagram.

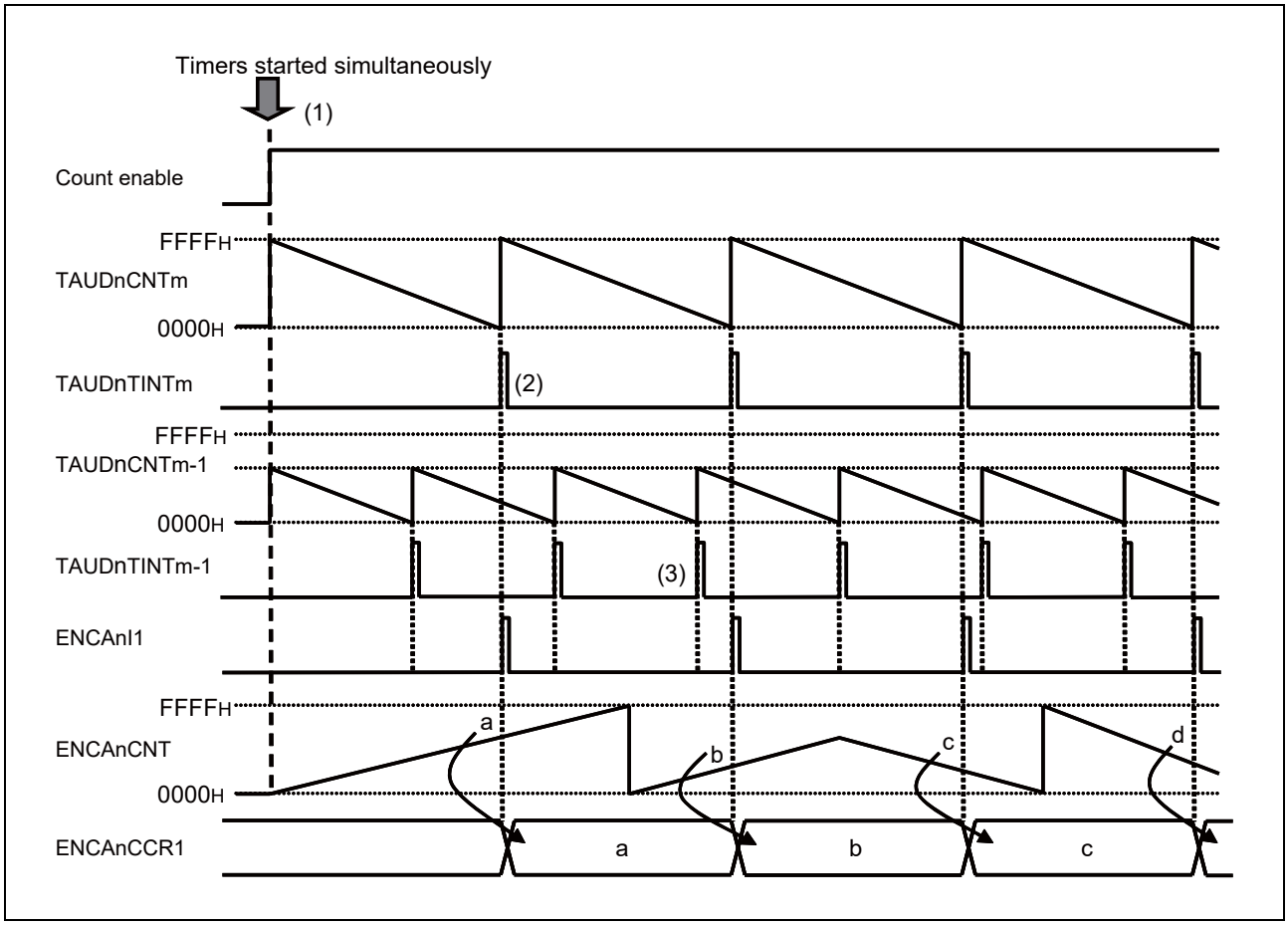


Figure 23.24 Timing Diagram of Encoder Capture Trigger Select Function (TAUDnTINTm)

(1) Using the simultaneous start trigger function, the timers to be used are started simultaneously.

(2) On generation of an effective edge of TAUDnTINTm, ENCAn captures ENCAnCNT.

Do not select ENCAn interrupt trigger signal (INTENCATnI1) as the ADCCn trigger described in **Section 23.3.3.1, ADCC Trigger Select Function**, the correct operation cannot be performed because the following loop occurs: ADCCnTRG1 generation à ENCAn capture operation à INTENCAnI1 generation by capture operation ADCCnTRG1 generation.

The following figure shows a timing chart of the loop paths of PIC1A, PIC2B, and ENCAn.

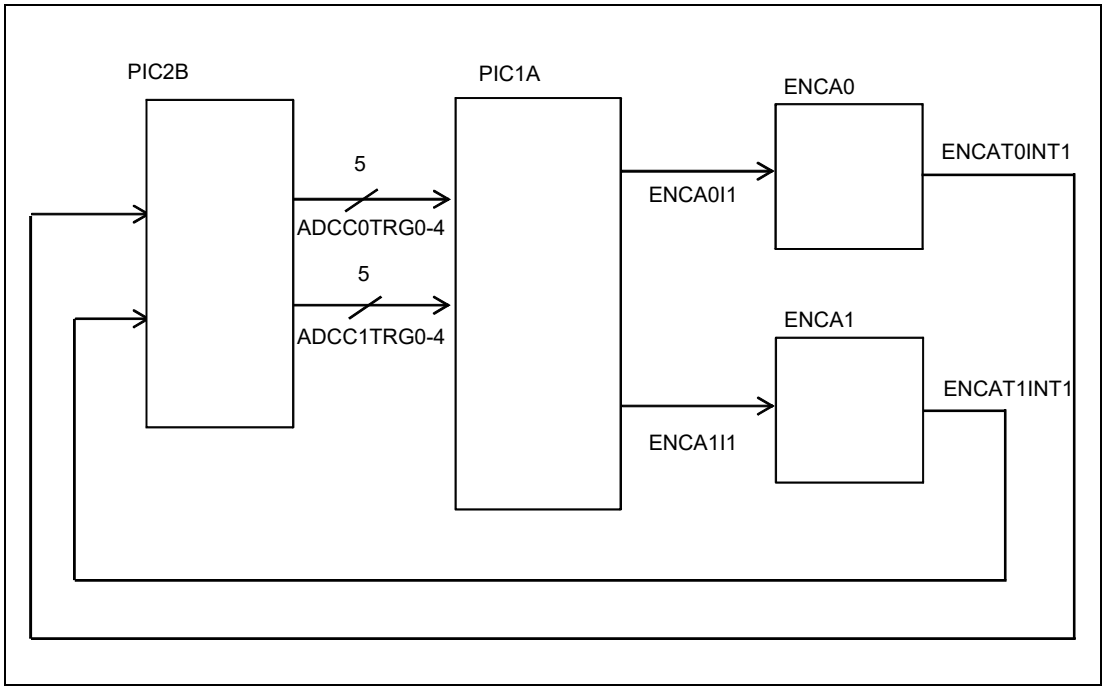


Figure 23.25 Timing Chart of Loop Paths of PIC1A, PIC2B, and ENCAn

(5) Flow Chart

Select the encoder capture trigger before starting the encoder timer.

ENCAn settings to use this function are described as follows.

ENCAnCTL[15:0] = 0000_001x_0 00x_xxxx$_B$

ENCAnIOC0[7:0] = 0000_01xx$_B$

ENCAnIOC1[7:0] = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of ENCA.

23.2.3.7 Two-Phase Encoder Control Function (Control Method 1)

(1) Overview

This function allows switching of the output patterns of the motor control function (TSG3n) using the two-phase encoder control function (ENCAn).

(2) Configuration

Switching of output pattern in 120-DC mode by encoder result is realized by using ENCAn and TSG3n, and PIC1A in combination. The following figure describes the block diagram of two-phase encoder control function (control method 1).

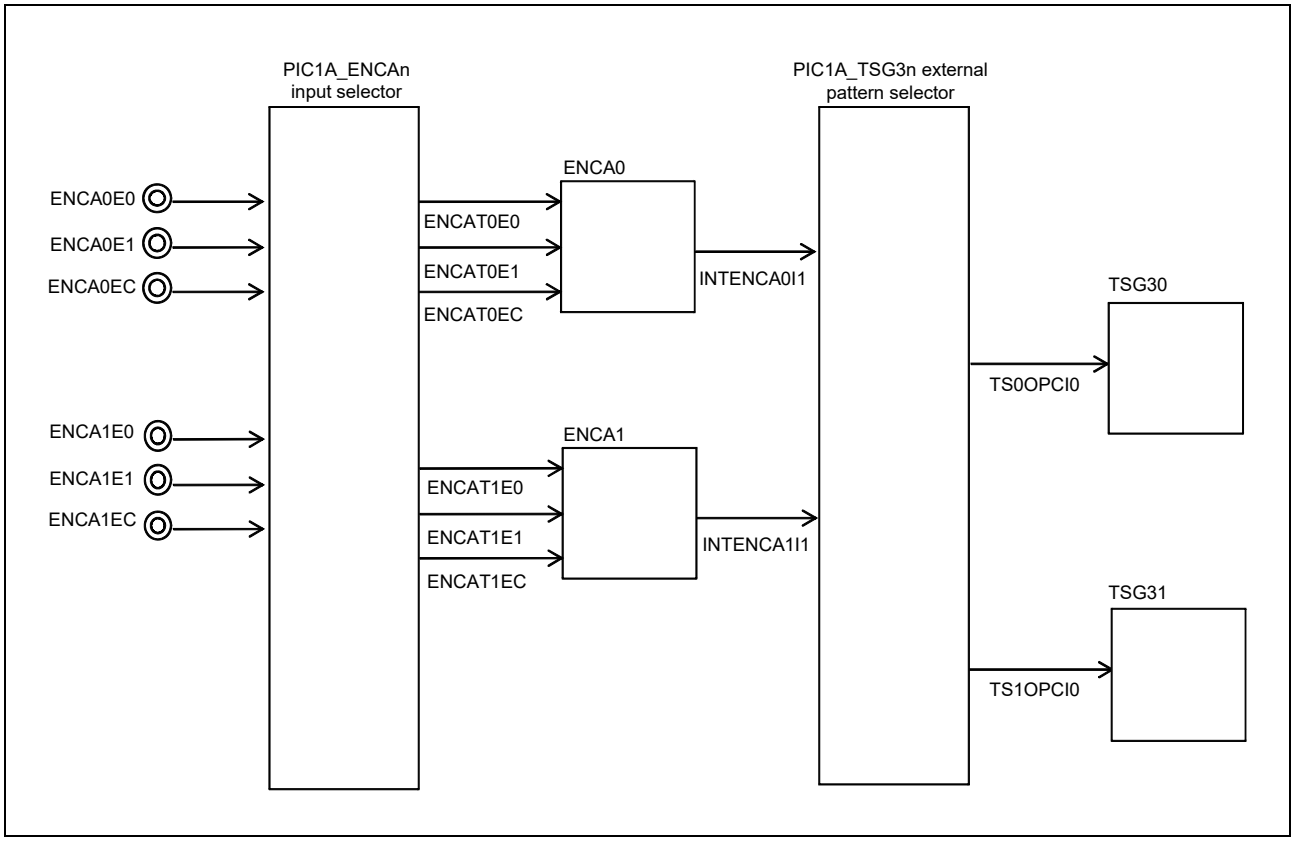


Figure 23.26 Block Diagram of Two-Phase Encoder Control Function (Control Method 1)

The configuration of two-phase encoder control function (control method 1) is described as follows.

- [PIC1A_ENCAn input selector]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output as ENCATnE0, ENCATnE1, and ENCATnEC.
- [ENCAn]
  INTENCAnI1 is output by two-phase encoder processing.
- [PIC1A_TSG3n external pattern selector]
  INTENCAnI1 is selected and output to TS0OPCI0 or TS1OPCI0.
- [TSG3n]
  Output pattern in 120-DC mode is switched by TSG3nOPCI0.

(3) Registers

The values of PIC1A registers used in this function are as follows.

PIC1A_ENCAn input selector

The register values to output ENCAn pin inputs (ENCAnE0, ENCAnE1, and ENCAnEC) as ENCATnE0, ENCATnE1, and ENCATnEC.

PIC1AREG30[22] = 0_B

PIC1AREG30[20:19] = 00_B

PIC1AREG30[17:16] = 00_B

PIC1AREG30[11:6] = 000000_B

PIC1AREG30[1:0] = 00_B

PIC1A_TSG3n external pattern selection

The register values to select the interrupt signal to be input as TSG30 external pattern.

Register Value				
PIC1AREG50				
10	8	6	5	TS0OPCI0
X	0	0	1	INTENCA0I1
0	X	1	0	INTENCA1I1

Note: Do not set the values other than the settings listed above for this function.

The register values to select the interrupt signal to be input as TSG31 external pattern.

Register Value				
PIC1AREG51				
10	8	6	5	TS0OPCI0
X	0	0	1	INTENCA0I1
0	X	1	0	INTENCA1I1

Note: Do not set the values other than the settings listed above for this function.

(4) Function

Detail of the function is described using the two-phase encoder control function (method 1) at up count (normal rotation) as an example.

The following figure shows the timing diagram.

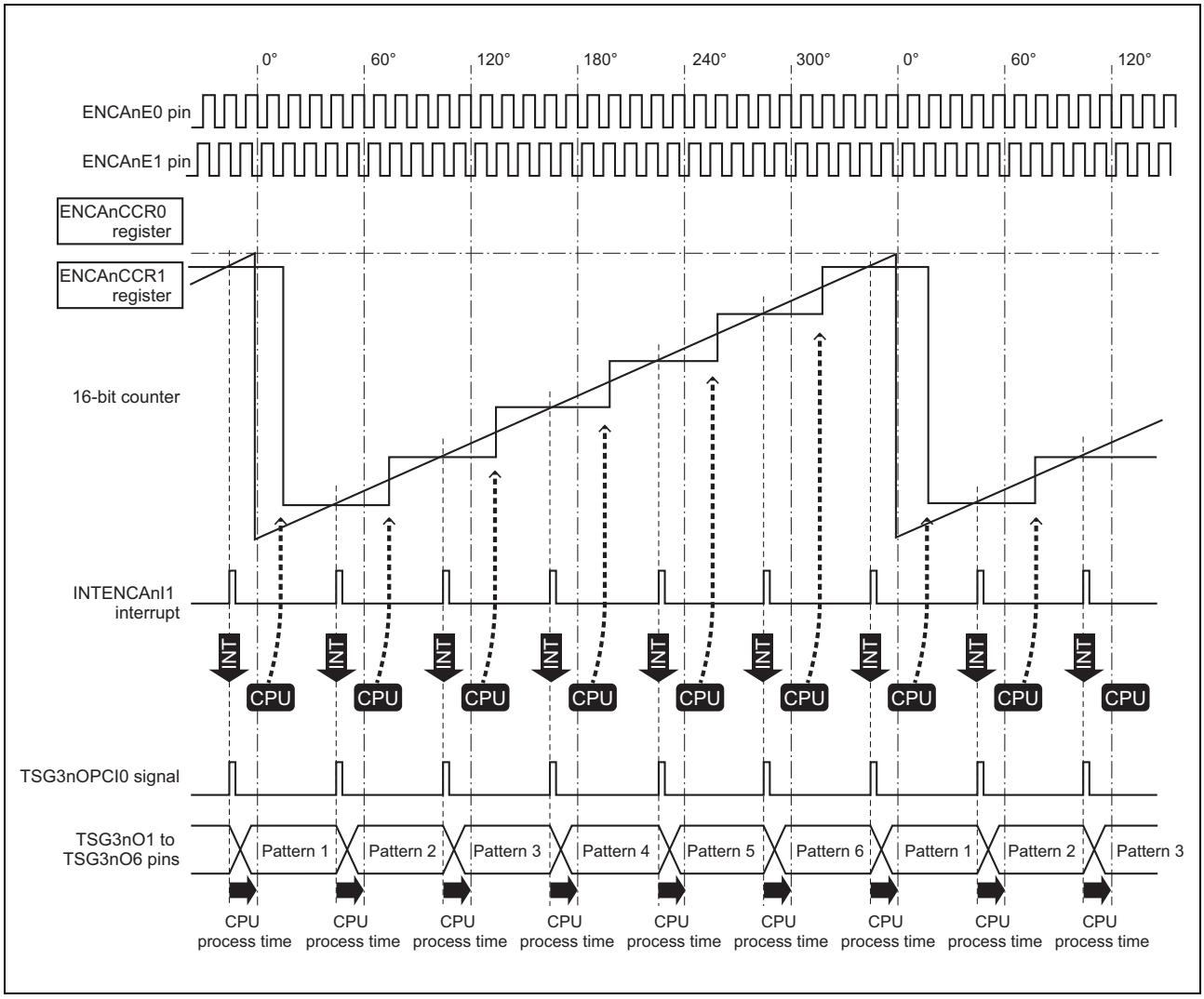


Figure 23.27 Timing Diagram of Two-Phase Encoder Control Function (Control Method 1) at Up Count (Normal Rotation)

(1) A match of the value of the encoder counter and ENCAnCCR1 allows INTENCAnI1 to be generated, and the pattern is output from the TSG3nO1 to TSG3nO6 pins.

(2) CPU calculates the next timing to switch output patterns by an interrupt processing and set the value to ENCAnCCR1.

(3) With a match of the value of the encoder counter and ENCAnCCR0 the encoder counter is cleared.

CAUTION

It is necessary to set ENCAnCCR1 at each pattern switch (each INTENCAnI1 interrupt). It is necessary to match the initial output pattern of timer TSG3n to the set value of ENCAnCCR1 before start.

Switching between normal and reverse rotations of output patterns should be set with the TSG3nPSC bit of the TSG3nOPT0 register.

The following figure shows the timing diagram.

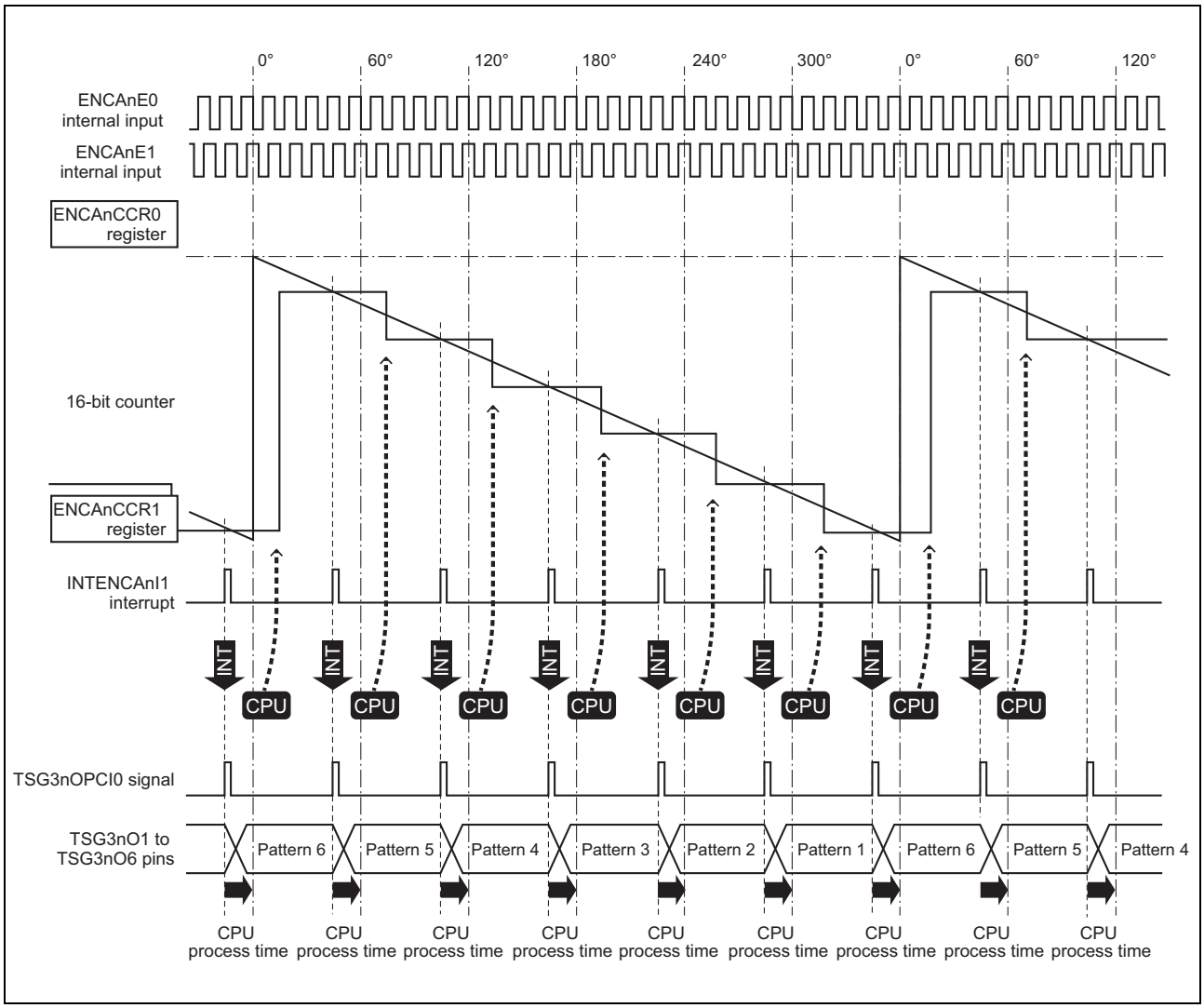
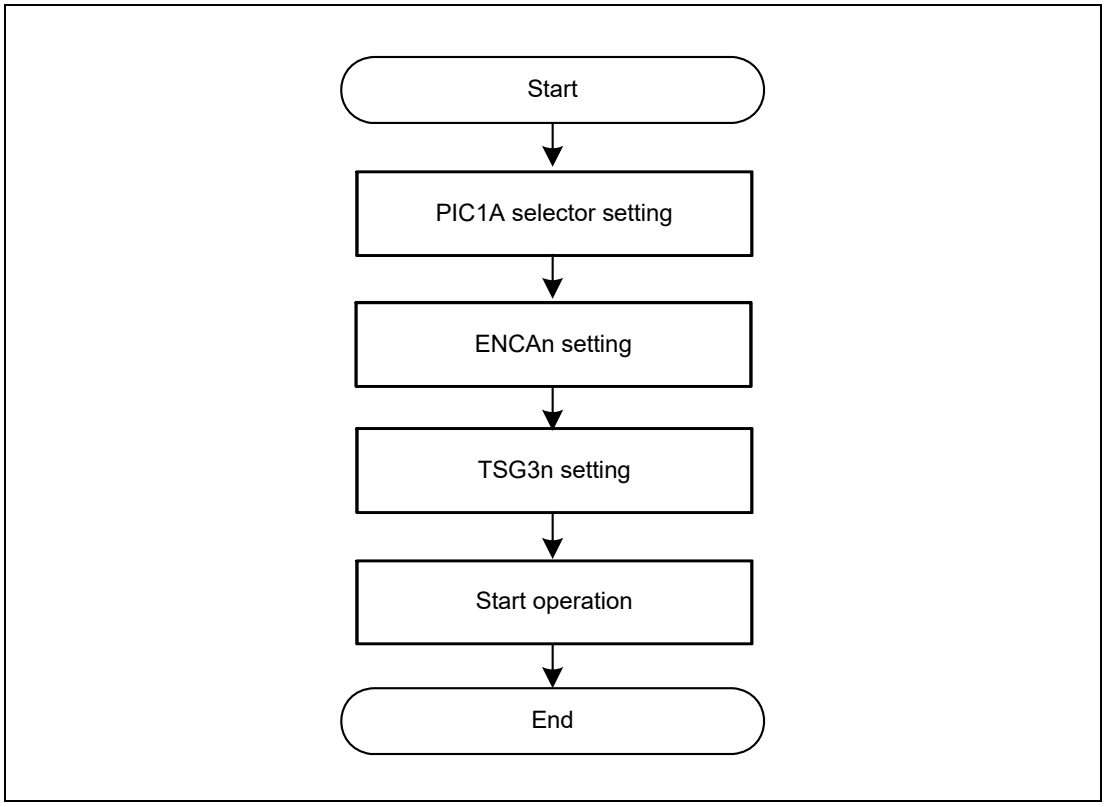


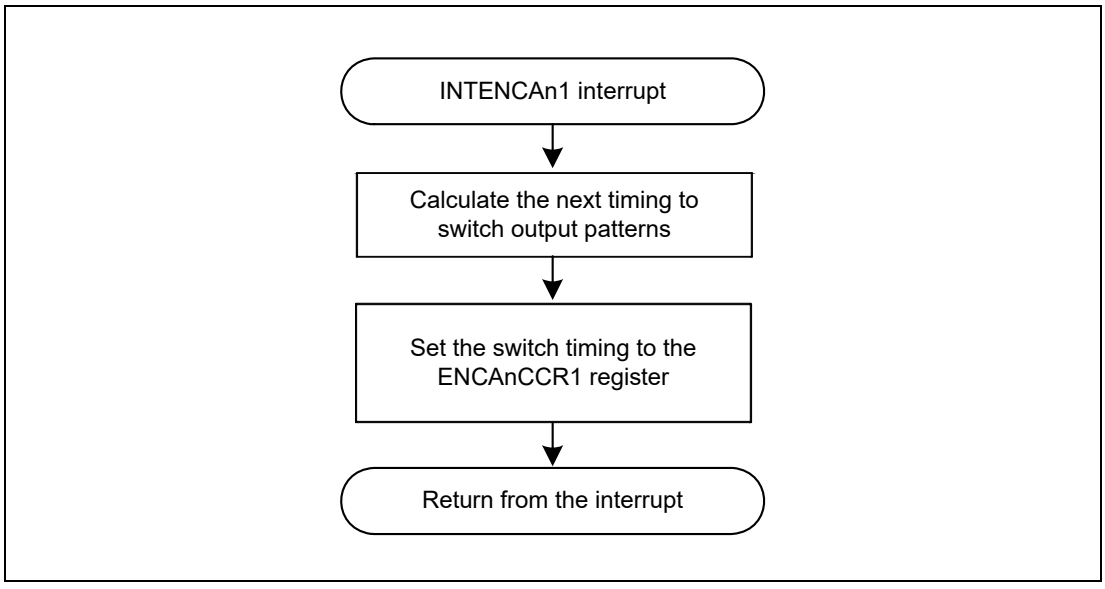
Figure 23.28 Timing Diagram of Two-Phase Encoder Control Function (Control Method 1) at Down Count (Reverse Rotation)

(5) Flow Chart

The following figure shows the setting flow of this function.



The following figure shows the flow chart after interrupt processing.



The values of ENCAn registers used in this function are as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of ENCA

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of TSG3.

23.2.3.8 Two-Phase Encoder Control Function (Control Method 2)

(1) Overview

This function allows switching the advance and retard control of the motor control function (TSG3n) output patterns in 120-DC mode using the two-phase encoder control function (ENCAn).

(2) Configuration

The similar configuration as the **Section 23.2.3.7, Two-Phase Encoder Control Function (Control Method 1)** is applied this function. See **(2) Configuration** of **Section 23.2.3.7, Two-Phase Encoder Control Function (Control Method 1)**.

(3) Registers

The similar configuration as the **Section 23.2.3.7, Two-Phase Encoder Control Function (Control Method 1)** is applied this function. See **(3) Registers** of **Section 23.2.3.7, Two-Phase Encoder Control Function (Control Method 1)**.

(4) Function

Detail of the function is described using the two-phase encoder control function (method 2) at advance control (normal rotation) as an example.

The following figure shows the timing diagram.

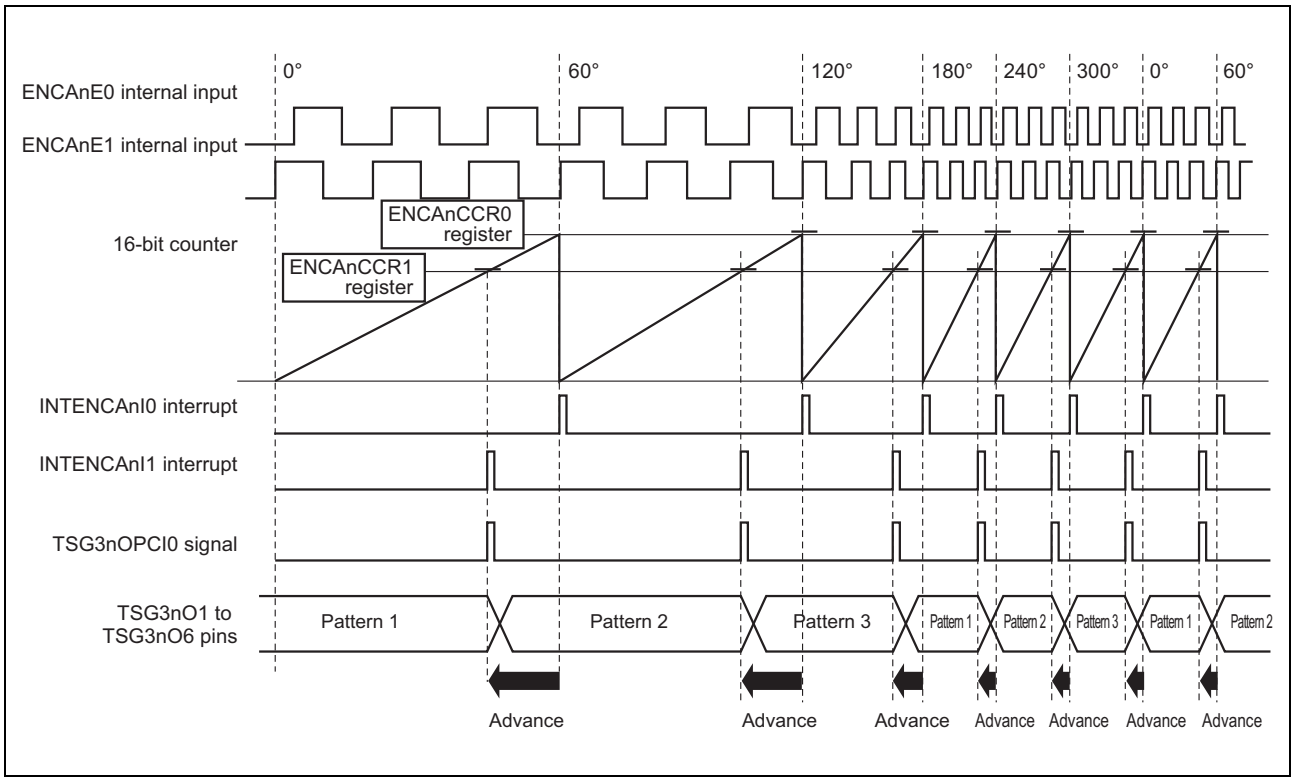


Figure 23.29 Timing Diagram of Two-Phase Encoder Control Function (Method 2) at Advance Control (Normal Rotation)

(1) A match of the value of the encoder counter and ENCAnCCR1 (corresponds to the output pattern switch position for TSG3n) allows INTENCAnI1 to be generated, and the pattern is output from the TSG3nO1 to 6 pins.

(2) With a match of the value of the encoder counter and ENCAnCCR0 (corresponds to the phase advance and retard of the switch position), INTENCAnI1 is generated and the encoder counter is cleared.

CAUTION

It is necessary to set ENCAnCCR1 at each pattern switch (each INTENCAnI1 interrupt).
It is necessary to match the initial output pattern of timer TSG3n to the set value of ENCAnCCR0 before start.

Switching between normal and reverse rotations of output patterns should be set with the TSG3nPSC bit of the TSG3nOPT0 register.

The following figure shows the timing diagram at retard control (normal rotation).

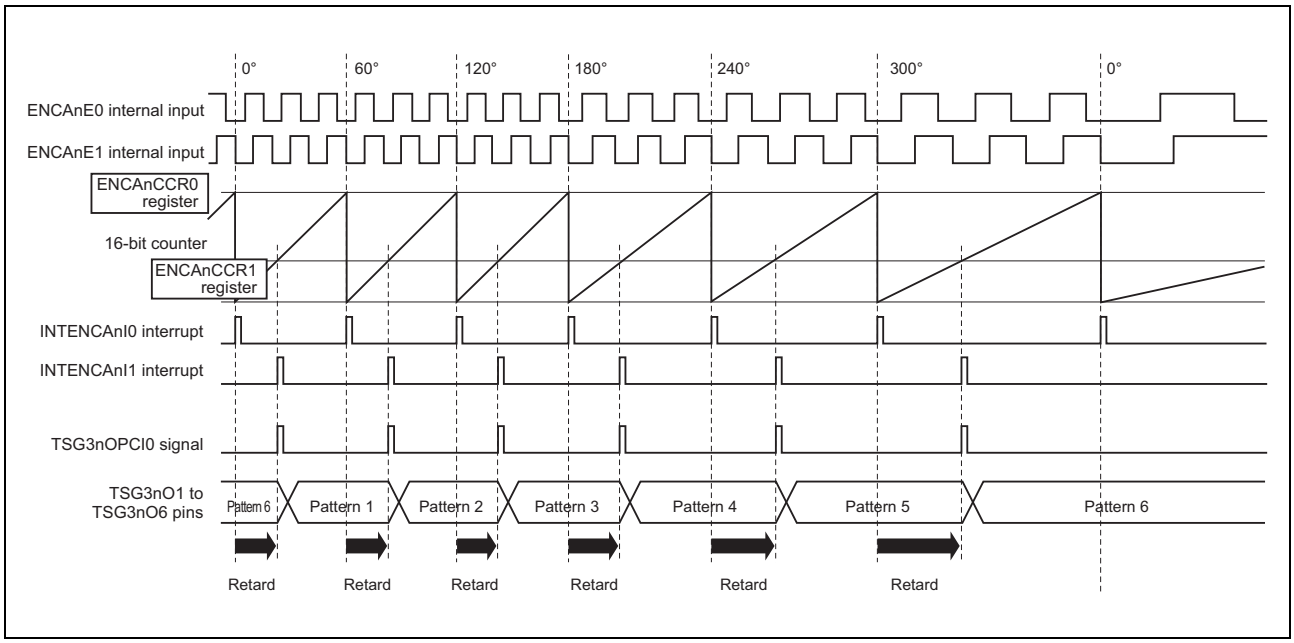


Figure 23.30 Timing Diagram of Two-Phase Encoder Control Function (Method 2) at Retard Control (Normal Rotation)

By setting greater value to ENCAnCCR1 than that of ENCAnCCR0, TSG3n output pattern phase can be retarded.

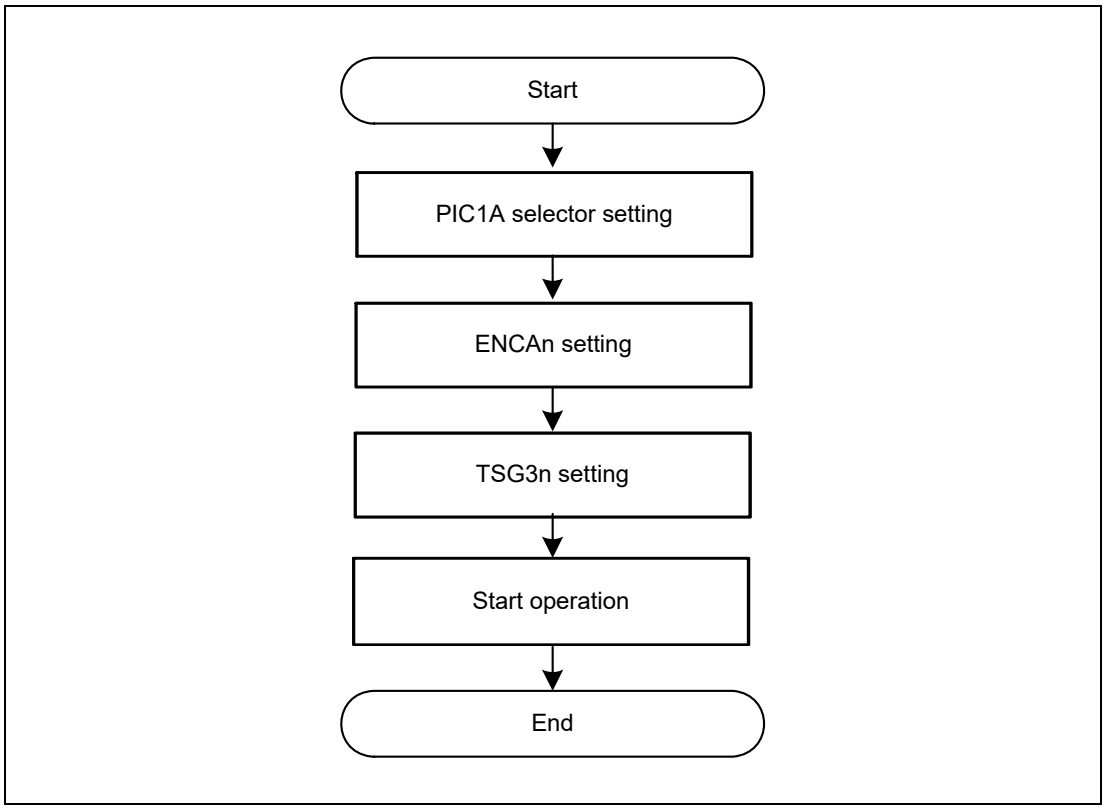
NOTE

This function can be used for advance control and retard control both at up count and down count.

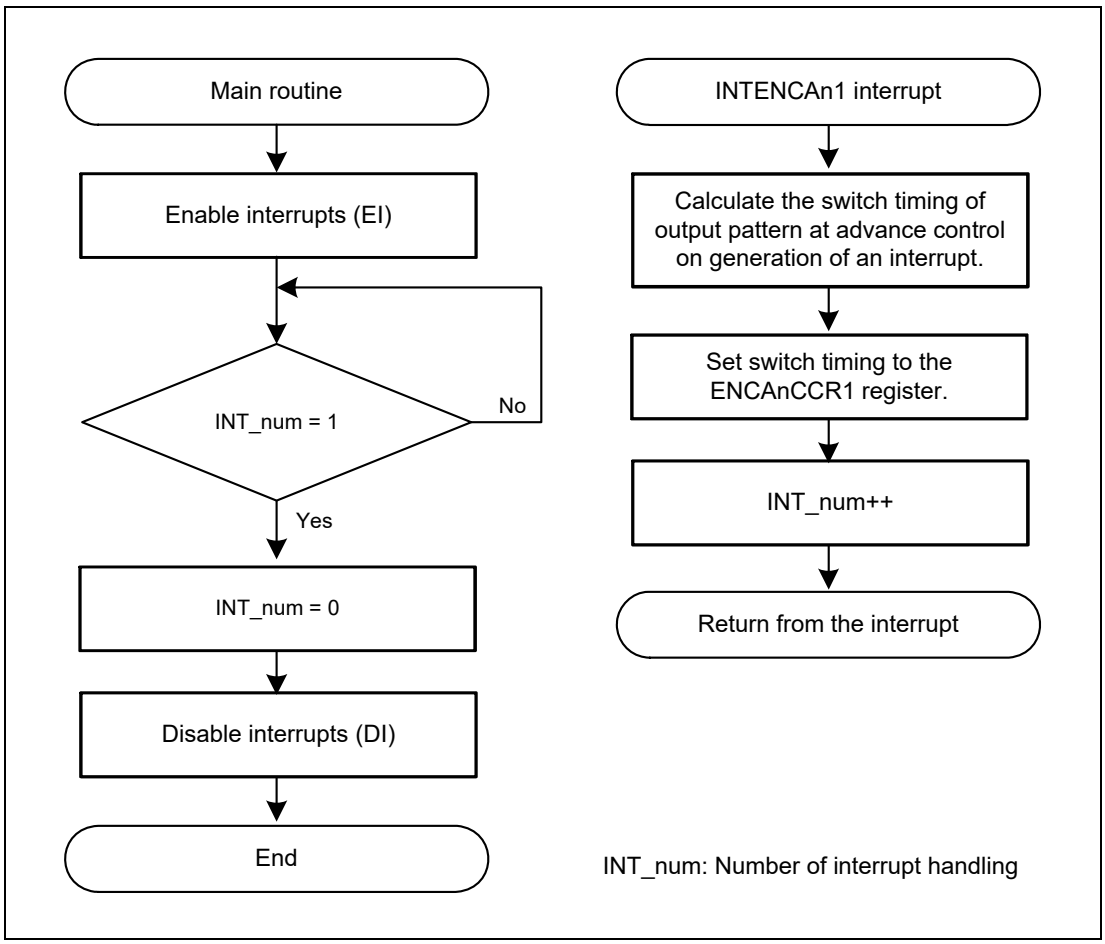
(5) Flow Chart

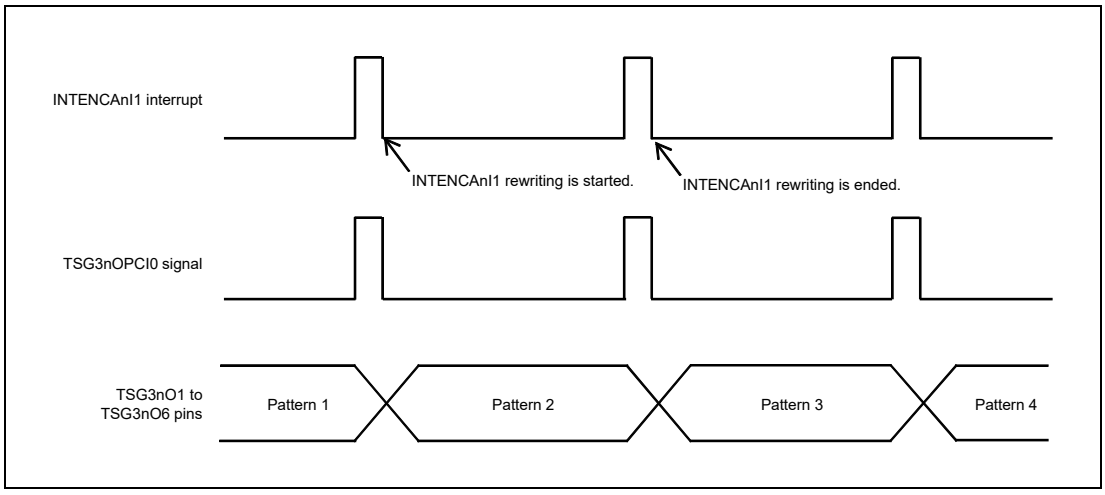
The flow charts for this function are shown as follows.

Flow chart of main routine

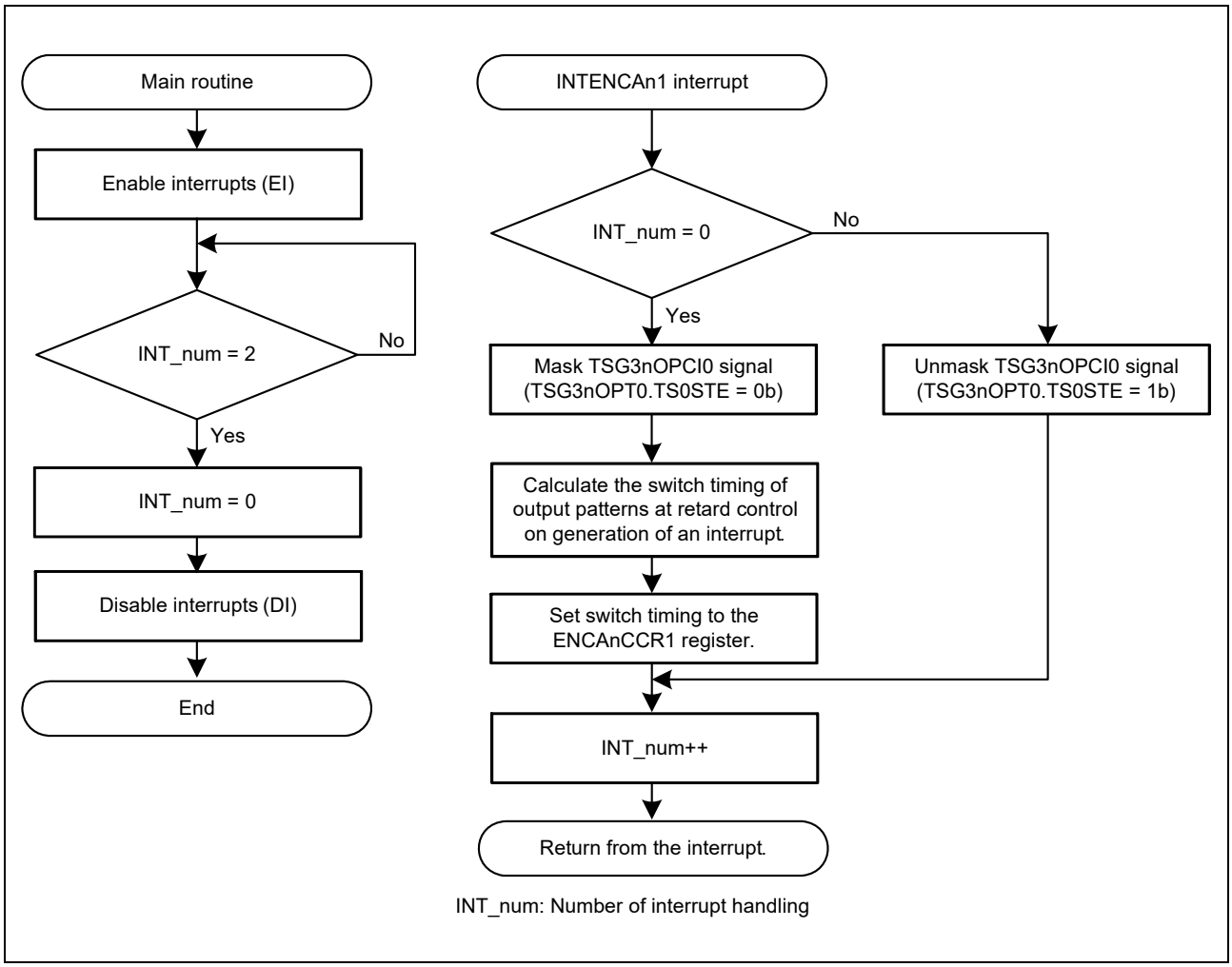


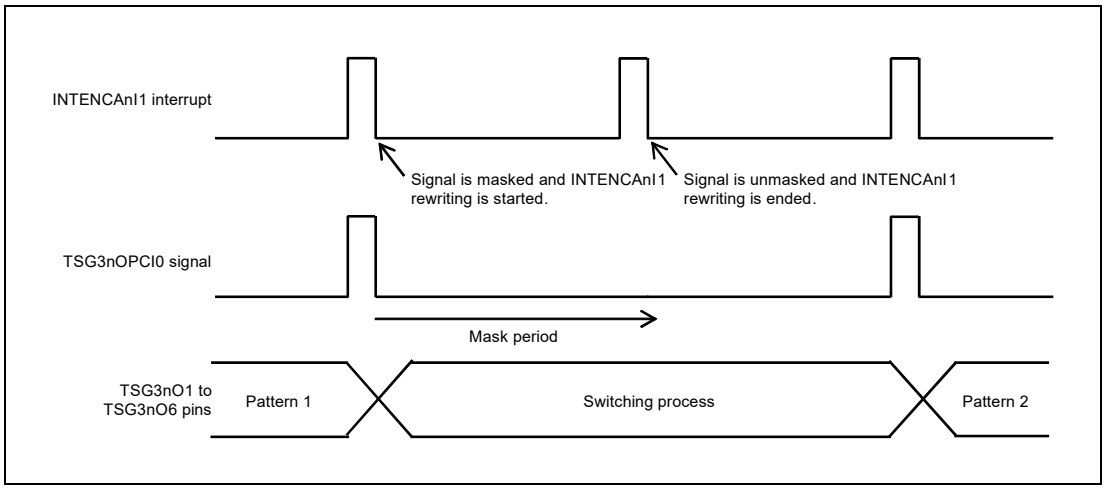
Flow chart of ENCAnCCR1 rewrite processing at advance control





Flow chart of ENCAnCCR1 rewrite processing at retard control





The register settings for ENCAn to use this function are described as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of ENCA

The register settings for TSG3n to use this function are described as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of TSG3.

23.2.3.9 Two-Phase Encoder Control Function (Control Method 3)

(1) Overview

Using the two-phase encoder control function (ENCAn), variation of the desired angle and phase (up to ±60°) by pattern output control is available regarding the angle of motor rotation indicated by the ENCA0 and ENCA1 timers.

(2) Configuration

Variation of desired angle and phase control by output pattern in 120-DC mode is realized by using ENCAn and TSG3n, and PIC1A in combination.

The following figure describes the block diagram of two-phase encoder control function (control method 3).

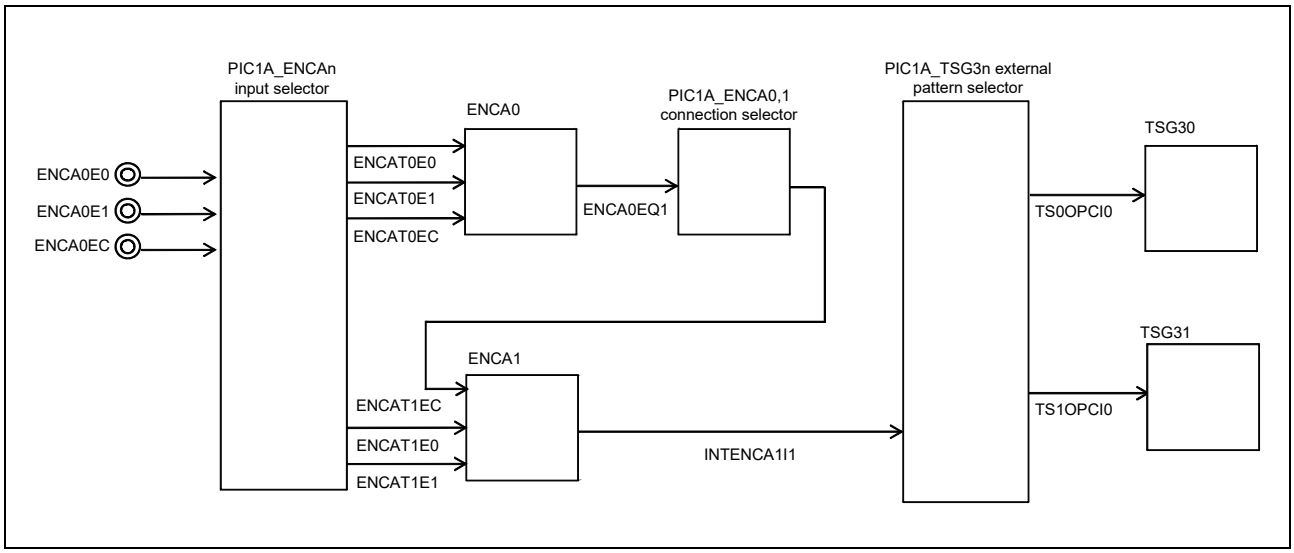


Figure 23.31 Block Diagram of Two-Phase Encoder Control Function (Control Method 3)

The configuration of this function is described as follows.

- [PIC1A_ENCAn input selector]
  ENCA0E0, ENCA0E1, and ENCA0EC pin inputs are selected and output to ENCATnE0, ENCATnE1, and ENCATnEC.
- [ENCA0]
  ENCA0EQ1 is output by two-phase encoder processing.
- [PIC1A_ENCA0, 1 connection selector]
  ENCA0EQ1 is selected and output to ENCAT1EC.
- [ENCA1]
  INTENCA1I1 is output by two-phase encoder processing. If an active level signal is input to ENCAT1EC, the timer count value is cleared.
- [PIC1A_TSG3n external pattern selector]
  INTENCA1I1 is selected and output to TSG3nOPCI0.
- [TSG3n]
  Output patterns in 120-DC mode are switched by TSG3nOPCI0.

(3) Registers

Block diagram of PIC1A is shown in the following figure.

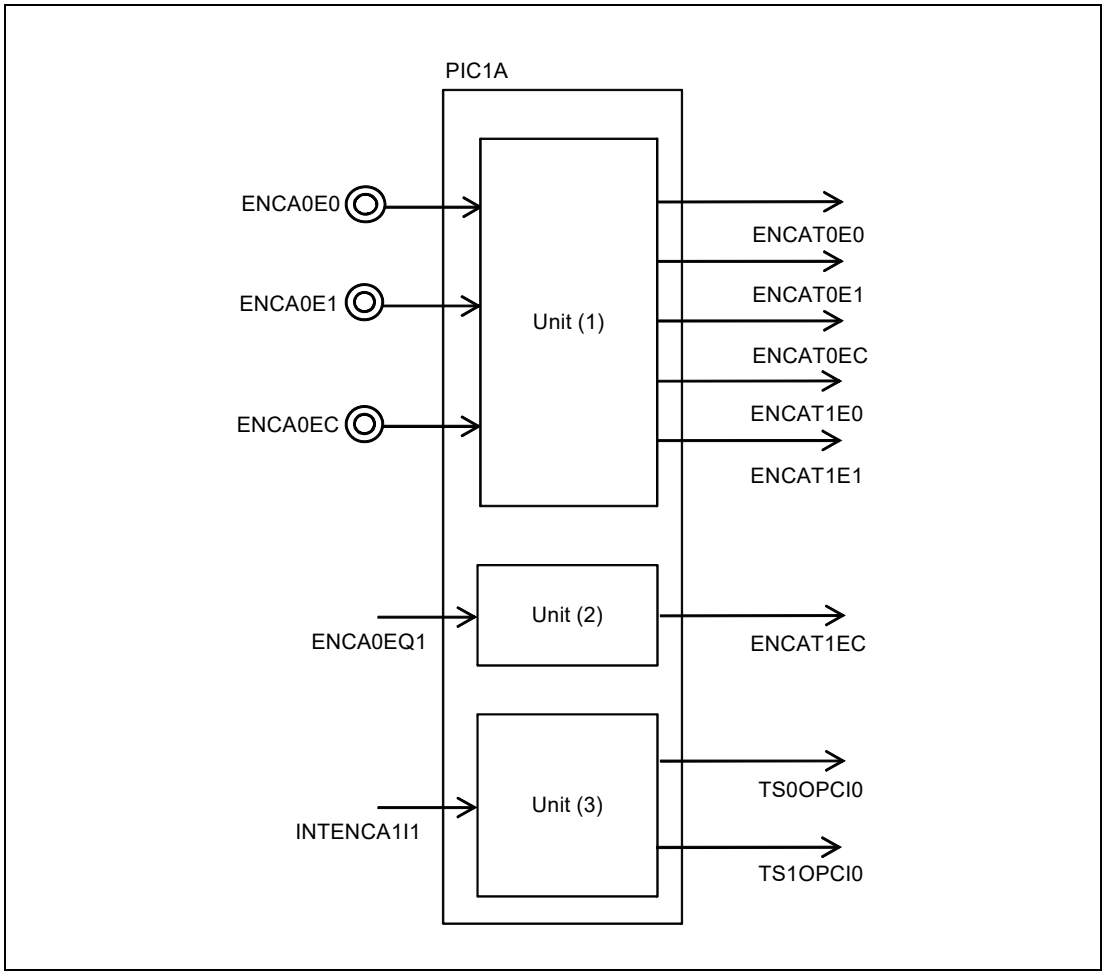


Figure 23.32 Block Diagram of PIC1A

The register settings for PIC1A to use this function are described as follows.

Unit (1): PIC1A_ENCAn input selector

PIC1AREG30[22] = 0_B

PIC1AREG30[17:16] = 00_B

PIC1AREG30[9:6] = 0101_B

PIC1AREG30[1:0] = 00_B

Unit (2): PIC1A_ENCA0, 1 connection selector

PIC1AREG30[11:10] = 11_B

Unit (3): PIC1A_TSG3n external pattern selector

PIC1AREG5n[10] = 0_B

PIC1AREG5n[6:5] = 10_B

(n = 0 when TSG30 is selected and n = 1 when TSG31 is selected.)

(4) Function

Detail of the function is described using the two-phase encoder control function (control method 3) at advance control (normal rotation) as an example.

The following figure shows the timing diagram.

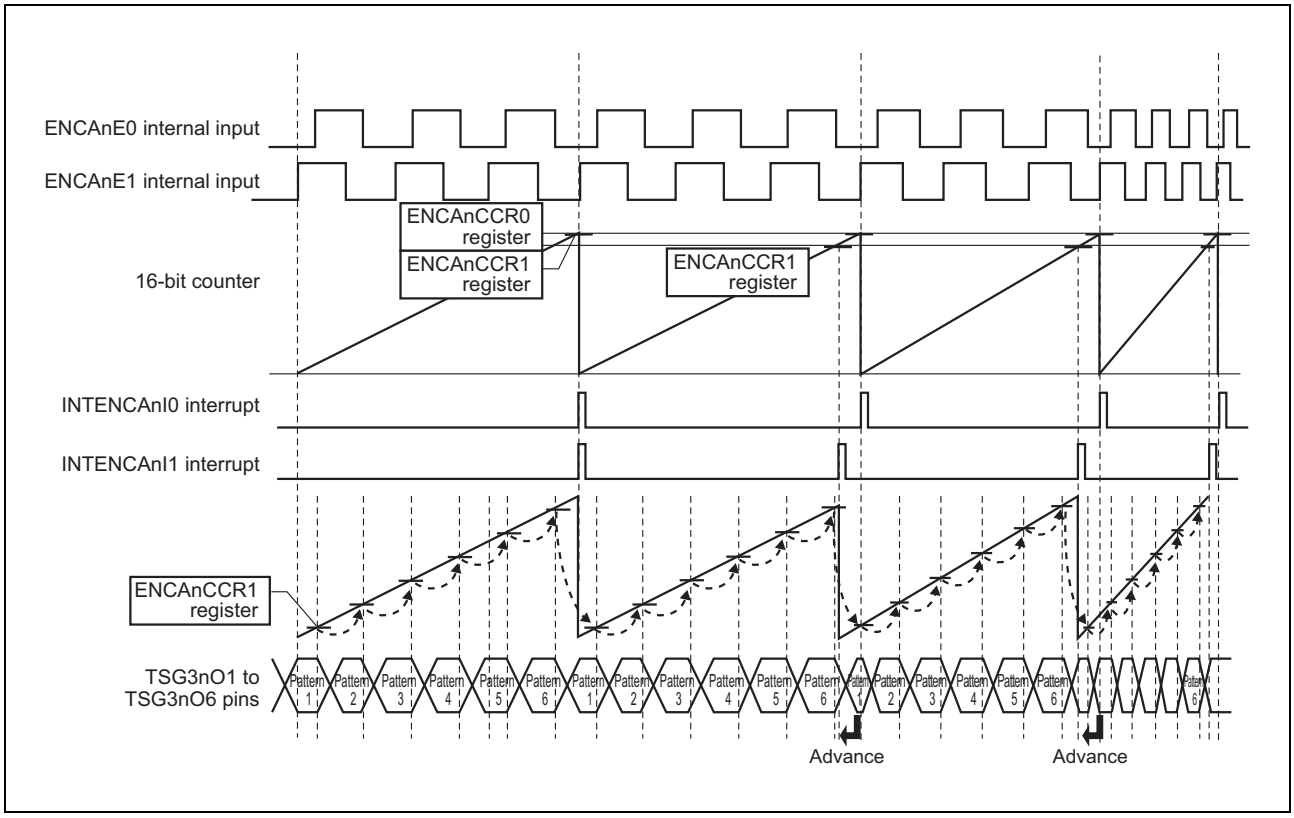


Figure 23.33 Timing Diagram of Two-Phase Encoder Control Function (Control Method 3) at Advance Control (Normal Rotation)

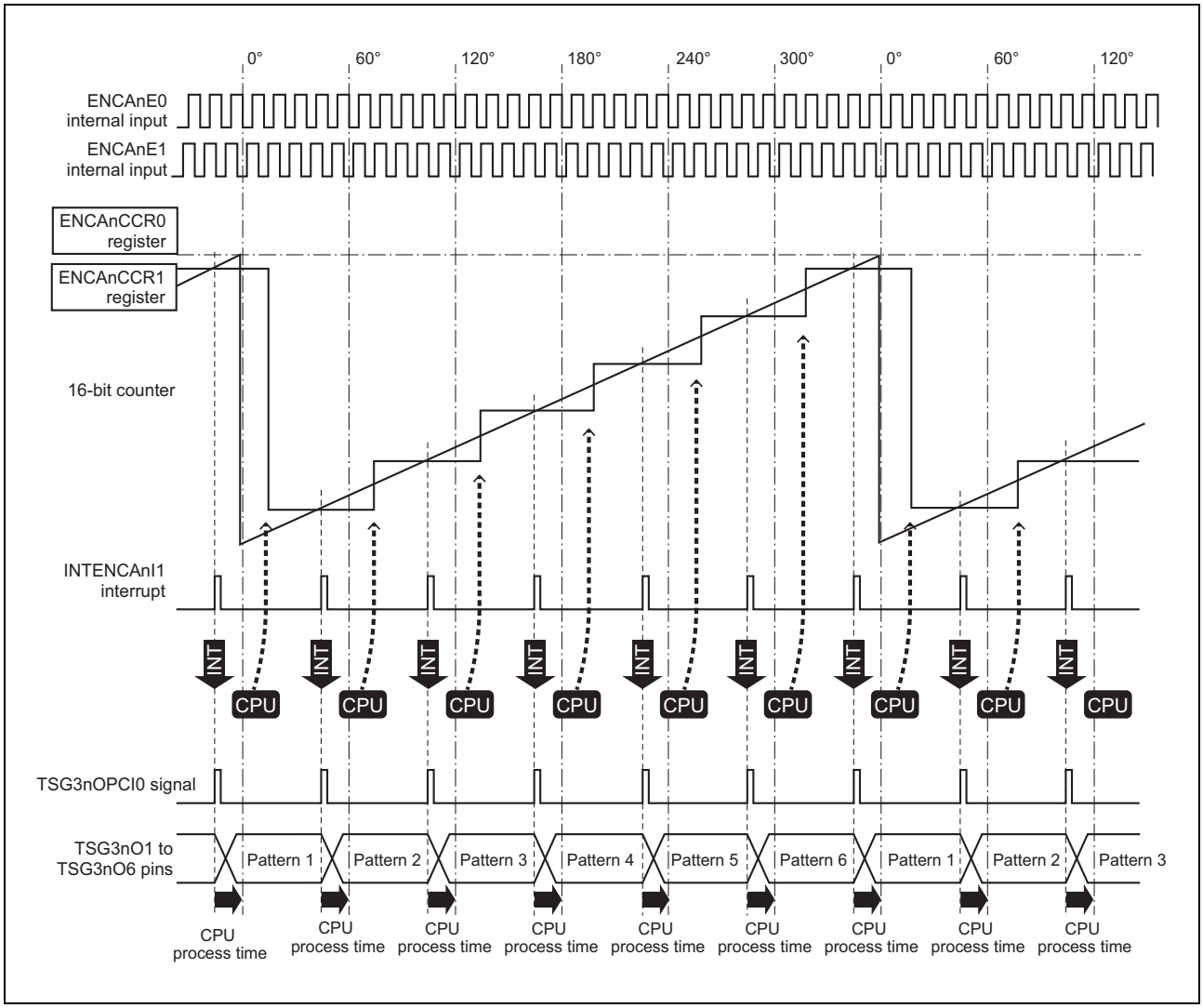


Figure 23.34 Timing Diagram of TSG3n Output Pattern Control (ENCA1) (Normal Rotation)

(1) A match of the value of the ENCA0 encoder counter and ENCA0CCR1 allows INTENCA0I1 to be generated and ENCA1 encoder counter is cleared.

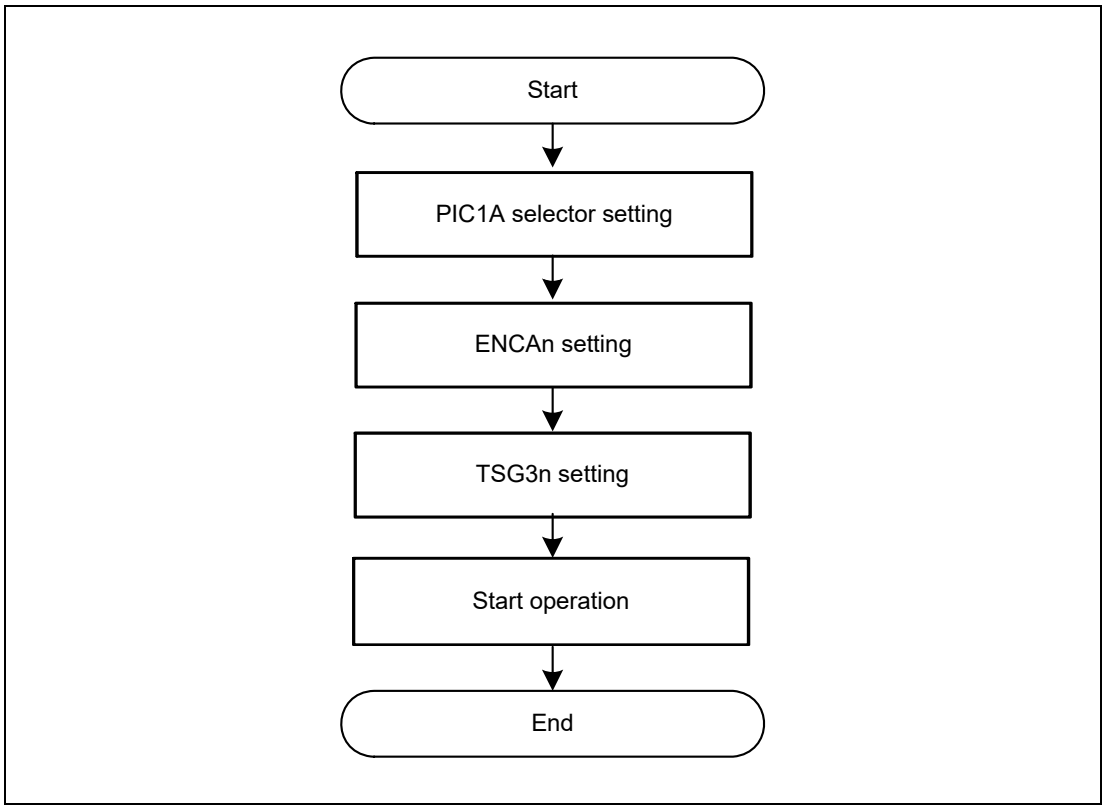
(2) A match of the value of the ENCA1 encoder counter and ENCA1CCR1 allows INTENCA1I1 to be generated, and the pattern is output from the TSG3nO1 to 6 pins.

(3) With a match of the value of the ENCA0 encoder counter and ENCA0CCR0, INTENCA0I0 is generated and the encoder counter is cleared.

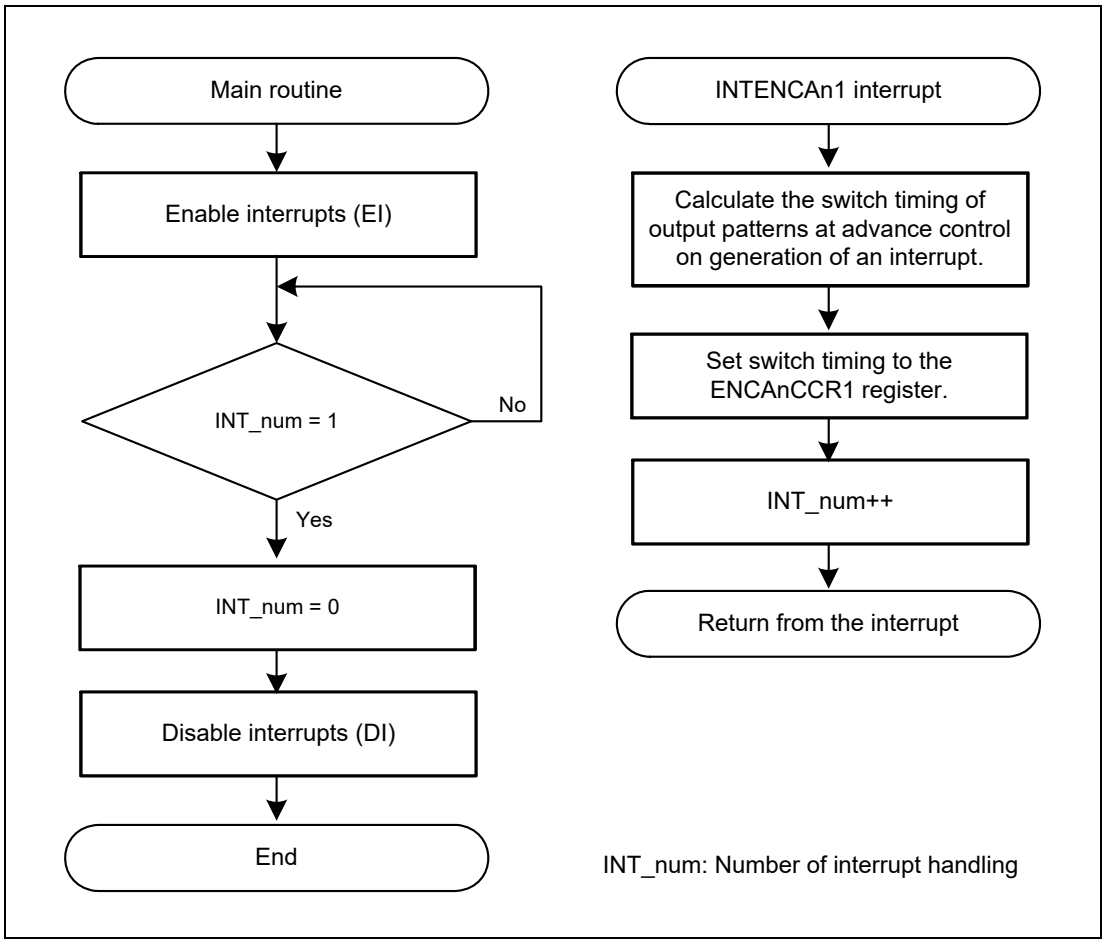
(5) Flow Chart

The flow charts for this function are shown as follows.

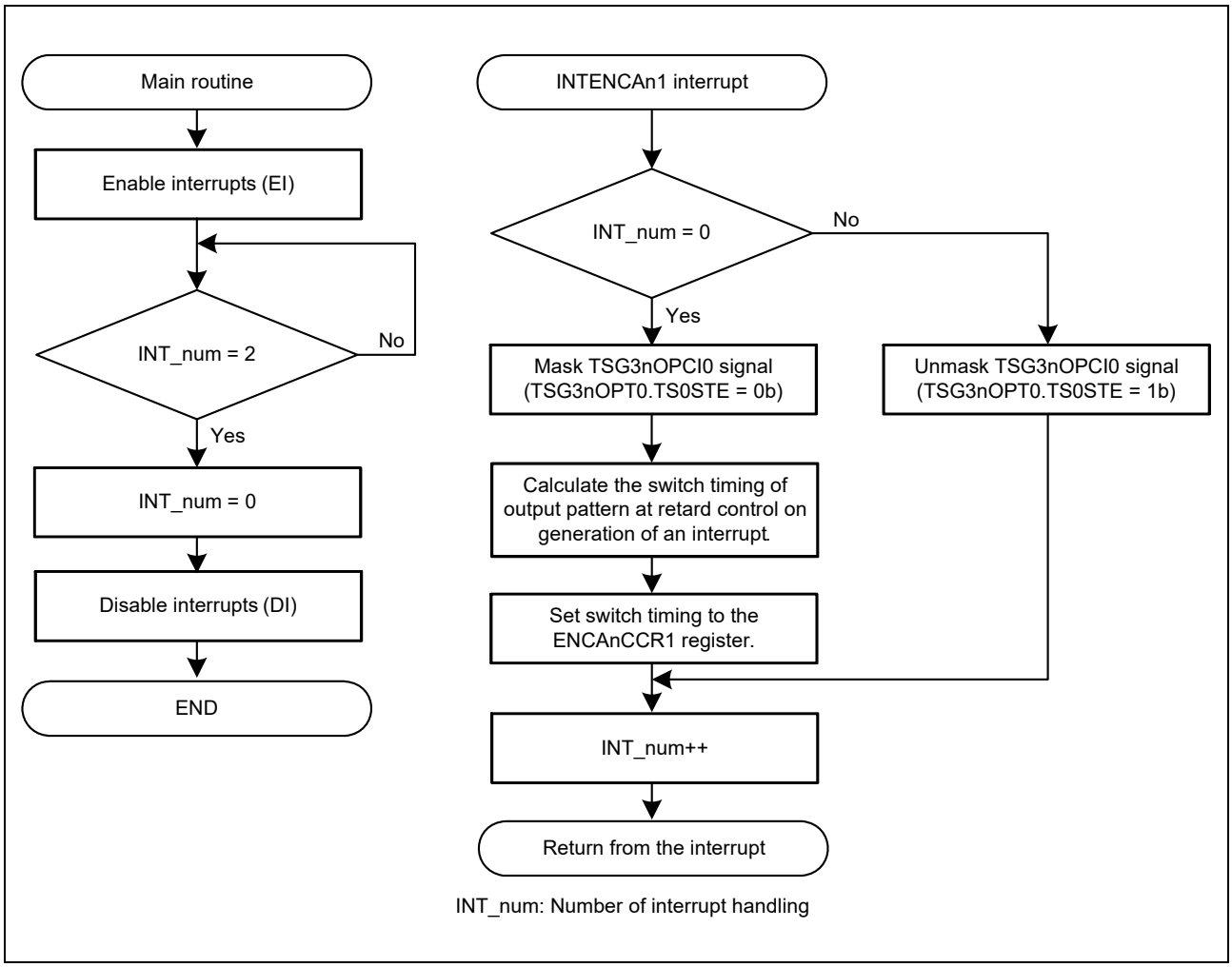
Flow chart of main routine



Flow chart of ENCAnCCR1 rewrite processing at advance control



Flow chart of ENCAnCCR1 rewrite processing at retard control



The register settings for ENCAn to use this function are described as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_01xx_B$

ENCAnIOC1[7:0] = 0000_00xx_B

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of ENCA.

The register settings for TSG3n to use this function are described as follows.

TSG3nCTL0[7:0] = $000x\_0011_B$

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see the section of TSG3.

23.2.3.10 Three-Phase Pulse Input Control Function

(1) Overview

This function allows variable phase control of TSG3n pattern output in 120-DC mode using TSG3n and TAUDn.

The following diagram shows the method of three-phase pulse input control.

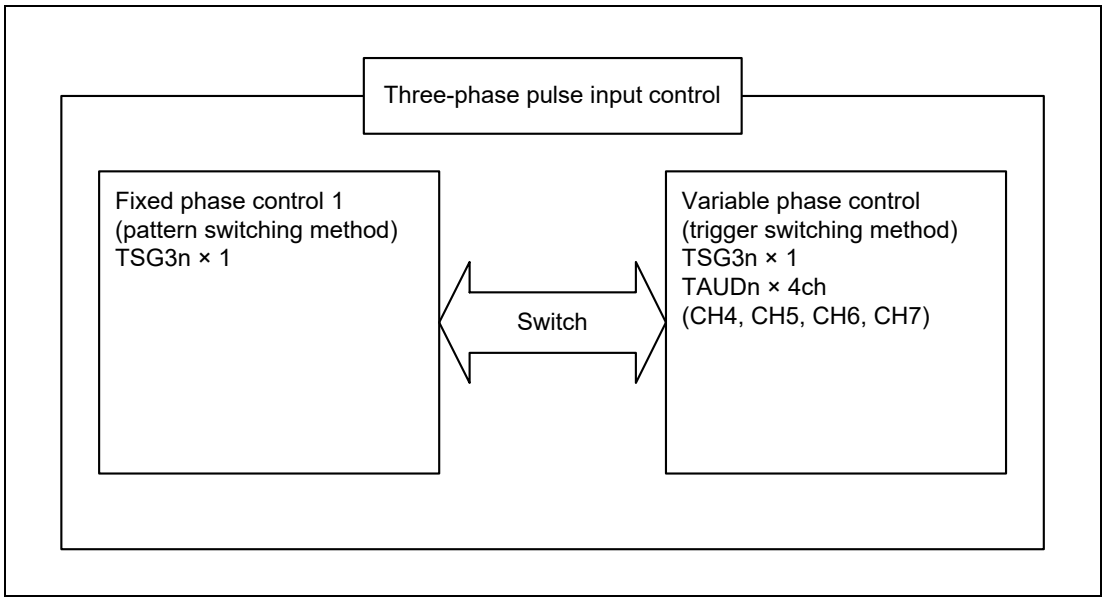


Figure 23.35 Control Method of Three-Phase Pulse Input Control Function

Control Method	Function
Fixed phase control 1 (pattern switching method)	Outputs a fixed pattern at constant rotation angle.
Variable phase control (trigger switching method)	Varies the phase by arbitrary angle (or time) up to ±60 degrees with reference to the rotation angle and outputs the pattern.

(2) Configuration

Three-phase pulse input control function is realized by using three-phase pulse input and TAUDn offset trigger mode, and PIC1A in combination.

The following figure shows the block diagram of three-phase pulse input control function.

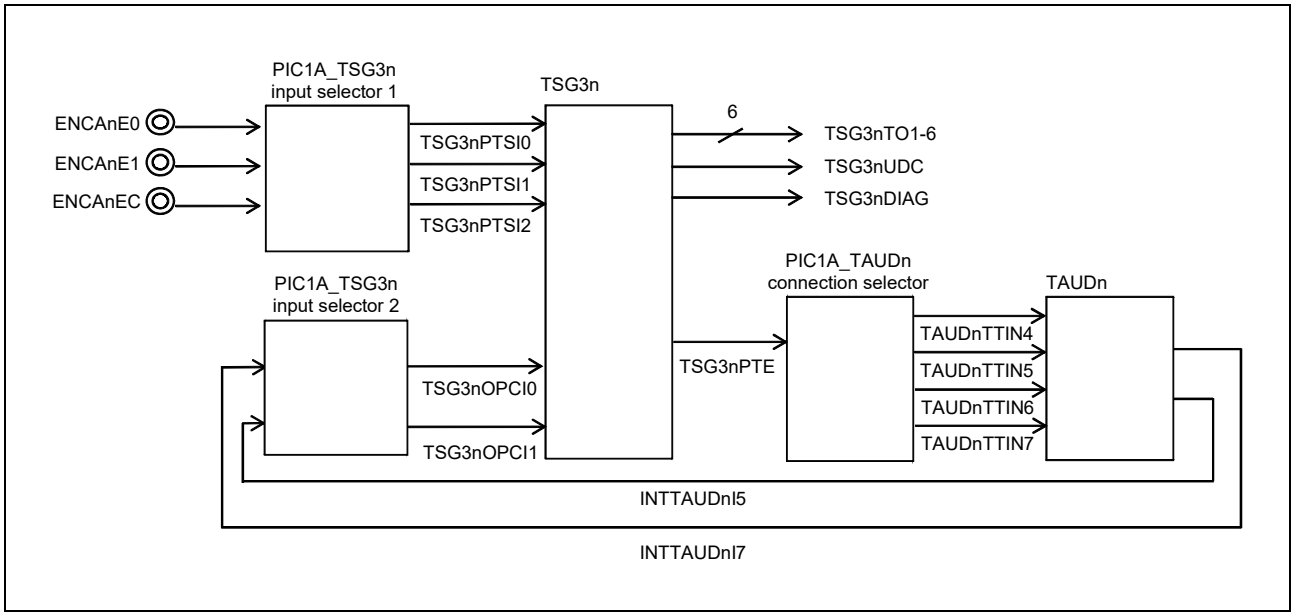


Figure 23.36 Block Diagram of Three-Phase Pulse Input Control

The configuration of this function is described as follows.

- [PIC1A_TSG3n input selector 1]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output to TSG3nPTSI0 to TSG3nPTSI02.
- [TSG3n]
  Patterns set in TSG3nTO1 to TSG3nTO6 are output in response to the input of TSG3nPTSI0 to TSG3nPTSI02 signals. TSG3nPTE is toggled each time the output patterns are switched.
- [PIC1A_TAUDn connection selector]
  TSG3nPTE input is selected and output to TAUDnTTIN4 to TAUDnTTIN7.
- [TAUDn]
  Interrupt signals INTTAUDnI5 and INTTAUDnI7 for output pattern phase generation are output with the offset trigger mode.
- [PIC1A_TSG3n input selector 2]
  INTTAUDnI5 and INTTAUDnI7 inputs are selected and output as TSG3nOPCI0 and TSG3nOPCI1.

(3) Registers

Block diagram of PIC1A is shown in the following figure.

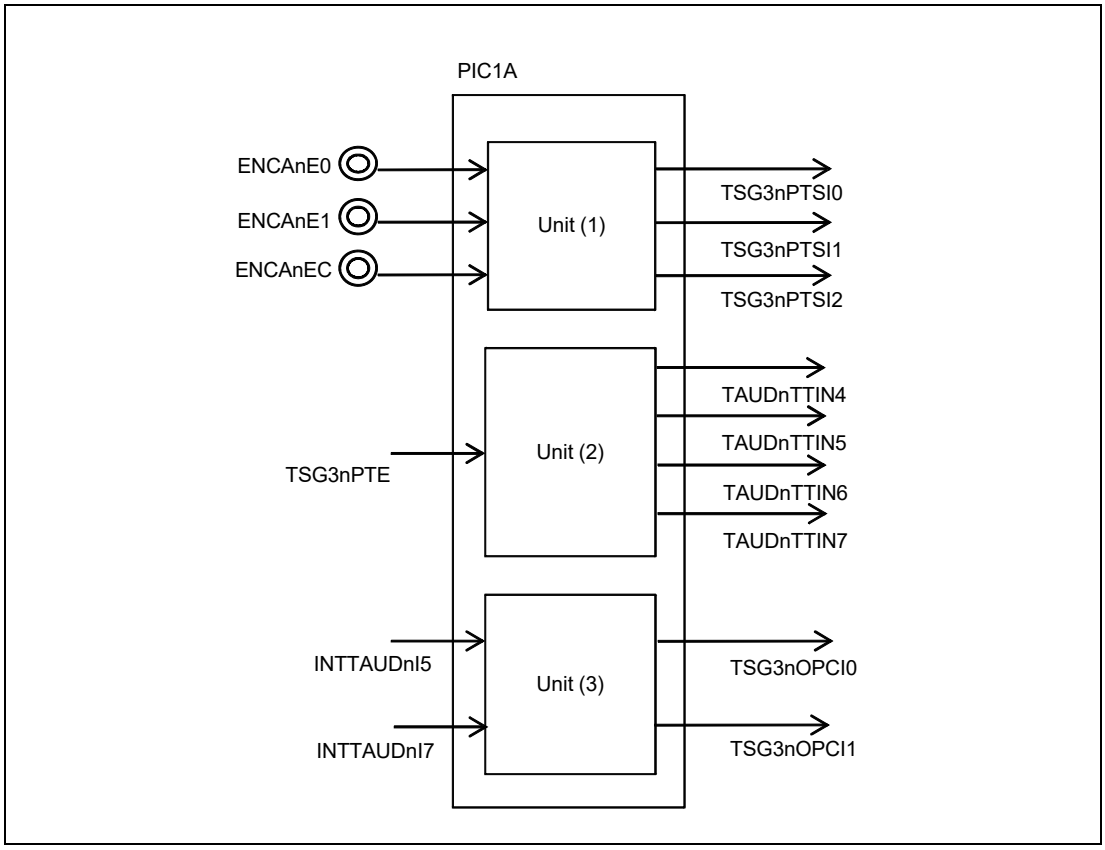


Figure 23.37 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

Unit (1): PIC1A_ TSG3n input selector 1

The values to output ENCA0E0, ENCA0E1, and ENCA0EC as TSG30PTSI0 to TSG30PTSI2

PIC1ATSGHALLSEL[0] = 1_B

PIC1AREG50[0] = 0_B

The values to output ENCA1E0, ENCA1E1, and ENCA1EC as TSG31PTSI0 to TSG31PTSI2

PIC1ATSGHALLSEL[1] = 1_B

PIC1AREG51[0] = 1_B

Unit (2): PIC1A_TAUDn connection selector

The values to output TSG3nPTE to TAUDnTTIN4 to TAUDnTTIN7.

PIC1AREG2n0[11:8] =1010_B

PIC1AREG2n0[3:0] = 1111_B

PIC1ATAUDnSEL[15:8] = 00_H

Unit (3): PIC1A_ENCAn input selector 2

The values to output INTTAUDnI5 and INTTAUDnI7 to TSG3nOPCI0 and TSG3nOPCI1.

PIC1AREG5n[7:5] = 011_B

(4) Function

Detail of the three-phase pulse input control function is described here.

The following figure shows the timing diagram.

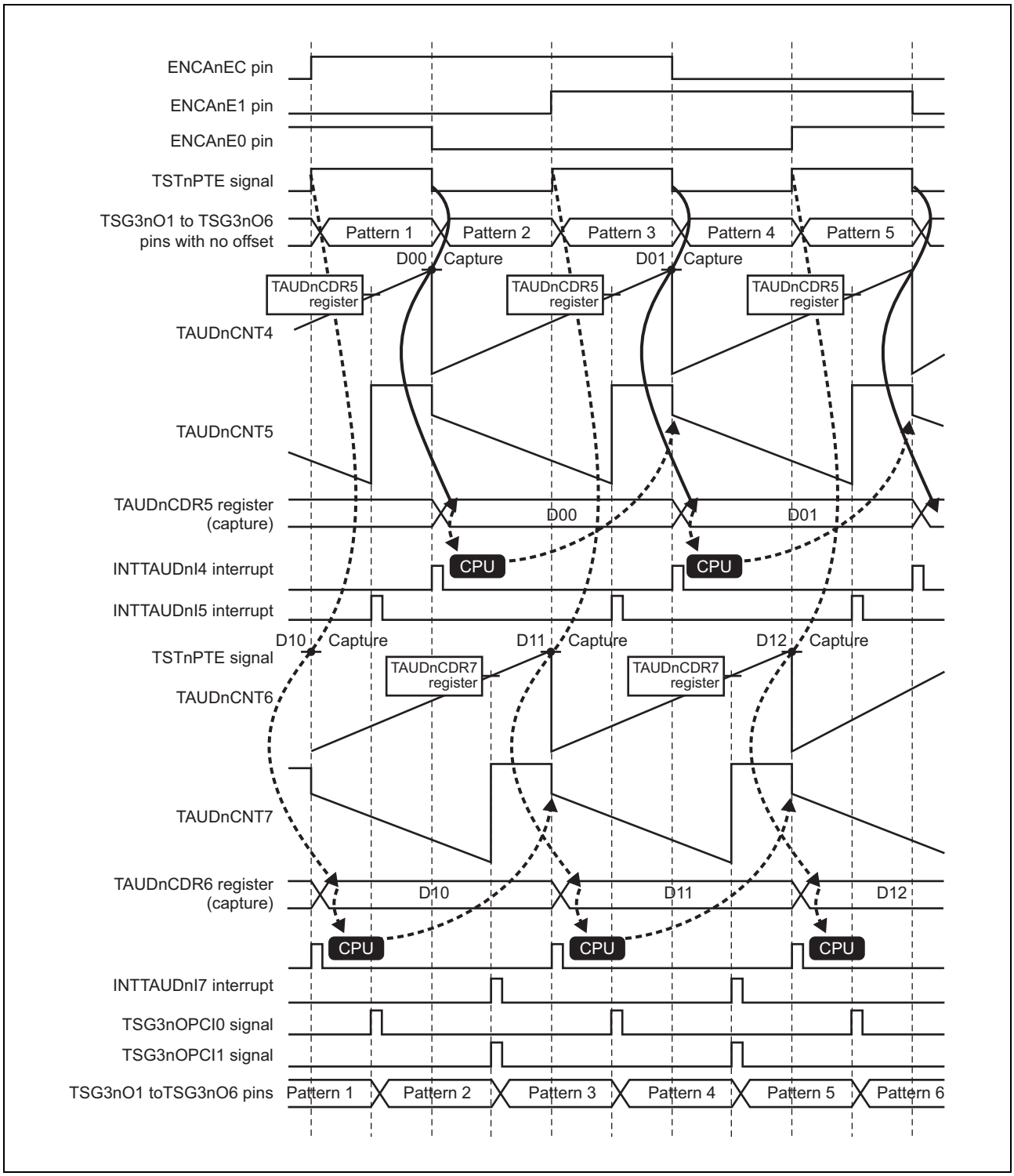


Figure 23.38 Three-Phase Pulse Input Control Function

(1) The output patterns are switched by the TSG pattern switching method on detection of ENCAn pin signal. The TSTnPTE signal toggles each time the patterns are switched.

(2) TAUDnCNT6 and TAUDnCNT4 are captured at the rising edge and falling edge of the TSTnPTE signal. The interval to switch patterns is calculated according to the captured value.

(3) CPU calculates the phase of the next output pattern and set the values to TAUDnCNT5 and TAUDnCNT7. The signals corresponding to the values are output as TSG3nOPCI0 and TSG3nOPCI1. At this time, the patterns delayed for the set phase are output by switching the output patterns by the trigger switching method.

The following table lists the relation between the value set to TAUDnCNTm and the captured value in TAUDnCDR (m-1) (m = 5, 7).

TAUDnCNTm Register Setting	TSG3n Pattern Output Switch Timing
TAUDnCNTm = 0000_H	The patterns switched on detection of an edge of TSTnPTE signal with a delay of up to one cycle of the clock signal being counted by TAUDn is generated.
TAUDnCNTm = captured value	The patterns are switched on detection of an edge of TSTnPTE signal.
TAUDnCNTm < captured value	The patterns are switched on the timing after the phase delayed from detection of an edge of TSTnPTE signal.
TAUDnCNTm > captured value	Setting prohibited

An example of switch operation from fixed phase control 1 to variable phase control

The following figure shows an example of switch operation from fixed phase control 1 to variable phase control.

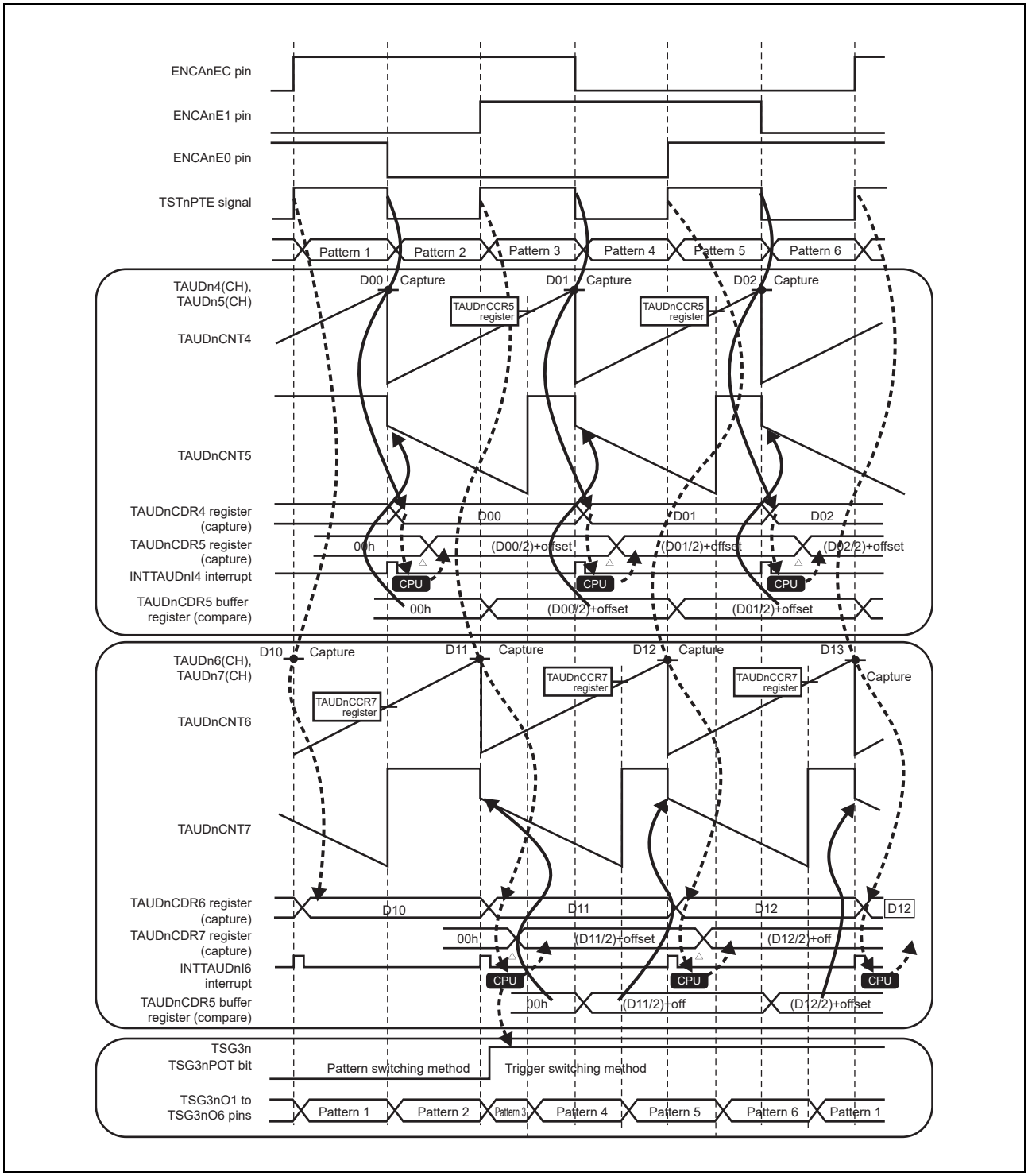


Figure 23.39 An Example of Switch Operation from Fixed Phase Control 1 to Variable Phase Control

The output pattern is changed to the trigger switching method by changing the TSG3nPOT bit from low level to high level, and the variable phase control is enabled.

An example of switch operation from variable phase control to fixed phase control 1

The following figure shows an example of switch operation from variable phase control to fixed phase control 1.

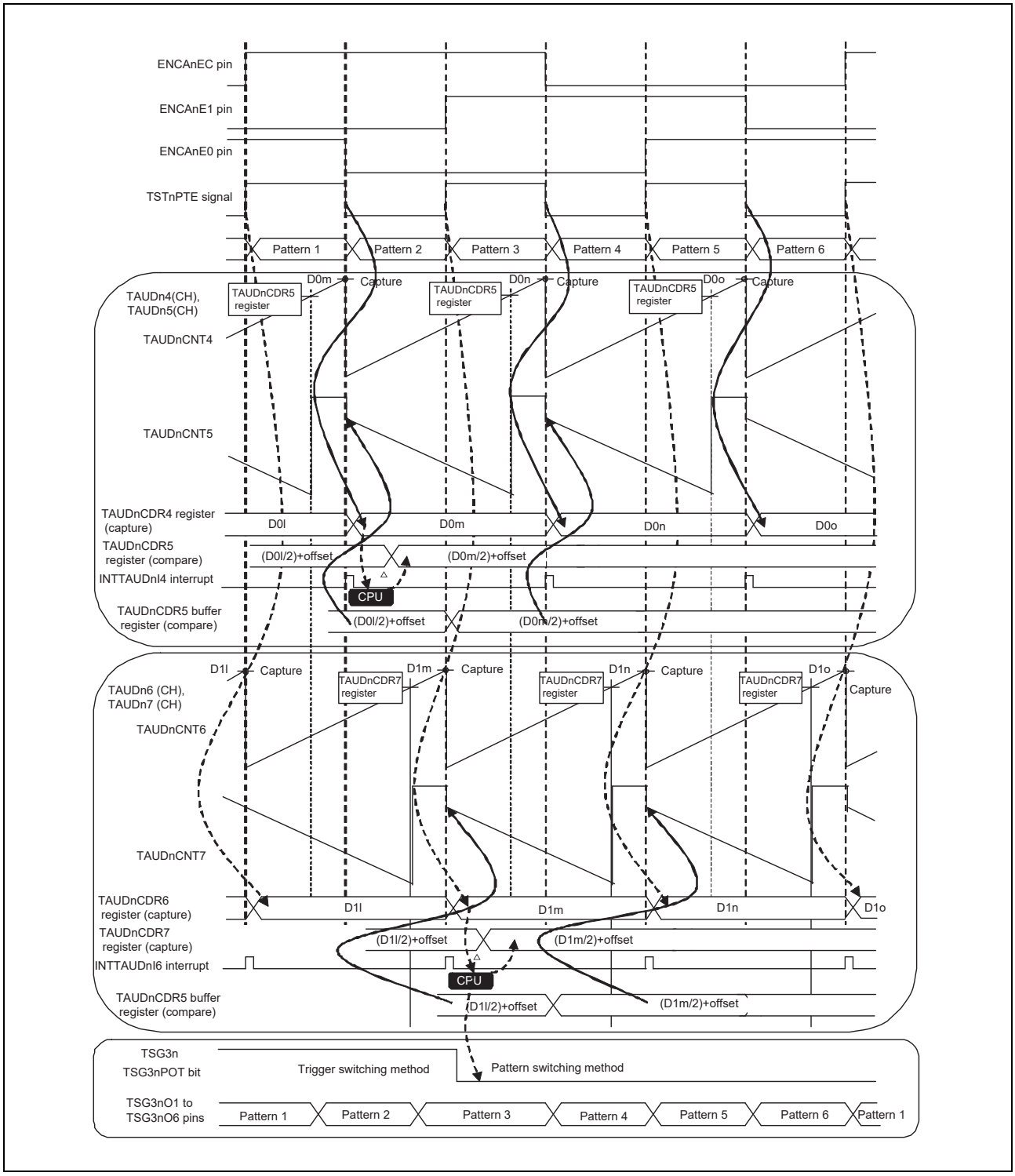


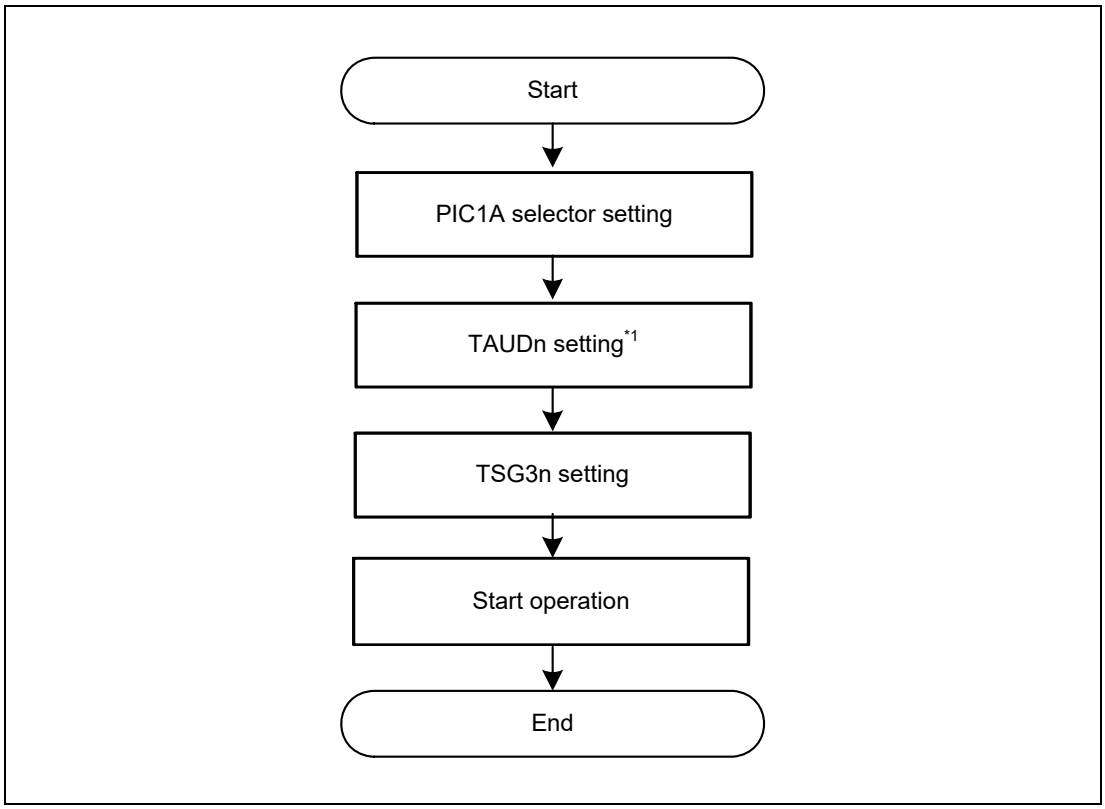
Figure 23.40 An Example of Switch Operation from Variable Phase Control to Fixed Phase Control 1

The output pattern is changed to the pattern switching method by changing the TSG3nPOT from high level to low level, and the fixed phase control 1 is enabled.

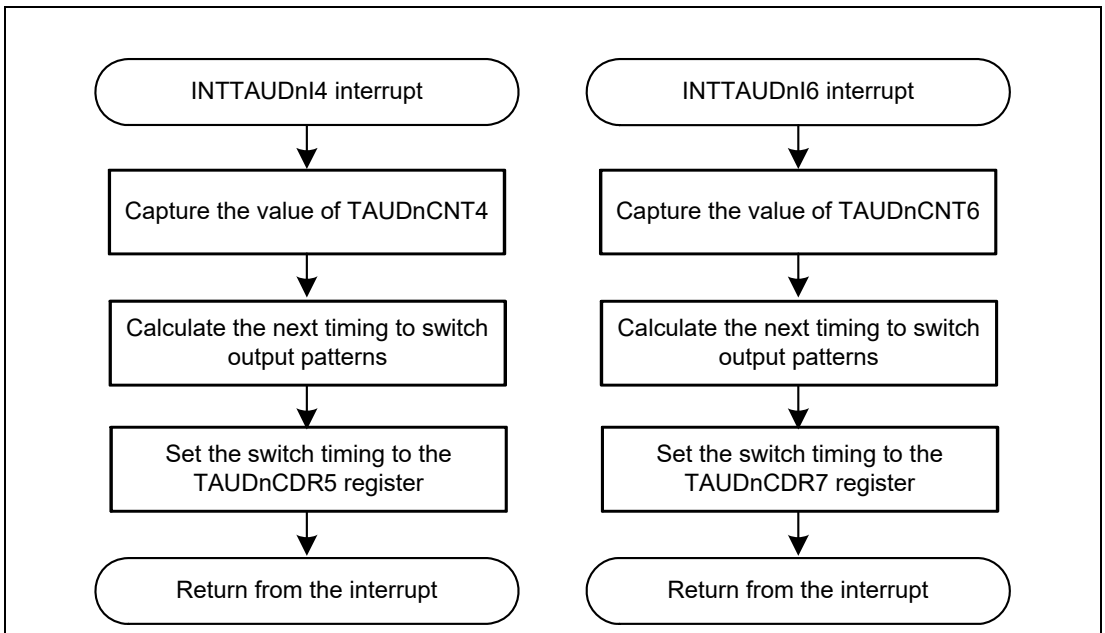
(5) Flow Chart

The flow charts for this function are shown as follows.

Flow chart of main routine



Flow chart of rewriting of TAUDnCDR5 and TAUDnCDR7 during operation



Note: For the TAUDn settings, see **Section 17.4.12.5, Offset Trigger Output Function**.

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = 0000_0011_B

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_0001\_xxx0\_0000_B$

TSG3nIOC0[7:0] = 0111_1110_B

TSG3nIOC1[7:0] = 0001_xxxx_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = 0011_1xx0_B

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

“x” can be set arbitrarily. For the registers specifications, see the section of TSG3.

23.2.3.11 Three-Phase Encoder Function

(1) Overview

The function allows three-phase external pattern inputs (TSG3nPTSI0 to TSG3nPTSI2) using ENCAn.

(2) Configuration

The three-phase encoder control function is realized by using TSG3n, ENCAn, and PIC1A in combination. The following figure shows the block diagram of three-phase encoder control function.

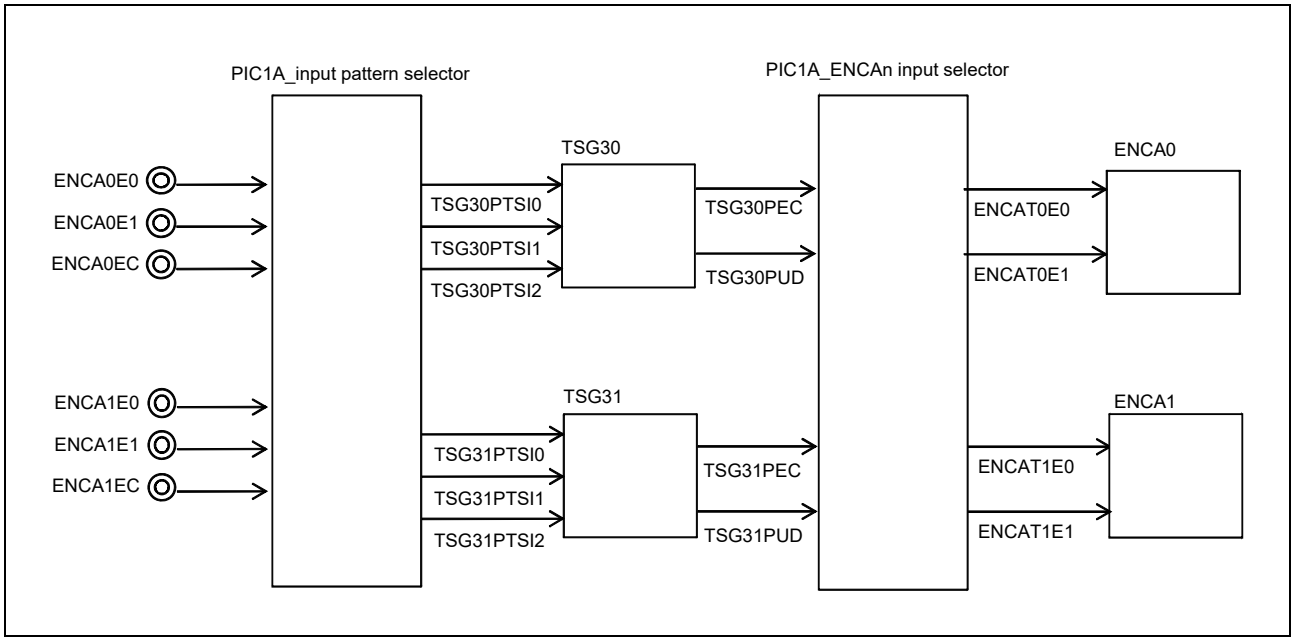


Figure 23.41 Block Diagram of Three-Phase Encoder Control Function

The configuration of this function is described as follows.

- [PIC1A_input pattern selector]
  ENCAnE0, ENCAnE1, and ENCAnEC pin inputs are selected and output to TSG3nPTSI0 to TSG3nPTSI2.
- [TSG3n]
  Patterns set in TSG3nPEC are output in response to the input of the TSG3nPTSI0 to TSG3nPTSI2 signals. TSG3nPUD is output depending on rotation in normal or reverse.
- [PIC1A_ENCAn input selector]
  TSG3nPEC is selected and output to ENCATnE0. TSG3nPUD is selected and output to ENCATnE1.
- [ENCAn]
  ENCATnE0 and ENCATnE1 are encoded.

(3) Registers

Block diagram of PIC1A is shown in the following figure.

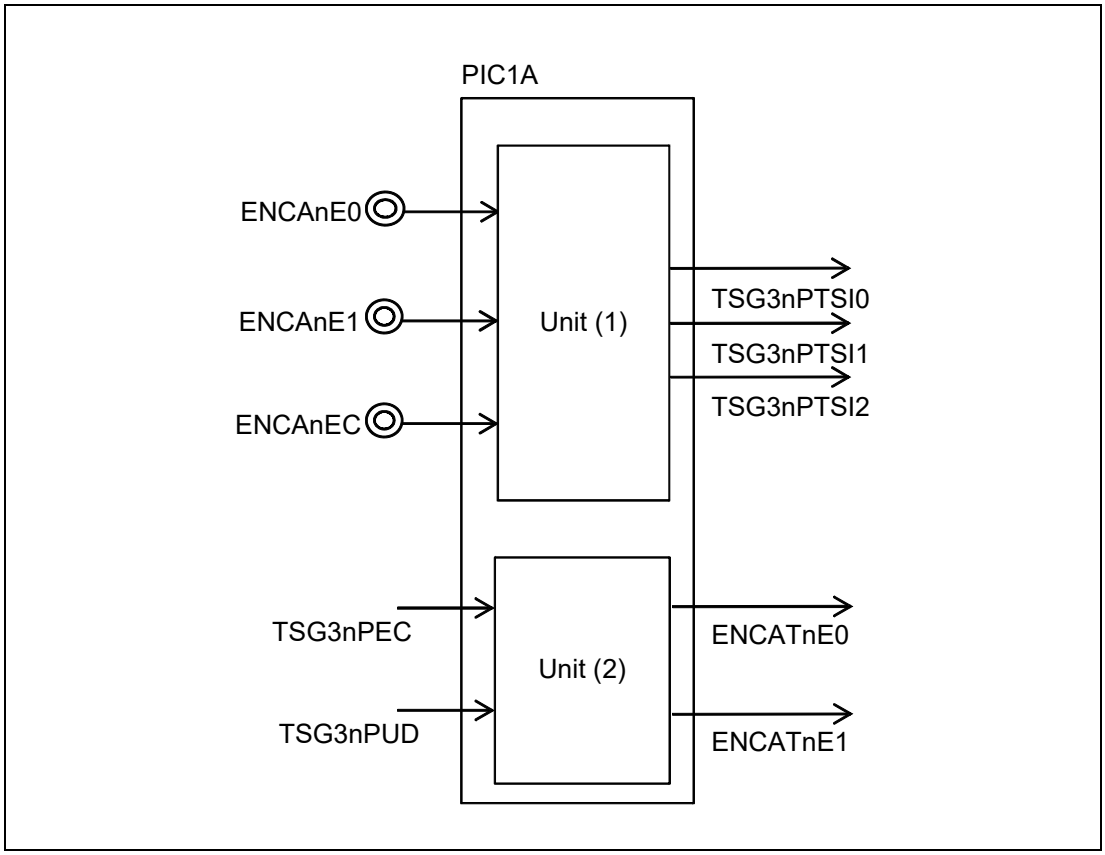


Figure 23.42 Block Diagram of PIC1A

The register settings for PIC1A to use this function are described as follows.

Unit (1): PIC1A_ input pattern selector

The values to output ENCA0E0, ENCA0E1, and ENCA0EC as TSG30PTSI0 to TSG30PTSI2

PIC1ATSGHALLSEL[0] = 1_B

PIC1AREG50[0] = 0_B

The value to output ENCA1E0, ENCA1E1, and ENCA1EC as TSG31PTSI0 to TSG31PTSI2

PIC1ATSGHALLSEL[1] = 1_B

PIC1AREG51[0] = 1_B

Unit (2): PIC1A_ENCAn input selector

The values to output TSG30PEC and TSG30PUD as ENCAT0E0 and ENCAT0E1, respectively

PIC1AREG30[22] = 0_B

PIC1AREG30[1:0] = 11_B

The values to output TSG31PEC and TSG31PUD as ENCAT1E0 and ENCAT1E1, respectively

PIC1AREG30[9:8] = 10_B

PIC1AREG30[7:6] = 10_B

(4) Function

Detail of the three-phase encoder function is described as follows.

The following figure shows the timing diagram.

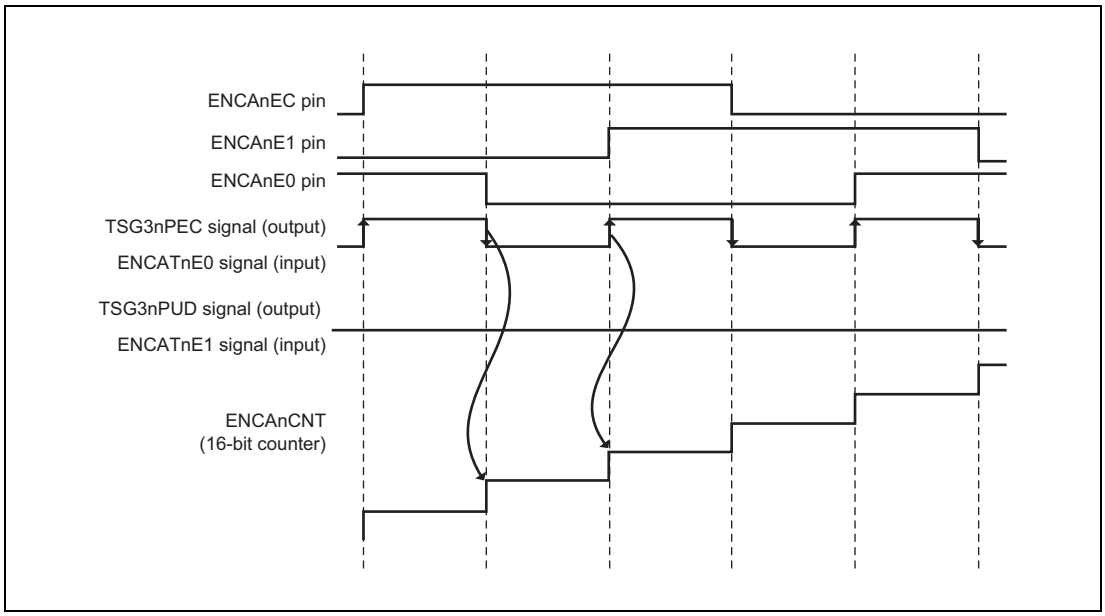


Figure 23.43 Timing Diagram of Three-Phase Encoder Function_ENCAnUDS1 and ENCAnUDS0 = 00_B

(1) When the low level is input to ENCAnE1, the count is incremented each time an active edge is input to ENCAnE0.

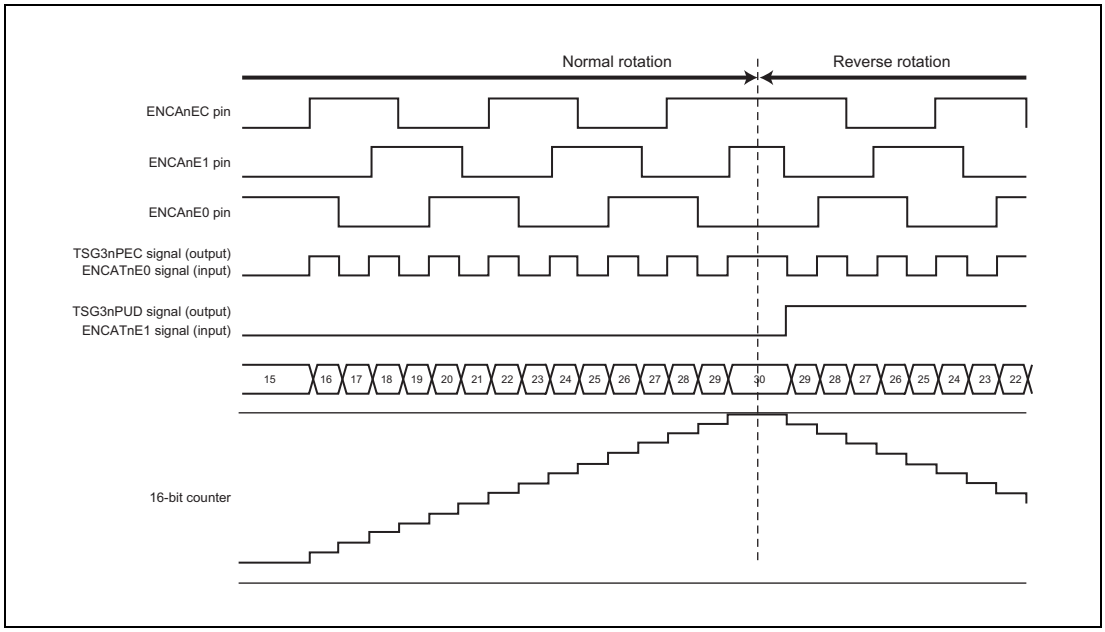


Figure 23.44 Timing Diagram of Three-Phase Encoder Function (Normal Rotation/ Reverse Rotation)

(5) Flow Chart

Set PIC before using the three-phase encoder control function.

The values of ENCAn registers used in this function are as follows.

ENCAnCTL[15:0] = $xx00\_00xx\_000x\_xx00_B$

ENCAnIOC1[7:0] = 0000_00xx_B*1

ENCAnCCR0 = (set any value)

ENCAnCCR1 = (set any value)

ENCAnCNT = (set any value)

"x" can be set arbitrarily. For the registers specifications, see the section of ENCA.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[1:0] because edge detection is necessary.

The values of TSG3n registers used in this function are as follows.

TSG3nCTL0[7:0] = 0000_0001_B

TSG3nCTL3[7:0] = 0000_00xx_B

TSG3nCTL4[15:0] = $0000\_000x\_xxxx\_xxxx_B$

TSG3nIOC0[7:0] = $0xxx\_xxx0_B$

TSG3nIOC1[7:0] = 0001_xxxx_B

TSG3nIOC2[15:0] = $0xxx\_xxx0\_0000\_0000_B$

TSG3nOPT0[7:0] = $0xxx\_xxx0_B$

TSG3nOPT1[7:0] = 0000_0xxx_B

TSG3nCMP0 = (set any value)

TSG3nCMP1W, 5W, 9W = (set any value)

TSG3nCMP1, 5, 9 = (set any value)

TSG3nPAT0W, 1W = (set any value)

TSG3nDTC0W, 1W = (set any value)

"x" can be set arbitrarily. For the registers specifications, see **Section 19, Motor Control Timer (TSG3)**.

23.2.3.12 ENCA Input Select Function

(1) Overview

The function selects ENCAn input signals from among RDC0, RDC1, encoder signal group 0 pin, and encoder signal group1 pin.

Connecting the same encoder signals to both ENCAn modules so that operation of the two is synchronized virtually increases the number of registers available for comparison and capture.

(2) Configuration

The ENCA input select function is realized by using RDCn output signal and ENCAn input pin signal, and PIC1A in combination. The following figure shows the block diagram of ENCA input select function.

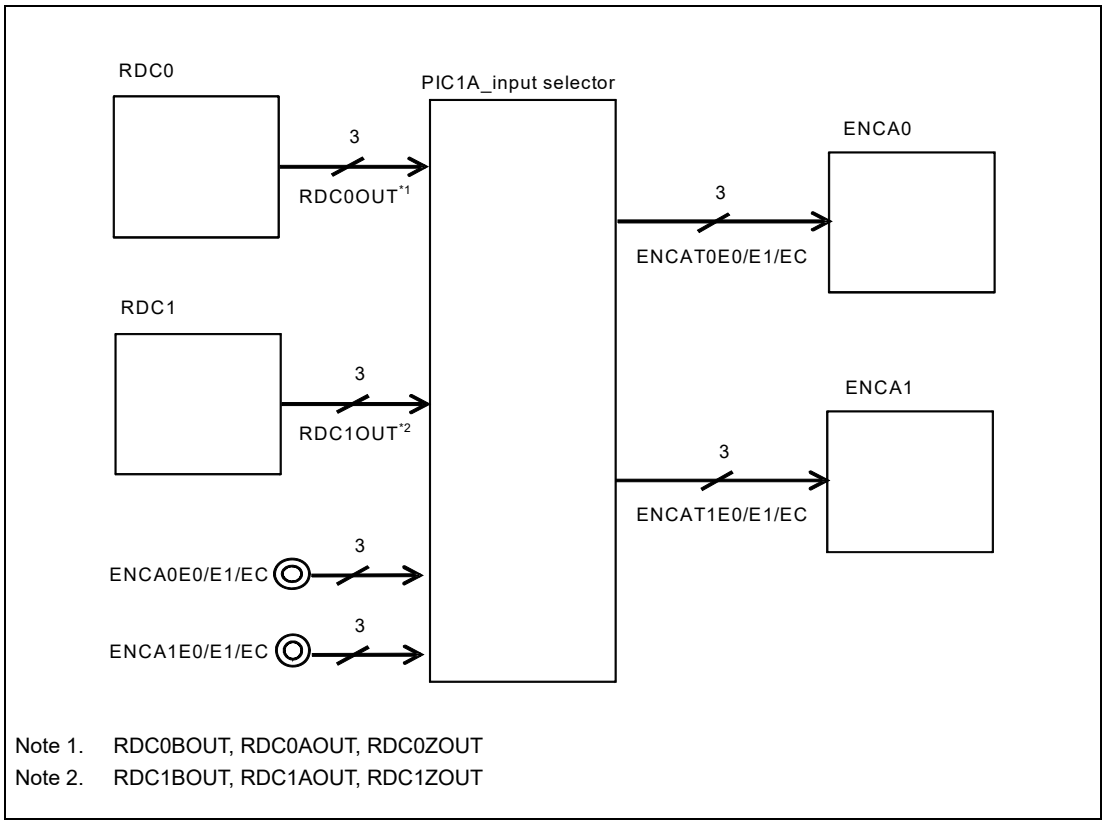
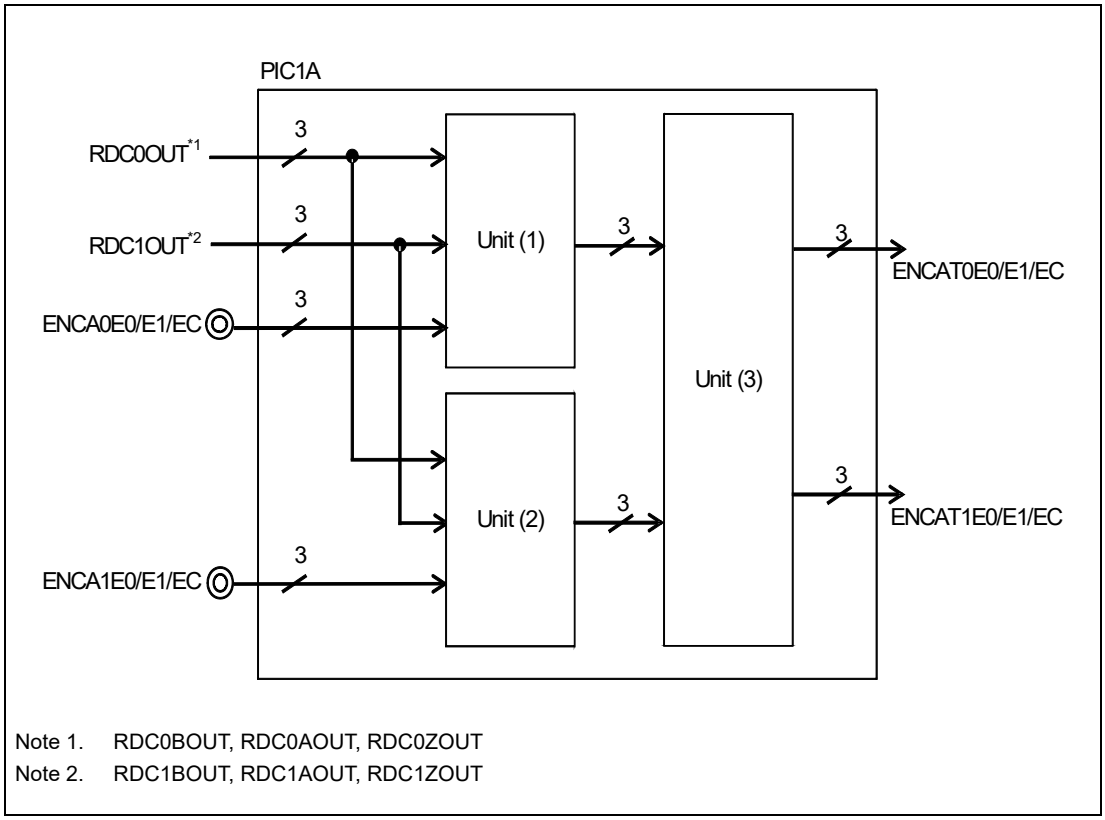


Figure 23.45 Block Diagram of ENCA Input Select Function

The signals to be output to ENCA0 and ENCA1 can be selected from the signals input to PIC1A.

(3) Registers

Block diagram of PIC1A is shown in the following figure.



Note 1. RDC0BOUT, RDC0AOUT, RDC0ZOUT
Note 2. RDC1BOUT, RDC1AOUT, RDC1ZOUT

Figure 23.46 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

PIC1A_input selector

- ENCA0 connection selector

The value to select the signal to input to ENCAT0E0, ENCAT0E1, and ENCAT0EC in unit (3) via unit (1)

Register Setting					
PIC1AREG30					
22	[17:16]	[1:0]	ENCAT0E0	ENCAT0E1	ENCAT0EC
0	01	00	RDC0BOUT	RDC0AOUT	RDC0ZOUT
0	10	00	RDC1BOUT	RDC1AOUT	RDC1ZOUT
0	00	00	ENCA0E0 pin input	ENCA0E1 pin input	ENCA0EC pin input

Note: Do not set the values other than the settings listed above for this function.

The value to select the signal to input to ENCAT0E0, ENCAT0E1, and ENCAT0EC in unit (3) via unit (2)

Register Setting				
PIC1AREG30				
22	[20:19]	ENCAT0E0	ENCAT0E1	ENCAT0EC
1	10	RDC0BOUT	RDC0AOUT	RDC0ZOUT
1	01	RDC1BOUT	RDC1AOUT	RDC1ZOUT
1	00	ENCA1E0 pin input	ENCA1E1 pin input	ENCA1EC pin input

Note: Do not set the values other than the settings listed above for this function.

- ENCA1 connection selector

The value to select the signal to input to ENCAT1E0, ENCAT1E1, and ENCAT1EC in unit (3) via unit (1)

Register Setting				
PIC1AREG30				
[17:16]	[11:6]	ENCAT1E0	ENCAT1E1	ENCAT1EC
10	100101	RDC0BOUT	RDC0AOUT	RDC0ZOUT
01	100101	RDC1BOUT	RDC1AOUT	RDC1ZOUT
00	100101	ENCA0E0 pin input	ENCA0E1 pin input	ENCA0EC pin input

Note: Do not set the values other than the settings listed above for this function.

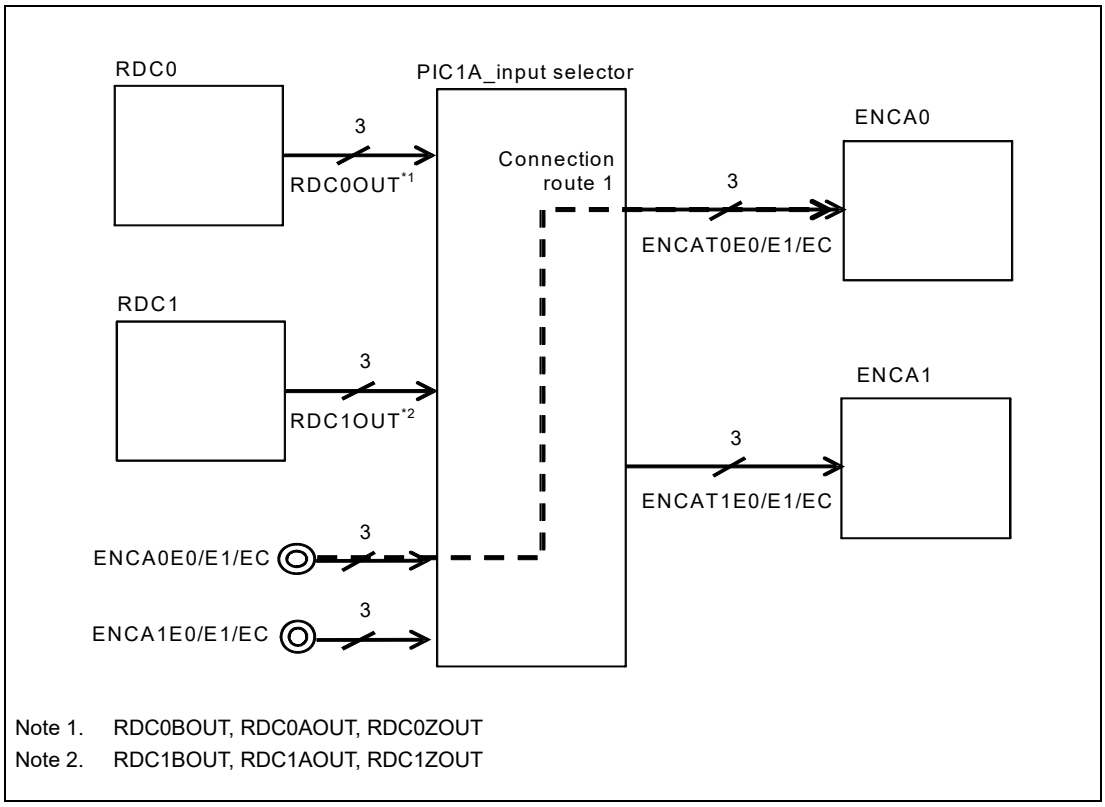
The value to select the signal to input to ENCAT1E0, ENCAT1E1, and ENCAT1EC in unit (3) via unit (2)

Register Setting				
PIC1AREG30				
[20:19]	[11:6]	ENCAT1E0	ENCAT1E1	ENCAT1EC
10	000000	RDC0BOUT	RDC0AOUT	RDC0ZOUT
01	000000	RDC1BOUT	RDC1AOUT	RDC1ZOUT
00	000000	ENCA1E0 pin input	ENCA1E1 pin input	ENCA1EC pin input

Note: Do not set the values other than the settings listed above for this function.

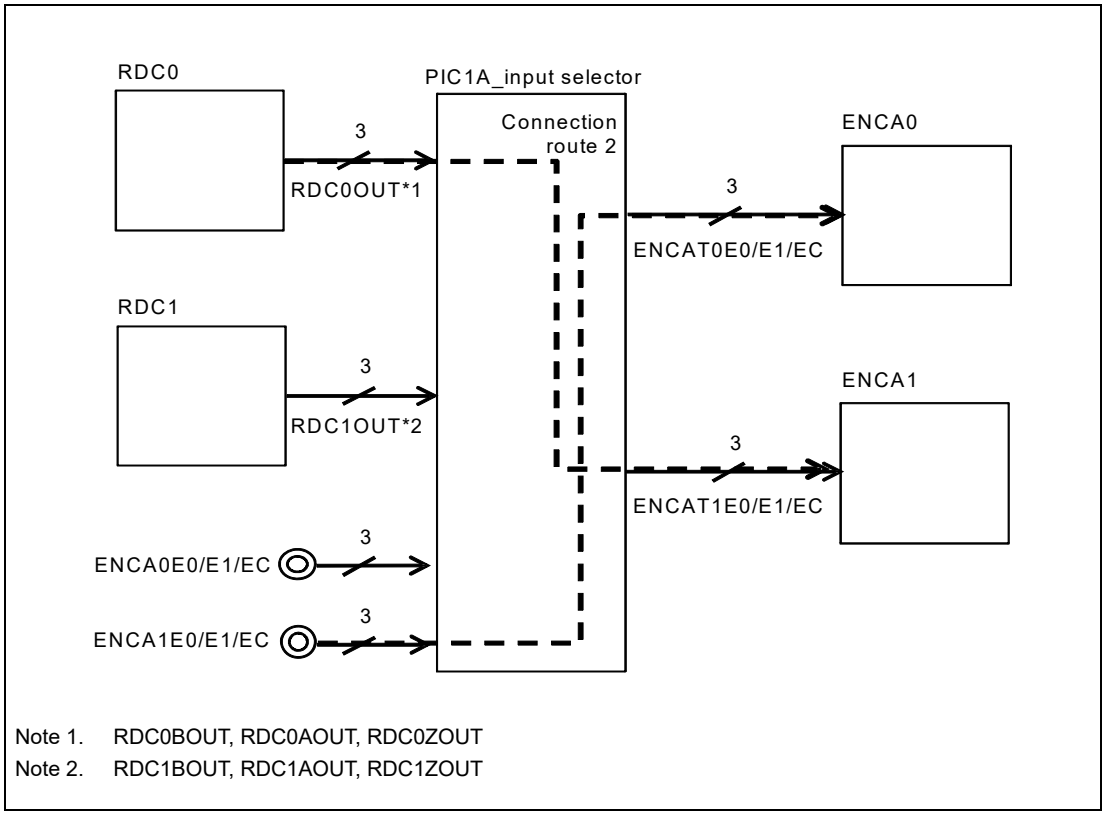
(4) Function

This function allows the connection route 1 and the connection route 2 as shown in the flowing figures. Each connection route counts the selected input signal.



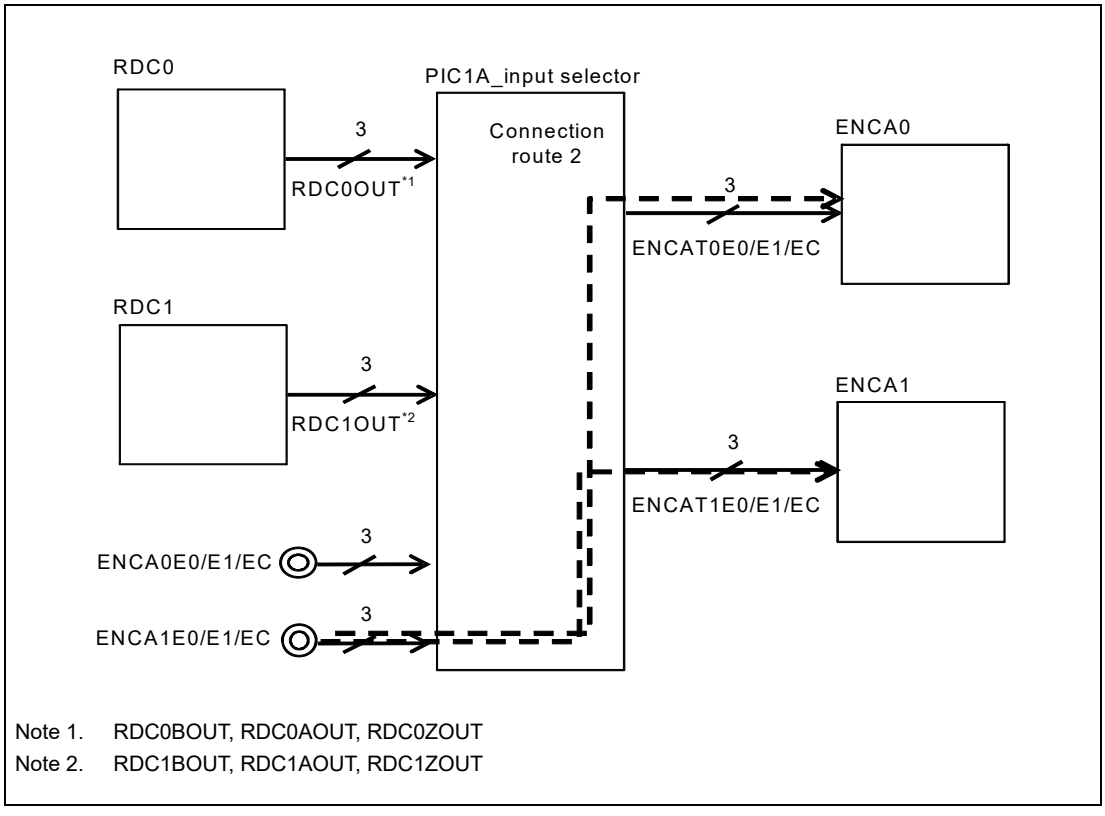
Note 1. RDC0BOUT, RDC0AOUT, RDC0ZOUT
Note 2. RDC1BOUT, RDC1AOUT, RDC1ZOUT

Figure 23.47 Example of Connection Route 1 (ENCA0 Pin Connected to ENCA0 Timer)



Note 1. RDC0BOUT, RDC0AOUT, RDC0ZOUT
Note 2. RDC1BOUT, RDC1AOUT, RDC1ZOUT

Figure 23.48 Example of Connection Route 2 (ENCA1 Pin and RDC0 Connected to ENCA0 and ENCA1, Respectively)



Note 1. RDC0BOUT, RDC0AOUT, RDC0ZOUT
Note 2. RDC1BOUT, RDC1AOUT, RDC1ZOUT

Figure 23.49 Example of Connection Route 2 (ENCA1 Pin Connected to ENCA0 and ENCA1)

(5) Flow Chart

Set PIC before starting the encoder timer.

The values of ENCAn registers when selecting different signals for input to ENCA0 and ENCA1 are as follows.

ENCAnCTL[15:0] = $1000\_0000\_000x\_0101_B$ (ENCA pin input), $1000\_0000\_000x\_0111_B$ (RDC input)

ENCAnIOC0[7:0] = 0000_xxxx_B

ENCAnIOC1[7:0] = $xxxx\_xxxx_B$*1

"x" can be set arbitrarily. For the registers specifications, see the section of ENCA.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[3:2] and [1:0] because edge detection is necessary.

The values of ENCAn registers when selecting same signals for input to ENCA0 and ENCA1 are as follows.

ENCAnCTL[15:0] = $0x00\_000x\_x00x\_0xxx_B$ (ENCA pin input), $0x00\_000x\_x00x\_0x11_B$ (RDC input)

ENCAnIOC0[7:0] = 0000_xxxx_B

ENCAnIOC1[7:0] = $xxxx\_xxxx_B$*1

"x" can be set arbitrarily. For the registers specifications, see the section of ENCA.

Note 1. Except for 00_B (no edge detected) for ENCAnIOC1[3:2] and [1:0] because edge detection is necessary.

23.2.3.13 TAUD Input Select Function

(1) Overview

The function selects TAUDn input signal to be input as TAUDnTTINm/m+1 signal from either TAUD0Im/m+1 signal or TAUD1Im/m+1 signal (m is an even number between 0 and 14).

(2) Configuration

The TAUD input select function is realized by using TAUDn input signals and PIC1A in combination.

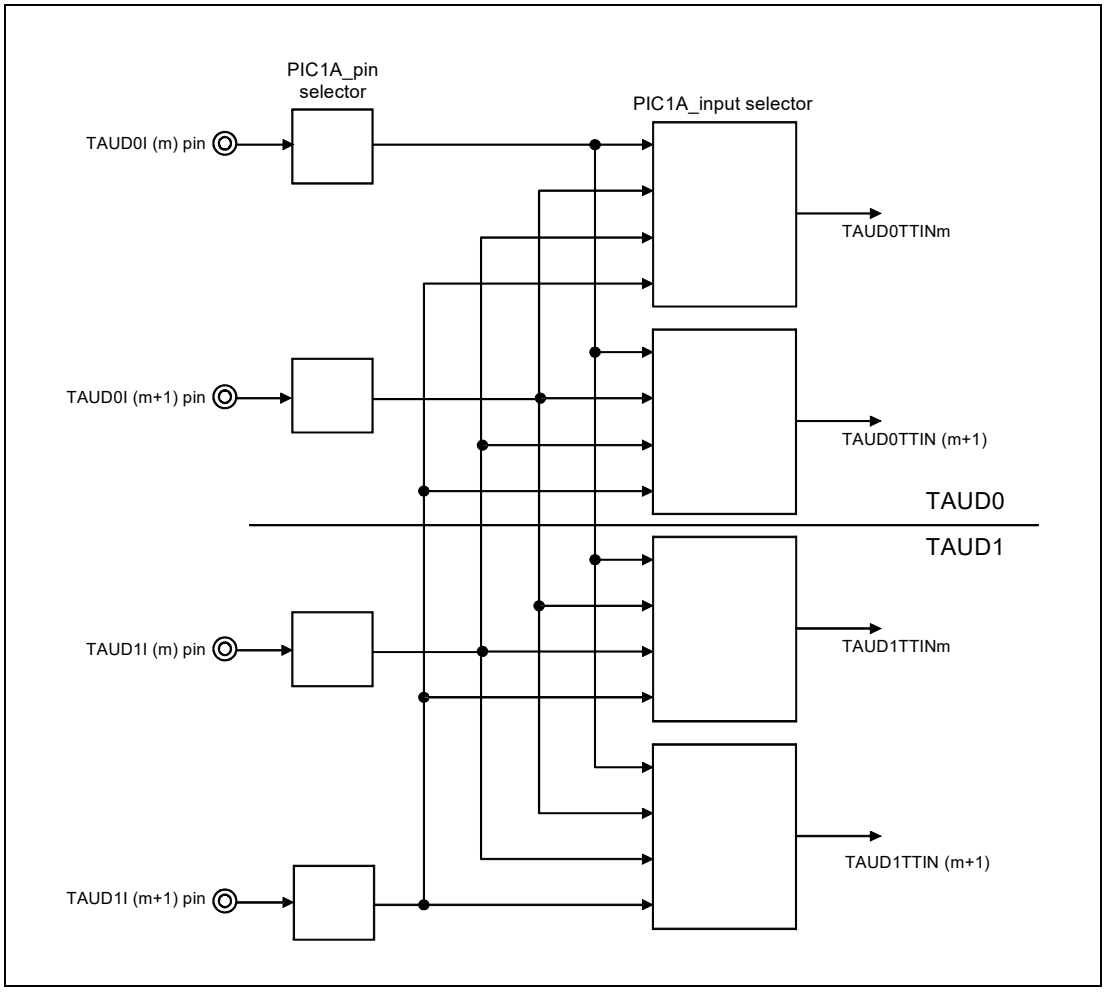


Figure 23.50 Block Diagram of TAUD Input Select Function

(3) Registers

The register settings for PIC1A to use this function are described as follows.

PIC1A_ pin selector

Select TAUDn external channel input pin for output from PIC1A pin selector.

PIC1A_input selector

Select by the registers listed as follows.

PIC1ATAUD0SEL

PIC1ATAUD1SEL

For the details of the register settings, see **Section 23.2.2.14**, **23.2.2.15**, **23.2.2.22**, **23.2.2.23**, **23.2.2.25**, and **23.2.2.28**.

(4) Function

Detail of the function is described using selection of the TAUD0TTIN[1:0] signal as an example.

The following table lists an example of selection of the TAUD0TTIN[1:0] signals. Setting 000000 to PIC1AREG31[11:6], and setting 01 to PIC1ATAUD0SEL[3:2] and PIC1ATAUD0SEL[1:0] allows TAUD0I0 and TAUD0I1 signals to be input to TAUD0TTIN1 and TAUD0TTIN0 input pins of the TAUD0 timer. Setting 1 to PIC1ATAUD0SEL[3] and PIC1ATAUD0SEL[1] selects TIN pin signal of TAUD1.

Register Setting PIC1ATAUD0SEL [1:0]	TAUD0TTIN0
00_B	TAUD0I0 pin
01_B	TAUD0I1 pin
10_B	TAUD1I0 pin
11_B	TAUD1I1 pin

Register Setting PIC1ATAUD0SEL [3:2]	TAUD0TTIN1
00_B	TAUD0I1 pin
01_B	TAUD0I0 pin
10_B	TAUD1I1 pin
11_B	TAUD1I0 pin

(5) Flow Chart

Set PIC1A before starting the TAUDn timer.

23.2.3.14 Switch Function Between TSG Output and Low/High Level Output

(1) Overview

The function allows switching of outputs between TSG output and low/high level output at desired timing.

(2) Configuration

TSG is output via PIC1A. With this function, the TSG output is delayed for one cycle of the clock (CLKC_HSB).

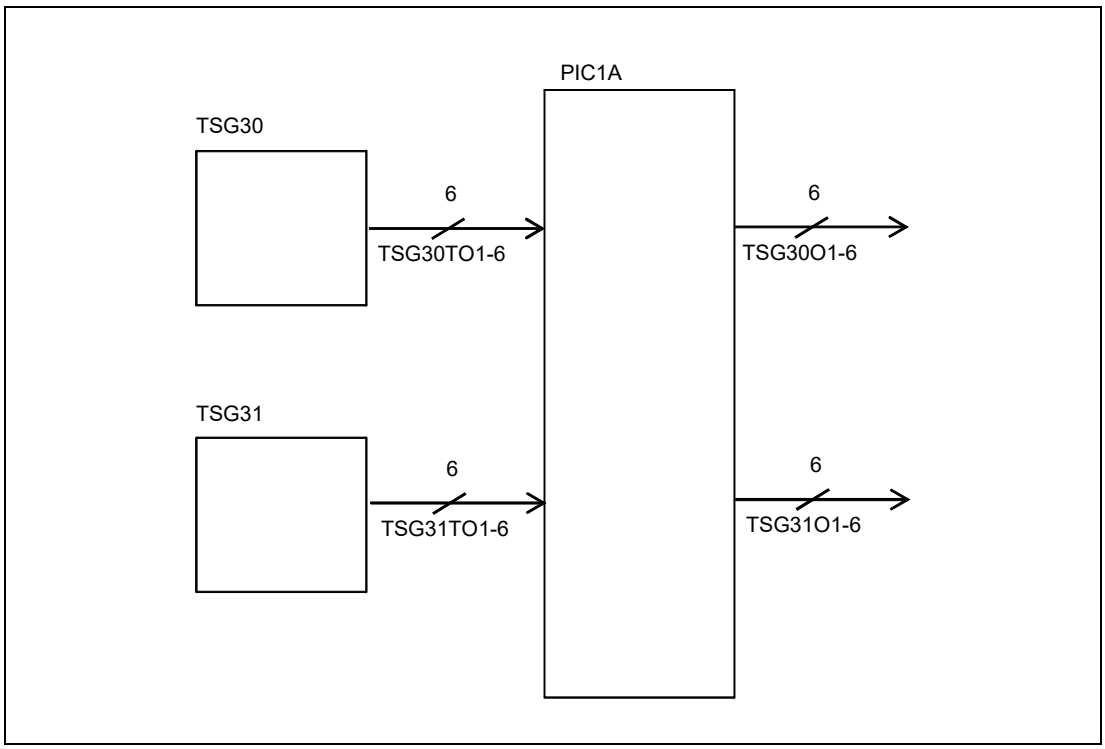


Figure 23.51 Block Diagram of Switch Function Between TSG Output and Low/High-Level Output

(3) Registers

The block diagram of PIC1A is shown in the following figure.

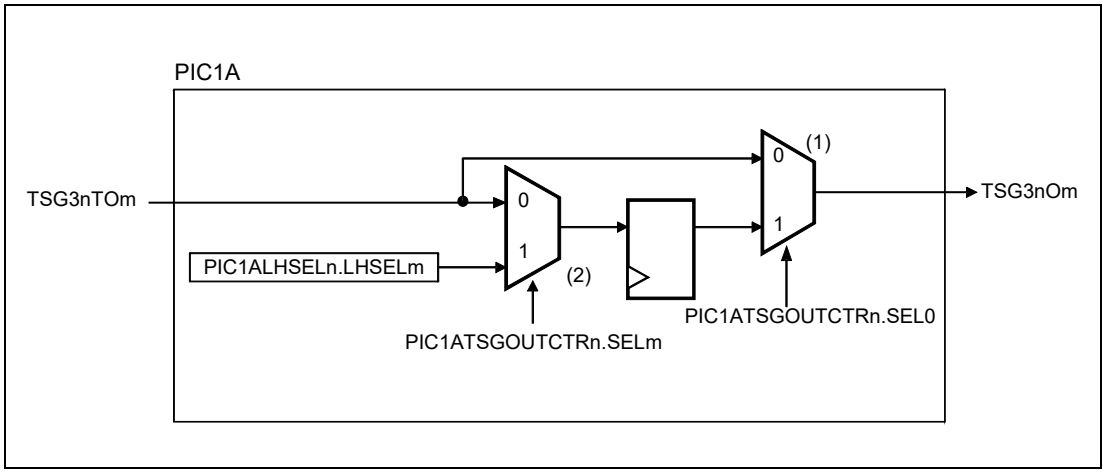


Figure 23.52 Block Diagram of PIC1A

The values of PIC1A registers used in this function are as follows.

(1) Switch on and off the function
PIC1ATSGOUTCTRn[0] = 0_B (turned off), 1_B (turned on)

(2) Switch between low level output and high level output
PIC1ALHSELn[m] = 0_B (output of low level), 1_B (output of high level)
PIC1ATSGOUTCTRn.SELm = 0_B (TSG3n output), 1_B (output of low /high level)

CAUTION

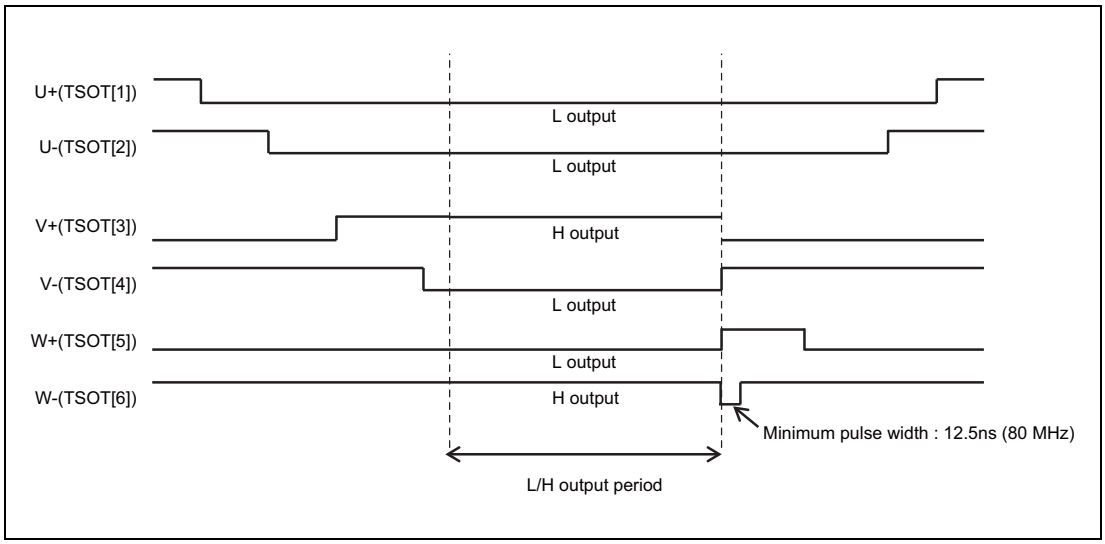
Set (1) before starting TSG3n.

Changing at operation is prohibited.

(4) Function

Detail of the function is described using an example timing of switch function.

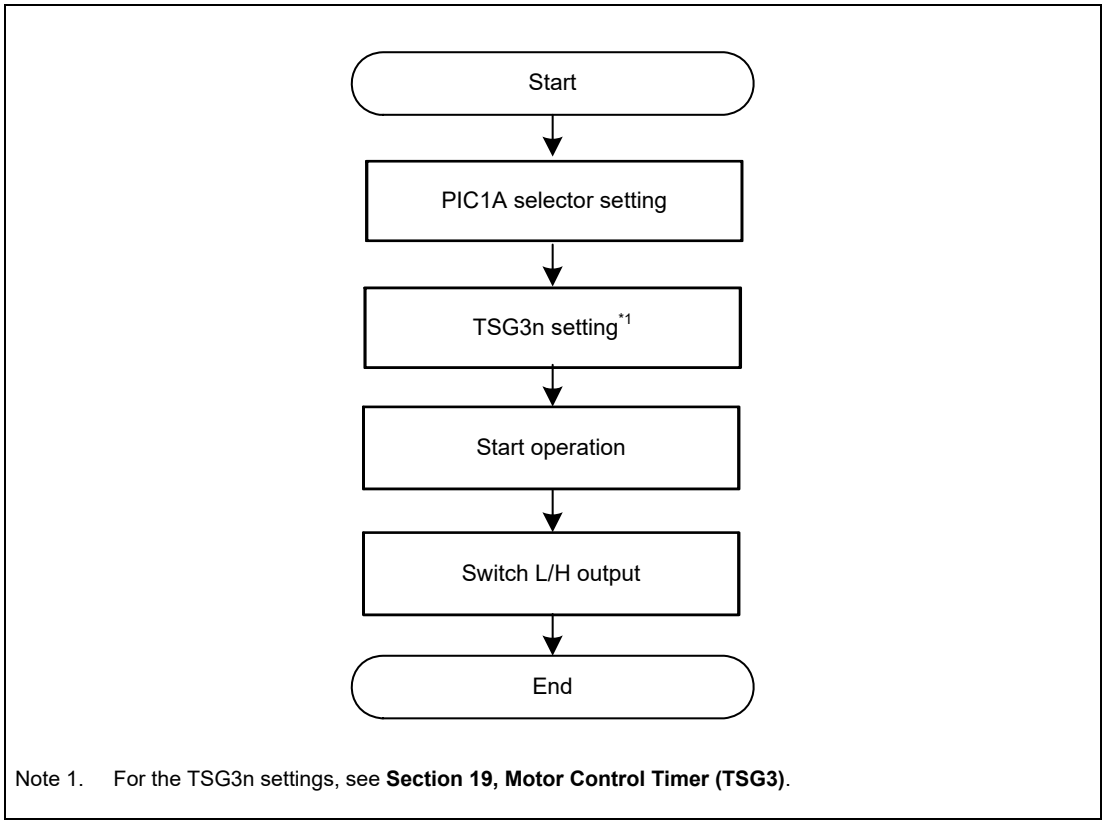
The following figure shows the timing diagram.



The function allows switching of outputs between TSG output and low/high level output at desired timing. The six output phases can be switched between high level and low level both individually and simultaneously. The minimum pulse (internal pulse) of CLKC_HSB may be generated depending on the switch timing.

(5) Flow Chart

The flow charts for this function are shown as follows.



Note 1. For the TSG3n settings, see **Section 19, Motor Control Timer (TSG3)**.

23.2.3.15 Hi-Z Control Function

(1) Overview

The function disconnects three-phase output signal and changes to Hi-Z state.

For the detail of the purpose and operation of the Hi-Z control function, see **Section 20.4.1, Asynchronous Hi-Z Control Functions**.

(2) Configuration

The ESOn, ERROROUTZ, INTTSG30IER, and INTTSG31IER signals are masked and OR'ed in PIC1A, and output to TAPAn as the signal for Hi-Z control.

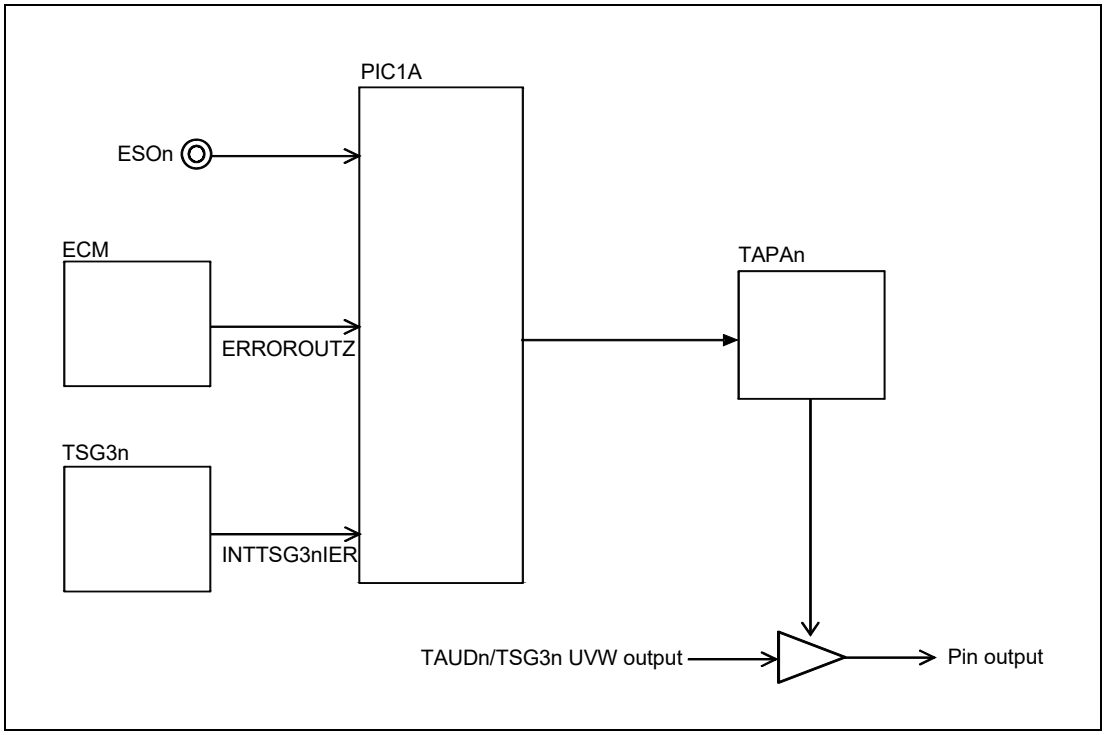


Figure 23.53 Block Diagram of Hi-Z Control

(3) Registers

The block diagram of PIC1A is shown in the following figure.

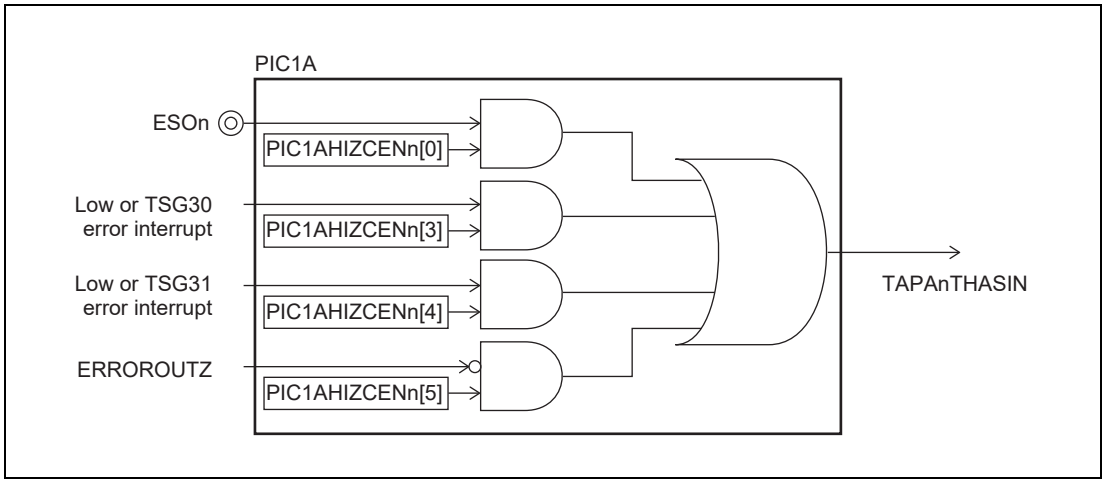


Figure 23.54 Block Diagram of PIC1A

The register settings for PIC1A to use this function are described as follows.

PIC1AHIZCENn[5] = 1_B (enabled), 0_B (disabled): ERROROUTZ

PIC1AHIZCENn[0] = 1_B (enabled), 0_B (disabled): ESOn

PIC1AHIZCENn[3] = 1_B (enabled), 0_B (disabled): INTTSG30IER

PIC1AHIZCENn[4] = 1_B (enabled), 0_B(disabled): INTTSG31IER

n = 0, 1 is available for TAUD0/TAUD1 and not available for PIC1AHIZCENn[3] and PIC1AHIZCENn[4].

n = 2 is available for TSG30 and not available for PIC1AHIZCEn[4].

n = 3 is available for TSG31 and not available for PIC1AHIZCEn[3].

(4) Function

The signals ESOn pin, ERROROUTZ, INTTSG30IER, and INTTSG31IER are masked and OR'ed in PIC1A, and output to TAPAn. For Hi-Z control by TAPA, see **Section 20.4.2.2, Basic Operation**.

(5) Flow Chart

Set PIC1A before starting Hi-Z control.

For the operation flow of TAPA, see **Section 20.4.2.3, Operating Procedure**.

23.3 Peripheral Interconnection 2 (PIC2B)

23.3.1 Overview

23.3.1.1 Functional Overview

The peripheral interconnection 2 (PIC2B) allows ADCC hardware trigger signal to be generated using the internal and external trigger signals output from individual IPs.

23.3.2 Registers

23.3.2.1 List of Registers

PIC2B registers are listed in the following table.

See **Section 23.1.2, Register Base Address** for <PIC2B_base>.

Table 23.40 List of Registers

Register Name	Symbol	Address
A/D converter 0 trigger select control register 0	PIC2BADCC0TSEL0	<PIC2B_base> + 00_H
A/D converter 0 trigger select control register 1	PIC2BADCC0TSEL1	<PIC2B_base> + 04_H
A/D converter 0 trigger select control register 2	PIC2BADCC0TSEL2	<PIC2B_base> + 08_H
A/D converter 0 trigger select control register 3	PIC2BADCC0TSEL3	<PIC2B_base> + $0C_H$
A/D converter 0 trigger select control register 4	PIC2BADCC0TSEL4	<PIC2B_base> + 10_H
A/D converter 0 trigger edge select control register	PIC2BADCC0EDGSEL	<PIC2B_base> + $1C_H$
A/D converter 1 trigger select control register 0	PIC2BADCC1TSEL0	<PIC2B_base> + 20_H
A/D converter 1 trigger select control register 1	PIC2BADCC1TSEL1	<PIC2B_base> + 24_H
A/D converter 1 trigger select control register 2	PIC2BADCC1TSEL2	<PIC2B_base> + 28_H
A/D converter 1 trigger select control register 3	PIC2BADCC1TSEL3	<PIC2B_base> + $2C_H$
A/D converter 1 trigger select control register 4	PIC2BADCC1TSEL4	<PIC2B_base> + 30_H
A/D converter 1 trigger edge select control register	PIC2BADCC1EDGSEL	<PIC2B_base> + $3C_H$
Common to ADCC0 and ADCC 1		
A/D converter trigger output control register 400	PIC2BADTEN400	<PIC2B_base> + 40_H
A/D converter trigger output control register 401	PIC2BADTEN401	<PIC2B_base> + 44_H
A/D converter trigger output control register 402	PIC2BADTEN402	<PIC2B_base> + 48_H
A/D converter trigger output control register 403	PIC2BADTEN403	<PIC2B_base> + $4C_H$
A/D converter trigger output control register 404	PIC2BADTEN404	<PIC2B_base> + 50_H
A/D converter trigger output control register 410	PIC2BADTEN410	<PIC2B_base> + 60_H
A/D converter trigger output control register 411	PIC2BADTEN411	<PIC2B_base> + 64_H
A/D converter trigger output control register 412	PIC2BADTEN412	<PIC2B_base> + 68_H
A/D converter trigger output control register 413	PIC2BADTEN413	<PIC2B_base> + $6C_H$
A/D converter trigger output control register 414	PIC2BADTEN414	<PIC2B_base> + 70_H

23.3.2.2 PIC2BADCCnTSELx — A/D Converter n Trigger Select Control Register x

The PIC2BADCCnTSELx register selects triggers for ADCCn scan group x (n = 0, 1; x = 0 to 4).

Address: Base = FFDD 1000_H
Base + 00_H (n = 0, x = 0), Base + 04_H (n = 0, x = 1), Base + 08_H (n = 0, x = 2),
Base + $0C_H$ (n = 0, x = 3), Base + 10_H(n = 0, x = 4),
Base + 20_H (n = 1, x = 0), Base + 24_H(n = 1, x = 1), Base + 28_H (n = 1, x = 2),
Base + $2C_H$ (n = 1, x = 3), Base + 30_H(n = 1, x = 4)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PIC2BADCCnTSELx[21:16]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PIC2BADCCnTSELx[12:0]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.41 PIC2BADCCnTSELx Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 22	—	Reserved These bits are read as 0. The write value should be 0.
21	PIC2BADCCnTSELx21	Selects RDC1 excitation timer (ET) A/D conversion start trigger signal output (RDC1ETCNT) as the trigger source for ADCCn scan group x.*1 0: RDC1ETCNT is not selected. 1: RDC1ETCNT is selected.
20	PIC2BADCCnTSELx20	Selects RDC1 Phi compare 0 match interrupt output (RDC1COMP0INT) as the trigger source for ADCCn scan group x.*1 0: RDC1COMP0INT is not selected. 1: RDC1COMP0INT is selected.
19	PIC2BADCCnTSELx19	Selects RDC0 excitation timer (ET) A/D conversion start trigger signal output (RDC0ETCNT) as the trigger source for ADCCn scan group x. 0: RDC0ETCNT is not selected. 1: RDC0ETCNT is selected.
18	PIC2BADCCnTSELx18	Selects RDC0 Phi compare 0 match interrupt output (RDC0COMP0INT) as the trigger source for ADCCn scan group x. 0: RDC0COMP0INT is not selected. 1: RDC0COMP0INT is selected.
17	PIC2BADCCnTSELx17	Selects TAPA1TADOUT1 pin as the trigger source for ADCCn scan group x. 0: TAPA1TADOUT1 pin is not selected. 1: TAPA1TADOUT1 pin is selected.
16	PIC2BADCCnTSELx16	Selects TAPA0TADOUT1 pin as the trigger source for ADCCn scan group x. 0: TAPA0TADOUT1 pin is not selected. 1: TAPA0TADOUT1 pin is selected.
15 to 13	—	Reserved These bits are read as 0. The write value should be 0.
12	PIC2BADCCnTSELx12	Selects ADC scan group conversion start signal of EMU channel 1 (EMU1ADT) as the trigger source for ADCCn scan group x. 0: EMU1ADT is not selected. 1: EMU1ADT is selected.

Table 23.41 PIC2BADCCnTSELx Register Contents (2/2)

Bit Position	Bit Name	Function
11	PIC2BADCCnTSELx11	Selects ADC scan group conversion start signal of EMU channel 0 (EMU0ADT) as the trigger source for ADCCn scan group x. 0: EMU0ADT is not selected. 1: EMU0ADT is selected.
10	PIC2BADCCnTSELx10	Selects the TAPA1TADOUT0 pin as the trigger source for ADCCn scan group x. 0: TAPA1TADOUT0 pin is not selected. 1: TAPA1TADOUT0 pin is selected.
9	PIC2BADCCnTSELx09	Selects the TAPA0TADOUT0 pin as the trigger source for ADCCn scan group x. 0: TAPA0TADOUT0 pin is not selected. 1: TAPA0TADOUT0 pin is selected.
8	PIC2BADCCnTSELx08	Selects the ADTRGnZ pin as the trigger source for ADCCn scan group x. 0: ADTRGnZ pin is not selected. 1: ADTRGnZ pin is selected.
7	PIC2BADCCnTSELx07	Selects TSG1TSTADT1 as the trigger source for ADCCn scan group x (the TSG1TSTADT1 signal). 0: TSG1TSTADT1 is not selected. 1:TSG1TSTADT1 is selected.
6	PIC2BADCCnTSELx06	Selects TSG1TSTADT0 as the trigger source for ADCCn scan group x (the TSG1TSTADT0 signal). 0: TSG1TSTADT0 is not selected. 1: TSG1TSTADT0 is selected.
5	PIC2BADCCnTSELx05	Selects TSG0TSTADT1 as the trigger source for ADCCn scan group x (the TSG0TSTADT1 signal). 0: TSG0TSTADT1 is not selected. 1: TSG0TSTADT1 is selected.
4	PIC2BADCCnTSELx04	Selects TSG0TSTADT0 as the trigger source for ADCCn scan group x (the TSG0TSTADT0 signal). 0: TSG0TSTADT0 is not selected. 1: TSG0TSTADT0 is selected.
3	PIC2BADCCnTSELx03	Selects ENCAT1INT1 as the trigger source for ADCCn scan group x. 0: ENCAT1INT1 is not selected. 1: ENCAT1INT1 is selected.
2	PIC2BADCCnTSELx02	Selects ENCAT0INT1 as the trigger source for ADCCn scan group x. 0: ENCAT0INT1 is not selected. 1: ENCAT0INT1 is selected.
1	PIC2BADCCnTSELx01	Selects the trigger selected by the PIC2ADTEN41x register as the trigger source for ADCCn scan group x. 0: The trigger selected by the PIC2ADTEN41x register is not selected. 1: The trigger selected by the PIC2ADTEN41x register is selected.
0	PIC2BADCCnTSELx00	Selects the trigger selected by PIC2ADTEN40x register as the trigger source for ADCCn scan group x. 0: The trigger selected by the PIC2ADTEN40x register is not selected. 1: The trigger selected by the PIC2ADTEN40x register is selected.

Note 1. Set 0 to C1M as RDC1 is not equipped.

23.3.2.3 PIC2BADCCnEDGSEL — A/D Converter Trigger Edge Control Register

The PIC2BADCCnEDGSEL register selects an effective edge for the one-shot pulse generation circuit which generates an ADCC trigger.

The ADC external pin trigger is input in negative logic, but it is converted to positive logic for trigger source selection. Since edge detection is made for a trigger source after selection, note that the definition of an edge is reversed for an ADC external pin signal.

(Setting 00 enables selection of a falling edge of ADC external pin triggers ADTRG0Z and ADTRG1Z).

Address: Base = FFDD 1000_H
Base+$1C_H$ (n = 0), Base+$3C_H$ (n = 1)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC2BADCCnEDGSEL[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.42 PIC2BADCCnEDGSEL Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Reserved These bits are read as 0. The write value should be 0.
9, 8	PIC2BADCCnEDGSEL[9:8]	Select an effective edge of ADCCn scan group 4. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
7, 6	PIC2BADCCnEDGSEL[7:6]	Select an effective edge of ADCCn scan group 3. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
5, 4	PIC2BADCCnEDGSEL[5:4]	Select an effective edge of ADCCn scan group 2. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
3, 2	PIC2BADCCnEDGSEL[3:2]	Select an effective edge of ADCCn scan group 1. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
1, 0	PIC2BADCCnEDGSEL[1:0]	Select an effective edge of ADCCn scan group 0. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)

23.3.2.4 PIC2BADTEN4nx — A/D Converter Trigger Output Select Control Register

The PIC2BADTEN4nx register enables selecting a trigger source from TAUDn channel m as the ADCC trigger. (n = 0, 1; x = 0 to 4). This register is common to ADCC0 and ADCC1.

Address: Base = FFDD 1000_H
Base + 40_H (n = 0, x = 0), Base + 44_H (n = 0, x = 1), Base + 48_H (n = 0, x = 2),
Base + $4C_H$ (n = 0,x = 3), Base + 50_H (n = 0, x = 4),
Base + 60_H (n = 1, x = 0), Base + 64_H (n = 1, x = 1), Base + 68_H (n = 1, x = 2),
Base + $6C_H$ (n = 1, x = 3), Base + 70_H (n = 1, x = 4)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2BADTEN4nx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.43 PIC2BADTEN4nx Register Contents

Bit Position	Bit Name	Function
15 to 0	PIC2BADTEN4nxm (m = 0 to15)	Set a trigger source from TAUDn channel m. 0: Trigger source of TAUDn channel m cannot be selected as the ADCC trigger. 1: Trigger source of TAUDn channel m can be selected as the ADCC trigger.

23.3.3 Function

23.3.3.1 ADCC Trigger Select Function

(1) Overview

The function allows generation of ADCC hardware trigger signal for individual channel groups by the signals from each IP. The IPs can be selected from among TAUD0, TAUD1, ENCA0, ENCA1, TSG30, TSG31, TAPA0, TAPA1, EMU, RDC0, and RDC1*1.

The external trigger signal (ADTRG) is active low and it is converted by PIC2B.

Note 1. C1M is not equipped with RDC1.

(2) Configuration

The ADCC trigger select function is realized by using individual IPs and PIC2B in combination. The following figure shows the block diagram of ADCC trigger select function.

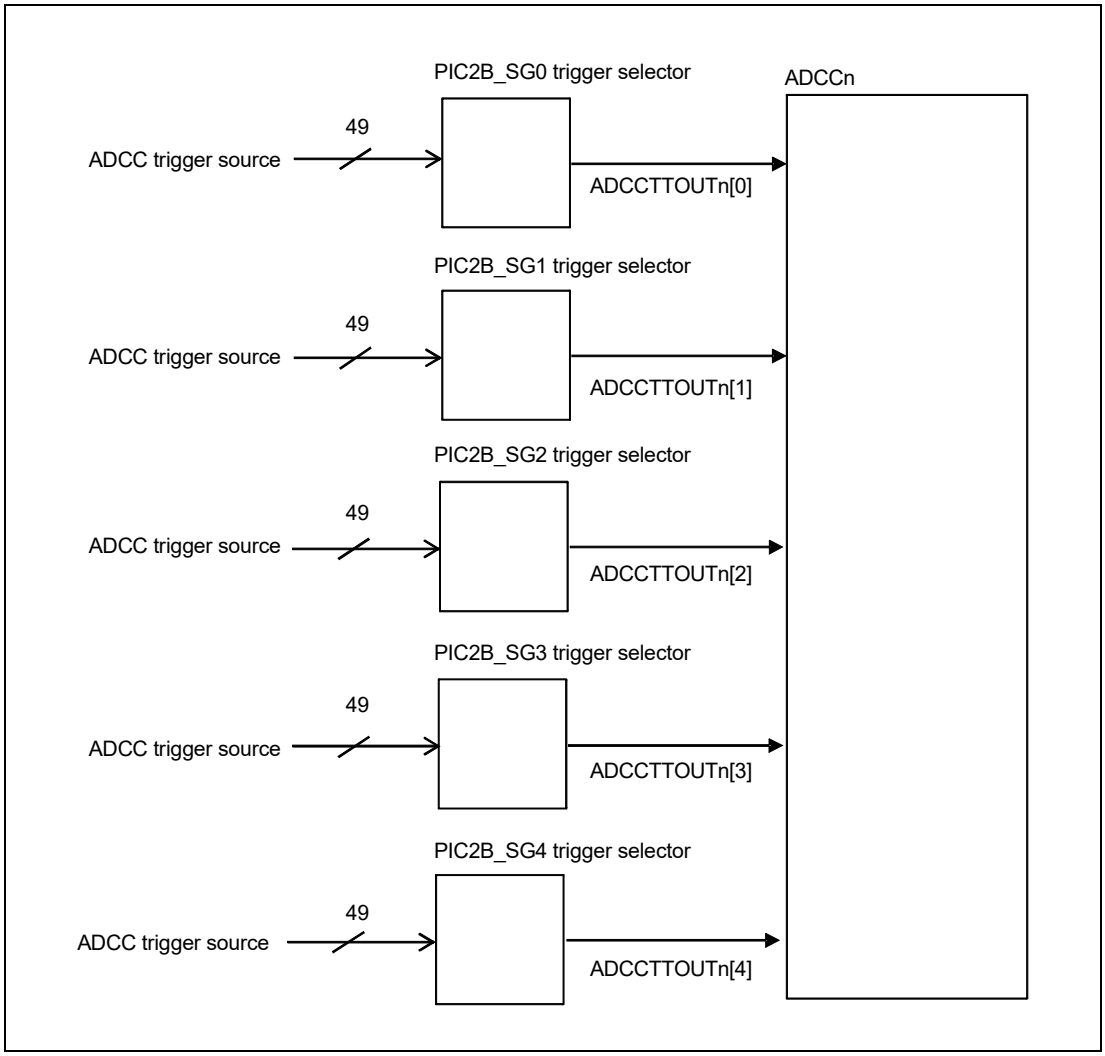


Figure 23.55 Block Diagram of ADCC Trigger Select Function

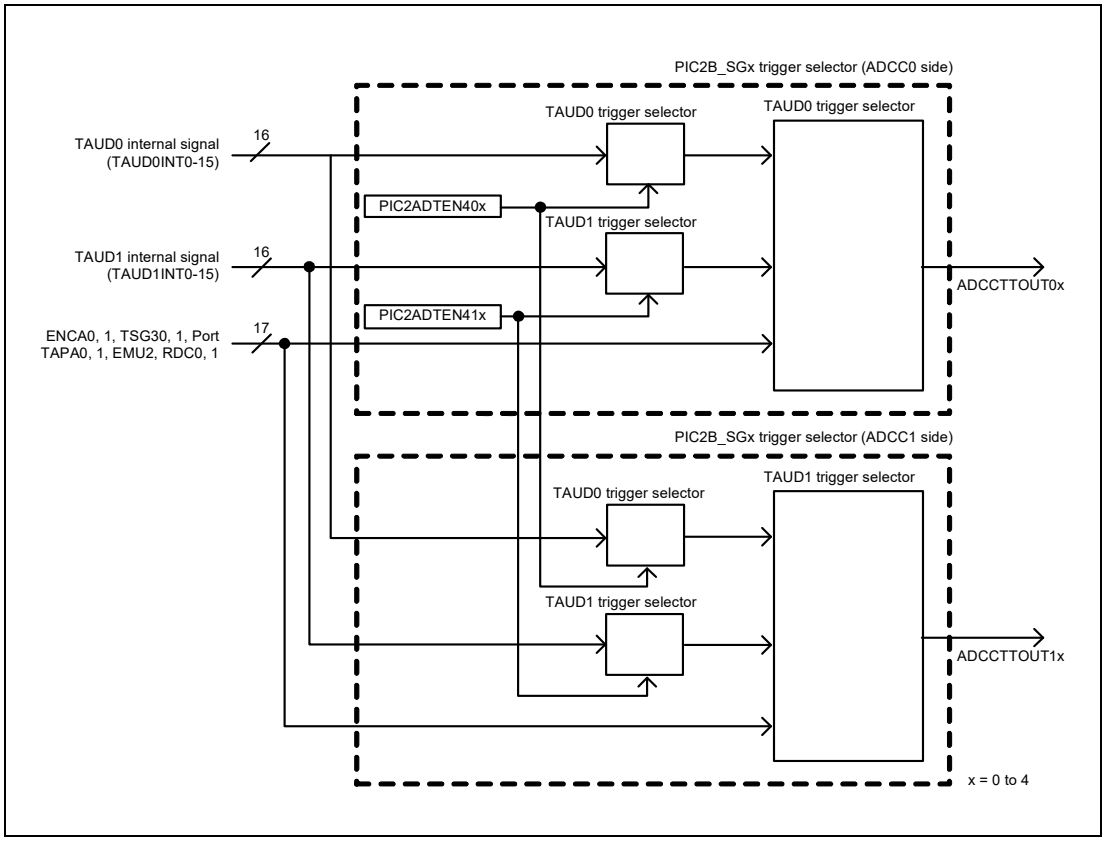
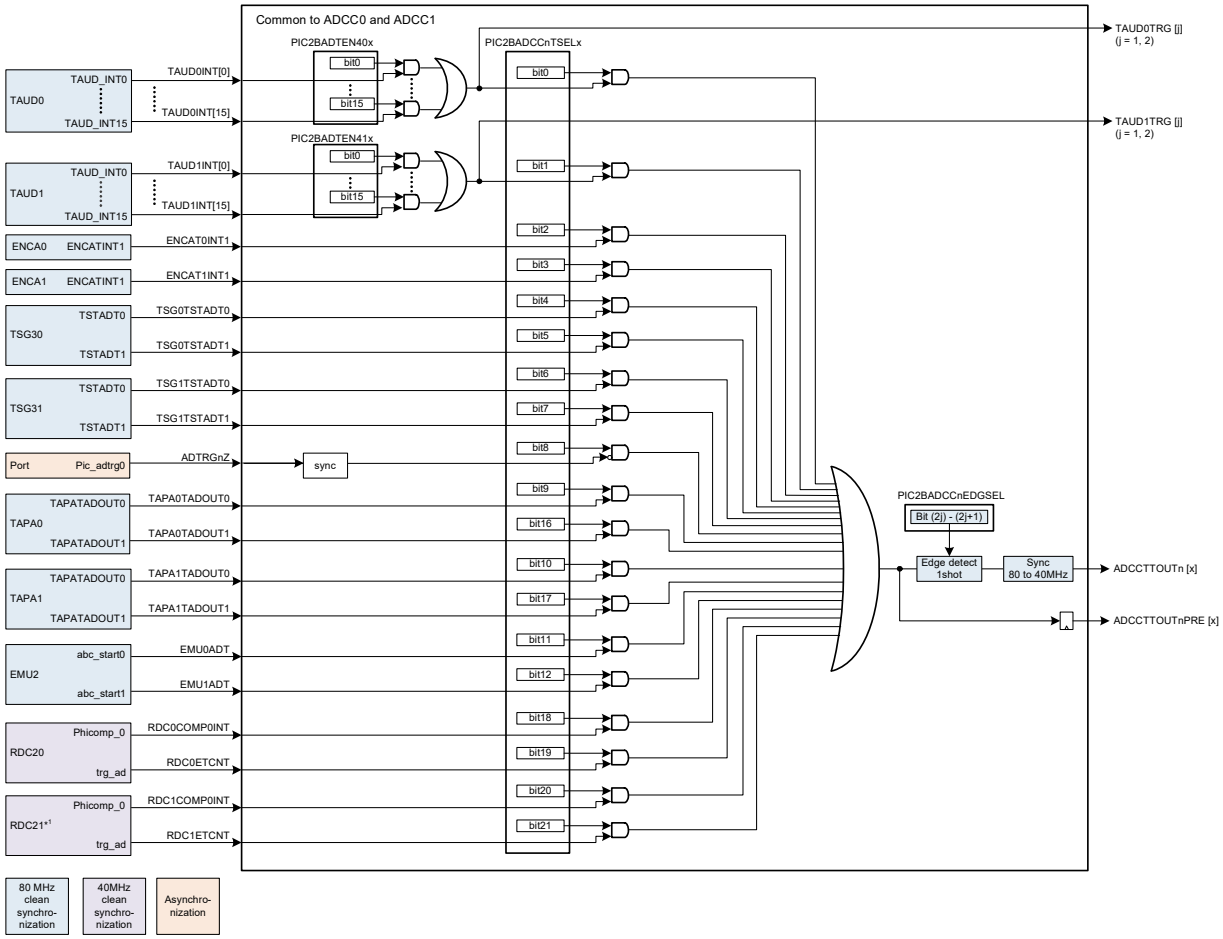


Figure 23.56 Block Diagram of PIC2B_SGx

(3) Registers

For the settings of the registers used in this function, see **Figure 23.57, Block Diagram of PIC2B**, and sections from **23.3.2.2** to **23.3.2.4**.



Note 1. C1M is not equipped.

Note 2. For external trigger signal, a delay of maximum two cycles of the clock (12.5 ns (80 MHz) @1clk) is generated because of the synchronization circuit (Sync) which needs to be considered when selecting an external trigger signal as the ADCC trigger.

Figure 23.57 Block Diagram of PIC2B

(4) Function

AD trigger signal can be selected for individual ADCCn channel groups and output can be selected from rising edge, falling edge, and both edges. The TAUD trigger for the scan group with the same number is shared among ADCC0 and ADCC1.

(5) Flow Chart

Set this function before starting A/D converter.

23.3.3.2 TAUD Trigger Output Function

(1) Overview

The interrupt signal of each channel of TAUD is masked and OR'ed, and output to TAPA as the TAUD trigger signal.

This function is available only for the scan groups 1 and 2.

The TAUD trigger signal is used as the trigger source for A/D converter conversion trigger signal for TAPA. For the detail, see Section of TAPA, A/D Converter Conversion Trigger Select Function.

(2) Configuration

The TAUD trigger output function is realized by PIC2B. The following figure shows the block diagram of the TAUD trigger output function.

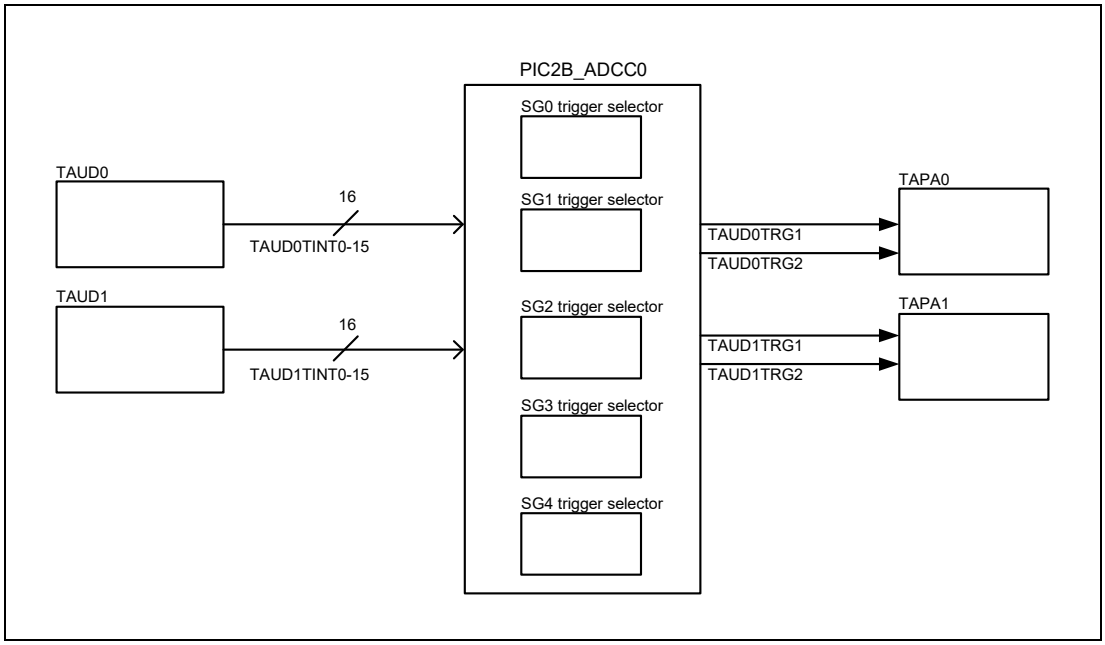


Figure 23.58 Block Diagram of TAUD Trigger Output Function

(3) Registers

For the settings of the registers used in this function, see **Figure 23.57, Block Diagram of PIC2B**, and **Section 23.3.2.4, PIC2BADTEN4nx — A/D Converter Trigger Output Select Control Register**.

The values of PIC2B registers used in this function are as follows.

PIC2BADTEN40x = (set any value)

PIC2BADTEN41x = (set any value)

PIC2BADCC0TSELx [1:0] = 00_B

x = 1, 2

(4) Function

The interrupt signal of each channel of TAUD is masked and OR'ed, and output to TAPA as the TAUD trigger signal.

CAUTION

This function is only available for the scan groups 1 and 2. When using this function with the scan groups 1 and 2, select the TAUD trigger via TAPA. Do not select the TAUD trigger directly by the ADCC trigger select function.

(5) Flow Chart

Set this function before starting A/D converter.

Section 24 Enhanced Motor Control Unit (EMU2)

The enhanced motor control unit 2 (EMU2) can perform motor control with reduced CPU load by using combined with the A/D converter (ADC), R/D converter (RDC), and TSG3. With the two EMU2 channels incorporated, two motors can be simultaneously controlled.

24.1 Features of RH850/C1x EMU2

24.1.1 Units

This LSI has the following number of EMU2 units.

Table 24.1 Units

Product	RH850/C1x
Number of units	1 (2 channels)
Name	EMU2n (n = 0, 1)

Table 24.2 Index

Index	Meaning
n	Throughout this section, the individual EMU2 channels are identified by the index “n” (n = 0, 1), for example, EMU2nPRT for the EMU2n protect register.
i	Throughout this section, the individual EMU2n interrupts are identified by the index “i”(i = 0 to 4).
m	Throughout this section, the individual ADCC units are identified by the index “m” (m = 0,1).
k	Throughout this section, the individual ADCC virtual channels are identified by the index “k” (m = 0 to 2).

24.1.2 Register Base Addresses

EMU2 base addresses are listed in the following table.

EMU2 register addresses are given as offsets from the base addresses in general.

Table 24.3 Register Base Addresses

Base Address Name	Base Address
<EMU20_base>	FFF9 9000_H
<EMU21_base>	FFF9 $A000_H$

24.1.3 Clock Supply

Clock supply by and to EMU2 is listed in the following table.

Table 24.4 TAUDn Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
EMU2	PCLK	CLK_HSB (High-speed peripheral clock)
	CCLK	CLKC_HSB (Unmodulated high-speed peripheral clock)

24.1.4 Interrupt Requests

The EMU2 has EMU2n interrupts i (n = 0, 1) (i = 0 to 4), each of which is allocated to the separate vector address. Interrupt sources can be selected using the EMU2nINTi register.

When an EMU2n interrupt occurs, the corresponding interrupt request is output. If the interrupt request is used to invoke the CPU interrupt, the priority can be changed using the interrupt controller. For details, refer to the section of Interrupts (INTC).

All of the EMU2n interrupts 0, 1, 2, 3, and 4 can activate the DMAC.

Table 24.5 lists the EMU2n interrupt source select registers, and **Table 24.6** lists the sources that can be selected for the EMU2n interrupts.

Table 24.5 EMU2n Interrupt Source Select Registers

Interrupt Name	Interrupt Source Select Register	Interrupt Number	DMA Trigger Number	DTS Trigger Number
EMU20 interrupt 0	EMU20 interrupt source select register 0	56	96	78
EMU20 interrupt 1	EMU20 interrupt source select register 1	57	97	79
EMU20 interrupt 2	EMU20 interrupt source select register 2	58	98	80
EMU20 interrupt 3	EMU20 interrupt source select register 3	59	99	81
EMU20 interrupt 4	EMU20 interrupt source select register 4	60	100	82
EMU21 interrupt 0	EMU21 interrupt source select register 0	61	101	83
EMU21 interrupt 1	EMU21 interrupt source select register 1	62	102	84
EMU21 interrupt 2	EMU21 interrupt source select register 2	63	103	85
EMU21 interrupt 3	EMU21 interrupt source select register 3	64	104	86
EMU21 interrupt 4	EMU21 interrupt source select register 4	65	105	87

Table 24.6 Sources that can be Selected for EMU2n Interrupts

Register Name	Selectable Interrupt Sources
EMU20 interrupt source select register 0	Selectable from the following: Speed measurement timer overflow, independent rectangle W-phase compare match, independent rectangle V-phase compare match, independent rectangle U-phase compare match, angle compare 1 match, angle compare 0 match, carrier counter, verification buffering completion, equivalence check error, input IP completion, PI control IP completion, PWM IP completion, and rectangle IP completion
EMU20 interrupt source select register 1	
EMU20 interrupt source select register 2	
EMU20 interrupt source select register 3	
EMU20 interrupt source select register 4	
EMU21 interrupt source select register 0	Selectable from the following: Speed measurement timer overflow, independent rectangle W-phase compare match, independent rectangle V-phase compare match, independent rectangle U-phase compare match, angle compare 1 match, angle compare 0 match, carrier counter, verification buffering completion, equivalence check error, input IP completion, PI control IP completion, PWM IP completion, and rectangle IP completion
EMU21 interrupt source select register 1	
EMU21 interrupt source select register 2	
EMU21 interrupt source select register 3	
EMU21 interrupt source select register 4	

24.1.5 Reset Sources

EMU2 reset sources are listed in the following table.

Table 24.7 Reset Sources

Unit Name	Reset Source
EMU2	Any reset source

24.2 Overview

24.2.1 Functional Overview

This module can calculate the duty cycle of the 3-phase PWM waveform or generate the rectangular wave pattern at a high speed simply by hardware, based on the current measured by the A/D converter and the motor angle value obtained by the R/D converter. The TSG3 then outputs the PWM wave or rectangular wave based on the PWM duty cycle or rectangular wave pattern generated by this module.

This module has several IPs for calculation. An interrupt can be generated upon process completion of each IP and upon a compare match, which provides flexible control by letting the CPU perform some of the calculations.

Table 24.8 lists the EMU2 specifications, **Table 24.9** lists the maximum processing time of each IP in the calculation block, and **Figure 24.1** shows the overall configuration of the EMU2.

Table 24.8 EMU2 Specifications

Item	Function	Description
Input block	Inputs the U/V/W-phase currents from A/D.	Obtains the A/D conversion results of the U/V/W-phase currents.
	Inputs the resolver angle from R/D.	Obtains the R/D conversion results of the resolver angle and Z phase.
	Inputs the carrier crest and trough triggers from TSG3.	Obtains the crest and trough trigger signals of the carrier counter from TSG3.
	Speed measurement timer	Measures the interval between the Z-phase pulses.
Calculation block		Provided with a dedicated 16-/32-bit calculator allowing parallel processing with CPU.
	Angle generation function (Angle generation IP).	Generates the electrical angle based on the R/D converter angle data, and detects a compare match.
	Input data calculation function (Input IP).	Performs dq transformation based on the motor current and electrical angle.
	PI control function (PI control IP).	Performs PI control based on the dq-axis currents.
	PWM values calculation function (PWM IP).	Performs 3-phase transformation based on the dq-axis voltages and electrical angle to obtain the PWM duty cycle.
	Rectangular wave pattern calculation function (Rectangle IP).	Calculates the rectangular wave output level and compare value based on the voltage phase value.
Rectangular wave generation block	Batch rectangular wave control function (Batch rectangle IP).	Switches the rectangular wave output patterns of all U/V/W phases simultaneously, using the match between the compare value and electrical angle detected by the angle generation IP as a trigger.
	Independent rectangular wave control function (Independent rectangle IP).	Switches the rectangular wave output patterns of each U/V/W phase independently according to the three compare values and output pattern values specified for each of U/V/W phases.
Verification function		Verifies the calculated result by using the buffering function or the equivalence check function.
Interrupts/DMA		Interrupt can be generated and DMA transfer can be started by the following sources: • Input IP completion, PI control IP completion, PWM IP completion, and rectangle IP completion • Verification buffering completion and equivalence check error • Carrier signal (either crest or trough, or both can be selected) • Angle compare 0 match and angle compare 1 match • Independent rectangle U-phase angle compare match, independent rectangle V-phase angle compare match, independent rectangle W-phase angle compare match • Speed measurement timer overflow interrupt

Table 24.9 Maximum Processing Time of each IP in Calculation Block

Input IP	PI Control IP	PWM IP	Rectangle IP	Angle Generation IP
0.6µs	0.2µs	0.8µs	0.2µs	0.4µs

Note: The startup time and data transfer time for each IP are excluded.

24.2.2 Block Diagram

The following figure shows the block diagram of the main components of EMU2.

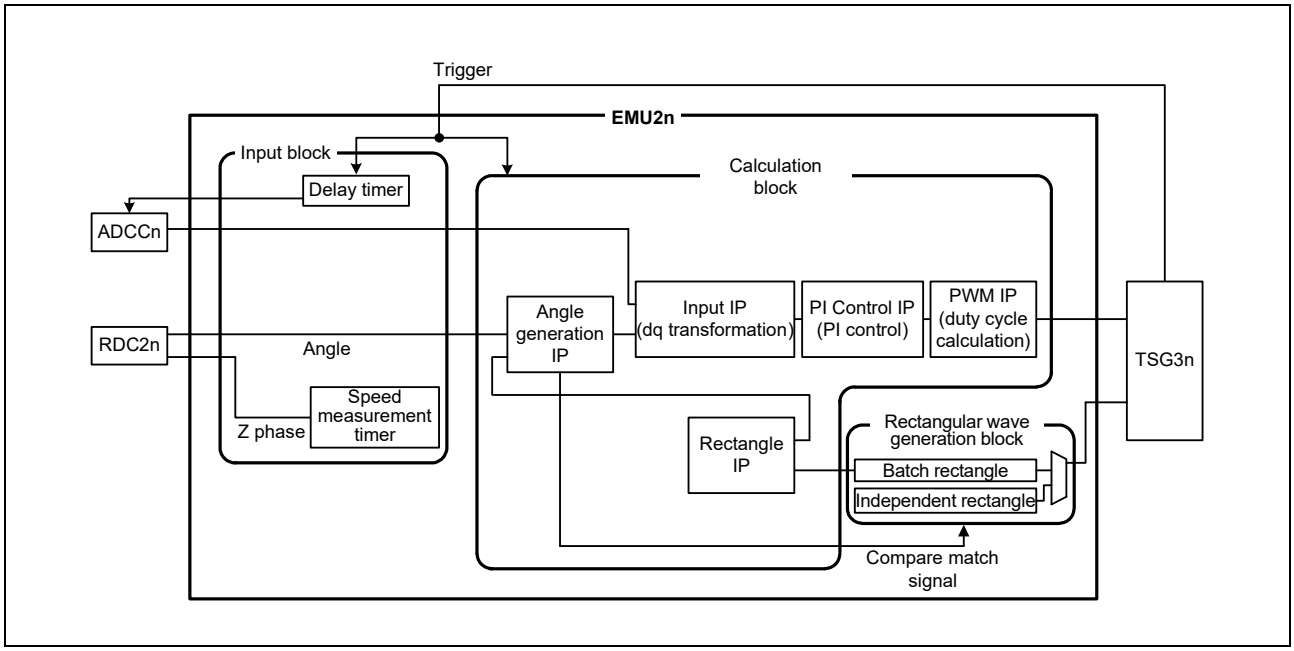


Figure 24.1 Overall Configuration of EMU2

The EMU2 consists of the input block, which controls the input signals from the various sensors; calculation block, which performs calculation for motor control based on the input data; and rectangular wave generation block, which generates rectangular waves.

In the EMU2, the input block receives the motor current from the A/D converter and the motor angle value from R/D converter; the calculation block calculates the duty cycle of the 3-phase PWM waveform based on those received values. Then, the TSG3 outputs the 3-phase PWM waveform based on the duty cycle. Alternatively, it is possible that the switching timing and levels of the rectangular wave are calculated based on the input data, and that the rectangular wave generation block generates the rectangular wave using the compare match signal detected by the angle generation IP as a trigger. Then the TSG3 outputs the rectangular wave with dead time.

24.3 Registers

24.3.1 List of Registers

Table 24.10 to **Table 24.25** list the EMU2 registers. **Table 24.26** to **Table 24.33** list the EMU2 register functions. Addresses to which no register is allocated are reserved; do not access them.

For <EMU2n_base>, see **Section 24.1.2, Register Base Addresses**.

Table 24.10 List of EMU2 Channel 0 Registers (General)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 protection register	EMU20PRT	<EMU20_base> + 0000_H
EMU20	EMU20 control register	EMU20CTR	<EMU20_base> + 0001_H
EMU20	EMU20 register value reflection control register	EMU20REFCTR	<EMU20_base> + 0002_H
EMU20	EMU20 IP activation trigger source select register	EMU20IPTRG	<EMU20_base> + 0004_H
EMU20	EMU20 IP software activation register	EMU20IPSFT	<EMU20_base> + 0005_H
EMU20	EMU20 A/D conversion trigger select register	EMU20ADTRG	<EMU20_base> + 0008_H
EMU20	EMU20 A/D conversion trigger source determination register	EMU20ADMON	<EMU20_base> + 0009_H
EMU20	EMU20 A/D conversion trigger source determination clear register	EMU20ADMONC	<EMU20_base> + $000A_H$
EMU20	EMU20 data delay counter value setting register	EMU20DDCNT	<EMU20_base> + $000C_H$
EMU20	EMU20 interrupt source select register 0	EMU20INT0	<EMU20_base> + 0010_H
EMU20	EMU20 interrupt source select register 1	EMU20INT1	<EMU20_base> + 0012_H
EMU20	EMU20 interrupt source select register 2	EMU20INT2	<EMU20_base> + 0014_H
EMU20	EMU20 interrupt source select register 3	EMU20INT3	<EMU20_base> + 0016_H
EMU20	EMU20 interrupt source select register 4	EMU20INT4	<EMU20_base> + 0018_H
EMU20	EMU20 interrupt source determination register	EMU20INTSD	<EMU20_base> + $001C_H$
EMU20	EMU20 interrupt source determination clear register	EMU20INTSDC	<EMU20_base> + $001E_H$
EMU20	EMU20 overflow detection result register	EMU20OFMON	<EMU20_base> + 0020_H
EMU20	EMU20 zero division detection result register	EMU20ZDMON	<EMU20_base> + 0021_H
EMU20	EMU20 overflow detection result clear register	EMU20OFMONC	<EMU20_base> + 0022_H
EMU20	EMU20 zero division detection result clear register	EMU20ZDMONC	<EMU20_base> + 0023_H
EMU20	EMU20 equivalence check function control register	EMU20SMLCTR	<EMU20_base> + 0028_H
EMU20	EMU20 verification buffer control register 0	EMU20CBCTR0	<EMU20_base> + $002C_H$
EMU20	EMU20 verification buffer control register 1	EMU20CBCTR1	<EMU20_base> + $002D_H$
EMU20	EMU20 verification buffer timing select register	EMU20CBTIM	<EMU20_base> + $002E_H$

Table 24.11 List of EMU2 Channel 0 Registers (Angle Generation IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 angle generation IP control register	EMU20ANGCTR	<EMU20_base> + 0040_H
EMU20	EMU20 compare judgment correction register 0	EMU20CPJUD0	<EMU20_base> + 0042_H
EMU20	EMU20 compare judgment correction register 1	EMU20CPJUD1	<EMU20_base> + 0043_H
EMU20	EMU20 resolver angle software input register	EMU20RESTHSFT	<EMU20_base> + 0044_H
EMU20	EMU20 angle generation IP offset register	EMU20ANGOFS	<EMU20_base> + 0046_H
EMU20	EMU20 electrical angle generation coefficient register	EMU20PXR	<EMU20_base> + 0048_H
EMU20	EMU20 resolver angle register	EMU20RESTHETA	<EMU20_base> + $004A_H$
EMU20	EMU20 electrical angle register	EMU20THTEFIX	<EMU20_base> + $004C_H$
EMU20	EMU20 resolver pole number setting register	EMU20RESRLD	<EMU20_base> + $004E_H$
EMU20	EMU20 resolver pole count register	EMU20RESCNT	<EMU20_base> + $004F_H$
EMU20	EMU20 speed measurement timer control register	EMU20VMTCTR	<EMU20_base> + 0054_H
EMU20	EMU20 speed measurement timer counter register	EMU20VMTCNT	<EMU20_base> + 0058_H
EMU20	EMU20 speed measurement timer capture register	EMU20VMTCAP	<EMU20_base> + $005C_H$
EMU20	EMU20 speed measurement timer overflow register	EMU20VMTOF	<EMU20_base> + 0060_H

Table 24.12 List of EMU2 Channel 0 Registers (Input IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 input IP control register	EMU20CTRINMD	<EMU20_base> + 0080_H
EMU20	EMU20 AD0 data register 0	EMU20AD00	<EMU20_base> + 0084_H
EMU20	EMU20 AD0 channel 0 origin correction value register	EMU20ADDOFS00	<EMU20_base> + 0086_H
EMU20	EMU20 AD0 data register 1	EMU20AD01	<EMU20_base> + 0088_H
EMU20	EMU20 AD0 channel 1 origin correction value register	EMU20ADDOFS01	<EMU20_base> + $008A_H$
EMU20	EMU20 AD0 data register 2	EMU20AD02	<EMU20_base> + $008C_H$
EMU20	EMU20 AD0 channel 2 origin correction value register	EMU20ADDOFS02	<EMU20_base> + $008E_H$
EMU20	EMU20 AD0 channel 0 conversion value register	EMU20ADFIX00	<EMU20_base> + 0090_H
EMU20	EMU20 electrical angle software input register	EMU20THTESFT	<EMU20_base> + 0092_H
EMU20	EMU20 AD0 channel 1 conversion value register	EMU20ADFIX01	<EMU20_base> + 0094_H
EMU20	EMU20 electrical angle response delay correction variable register	EMU20EARD	<EMU20_base> + 0096_H
EMU20	EMU20 AD0 channel 2 conversion value register	EMU20ADFIX02	<EMU20_base> + 0098_H
EMU20	EMU20 electrical angle retention register	EMU20THTE	<EMU20_base> + $009A_H$
EMU20	EMU20 resolver angle monitor register	EMU20THTREFIXIN	<EMU20_base> + $009C_H$
EMU20	EMU20 dq-axis current transformation coefficient register	EMU20SR2	<EMU20_base> + $00A0_H$
EMU20	EMU20 LSB adjustment register	EMU20DIVLSB	<EMU20_base> + $00A4_H$
EMU20	EMU20 U-phase current value register	EMU20IUFIX	<EMU20_base> + $00A8_H$
EMU20	EMU20 V-phase current value register	EMU20IVFIX	<EMU20_base> + $00AC_H$
EMU20	EMU20 W-phase current value register	EMU20IWFIX	<EMU20_base> + $00B0_H$
EMU20	EMU20 d-axis current value register	EMU20IDFIX	<EMU20_base> + $00B4_H$
EMU20	EMU20 q-axis current value register	EMU20IQFIX	<EMU20_base> + $00B8_H$

Table 24.13 List of EMU2 Channel 0 Registers (PI control IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 PI control register	EMU20PICTR	<EMU20_base> + $00C0_H$
EMU20	EMU20 d-axis current target value register	EMU20IDIN	<EMU20_base> + $00C4_H$
EMU20	EMU20 q-axis current target value register	EMU20IQIN	<EMU20_base> + $00C8_H$
EMU20	EMU20 id register	EMU20ID	<EMU20_base> + $00CC_H$
EMU20	EMU20 iq register	EMU20IQ	<EMU20_base> + $00D0_H$
EMU20	EMU20 GPD0 register	EMU20GPD0	<EMU20_base> + $00D4_H$
EMU20	EMU20 GPQ0 register	EMU20GPQ0	<EMU20_base> + $00D8_H$
EMU20	EMU20 GPD register	EMU20GPD	<EMU20_base> + $00DC_H$
EMU20	EMU20 GPQ register	EMU20GPQ	<EMU20_base> + $00E0_H$
EMU20	EMU20 GID register	EMU20GID	<EMU20_base> + $00E4_H$
EMU20	EMU20 GIQ register	EMU20GIQ	<EMU20_base> + $00E8_H$
EMU20	EMU20 GID_MAX register	EMU20GIDMAX	<EMU20_base> + $00EC_H$
EMU20	EMU20 GIQ_MAX register	EMU20GIQMAX	<EMU20_base> + $00F0_H$
EMU20	EMU20 vd_MAX register	EMU20VDMAX	<EMU20_base> + $00F4_H$
EMU20	EMU20 vq_MAX register	EMU20VQMAX	<EMU20_base> + $00F8_H$
EMU20	EMU20 sum_id register	EMU20SUMID	<EMU20_base> + 0100_H
EMU20	EMU20 sum_id monitor register	EMU20SUMIDM	<EMU20_base> + 0104_H
EMU20	EMU20 sum_iq register	EMU20SUMIQ	<EMU20_base> + 0108_H
EMU20	EMU20 sum_iq monitor register	EMU20SUMIQM	<EMU20_base> + $010C_H$
EMU20	EMU20 d-axis voltage register	EMU20VD	<EMU20_base> + 0110_H
EMU20	EMU20 q-axis voltage register	EMU20VQ	<EMU20_base> + 0114_H

Table 24.14 List of EMU2 Channel 0 Registers (PWM IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 PWM IP control register	EMU20PWMCTR	<EMU20_base> + 0120_H
EMU20	EMU20 PWM data transfer register	EMU20PWMDT	<EMU20_base> + 0122_H
EMU20	EMU20 d-axis voltage correction value register	EMU20VDCRCT	<EMU20_base> + 0124_H
EMU20	EMU20 q-axis voltage correction value register	EMU20VQCRCT	<EMU20_base> + 0128_H
EMU20	EMU20 three-phase voltage transformation coefficient register	EMU20SR23	<EMU20_base> + $012C_H$
EMU20	EMU20 U-phase voltage offset register	EMU20UVOFS	<EMU20_base> + 0130_H
EMU20	EMU20 W-phase voltage offset register	EMU20WVOFS	<EMU20_base> + 0134_H
EMU20	EMU20 d-axis reference voltage register	EMU20PHI	<EMU20_base> + 0138_H
EMU20	EMU20 electrical angle adjustment register	EMU20GTHT	<EMU20_base> + $013C_H$
EMU20	EMU20 predicted electrical angle software input register	EMU20THTFORESFT	<EMU20_base> + $013E_H$
EMU20	EMU20 digit alignment register 1	EMU20PWMK1	<EMU20_base> + 0140_H
EMU20	EMU20 digit alignment register 2	EMU20PWMK2	<EMU20_base> + 0144_H
EMU20	EMU20 input voltage register	EMU20VOLV	<EMU20_base> + 0146_H
EMU20	EMU20 duty cycle upper limit register	EMU20DTUL	<EMU20_base> + 0148_H
EMU20	EMU20 duty cycle lower limit register	EMU20DTLL	<EMU20_base> + $014C_H$
EMU20	EMU20 PWM upper limit register	EMU20PWMUL	<EMU20_base> + 0150_H
EMU20	EMU20 PWM lower limit register	EMU20PWMLL	<EMU20_base> + 0152_H
EMU20	EMU20 dead time setting register	EMU20DTT	<EMU20_base> + 0154_H
EMU20	EMU20 carrier cycle register	EMU20CARR	<EMU20_base> + 0156_H
EMU20	EMU20 U-phase PWM register	EMU20UPWM	<EMU20_base> + 0158_H
EMU20	EMU20 V-phase PWM register	EMU20VPWM	<EMU20_base> + $015A_H$
EMU20	EMU20 W-phase PWM register	EMU20WPWM	<EMU20_base> + $015C_H$
EMU20	EMU20 U-phase output voltage for duty cycle calculation register	EMU20VUFIX	<EMU20_base> + 0160_H
EMU20	EMU20 V-phase output voltage for duty cycle calculation register	EMU20VVFIX	<EMU20_base> + 0164_H
EMU20	EMU20 W-phase output voltage for duty cycle calculation register	EMU20VWFIX	<EMU20_base> + 0168_H
EMU20	EMU20 U-phase compare value register	EMU20PWMUIP	<EMU20_base> + $016C_H$
EMU20	EMU20 V-phase compare value register	EMU20PWMVIP	<EMU20_base> + $016E_H$
EMU20	EMU20 W-phase compare value register	EMU20PWMWIP	<EMU20_base> + 0170_H
EMU20	EMU20 U-phase output voltage correction amount register	EMU20VUOFS	<EMU20_base> + 0174_H
EMU20	EMU20 V-phase output voltage correction amount register	EMU20VVOFS	<EMU20_base> + 0176_H
EMU20	EMU20 W-phase output voltage correction amount register	EMU20VWOFS	<EMU20_base> + 0178_H

Table 24.15 List of EMU2 Channel 0 Registers (Rectangle IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 rectangle IP control register	EMU20RECCTR	<EMU20_base> + 0180_H
EMU20	EMU20 batch rectangle output register	EMU20PTNN	<EMU20_base> + 0184_H
EMU20	EMU20 batch rectangle output pattern AB register	EMU20PTNAB	<EMU20_base> + 0185_H
EMU20	EMU20 batch rectangle output pattern CD register	EMU20PTNCD	<EMU20_base> + 0186_H
EMU20	EMU20 batch rectangle output pattern EF register	EMU20PTNEF	<EMU20_base> + 0187_H
EMU20	EMU20 compare register 0	EMU20CMP0	<EMU20_base> + 0188_H
EMU20	EMU20 compare register 1	EMU20CMP1	<EMU20_base> + $018A_H$
EMU20	EMU20 q-axis reference voltage phase software input register	EMU20PHQSFT	<EMU20_base> + $018C_H$
EMU20	EMU20 phase differential software input register	EMU20PSWSFT	<EMU20_base> + $018E_H$
EMU20	EMU20 phase differential register	EMU20PSW	<EMU20_base> + $018F_H$
EMU20	EMU20 IP compare value 0 register	EMU20IPCMP0	<EMU20_base> + 0190_H

Table 24.16 List of EMU2 Channel 0 Registers (Independent Rectangle IP)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 independent rectangle IP control register	EMU20IRECCTR	<EMU20_base> + 0198_H
EMU20	EMU20 independent rectangle output pattern update register	EMU20IRPTN	<EMU20_base> + $019C_H$
EMU20	EMU20 independent rectangle IP flag/select signal reset register	EMU20IRCTRST	<EMU20_base> + $019E_H$
EMU20	EMU20 independent rectangle IP U-phase compare/pattern setting register 0	EMU20IRUCPPN0	<EMU20_base> + $01A0_H$
EMU20	EMU20 independent rectangle IP U-phase compare/pattern setting register 1	EMU20IRUCPPN1	<EMU20_base> + $01A2_H$
EMU20	EMU20 independent rectangle IP U-phase compare/pattern setting register 2	EMU20IRUCPPN2	<EMU20_base> + $01A4_H$
EMU20	EMU20 independent rectangle IP V-phase compare/pattern setting register 0	EMU20IRVCPPN0	<EMU20_base> + $01A8_H$
EMU20	EMU20 independent rectangle IP V-phase compare/pattern setting register 1	EMU20IRVCPPN1	<EMU20_base> + $01AA_H$
EMU20	EMU20 independent rectangle IP V-phase compare/pattern setting register 2	EMU20IRVCPPN2	<EMU20_base> + $01AC_H$
EMU20	EMU20 independent rectangle IP W-phase compare/pattern setting register 0	EMU20IRWCPPN0	<EMU20_base> + $01B0_H$
EMU20	EMU20 independent rectangle IP W-phase compare/pattern setting register 1	EMU20IRWCPPN1	<EMU20_base> + $01B2_H$
EMU20	EMU20 independent rectangle IP W-phase compare/pattern setting register 2	EMU20IRWCPPN2	<EMU20_base> + $01B4_H$
EMU20	EMU20 independent rectangle IP flag monitor register	EMU20IRFLGM	<EMU20_base> + $01B8_H$
EMU20	EMU20 independent rectangle IP select signal monitor register	EMU20IRSELM	<EMU20_base> + $01BA_H$

Table 24.17 List of EMU2 Channel 0 Registers (Verification Function)

Module Name	Register Name	Symbol	Address
EMU20	EMU20 AD0 data 0 verification buffer register	EMU20CBAD00	<EMU20_base> + $01C0_H$
EMU20	EMU20 AD0 data 1 verification buffer register	EMU20CBAD01	<EMU20_base> + $01C2_H$
EMU20	EMU20 AD0 data 2 verification buffer register	EMU20CBAD02	<EMU20_base> + $01C4_H$
EMU20	EMU20 resolver angle verification buffer register	EMU20CBTHTREFIXIN	<EMU20_base> + $01C6_H$
EMU20	EMU20 d-axis current value verification buffer register	EMU20CBIDFIX	<EMU20_base> + $01C8_H$
EMU20	EMU20 q-axis current value verification buffer register	EMU20CBIQFIX	<EMU20_base> + $01CC_H$
EMU20	EMU20 U-phase compare value verification buffer register	EMU20CBPWMUIP	<EMU20_base> + $01D0_H$
EMU20	EMU20 V-phase compare value verification buffer register	EMU20CBPWMVIP	<EMU20_base> + $01D4_H$
EMU20	EMU20 W-phase compare value verification buffer register	EMU20CBPWMWIP	<EMU20_base> + $01D8_H$
EMU20	EMU20 batch rectangle pattern verification buffer register	EMU20CBBREC	<EMU20_base> + $01DC_H$
EMU20	EMU20 independent rectangle pattern verification buffer register	EMU20CBIREC	<EMU20_base> + $01DD_H$

Table 24.18 List of EMU2 Channel 1 Registers (General)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 protection register	EMU21PRT	<EMU21_base> + 0000_H
EMU21	EMU21 control register	EMU21CTR	<EMU21_base> + 0001_H
EMU21	EMU21 register value reflection control register	EMU21REFCTR	<EMU21_base> + 0002_H
EMU21	EMU21 IP activation trigger source select register	EMU21IPTRG	<EMU21_base> + 0004_H
EMU21	EMU21 IP software activation register	EMU21IPSFT	<EMU21_base> + 0005_H
EMU21	EMU21 A/D conversion trigger select register	EMU21ADTRG	<EMU21_base> + 0008_H
EMU21	EMU21 A/D conversion trigger source determination register	EMU21ADMON	<EMU21_base> + 0009_H
EMU21	EMU21 A/D conversion trigger source determination clear register	EMU21ADMONC	<EMU21_base> + $000A_H$
EMU21	EMU21 data delay counter value setting register	EMU21DDCNT	<EMU21_base> + $000C_H$
EMU21	EMU21 interrupt source select register 0	EMU21INT0	<EMU21_base> + 0010_H
EMU21	EMU21 interrupt source select register 1	EMU21INT1	<EMU21_base> + 0012_H
EMU21	EMU21 interrupt source select register 2	EMU21INT2	<EMU21_base> + 0014_H
EMU21	EMU21 interrupt source select register 3	EMU21INT3	<EMU21_base> + 0016_H
EMU21	EMU21 interrupt source select register 4	EMU21INT4	<EMU21_base> + 0018_H
EMU21	EMU21 interrupt source determination register	EMU21INTSD	<EMU21_base> + $001C_H$
EMU21	EMU21 interrupt source determination clear register	EMU21INTSDC	<EMU21_base> + $001E_H$
EMU21	EMU21 overflow detection result register	EMU21OFMON	<EMU21_base> + 0020_H
EMU21	EMU21 zero division detection result register	EMU21ZDMON	<EMU21_base> + 0021_H
EMU21	EMU21 overflow detection result clear register	EMU21OFMONC	<EMU21_base> + 0022_H
EMU21	EMU21 zero division detection result clear register	EMU21ZDMONC	<EMU21_base> + 0023_H
EMU21	EMU21 equivalence check function control register	EMU21SMLCTR	<EMU21_base> + 0028_H
EMU21	EMU21 verification buffer control register 0	EMU21CBCTR0	<EMU21_base> + $002C_H$
EMU21	EMU21 verification buffer control register 1	EMU21CBCTR1	<EMU21_base> + $002D_H$
EMU21	EMU21 verification buffer timing select register	EMU21CBTIM	<EMU21_base> + $002E_H$

Table 24.19 List of EMU2 Channel 1 Registers (Angle Generation IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 angle generation IP control register	EMU21ANGCTR	<EMU21_base> + 0040_H
EMU21	EMU21 compare judgment correction register 0	EMU21CPJUD0	<EMU21_base> + 0042_H
EMU21	EMU21 compare judgment correction register 1	EMU21CPJUD1	<EMU21_base> + 0043_H
EMU21	EMU21 resolver angle software input register	EMU21RESTHSFT	<EMU21_base> + 0044_H
EMU21	EMU21 angle generation IP offset register	EMU21ANGOFS	<EMU21_base> + 0046_H
EMU21	EMU21 electrical angle generation coefficient register	EMU21PXR	<EMU21_base> + 0048_H
EMU21	EMU21 resolver angle register	EMU21RESTHETA	<EMU21_base> + $004A_H$
EMU21	EMU21 electrical angle register	EMU21THTEFIX	<EMU21_base> + $004C_H$
EMU21	EMU21 resolver pole number setting register	EMU21RESRLD	<EMU21_base> + $004E_H$
EMU21	EMU21 resolver pole count register	EMU21RESCNT	<EMU21_base> + $004F_H$
EMU21	EMU21 speed measurement timer control register	EMU21VMTCTR	<EMU21_base> + 0054_H
EMU21	EMU21 speed measurement timer counter register	EMU21VMTCNT	<EMU21_base> + 0058_H
EMU21	EMU21 speed measurement timer capture register	EMU21VMTCAP	<EMU21_base> + $005C_H$
EMU21	EMU21 speed measurement timer overflow register	EMU21VMTOF	<EMU21_base> + 0060_H

Table 24.20 List of EMU2 Channel 1 Registers (Input IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 input IP control register	EMU21CTRINMD	<EMU21_base> + 0080_H
EMU21	EMU21 AD1 data register 0	EMU21AD10	<EMU21_base> + 0084_H
EMU21	EMU21 AD1 channel 0 origin correction value register	EMU21ADDOFS10	<EMU21_base> + 0086_H
EMU21	EMU21 AD1 data register 1	EMU21AD11	<EMU21_base> + 0088_H
EMU21	EMU21 AD1 channel 1 origin correction value register	EMU21ADDOFS11	<EMU21_base> + $008A_H$
EMU21	EMU21 AD1 data register 2	EMU21AD12	<EMU21_base> + $008C_H$
EMU21	EMU21 AD1 channel 2 origin correction value register	EMU21ADDOFS12	<EMU21_base> + $008E_H$
EMU21	EMU21 AD1 channel 0 conversion value register	EMU21ADFIX10	<EMU21_base> + 0090_H
EMU21	EMU21 electrical angle software input register	EMU21THTESFT	<EMU21_base> + 0092_H
EMU21	EMU21 AD1 channel 1 conversion value register	EMU21ADFIX11	<EMU21_base> + 0094_H
EMU21	EMU21 electrical angle response delay correction variable register	EMU21EARD	<EMU21_base> + 0096_H
EMU21	EMU21 AD1 channel 2 conversion value register	EMU21ADFIX12	<EMU21_base> + 0098_H
EMU21	EMU21 electrical angle retention register	EMU21THTE	<EMU21_base> + $009A_H$
EMU21	EMU21 resolver angle monitor register	EMU21THTREFIXIN	<EMU21_base> + $009C_H$
EMU21	EMU21 dq-axis current transformation coefficient register	EMU21SR2	<EMU21_base> + $00A0_H$
EMU21	EMU21 LSB adjustment register	EMU21DIVLSB	<EMU21_base> + $00A4_H$
EMU21	EMU21 U-phase current value register	EMU21IUFIX	<EMU21_base> + $00A8_H$
EMU21	EMU21 V-phase current value register	EMU21IVFIX	<EMU21_base> + $00AC_H$
EMU21	EMU21 W-phase current value register	EMU21IWFIX	<EMU21_base> + $00B0_H$
EMU21	EMU21 d-axis current value register	EMU21IDFIX	<EMU21_base> + $00B4_H$
EMU21	EMU21 q-axis current value register	EMU21IQFIX	<EMU21_base> + $00B8_H$

Table 24.21 List of EMU2 Channel 1 Registers (PI control IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 PI control register	EMU21PICTR	<EMU21_base> + $00C0_H$
EMU21	EMU21 d-axis current target value register	EMU21IDIN	<EMU21_base> + $00C4_H$
EMU21	EMU21 q-axis current target value register	EMU21IQIN	<EMU21_base> + $00C8_H$
EMU21	EMU21 id register	EMU21ID	<EMU21_base> + $00CC_H$
EMU21	EMU21 iq register	EMU21IQ	<EMU21_base> + $00D0_H$
EMU21	EMU21 GPD0 register	EMU21GPD0	<EMU21_base> + $00D4_H$
EMU21	EMU21 GPQ0 register	EMU21GPQ0	<EMU21_base> + $00D8_H$
EMU21	EMU21 GPD register	EMU21GPD	<EMU21_base> + $00DC_H$
EMU21	EMU21 GPQ register	EMU21GPQ	<EMU21_base> + $00E0_H$
EMU21	EMU21 GID register	EMU21GID	<EMU21_base> + $00E4_H$
EMU21	EMU21 GIQ register	EMU21GIQ	<EMU21_base> + $00E8_H$
EMU21	EMU21 GID_MAX register	EMU21GIDMAX	<EMU21_base> + $00EC_H$
EMU21	EMU21 GIQ_MAX register	EMU21GIQMAX	<EMU21_base> + $00F0_H$
EMU21	EMU21 vd_MAX register	EMU21VDMAX	<EMU21_base> + $00F4_H$
EMU21	EMU21 vq_MAX register	EMU21VQMAX	<EMU21_base> + $00F8_H$
EMU21	EMU21 sum_id register	EMU21SUMID	<EMU21_base> + 0100_H
EMU21	EMU21 sum_id monitor register	EMU21SUMIDM	<EMU21_base> + 0104_H
EMU21	EMU21 sum_iq register	EMU21SUMIQ	<EMU21_base> + 0108_H
EMU21	EMU21 sum_iq monitor register	EMU21SUMIQM	<EMU21_base> + $010C_H$
EMU21	EMU21 d-axis voltage register	EMU21VD	<EMU21_base> + 0110_H
EMU21	EMU21 q-axis voltage register	EMU21VQ	<EMU21_base> + 0114_H

Table 24.22 List of EMU2 Channel 1 Registers (PWM IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 PWM IP control register	EMU21PWMCTR	<EMU21_base> + 0120_H
EMU21	EMU21 PWM data transfer register	EMU21PWMDT	<EMU21_base> + 0122_H
EMU21	EMU21 d-axis voltage correction value register	EMU21VDCRCT	<EMU21_base> + 0124_H
EMU21	EMU21 q-axis voltage correction value register	EMU21VQCRCT	<EMU21_base> + 0128_H
EMU21	EMU21 three-phase voltage transformation coefficient register	EMU21SR23	<EMU21_base> + $012C_H$
EMU21	EMU21 U-phase voltage offset register	EMU21UVOFS	<EMU21_base> + 0130_H
EMU21	EMU21 W-phase voltage offset register	EMU21WVOFS	<EMU21_base> + 0134_H
EMU21	EMU21 d-axis reference voltage register	EMU21PHI	<EMU21_base> + 0138_H
EMU21	EMU21 electrical angle adjustment register	EMU21GTHT	<EMU21_base> + $013C_H$
EMU21	EMU21 predicted electrical angle software input register	EMU21THTFORESFT	<EMU21_base> + $013E_H$
EMU21	EMU21 digit alignment register 1	EMU21PWMK1	<EMU21_base> + 0140_H
EMU21	EMU21 digit alignment register 2	EMU21PWMK2	<EMU21_base> + 0144_H
EMU21	EMU21 input voltage register	EMU21VOLV	<EMU21_base> + 0146_H
EMU21	EMU21 duty cycle upper limit register	EMU21DTUL	<EMU21_base> + 0148_H
EMU21	EMU21 duty cycle lower limit register	EMU21DTLL	<EMU21_base> + $014C_H$
EMU21	EMU21 PWM upper limit register	EMU21PWMUL	<EMU21_base> + 0150_H
EMU21	EMU21 PWM lower limit register	EMU21PWMLL	<EMU21_base> + 0152_H
EMU21	EMU21 dead time setting register	EMU21DTT	<EMU21_base> + 0154_H
EMU21	EMU21 carrier cycle register	EMU21CARR	<EMU21_base> + 0156_H
EMU21	EMU21 U-phase PWM register	EMU21UPWM	<EMU21_base> + 0158_H
EMU21	EMU21 V-phase PWM register	EMU21VPWM	<EMU21_base> + $015A_H$
EMU21	EMU21 W-phase PWM register	EMU21WPWM	<EMU21_base> + $015C_H$
EMU21	EMU21 U-phase output voltage for duty cycle calculation register	EMU21VUFIX	<EMU21_base> + 0160_H
EMU21	EMU21 V-phase output voltage for duty cycle calculation register	EMU21VVFIX	<EMU21_base> + 0164_H
EMU21	EMU21 W-phase output voltage for duty cycle calculation register	EMU21VWFIX	<EMU21_base> + 0168_H
EMU21	EMU21 U-phase compare value register	EMU21PWMUIP	<EMU21_base> + $016C_H$
EMU21	EMU21 V-phase compare value register	EMU21PWMVIP	<EMU21_base> + $016E_H$
EMU21	EMU21 W-phase compare value register	EMU21PWMWIP	<EMU21_base> + 0170_H
EMU21	EMU21 U-phase output voltage correction amount register	EMU21VUOFS	<EMU21_base> + 0174_H
EMU21	EMU21 V-phase output voltage correction amount register	EMU21VVOFS	<EMU21_base> + 0176_H
EMU21	EMU21 W-phase output voltage correction amount register	EMU21VWOFS	<EMU21_base> + 0178_H

Table 24.23 List of EMU2 Channel 1 Registers (Rectangle IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 rectangle IP control register	EMU21RECCTR	<EMU21_base> + 0180_H
EMU21	EMU21 batch rectangle output register	EMU21PTNN	<EMU21_base> + 0184_H
EMU21	EMU21 batch rectangle output pattern AB register	EMU21PTNAB	<EMU21_base> + 0185_H
EMU21	EMU21 batch rectangle output pattern CD register	EMU21PTNCD	<EMU21_base> + 0186_H
EMU21	EMU21 batch rectangle output pattern EF register	EMU21PTNEF	<EMU21_base> + 0187_H
EMU21	EMU21 compare register 0	EMU21CMP0	<EMU21_base> + 0188_H
EMU21	EMU21 compare register 1	EMU21CMP1	<EMU21_base> + $018A_H$
EMU21	EMU21 q-axis reference voltage phase software input register	EMU21PHQSFT	<EMU21_base> + $018C_H$
EMU21	EMU21 phase differential software input register	EMU21PSWSFT	<EMU21_base> + $018E_H$
EMU21	EMU21 phase differential register	EMU21PSW	<EMU21_base> + $018F_H$
EMU21	EMU21 IP compare value 0 register	EMU21IPCMP0	<EMU21_base> + 0190_H

Table 24.24 List of EMU2 Channel 1 Registers (Independent Rectangle IP)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 independent rectangle IP control register	EMU21IRECCTR	<EMU21_base> + 0198_H
EMU21	EMU21 independent rectangle output pattern update register	EMU21IRPTN	<EMU21_base> + $019C_H$
EMU21	EMU21 independent rectangle IP flag/select signal reset register	EMU21IRCTRST	<EMU21_base> + $019E_H$
EMU21	EMU21 independent rectangle IP U-phase compare/pattern setting register 0	EMU21IRUCPPN0	<EMU21_base> + $01A0_H$
EMU21	EMU21 independent rectangle IP U-phase compare/pattern setting register 1	EMU21IRUCPPN1	<EMU21_base> + $01A2_H$
EMU21	EMU21 independent rectangle IP U-phase compare/pattern setting register 2	EMU21IRUCPPN2	<EMU21_base> + $01A4_H$
EMU21	EMU21 independent rectangle IP V-phase compare/pattern setting register 0	EMU21IRVCPPN0	<EMU21_base> + $01A8_H$
EMU21	EMU21 independent rectangle IP V-phase compare/pattern setting register 1	EMU21IRVCPPN1	<EMU21_base> + $01AA_H$
EMU21	EMU21 independent rectangle IP V-phase compare/pattern setting register 2	EMU21IRVCPPN2	<EMU21_base> + $01AC_H$
EMU21	EMU21 independent rectangle IP W-phase compare/pattern setting register 0	EMU21IRWCPPN0	<EMU21_base> + $01B0_H$
EMU21	EMU21 independent rectangle IP W-phase compare/pattern setting register 1	EMU21IRWCPPN1	<EMU21_base> + $01B2_H$
EMU21	EMU21 independent rectangle IP W-phase compare/pattern setting register 2	EMU21IRWCPPN2	<EMU21_base> + $01B4_H$
EMU21	EMU21 independent rectangle IP flag monitor register	EMU21IRFLGM	<EMU21_base> + $01B8_H$
EMU21	EMU21 independent rectangle IP select signal monitor register	EMU21IRSELM	<EMU21_base> + $01BA_H$

Table 24.25 List of EMU2 Channel 1 Registers (Verification Function)

Module Name	Register Name	Symbol	Address
EMU21	EMU21 AD1 data 0 verification buffer register	EMU21CBAD10	<EMU21_base> + $01C0_H$
EMU21	EMU21 AD1 data 1 verification buffer register	EMU21CBAD11	<EMU21_base> + $01C2_H$
EMU21	EMU21 AD1 data 2 verification buffer register	EMU21CBAD12	<EMU21_base> + $01C4_H$
EMU21	EMU21 resolver angle verification buffer register	EMU21CBTHTREFIXIN	<EMU21_base> + $01C6_H$
EMU21	EMU21 d-axis current value verification buffer register	EMU21CBIDFIX	<EMU21_base> + $01C8_H$
EMU21	EMU21 q-axis current value verification buffer register	EMU21CBIQFIX	<EMU21_base> + $01CC_H$
EMU21	EMU21 U-phase compare value verification buffer register	EMU21CBPWMUIP	<EMU21_base> + $01D0_H$
EMU21	EMU21 V-phase compare value verification buffer register	EMU21CBPWMVIP	<EMU21_base> + $01D4_H$
EMU21	EMU21 W-phase compare value verification buffer register	EMU21CBPWMWIP	<EMU21_base> + $01D8_H$
EMU21	EMU21 batch rectangle pattern verification buffer register	EMU21CBBREC	<EMU21_base> + $01DC_H$
EMU21	EMU21 independent rectangle pattern verification buffer register	EMU21CBIREC	<EMU21_base> + $01DD_H$

Table 24.26 List of EMU2 Register Functions (General)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer √: Provided, —: Not provided	Protection*1 √: Applicable, —: Not applicable
EMU2n	EMU2n protection register	EMU2nPRT	U8	—	—
EMU2n	EMU2n control register	EMU2nCTR	U8	—	√
EMU2n	EMU2n register value reflection control register	EMU2nREFCTR	U8	—	—
EMU2n	EMU2n IP activation trigger source select register	EMU2nIPTRG	U8	—	—
EMU2n	EMU2n IP software activation register	EMU2nIPSFT	U8	—	—
EMU2n	EMU2n A/D conversion trigger select register	EMU2nADTRG	U8	—	—
EMU2n	EMU2n A/D conversion trigger source determination register	EMU2nADMON	U8	—	—
EMU2n	EMU2n A/D conversion trigger source determination clear register	EMU2nADMONC	U8	—	—
EMU2n	EMU2n data delay counter value setting register	EMU2nDDCNT	U32	—	—
EMU2n	EMU2n interrupt source select register 0	EMU2nINT0	U16	—	—
EMU2n	EMU2n interrupt source select register 1	EMU2nINT1	U16	—	—
EMU2n	EMU2n interrupt source select register 2	EMU2nINT2	U16	—	—
EMU2n	EMU2n interrupt source select register 3	EMU2nINT3	U16	—	—
EMU2n	EMU2n interrupt source select register 4	EMU2nINT4	U16	—	—
EMU2n	EMU2n interrupt source determination register	EMU2nINTSD	U16	—	—
EMU2n	EMU2n interrupt source determination clear register	EMU2nINTSDC	U16	—	—
EMU2n	EMU2n overflow detection result register	EMU2nOFMON	U8	—	—
EMU2n	EMU2n zero division detection result register	EMU2nZDMON	U8	—	—
EMU2n	EMU2n overflow detection result clear register	EMU2nOFMONC	U8	—	—
EMU2n	EMU2n zero division detection result clear register	EMU2nZDMONC	U8	—	—
EMU2n	EMU2n equivalence check function control register	EMU2nSMLCTR	U8	—	—
EMU2n	EMU2n verification buffer control register 0	EMU2nCBCTR0	U8	—	—
EMU2n	EMU2n verification buffer control register 1	EMU2nCBCTR1	U8	—	—
EMU2n	EMU2n verification buffer timing select register	EMU2nCBTIM	U16	—	—

Note 1. To remove protection, refer to **Section 24.3.2**, EMU2n protection register (EMU2nPRT) (n = 0, 1).

Table 24.27 List of EMU2 Register Functions (Angle Generation IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n angle generation IP control register	EMU2nANGCTR	U8	—	—
EMU2n	EMU2n compare judgment correction register 0	EMU2nCPJUD0	U8	—	—
EMU2n	EMU2n compare judgment correction register 1	EMU2nCPJUD1	U8	—	—
EMU2n	EMU2n resolver angle software input register	EMU2nRESTHSFT	U12	—	√*2
EMU2n	EMU2n angle generation IP offset register	EMU2nANGOFS	S16	√	—
EMU2n	EMU2n electrical angle generation coefficient register	EMU2nPXR	S16	—	—
EMU2n	EMU2n resolver angle register	EMU2nRESTHETA	U12	—	—
EMU2n	EMU2n electrical angle register	EMU2nTHTEFIX	U12	—	—
EMU2n	EMU2n resolver pole number setting register	EMU2nRESRLD	U3	—	—
EMU2n	EMU2n resolver pole count register	EMU2nRESCNT	U3	—	—
EMU2n	EMU2n speed measurement timer control register	EMU2nVMTCTR	U32	—	—
EMU2n	EMU2n speed measurement timer counter register	EMU2nVMTCNT	U25	—	—
EMU2n	EMU2n speed measurement timer capture register	EMU2nVMTCAP	U25	—	—
EMU2n	EMU2n speed measurement timer overflow register	EMU2nVMTOF	U32	—	—

Note 1. The user-set values are stored in the buffer upon angle generation IP activation, and the EMU calculation results are stored in the buffer upon angle generation IP completion.

Note 2. The user-set value is used by the EMU when the PHISEL bit in the EMU2nANGCTR register is set to 1.

Table 24.28 List of EMU2 Register Functions (Input IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n input IP control register	EMU2nCTRINMD	U16	√(b0 only)	—
EMU2n	EMU2n ADm data register 0	EMU2nADm0	U12	√	—
EMU2n	EMU2n ADm channel 0 origin correction value register	EMU2nADDOFSm0	S16	√	—
EMU2n	EMU2n ADm data register 1	EMU2nADm1	U12	√	—
EMU2n	EMU2n ADm channel 1 origin correction value register	EMU2nADDOFSm1	S16	√	—
EMU2n	EMU2n ADm data register 2	EMU2nADm2	U12	√	—
EMU2n	EMU2n ADm channel 2 origin correction value register	EMU2nADDOFSm2	S16	√	—
EMU2n	EMU2n ADm channel 0 conversion value register	EMU2nADFIXm0	S16	√	—
EMU2n	EMU2n electrical angle software input register	EMU2nTHTESFT	U12	√	√*3
EMU2n	EMU2n ADm channel 1 conversion value register	EMU2nADFIXm1	S16	√	—
EMU2n	EMU2n electrical angle response delay correction variable register	EMU2nEARD	U12	√	—
EMU2n	EMU2n ADm channel 2 conversion value register	EMU2nADFIXm2	S16	√	—
EMU2n	EMU2n electrical angle retention register	EMU2nTHTE	U12	√*2	—
EMU2n	EMU2n resolver angle monitor register	EMU2nTHTREFIXIN	U12	√	—
EMU2n	EMU2n dq-axis current transformation coefficient register	EMU2nSR2	S32	—	—
EMU2n	EMU2n LSB adjustment register	EMU2nDIVLSB	S32	√	—
EMU2n	EMU2n U-phase current value register	EMU2nIUFIX	S32	√	—
EMU2n	EMU2n V-phase current value register	EMU2nIVFIX	S32	√	—
EMU2n	EMU2n W-phase current value register	EMU2nIWFIX	S32	√	—
EMU2n	EMU2n d-axis current value register	EMU2nIDFIX	S32	√	—
EMU2n	EMU2n q-axis current value register	EMU2nIQFIX	S32	√	—

Note 1. The user-set values are stored in the buffer upon input IP activation, and the EMU calculation results are stored in the buffer upon input IP completion.

Note 2. The electrical angle output from the angle generation IP is stored in the buffer when the time set to the RDDATA bits in the EMU2nDDCNT register has elapsed since the trigger event enabled in the EMU2nADTRG register occurs.

Note 3. The user-set value is used by the EMU when the FREGIN bit in the EMU2nCTRINMD register is set to 0.

Table 24.29 List of EMU2 Register Functions (PI control IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n PI control register	EMU2nPICTR	U8	√	—
EMU2n	EMU2n d-axis current target value register	EMU2nIDIN	S32	√	—
EMU2n	EMU2n q-axis current target value register	EMU2nIQIN	S32	√	—
EMU2n	EMU2n id register	EMU2nID	S32	√	—
EMU2n	EMU2n iq register	EMU2nIQ	S32	√	—
EMU2n	EMU2n GPD0 register	EMU2nGPD0	S32	√	—
EMU2n	EMU2n GPQ0 register	EMU2nGPQ0	S32	√	—
EMU2n	EMU2n GPD register	EMU2nGPD	S32	√	—
EMU2n	EMU2n GPQ register	EMU2nGPQ	S32	√	—
EMU2n	EMU2n GID register	EMU2nGID	S32	√	—
EMU2n	EMU2n GIQ register	EMU2nGIQ	S32	√	—
EMU2n	EMU2n GID_MAX register	EMU2nGIDMAX	U31	√	—
EMU2n	EMU2n GIQ_MAX register	EMU2nGIQMAX	U31	√	—
EMU2n	EMU2n vd_MAX register	EMU2nVDMAX	U29	√	—
EMU2n	EMU2n vq_MAX register	EMU2nVQMAX	U29	√	—
EMU2n	EMU2n sum_id register	EMU2nSUMID	S32	√	—
EMU2n	EMU2n sum_id monitor register	EMU2nSUMIDM	S32	—	—
EMU2n	EMU2n sum_iq register	EMU2nSUMIQ	S32	√	—
EMU2n	EMU2n sum_iq monitor register	EMU2nSUMIQM	S32	—	—
EMU2n	EMU2n d-axis voltage register	EMU2nVD	S32	—	—
EMU2n	EMU2n q-axis voltage register	EMU2nVQ	S32	—	—

Note 1. The user-set values are stored in the buffer upon PI control IP activation, and the EMU calculation results are stored in the buffer upon PI control IP completion.

Table 24.30 List of EMU2 Register Functions (PWM IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value*4 √: Applicable, —: Not applicable
EMU2n	EMU2n PWM IP control register	EMU2nPWMCTR	U8	√(b5 only)	—
EMU2n	EMU2n PWM data transfer register	EMU2nPWMDT	U8	—	√*5
EMU2n	EMU2n d-axis voltage correction value register	EMU2nVDCRCT	S32	√	√
EMU2n	EMU2n q-axis voltage correction value register	EMU2nVQCRCT	S32	√	√
EMU2n	EMU2n three-phase voltage transformation coefficient register	EMU2nSR23	S32	—	—
EMU2n	EMU2n U-phase voltage offset register	EMU2nUVOFS	S32	√	√
EMU2n	EMU2n W-phase voltage offset register	EMU2nWVOFS	S32	√	√
EMU2n	EMU2n d-axis reference voltage register	EMU2nPHI	S16	√	√
EMU2n	EMU2n electrical angle adjustment register	EMU2nGTHT	S16	√	—
EMU2n	EMU2n predicted electrical angle software input register	EMU2nTHTFORESFT	U12	√	—
EMU2n	EMU2n digit alignment register 1	EMU2nPWMK1	S32	—	—
EMU2n	EMU2n digit alignment register 2	EMU2nPWMK2	S16	—	—
EMU2n	EMU2n input voltage register	EMU2nVOLV	S16	√	—
EMU2n	EMU2n duty cycle upper limit register	EMU2nDTUL	S32	—	—
EMU2n	EMU2n duty cycle lower limit register	EMU2nDTLL	S32	—	—
EMU2n	EMU2n PWM upper limit register	EMU2nPWMUL	U16	—	—
EMU2n	EMU2n PWM lower limit register	EMU2nPWMLL	U16	—	—
EMU2n	EMU2n dead time setting register	EMU2nDTT	U16	—	—
EMU2n	EMU2n carrier cycle register	EMU2nCARR	U16	√*2	—
EMU2n	EMU2n U-phase PWM register	EMU2nUPWM	U16	√*3	√*5
EMU2n	EMU2n V-phase PWM register	EMU2nVPWM	U16	√*3	√*5
EMU2n	EMU2n W-phase PWM register	EMU2nWPWM	U16	√*3	√*5
EMU2n	EMU2n U-phase output voltage for duty cycle calculation register	EMU2nVUFIX	S32	—	—
EMU2n	EMU2n V-phase output voltage for duty cycle calculation register	EMU2nVVFIX	S32	—	—
EMU2n	EMU2n W-phase output voltage for duty cycle calculation register	EMU2nVWFIX	S32	—	—
EMU2n	EMU2n U-phase compare value register	EMU2nPWMUIP	U16	—	—
EMU2n	EMU2n V-phase compare value register	EMU2nPWMVIP	U16	—	—
EMU2n	EMU2n W-phase compare value register	EMU2nPWMWIP	U16	—	—
EMU2n	EMU2n U-phase output voltage correction amount register	EMU2nVUOFS	S16	√	—
EMU2n	EMU2n V-phase output voltage correction amount register	EMU2nVVOFS	S16	√	—
EMU2n	EMU2n W-phase output voltage correction amount register	EMU2nVWOFS	S16	√	—

Note 1. The user-set values are stored in the buffer upon PWM IP activation, and the EMU calculation results are stored in the buffer upon PWM IP completion.

Note 2. When the SETPWM bit in the EMU2nPWMCTR register is set to 0, the user-set values are stored in the buffer by writing 1 to the PWMDT bit in the EMU2nPWMDT register. When the SETPWM bit is set to 1, the user-set values are stored in the buffer upon PWM IP activation.

Note 3. The user-set values are stored in the buffer by writing 1 to the PWMDT bit in the EMU2nPWMDT register.

Note 4. Setting the FPWMREFPER bit in the EMU2nREFCTR register to 1 allows the user-set values to be reflected in the EMU. For details, refer to **Section 24.3.4, EMU2nREFCTR — EMU2n Register Value Reflection Control Register (n = 0, 1)**.

Note 5. The user-set values are used by the EMU when the SETPWM bit in the EMU2nPWMCTR register is set to 0.

Table 24.31 List of EMU2 Register Functions (Rectangle IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n rectangle IP control register	EMU2nRECCTR	U8	√*2	—
EMU2n	EMU2n batch rectangle output register	EMU2nPTNN	U8	—	√*3
EMU2n	EMU2n batch rectangle output pattern AB register	EMU2nPTNAB	U8	—	—
EMU2n	EMU2n batch rectangle output pattern CD register	EMU2nPTNCD	U8	—	—
EMU2n	EMU2n batch rectangle output pattern EF register	EMU2nPTNEF	U8	—	—
EMU2n	EMU2n compare register 0	EMU2nCMP0	U12	—	—
EMU2n	EMU2n compare register 1	EMU2nCMP1	U12	—	—
EMU2n	EMU2n q-axis reference voltage phase software input register	EMU2nPHQSFT	S16	√	—
EMU2n	EMU2n phase differential software input register	EMU2nPSWSFT	U3	√	—
EMU2n	EMU2n phase differential register	EMU2nPSW	U3	—	—
EMU2n	EMU2n IP compare value 0 register	EMU2nIPCMP0	U12	—	—

Note 1. The user-set values are stored in the buffer upon rectangle IP activation, and the EMU calculation results are stored in the buffer upon rectangle IP completion.

Note 2. The FIPPOSI bit and FDRCT bit values are stored in the buffer upon rectangle IP activation. No buffer is provided to the other bits.

Note 3. The user-set values are used by the EMU when the SETREC bit in the EMU2nRECCTR register is set to 0.

Table 24.32 List of EMU2 Register Functions (Independent Rectangle IP)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n independent rectangle IP control register	EMU2nIRECCTR	U8	—	—
EMU2n	EMU2n independent rectangle output pattern update register	EMU2nIRPTN	U8	—	—
EMU2n	EMU2n independent rectangle IP flag/select signal reset register	EMU2nIRCTRST	U8	—	—
EMU2n	EMU2n independent rectangle IP U-phase compare/ pattern setting register 0	EMU2nIRUCPPN0	U16	—	—
EMU2n	EMU2n independent rectangle IP U-phase compare/ pattern setting register 1	EMU2nIRUCPPN1	U16	—	—
EMU2n	EMU2n independent rectangle IP U-phase compare/ pattern setting register 2	EMU2nIRUCPPN2	U16	—	—
EMU2n	EMU2n independent rectangle IP V-phase compare/ pattern setting register 0	EMU2nIRVCPPN0	U16	—	—
EMU2n	EMU2n independent rectangle IP V-phase compare/ pattern setting register 1	EMU2nIRVCPPN1	U16	—	—
EMU2n	EMU2n independent rectangle IP V-phase compare/ pattern setting register 2	EMU2nIRVCPPN2	U16	—	—
EMU2n	EMU2n independent rectangle IP W-phase compare/ pattern setting register 0	EMU2nIRWCPPN0	U16	—	—
EMU2n	EMU2n independent rectangle IP W-phase compare/ pattern setting register 1	EMU2nIRWCPPN1	U16	—	—
EMU2n	EMU2n independent rectangle IP W-phase compare/ pattern setting register 2	EMU2nIRWCPPN2	U16	—	—
EMU2n	EMU2n independent rectangle IP flag monitor register	EMU2nIRFLGM	U16	—	—
EMU2n	EMU2n independent rectangle IP select signal monitor register	EMU2nIRSELM	U16	—	—

Table 24.33 List of EMU2 Register Functions (Verification Function)

Module Name	Register Name	Symbol	Sign/Data Size U: Unsigned S: Signed	Buffer*1 √: Provided, —: Not provided	Reflection of set value √: Applicable, —: Not applicable
EMU2n	EMU2n AD0 data 0 verification buffer register	EMU2nCBAD00	U12	√	—
EMU2n	EMU2n AD0 data 1 verification buffer register	EMU2nCBAD01	U12	√	—
EMU2n	EMU2n AD0 data 2 verification buffer register	EMU2nCBAD02	U12	√	—
EMU2n	EMU2n resolver angle verification buffer register	EMU2nCBTHTREFIXIN	U12	√	—
EMU2n	EMU2n d-axis current value verification buffer register	EMU2nCBIDFIX	S32	√	—
EMU2n	EMU2n q-axis current value verification buffer register	EMU2nCBIQFIX	S32	√	—
EMU2n	EMU2n U-phase compare value verification buffer register	EMU2nCBPWMUIP	U18	√	—
EMU2n	EMU2n V-phase compare value verification buffer register	EMU2nCBPWMVIP	U18	√	—
EMU2n	EMU2n W-phase compare value verification buffer register	EMU2nCBPWMWIP	U18	√	—
EMU2n	EMU2n batch rectangle pattern verification buffer register	EMU2nCBBREC	U8	√	—
EMU2n	EMU2n independent rectangle pattern verification buffer register	EMU2nCBIREC	U8	√	—

Note 1. The values are stored in the buffer at the timing specified by the EMU2nCMTIM register.

24.3.2 EMU2nPRT — EMU2n Protection Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0000_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PRTCT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.34 EMU2nPRT Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	PRTCT	Protection Remove Releases protection of the write-protected register 0: Protected 1: Protection released

This register sets the protection function to prevent the EMU2nCTR register from being inadvertently rewritten.

PRTCT Bit

Use the following procedure to modify the EMU2nCTR register.

(1) Write 01_H to the EMU2nPRT register (to enable writing to registers).

(2) Modify the EMU2nCTR register.

(3) Write 00_H in the EMU2nPRT register (to disable writing to registers).

24.3.3 EMU2nCTR — EMU2n Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0001_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EMUST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.35 EMU2nCTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are read as 0. The write value should be 0
0	EMUST	EMU Operation*1 0: EMU2n is in reset 1: EMU2n operates

Note 1. Setting the EMUST bit to 0 resets the EMU2 module. The initialized objects are EMU2n related registers and internal registers of EMU2 module except for EMU2n control registers (EMU2nCTR) and EMU2n protect registers (EMU2nPRT).

Note 2. The EMU2nCTR register is protected after a reset. To modify the EMU2nCTR register, remove the protection using the EMU2nPRT register. For EMU2nPRT, see **Section 24.3.2, EMU2nPRT — EMU2n Protection Register (n = 0, 1)**.

24.3.4 EMU2nREFCTR — EMU2n Register Value Reflection Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0002_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	FPWMREFPER
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.36 EMU2nREFCTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are read as 0. The write value should be 0
0	FPWMREFPER	PWM IP Register Value Reflection Control Enables reflection of register values in the EMU internal circuit (PWM IP) 0: Disables reflection 1: Enables reflection

This register selects whether or not to reflect values of the registers listed in **Table 24.37** in the EMU internal circuit.

FPWMREFPER Bit

Selects whether or not to reflect values of the registers listed in **Table 24.37** in the EMU internal circuit. These registers can be written any time.

The values set in the registers of the following table are not reflected to the IP internal circuit when the EMU2nREFCTR register is at the value after reset. After reset, the values of the IP internal circuit are the value after reset of each register.

Figure 24.2 shows an example of the register value reflection by the FPWMREFPER bit.

Table 24.37 Registers for Which Reflection can be Enabled or Disabled by FPWMREFPER

Register Name	Symbol
EMU2n d-axis voltage correction value register	EMU2nVDCRCT
EMU2n q-axis voltage correction value register	EMU2nVQCRCT
EMU2n U-phase voltage offset register	EMU2nUVOFS
EMU2n W-phase voltage offset register	EMU2nWVOFS
EMU2n d-axis reference voltage register	EMU2nPHI

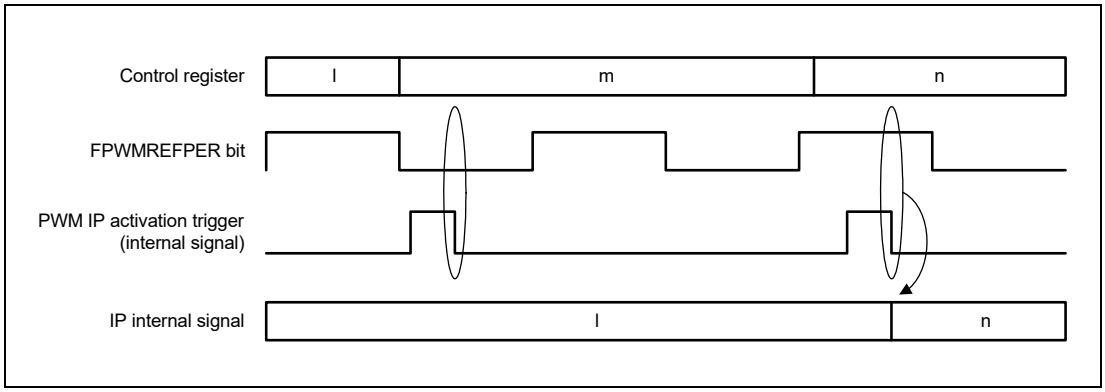


Figure 24.2 Example of Register Value Reflection by FPWMREFPER (for PWM IP)

24.3.5 EMU2nIPTRG — EMU2n IP Activation Trigger Source Select Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0004_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	RECIPTRG	PWMIPTRG	PIIPTRG	NIPTRG1	NIPTRG0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 24.38 EMU2nIPTRG Register Contents

Bit Position	Bit Name	Function
7 to 5	—	These bits are read as 0. The write value should be 0.
4	RECIPTRG	Rectangle IP Activation Trigger Select Selects the rectangle IP activation source. 0: Software trigger (the RECIPSFT bit in the EMU2nIPSFT register) 1: Angle compare 0 match*1
3	PWMIPTRG	PWM IP Activation Trigger Select Selects the PWM IP activation source. 0: Software trigger (the PWMIPSFT bit in the EMU2nIPSFT register) 1: PI control IP completion
2	PIIPTRG	PI Control IP Activation Trigger Select Selects the PI control IP activation source. 0: Software trigger (the PIIPSFT bit in the EMU2nIPSFT register) 1: Input IP completion
1, 0	INIPTRG[1:0]	Input IP Activation Trigger Select Selects the input IP activation source. 0 0: Software trigger (the INIPSFT bit in the EMU2nIPSFT register) 0 1: Angle Compare 0 match 1 0: A/D conversion completion timing selected by the INSTCTR[1:0] bits in the EMU2nCTRINMD register *2 1 1: Angle Compare 0 match or A/D conversion completion timing selected by the INSTCTR[1:0] bits in the EMU2nCTRINMD register *2

Note 1. For details on angle compare 0, see **Section 24.4.10, Batch Rectangle IP**.

Note 2. Set the ADIE bit in the ADCCmSGCRx register to 1 when the INIPTRG[1:0] bits is set to 10_B or 11_B.

Note 3. Do not change the activation trigger select bits at the same timing as the activation trigger generation of the corresponding IP.

24.3.6 EMU2nIPSFT — EMU2n IP Software Activation Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0005_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	RECIPSFT	PWMIPSFT	PIIPSFT	INIPSFT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 24.39 EMU2nIPSFT Register Contents

Bit Position	Bit Name	Function
7 to 4	—	These bits are read as 0. The write value should be 0.
3	RECIPSFT	Rectangle IP Software Activation Writing 1 to this bit activates the rectangle IP. Set the RECIPTRG bit in the EMU2nIPTRG register to 0 before writing 1 to this bit.
2	PWMIPSFT	PWM IP Software Activation Writing 1 to this bit activates the PWM IP. Set the PWMIPTRG bit in the EMU2nIPTRG register to 0 before writing 1 to this bit.
1	PIIPSFT	PI Control IP Software Activation Writing 1 to this bit activates the PI control IP. Set the PIIPTRG bit in the EMU2nIPTRG register to 0 before writing 1 to this bit.
0	INIPSFT	Input IP Software Activation Writing 1 to this bit activates the input IP. Set the INIPTRG[1:0] bits in the EMU2nIPTRG register to 00_B before writing 1 to this bit.

This register is used to activate each IP by software. After written 1, these bits become 0 on completion of the corresponding IP.

Before activating an IP by software, clear the corresponding IP bit in the EMU2nINTSD register to 0. The EMU2nINTSD register bit becomes 0 by writing 1 to the corresponding bit in the EMU2nINTSDC register.

24.3.7 EMU2nADTRG — EMU2n A/D Conversion Start Trigger Select Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0008_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPAD	CAVALAD	CAMOUAD
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.40 EMU2nADTRG Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CMPAD	A/D Start by Angle Compare 0 Match Enables A/D start by angle compare 0 match.*1 0: Disables 1: Enables
1	CAVALAD	A/D Start by Carrier Trough Timing Enables A/D start by carrier signal trough timing. 0: Disables 1: Enables
0	CAMOUAD	A/D Start by Carrier Crest Timing Enables A/D start by carrier signal crest timing. 0: Disables 1: Enables

Note 1. For details on angle compare 0, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set the A/D conversion trigger source. Do not change the EMU2nADTRG register at the same timing as the A/D conversion trigger generation.

24.3.8 EMU2nADMON — EMU2n A/D Conversion Trigger Source Determination Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + 0009_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMPADMO	CAVALADMO	CAMOUADMO
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.41 EMU2nADMON Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CMPADMO	A/D Conversion Trigger source determination (angle compare 0 match)*1 Indicates whether A/D conversion is triggered by the angle compare 0 match or not. 0: The source has not been generated. 1: The source has been generated.
1	CAVALADMO	A/D Conversion Trigger source determination (carrier signal trough timing) Indicates whether A/D conversion is triggered by the carrier signal trough timing or not. 0: The source has not been generated. 1: The source has been generated.
0	CAMOUADMO	A/D Conversion Trigger source determination (carrier signal crest timing) Indicates whether A/D conversion is triggered by the carrier signal crest timing or not. 0: The source has not been generated. 1: The source has been generated.

Note 1. For details on angle compare 0, see **Section 24.4.10, Batch Rectangle IP**

This register monitors the A/D conversion trigger source. This register bit becomes 1 when the corresponding A/D conversion source is generated. It becomes 0 by writing 1 to the corresponding bit in the EMU2nADMONC register.

24.3.9 EMU2nADMONC — EMU2n A/D Conversion Trigger Source Determination Clear Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $000A_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CMADMOCL	CVLADMOCL	CMOADMOCL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.42 EMU2nADMONC Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CMADMOCL	A/D Conversion Trigger Source Determination Register Clear (angle compare 0 match)*1 The CMPADMO bit in the EMU2nADMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
1	CVLADMOCL	A/D Conversion Trigger Source Determination Register Clear (carrier signal trough timing) The CAVALADMO bit in the EMU2nADMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
0	CMOADMOCL	A/D Conversion Trigger Source Determination Register Clear (carrier signal crest timing) The DAMOUADMO bit in the EMU2nADMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.

Note 1. For details on angle compare 0, see **Section 24.4.10, Batch Rectangle IP**.

The corresponding bit in the EMU2nADMON register becomes 0 by writing 1 to this register bit. This bit automatically becomes 0 after one PCLK cycle has elapsed since writing 1 to the bit.

24.3.10 EMU2nDDCNT — EMU2n Data Delay Counter Value Setting Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $000C_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.43 EMU2nDDCNT Register Contents

Bit Position	Bit Name	Function
31 to 16	RDDATA	R/D Data Set the R/D conversion trigger output delay time.
15 to 0	ADDATA	A/D Data Set the A/D conversion trigger output delay time.

This register sets delay time for A/D conversion and R/D conversion trigger outputs. The EMU2 can be operated with after reset value, 0000_H. The trigger is disabled while the corresponding delay counter is running.

24.3.11 EMU2nINT0 — EMU2n Interrupt Source Select Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0010_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTINT0	IRECWINT0	IRECVINT0	IRECUINT0	CMP1INT0	CMP0INT0	CARRINT0	SMLINT0	CBUFINT0	—	—	RECINT0	PWMINT0	PIINT0	ININT0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.44 EMU2nINT0 Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTINT0	Interrupt Source Setting 0 (speed measurement timer overflow) Enables the interrupt generation by a speed measurement timer overflow. 0: Disables the interrupt 1: Enables the interrupt
13	IRECWINT0	Interrupt Source Setting 0 (independent rectangle W-phase compare match) Enables the interrupt generation by an independent rectangle W-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
12	IRECVINT0	Interrupt Source Setting 0 (independent rectangle V-phase compare match) Enables the interrupt generation by an independent rectangle V-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
11	IRECUINT0	Interrupt Source Setting 0 (independent rectangle U-phase compare match) Enables the interrupt generation by an independent rectangle U-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
10	CMP1INT0	Interrupt Source Setting 0 (angle compare 1 match)*1 Enables the interrupt generation by an angle compare 1 match. 0: Disables the interrupt 1: Enables the interrupt
9	CMP0INT0	Interrupt Source Setting 0 (angle compare 0 match) *1 Enables the interrupt generation by an angle compare 0 match. 0: Disables the interrupt 1: Enables the interrupt
8	CARRINT0	Interrupt Source Setting 0 (carrier counter) Enables the interrupt generation by the carrier counter crest/trough timing. 0: Disables the interrupt 1: Enables the interrupt When the CARRINT0 bit is set to 1, an interrupt is generated at either crest or trough timing, or both crest and trough timings of the carrier counter depending on the CARRMOU and CARRVAL bit settings in the EMU2nPWMCTR register.
7	SMLINT0	Interrupt Source Setting 0 (equivalence check error) Enables the interrupt generation by equivalence check error occurrence. 0: Disables the interrupt 1: Enables the interrupt
6	CBUFINT0	Interrupt Source Setting 0 (verification buffering completion) Enables the interrupt generation by verification buffering completion. 0: Disables the interrupt 1: Enables the interrupt
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.44 EMU2nINT0 Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECINT0	Interrupt Source Setting 0 (rectangle IP completion) Enables the interrupt generation by rectangle IP completion. 0: Disables the interrupt 1: Enables the interrupt
2	PWMINT0	Interrupt Source Setting 0 (PWM IP completion) Enables the interrupt generation by PWM IP completion. 0: Disables the interrupt 1: Enables the interrupt
1	PIINT0	Interrupt Source Setting 0 (PI control IP completion) Enables the interrupt generation by PI control IP completion. 0: Disables the interrupt 1: Enables the interrupt
0	ININT0	Interrupt Source Setting 0 (input IP completion) Enables the interrupt generation by input IP completion. 0: Disables the interrupt 1: Enables the interrupt

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set sources for the EMU2n interrupt 0 (n = 0, 1). For details, see **Section 24.1.4, Interrupt Requests**.

24.3.12 EMU2nINT1 — EMU2n Interrupt Source Select Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0012_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTINT1	IRECWINT1	IRECVINT1	IRECUINT1	CMP1INT1	CMP0INT1	CARRINT1	SMLINT1	CBUFINT1	—	—	RECINT1	PWMINT1	PIINT1	ININT1
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.45 EMU2nINT1 Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTINT1	Interrupt Source Setting 1 (speed measurement timer overflow) Enables the interrupt generation by a speed measurement timer overflow. 0: Disables the interrupt 1: Enables the interrupt
13	IRECWINT1	Interrupt Source Setting 1 (independent rectangle W-phase compare match) Enables the interrupt generation by an independent rectangle W-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
12	IRECVINT1	Interrupt Source Setting 1 (independent rectangle V-phase compare match) Enables the interrupt generation by an independent rectangle V-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
11	IRECUINT1	Interrupt Source Setting 1 (independent rectangle U-phase compare match) Enables the interrupt generation by an independent rectangle U-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
10	CMP1INT1	Interrupt Source Setting 1 (angle compare 1 match)*1 Enables the interrupt generation by an angle compare 1 match. 0: Disables the interrupt 1: Enables the interrupt
9	CMP0INT1	Interrupt Source Setting 1 (angle compare 0 match) *1 Enables the interrupt generation by an angle compare 0 match. 0: Disables the interrupt 1: Enables the interrupt
8	CARRINT1	Interrupt Source Setting 1 (carrier counter) Enables the interrupt generation by the carrier counter crest/trough timing. 0: Disables the interrupt 1: Enables the interrupt When the CARRINT1 bit is set to 1, an interrupt is generated at either crest or trough timing, or both crest and trough timings of the carrier counter depending on the CARRMOU and CARRVAL bit settings in the EMU2nPWMCTR register.
7	SMLINT1	Interrupt Source Setting 1 (equivalence check error) Enables the interrupt generation by equivalence check error occurrence. 0: Disables the interrupt 1: Enables the interrupt
6	CBUFINT1	Interrupt Source Setting 1 (verification buffering completion) Enables the interrupt generation by verification buffering completion. 0: Disables the interrupt 1: Enables the interrupt
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.45 EMU2nINT1 Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECINT1	Interrupt Source Setting 1 (rectangle IP completion) Enables the interrupt generation by rectangle IP completion. 0: Disables the interrupt 1: Enables the interrupt
2	PWMINT1	Interrupt Source Setting 1 (PWM IP completion) Enables the interrupt generation by PWM IP completion. 0: Disables the interrupt 1: Enables the interrupt
1	PIINT1	Interrupt Source Setting 1 (PI control IP completion) Enables the interrupt generation by PI control IP completion. 0: Disables the interrupt 1: Enables the interrupt
0	ININT1	Interrupt Source Setting 1 (input IP completion) Enables the interrupt generation by input IP completion. 0: Disables the interrupt 1: Enables the interrupt

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set sources for the EMU2n interrupt 1 (n = 0, 1). For details, **Section 24.1.4, Interrupt Requests**.

24.3.13 EMU2nINT2 — EMU2n Interrupt Source Select Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0014_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTINT2	IRECWINT2	IRECVINT2	IRECUINT2	CMP1INT2	CMP0INT2	CARRINT2	SMLINT2	CBUFINT2	—	—	RECINT2	PWMINT2	PIINT2	ININT2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.46 EMU2nINT2 Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTINT2	Interrupt Source Setting 2 (speed measurement timer overflow) Enables the interrupt generation by a speed measurement timer overflow. 0: Disables the interrupt 1: Enables the interrupt
13	IRECWINT2	Interrupt Source Setting 2 (independent rectangle W-phase compare match) Enables the interrupt generation by an independent rectangle W-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
12	IRECVINT2	Interrupt Source Setting 2 (independent rectangle V-phase compare match) Enables the interrupt generation by an independent rectangle V-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
11	IRECUINT2	Interrupt Source Setting 2 (independent rectangle U-phase compare match) Enables the interrupt generation by an independent rectangle U-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
10	CMP1INT2	Interrupt Source Setting 2 (angle compare 1 match)*1 Enables the interrupt generation by an angle compare 1 match. 0: Disables the interrupt 1: Enables the interrupt
9	CMP0INT2	Interrupt Source Setting 2 (angle compare 0 match) *1 Enables the interrupt generation by an angle compare 0 match. 0: Disables the interrupt 1: Enables the interrupt
8	CARRINT2	Interrupt Source Setting 2 (carrier counter) Enables the interrupt generation by the carrier counter crest/trough timing. 0: Disables the interrupt 1: Enables the interrupt When the CARRINT2 bit is set to 1, an interrupt is generated at either crest or trough timing, or both crest and trough timings of the carrier counter depending on the CARRMOU and CARRVAL bit settings in the EMU2nPWMCTR register.
7	SMLINT2	Interrupt Source Setting 2 (equivalence check error) Enables the interrupt generation by equivalence check error occurrence. 0: Disables the interrupt 1: Enables the interrupt
6	CBUFINT2	Interrupt Source Setting 2 (verification buffering completion) Enables the interrupt generation by verification buffering completion. 0: Disables the interrupt 1: Enables the interrupt
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.46 EMU2nINT2 Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECINT2	Interrupt Source Setting 2 (rectangle IP completion) Enables the interrupt generation by rectangle IP completion. 0: Disables the interrupt 1: Enables the interrupt
2	PWMINT2	Interrupt Source Setting 2 (PWM IP completion) Enables the interrupt generation by PWM IP completion. 0: Disables the interrupt 1: Enables the interrupt
1	PIINT2	Interrupt Source Setting 2 (PI control IP completion) Enables the interrupt generation by PI control IP completion. 0: Disables the interrupt 1: Enables the interrupt
0	ININT2	Interrupt Source Setting 2 (input IP completion) Enables the interrupt generation by input IP completion. 0: Disables the interrupt 1: Enables the interrupt

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set sources for the EMU2n interrupt 2 (n = 0, 1). For details, see **Section 24.1.4, Interrupt Requests**.

24.3.14 EMU2nINT3 — EMU2n Interrupt Source Select Register 3 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0016_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTINT3	IRECWINT3	IRECVINT3	IRECUINT3	CMP1INT3	CMP0INT3	CARRINT3	SMLINT3	CBUFINT3	—	—	RECINT3	PWMINT3	PIINT3	ININT3
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.47 EMU2nINT3 Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTINT3	Interrupt Source Setting 3 (speed measurement timer overflow) Enables the interrupt generation by a speed measurement timer overflow. 0: Disables the interrupt 1: Enables the interrupt
13	IRECWINT3	Interrupt Source Setting 3 (independent rectangle W-phase compare match) Enables the interrupt generation by an independent rectangle W-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
12	IRECVINT3	Interrupt Source Setting 3 (independent rectangle V-phase compare match) Enables the interrupt generation by an independent rectangle V-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
11	IRECUINT3	Interrupt Source Setting 3 (independent rectangle U-phase compare match) Enables the interrupt generation by an independent rectangle U-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
10	CMP1INT3	Interrupt Source Setting 3 (angle compare 1 match)*1 Enables the interrupt generation by an angle compare 1 match. 0: Disables the interrupt 1: Enables the interrupt
9	CMP0INT3	Interrupt Source Setting 3 (angle compare 0 match) *1 Enables the interrupt generation by an angle compare 0 match. 0: Disables the interrupt 1: Enables the interrupt
8	CARRINT3	Interrupt Source Setting 3 (carrier counter) Enables the interrupt generation by the carrier counter crest/trough timing. 0: Disables the interrupt 1: Enables the interrupt When the CARRINT3 bit is set to 1, an interrupt is generated at either crest or trough timing, or both crest and trough timings of the carrier counter depending on the CARRMOU and CARRVAL bit settings in the EMU2nPWMCTR register.
7	SMLINT3	Interrupt Source Setting 3 (equivalence check error) Enables the interrupt generation by equivalence check error occurrence. 0: Disables the interrupt 1: Enables the interrupt
6	CBUFINT3	Interrupt Source Setting 3 (verification buffering completion) Enables the interrupt generation by verification buffering completion. 0: Disables the interrupt 1: Enables the interrupt
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.47 EMU2nINT3 Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECINT3	Interrupt Source Setting 3 (rectangle IP completion) Enables the interrupt generation by rectangle IP completion. 0: Disables the interrupt 1: Enables the interrupt
2	PWMINT3	Interrupt Source Setting 3 (PWM IP completion) Enables the interrupt generation by PWM IP completion. 0: Disables the interrupt 1: Enables the interrupt
1	PIINT3	Interrupt Source Setting 3 (PI control IP completion) Enables the interrupt generation by PI control IP completion. 0: Disables the interrupt 1: Enables the interrupt
0	ININT3	Interrupt Source Setting 3 (input IP completion) Enables the interrupt generation by input IP completion. 0: Disables the interrupt 1: Enables the interrupt

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set sources for the EMU2n interrupt 3 (n = 0, 1). For details, see **Section 24.1.4, Interrupt Requests**.

24.3.15 EMU2nINT4 — EMU2n Interrupt Source Select Register 4 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0018_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTINT4	IRECWINT4	IRECVINT4	IRECUINT4	CMP1INT4	CMP0INT4	CARRINT4	SMLINT4	CBUFINT4	—	—	RECINT4	PWMINT4	PIINT4	ININT4
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.48 EMU2nINT4 Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTINT4	Interrupt Source Setting 4 (speed measurement timer overflow) Enables the interrupt generation by a speed measurement timer overflow. 0: Disables the interrupt 1: Enables the interrupt
13	IRECWINT4	Interrupt Source Setting 4 (independent rectangle W-phase compare match) Enables the interrupt generation by an independent rectangle W-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
12	IRECVINT4	Interrupt Source Setting 4 (independent rectangle V-phase compare match) Enables the interrupt generation by an independent rectangle V-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
11	IRECUINT4	Interrupt Source Setting 4 (independent rectangle U-phase compare match) Enables the interrupt generation by an independent rectangle U-phase compare match. 0: Disables the interrupt 1: Enables the interrupt
10	CMP1INT4	Interrupt Source Setting 4 (angle compare 1 match)*1 Enables the interrupt generation by an angle compare 1 match. 0: Disables the interrupt 1: Enables the interrupt
9	CMP0INT4	Interrupt Source Setting 4 (angle compare 0 match) *1 Enables the interrupt generation by an angle compare 0 match. 0: Disables the interrupt 1: Enables the interrupt
8	CARRINT4	Interrupt Source Setting 4 (carrier counter) Enables the interrupt generation by the carrier counter crest/trough timing. 0: Disables the interrupt 1: Enables the interrupt When the CARRINT4 bit is set to 1, an interrupt is generated at either crest or trough timing, or both crest and trough timings of the carrier counter depending on the CARRMOU and CARRVAL bit settings in the EMU2nPWMCTR register.
7	SMLINT4	Interrupt Source Setting 4 (equivalence check error) Enables the interrupt generation by equivalence check error occurrence. 0: Disables the interrupt 1: Enables the interrupt
6	CBUFINT4	Interrupt Source Setting 4 (verification buffering completion) Enables the interrupt generation by verification buffering completion. 0: Disables the interrupt 1: Enables the interrupt
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.48 EMU2nINT4 Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECINT4	Interrupt Source Setting 4 (rectangle IP completion) Enables the interrupt generation by rectangle IP completion. 0: Disables the interrupt 1: Enables the interrupt
2	PWMINT4	Interrupt Source Setting 4 (PWM IP completion) Enables the interrupt generation by PWM IP completion. 0: Disables the interrupt 1: Enables the interrupt
1	PIINT4	Interrupt Source Setting 4 (PI control IP completion) Enables the interrupt generation by PI control IP completion. 0: Disables the interrupt 1: Enables the interrupt
0	ININT4	Interrupt Source Setting 4 (input IP completion) Enables the interrupt generation by input IP completion. 0: Disables the interrupt 1: Enables the interrupt

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

This register is used to set sources for the EMU2n interrupt 4 (n = 0, 1). For details, see **Section 24.1.4, Interrupt Requests**.

24.3.16 EMU2nINTSD — EMU2n Interrupt Source Determination Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $001C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTIF	IRECWIF	IRECVIF	IRECUIF	CMP1IF	CMP0IF	CARRIF	SMLIF	CBUFIF	—	—	RECIF	PWMIF	PIIF	INIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.49 EMU2nINTSD Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTIF	Interrupt Source Determination Flag (speed measurement timer overflow) 0: A speed measurement timer overflow has not occurred. 1: A speed measurement timer overflow has occurred.
13	IRECWIF	Interrupt Source Determination Flag (independent rectangle W-phase compare match) 0: An independent rectangle W-phase compare match has not occurred. 1: An independent rectangle W-phase compare match has occurred.
12	IRECVIF	Interrupt Source Determination Flag (independent rectangle V-phase compare match) 0: An independent rectangle V-phase compare match has not occurred. 1: An independent rectangle V-phase compare match has occurred.
11	IRECUIF	Interrupt Source Determination Flag (independent rectangle U-phase compare match) 0: An independent rectangle U-phase compare match has not occurred. 1: An independent rectangle U-phase compare match has occurred.
10	CMP1IF	Interrupt Source Determination Flag (angle compare 1 match)*1 0: An angle compare 1 match has not occurred. 1: An angle compare 1 match has occurred.
9	CMP0IF	Interrupt Source Determination Flag (angle compare 0 match)*1 0: An angle compare 0 match has not occurred. 1: An angle compare 0 match has occurred.
8	CARRIF	Interrupt Source Determination Flag (carrier counter) 0: A carrier counter interrupt has not occurred. 1: A carrier counter interrupt has occurred.
7	SMLIF	Interrupt Source Determination Flag (equivalence check error) 0: An equivalence check error interrupt has not occurred. 1: An equivalence check error interrupt has occurred.
6	CBUFIF	Interrupt Source Determination Flag (verification buffering completion) 0: Buffering to the verification buffer has not been completed. 1: Buffering to the verification buffer has been completed.
5, 4	—	These bits are read as 0. The write value should be 0.
3	RECIF	Interrupt Source Determination Flag (rectangle IP completion) 0: A rectangle IP completion interrupt has not occurred. 1: A rectangle IP completion interrupt has occurred.
2	PWMIF	Interrupt Source Determination Flag (PWM IP completion) 0: A PWM IP completion interrupt has not occurred. 1: A PWM IP completion interrupt has occurred.
1	PIIF	Interrupt Source Determination Flag (PI control IP completion) 0: A PI control IP completion interrupt has not occurred. 1: A PI control IP completion interrupt has occurred.

Table 24.49 EMU2nINTSD Register Contents (2/2)

Bit Position	Bit Name	Function
0	INIF	Interrupt Source Determination Flag (input IP completion) 0: An input IP completion interrupt has not occurred. 1: An input IP completion interrupt has occurred.

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

The register bit becomes 1 when the corresponding IP operation is completed or corresponding event has occurred. This register can be used to determine the source of generated interrupt.

24.3.17 EMU2nINTSDC — EMU2n Interrupt Source Determination Clear Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $001E_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VMTIFC	IRECWIFC	IRECVIFC	IRECUIFC	CMP1IFC	CMP0IFC	CARRIFC	SMLIFC	CBUFIFC	—	—	RECIFC	PWMIFC	PIIFC	INIFC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Table 24.50 EMU2nINTSDC Register Contents (1/2)

Bit Position	Bit Name	Function
15	—	This bit is read as 0. The write value should be 0.
14	VMTIFC	Interrupt Source Determination Flag Clear (speed measurement timer overflow) The VMTIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
13	IRECWIFC	Interrupt Source Determination Flag Clear (independent rectangle W-phase compare match) The IRECWIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
12	IRECVIFC	Interrupt Source Determination Flag Clear (independent rectangle V-phase compare match) The IRECVIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
11	IRECUIFC	Interrupt Source Determination Flag Clear (independent rectangle U-phase compare match) The IRECUIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
10	CMP1IFC	Interrupt Source Determination Flag Clear (angle compare 1 match)*1 The CMP1IF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
9	CMP0IFC	Interrupt Source Determination Flag Clear (angle compare 0 match)*1 The CMP0IF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
8	CARRIFC	Interrupt Source Determination Flag Clear (carrier counter) The CARRIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
7	SMLIFC	Interrupt Source Determination Flag Clear (equivalence check error) The SMLIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
6	CBUFIFC	Interrupt Source Determination Flag Clear (verification buffering completion) The CBUFIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
5, 4	—	These bits are read as 0. The write value should be 0.

Table 24.50 EMU2nINTSDC Register Contents (2/2)

Bit Position	Bit Name	Function
3	RECIFC	Interrupt Source Determination Flag Clear (rectangle IP completion) The RECIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
2	PWMIFC	Interrupt Source Determination Flag Clear (PWM IP completion) The PWMIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
1	PIIFC	Interrupt Source Determination Flag Clear (PI control IP completion) The PIIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
0	INIFC	Interrupt Source Determination Flag Clear (input IP completion) The INIF bit in the EMU2nINTSD register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.

Note 1. For details on angle compare 0 and 1, see **Section 24.4.10, Batch Rectangle IP**.

24.3.18 EMU2nOFMON — EMU2n Overflow Detection Result Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + 0020_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPOF	PIIPOF	INIPOF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.51 EMU2nOFMON Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPOF	PWM IP Overflow Detection Flag 0: PWM IP overflow has not been detected. 1: PWM IP overflow has been detected.
1	PIIPOF	PI Control IP Overflow Detection Flag 0: PI control IP overflow has not been detected. 1: PI control IP overflow has been detected.
0	INIPOF	Input IP Overflow Detection Flag 0: Input IP overflow has not been detected. 1: Input IP overflow has been detected.

24.3.19 EMU2nZDMON — EMU2n Zero Division Detection Result Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + 0021_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPZD	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.52 EMU2nZDMON Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPZD	PWM IP Zero Division Detection Flag 0: PWM IP zero division has not been detected. 1: PWM IP zero division has been detected.
1, 0	—	These bits are read as 0. The write value should be 0.

24.3.20 EMU2nOFMONC — EMU2n Overflow Detection Result Clear Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0022_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPOFC	PIIPOFC	INIPOFC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.53 EMU2nOFMONC Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPOFC	PWM IP Overflow Detection Flag Clear The PWMIPOF bit in the EMU2nOFMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
1	PIIPOFC	PI Control IP Overflow Detection Flag Clear The PIIPOF bit in the EMU2nOFMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
0	INIPOFC	Input IP Overflow Detection Flag Clear The INIPOF bit in the EMU2nOFMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.

24.3.21 EMU2nZDMONC — EMU2n Zero Division Detection Result Clear Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0023_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PWMIPZDC	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R

Table 24.54 EMU2nZDMONC Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMIPZDC	PWM IP Zero Division Detection Flag Clear The PWMIPZD bit in the EMU2nZDMON register becomes 0 by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
1, 0	—	These bits are read as 0. The write value should be 0.

24.3.22 EMU2nSMLCTR — EMU2n Equivalence Check Function Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0028_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DATTRG	SMLEN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.55 EMU2nSMLCTR Register Contents

Bit Position	Bit Name	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1	DATTRG	Data Trigger Signal Select For channel 0: 0: Uses data and trigger signal in channel 0. 1: Uses data and trigger signal in channel 1. For channel 1: 0: Uses data and trigger signal in channel 1. 1: Uses data and trigger signal in channel 0.
0	SMLEN	Equivalence Check Function Enable 0: Disables equivalence check function. 1: Enables equivalence check function.

Set the same value to the SMLEN bit in registers EMU20SMLCTR and EMU21SMLCTR.

Set the following combination to the DATTRG bit in registers EMU20SMLCTR and EMU21SMLCTR: 0 and 0, 0 and 1, or 1 and 0.

24.3.23 EMU2nCBCTR0 — EMU2n Verification Buffer Control Register 0 (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $002C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	CBMON	CBEN1	CBEN0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.56 EMU2nCBCTR0 Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	CBMON	Buffering Status Indicates the buffering status. 0: Buffering is completed or in idle state. 1: Waiting for buffering completion
1	CBEN1	Buffering Enable 1 Enables buffering. 0: Enables buffering using the CBEN0 bit. 1: Always enables buffering.
0	CBEN0	Buffering Enable 0 Buffering is enabled by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.

The values to be buffered are:

- EMU2nADm data register k (EMU2nADmk) (m = 0, 1, k = 0 to 2)
- EMU2n resolver angle monitor register (EMU2nTHTREFIXIN)
- EMU2n d-axis current value register (EMU2nIDFIX)
- EMU2n q-axis current value register (EMU2nIQFIX)
- U-phase compare value register (EMU2nPWMUIP)
- V-phase compare value register (EMU2nPWMVIP)
- W-phase compare value register (EMU2nPWMWIP)
- U/V/W-phase batch rectangle pattern values
- U/V/W-phase independent rectangle pattern values

CBEN0

When 1 is written to the CBEN0 bit, buffering to the verification buffers is performed only once at the first event timing among those enabled by the EMU2nCBTIM register. Write 1 to the CBEN0 bit again to perform another buffering. When multiple timings are enabled in the EMU2nCBTIM register, write 1 to the CBEN0 bit for each timing.

CBEN1 Bit

While the CBEN1 bit is set to 1, buffering to the verification buffers is always performed every time the timing enabled by the EMU2nCBTIM register is occurred.

CBMON Bit

When buffering is enabled with the CBEN0 bit, the CBMON bit becomes 1 from when 1 is written to the CBEN0 bit to when the buffering is completed. When buffering is enabled with the CBEN1 bit, the CBMON bit becomes 1 while the CBEN0 bit is set to 1.

24.3.24 EMU2nCBCTR1 — EMU2n Verification Buffer Control Register 1 (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $002D_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	IRECCEN	BRECCEN	PWMCEN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.57 EMU2nCBCTR1 Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	IRECCEN	Independent Rectangle Pattern Compare Enable Enables the check if independent rectangle patterns match between the channels upon completion of buffering. 0: Disables independent rectangle pattern compare. 1: Enables independent rectangle pattern compare.
1	BRECCEN	Batch Rectangle Pattern Compare Enable Enables the check if batch rectangle patterns match between the channels upon completion of buffering. 0: Disables batch rectangle pattern compare. 1: Enables batch rectangle pattern compare.
0	PWMCEN	PWM Compare Value Compare Enable Enables the check if PWM compare values match between the channels upon completion of buffering. 0: Disables PWM duty cycle compare. 1: Enables PWM duty cycle compare.

When the PWMCEN, BRECCEN, or IRECCEN bit is set to 1, set the same value to registers EMU20CBTIM and EMU21CBTIM. The compare result is output as the interrupt and indicated in the SMLIF bit in the EMU2nINTSD register.

24.3.25 EMU2nCBTIM — EMU2n Verification Buffer Timing Select Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $002E_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	IRECWBT	IRECVBT	IRECUBT	—	CMP0BT	—	—	—	—	—	—	PWMBT	—	INBT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R	R/W	R	R	R	R	R	R	R/W	R	R/W

Table 24.58 EMU2nCBTIM Register Contents

Bit Position	Bit Name	Function
15, 14	—	These bits are read as 0. The write value should be 0.
13	IRECWBT	Buffer Timing Setting (independent rectangle W-phase compare match)*1 Enables buffering to the verification buffer register on an independent rectangle W-phase compare match. 0: Disables buffering. 1: Enables buffering.
12	IRECVBT	Buffer Timing Setting (independent rectangle V-phase compare match)*1 Enables buffering to the verification buffer register on an independent rectangle V-phase compare match. 0: Disables buffering. 1: Enables buffering.
11	IRECUBT	Buffer Timing Setting (independent rectangle U-phase compare match)*1 Enables buffering to the verification buffer register on an independent rectangle U-phase compare match. 0: Disables buffering. 1: Enables buffering.
10	—	This bit is read as 0. The write value should be 0.
9	CMP0BT	Buffer Timing Setting (angle compare 0 match) Enables buffering to the verification buffer register on an angle compare 0 match. 0: Disables buffering. 1: Enables buffering.
8 to 3	—	These bits are read as 0. The write value should be 0.
2	PWMBT	Buffer Timing Setting (PWM IP completion) Enables buffering to the verification buffer register at PWM IP completion. 0: Disables buffering. 1: Enables buffering.
1	—	This bit is read as 0. The write value should be 0.
0	INBT	Buffer Timing Setting (Input IP completion) Enables buffering to the verification buffer register at input IP completion. 0: Disables buffering. 1: Enables buffering.

Note 1. Buffering on independent rectangle compare match timing is performed on each compare match of compare/pattern setting registers 0 to 2.

24.3.26 EMU2nANGCTR — EMU2n Angle Generation IP Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0040_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RDPHIEN	PHISEL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.59 EMU2nANGCTR Register Contents

Bit Position	Bit Name	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1	RDPHIEN	0: Angle input is fixed to 0. 1: Angle data/Z-phase signal from R/D converter is enabled.
0	PHISEL	0: Angle data/Z-phase signal from R/D converter is used. 1: EMU2 user defined value is used (EMU2nRESTHSFT register).

This register controls the angle generation IP calculation. For details on control of angle generation IP calculation, see **Section 24.4.4, Angle Generation IP**.

24.3.27 EMU2nCPJUD0 — EMU2n Compare Judgment Correction Register 0 (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0042_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	DATA							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.60 EMU2nCPJUD0 Register Contents

Bit Position	Bit Name	Function
7 to 0	DATA	Data Set data.

For details on calculation with the EMU2nCPJUD0 register setting, see **Section 24.4.4 (4) Determining Angle Compare 0 Match**.

24.3.28 EMU2nCPJUD1 — EMU2n Compare Judgment Correction Register 1 (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0043_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	DATA							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.61 EMU2nCPJUD1 Register Contents

Bit Position	Bit Name	Function
7 to 0	DATA	Data Set data.

For details on calculation with the EMU2nCPJUD1 register setting, see **Section 24.4.4 (5) Determining Angle Compare 1 Match**.

24.3.29 EMU2nRESTHSFT — EMU2n Resolver Angle Software Input Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0044_H

Value after reset: 0000_H

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

Table 24.62 EMU2nRESTHSFT Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

For details on calculation with the EMU2nRESTHSFT register setting, see **Section 24.4.4 (2) Obtaining Angle Data**.

24.3.30 EMU2nANGOFS — EMU2n Angle Generation IP Offset Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0046_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S				DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.63 EMU2nANGOFS Register Contents

Bit Position	Bit Name	Function
15 to 12	S	Sign Set the sign bits.
11 to 0	DATA	Data Set data.

For details on calculation with the EMU2nANGOFS register setting, see **Section 24.4.4 (2) Obtaining Angle Data**.

24.3.31 EMU2nPXR — EMU2n Electrical Angle Generation Coefficient Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0048_H

Value after reset: 0100_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.64 EMU2nPXR Register Contents

Bit Position	Bit Name	Function
15	S	Sign Fix to 0.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nPXR register setting, see **Section 24.4.4 (3) Generating Electrical Angle**.

24.3.32 EMU2nRESTHETA — EMU2n Resolver Angle Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $004A_H$

Value after reset: 0000_H

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

Table 24.65 EMU2nRESTHETA Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nRESTHETA register, see **Section 24.4.4 (2) Obtaining Angle Data** and **Section 24.4.4 (5) Determining Angle Compare 1 Match**.

24.3.33 EMU2nTHTEFIX — EMU2n Electrical Angle Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $004C_H$

Value after reset: 0000_H

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

Table 24.66 EMU2nTHTEFIX Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nTHTEFIX register, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results**, **Section 24.4.4 (3) Generating Electrical Angle**, and **Section 24.4.4 (4) Determining Angle Compare 0 Match**.

24.3.34 EMU2nRESRLD — EMU2n Resolver Pole Number Setting Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $004E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.67 EMU2nRESRLD Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data Set data.

For details on calculation with the EMU2nRESRLD register setting, see **Section 24.4.4 (3) Generating Electrical Angle**.

24.3.35 EMU2nRESCNT — EMU2n Resolver Pole Count Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $004F_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.68 EMU2nRESCNT Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data Set data.

For details on calculation with the EMU2nRESCNT register setting, see **Section 24.4.4 (3) Generating Electrical Angle**.

24.3.36 EMU2nVMTCTR — EMU2n Speed Measurement Timer Control Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0054_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STTRG	—	—	—	—	—	—	—	—	—	—	—	—	—	OVFSW	STR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 24.69 EMU2nVMTCTR Register Contents

Bit Position	Bit Name	Function
31 to 16	—	These bits are read as 0. The write value should be 0.
15	STTRG	Software Trigger When 1 is written to this bit, the EMU2nVMTCNT counter value is stored (captured) in the EMU2nVMNTCAP register. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
14 to 2	—	These bits are read as 0. The write value should be 0.
1	OVFSW	Overflow Signal Output Select 0: EMU2nVMTOF register 1: Interrupt request
0	STR	Count Start 0: Stops counting. 1: Starts counting.

STTRG Bit

Writing 1 to this bit stores (captures) the EMU2nVMTCNT counter value in the EMU2nVMTCAP register.

OVFSW Bit

This bit selects whether an interrupt request is generated or the OVF bit in the EMU2nVMTOF register becomes 1 when the EMU2nVMTCNT counter overflows.

STR Bit

This bit starts or stops the EMU2nVMTCNT counter operation.

24.3.37 EMU2nVMTCNT — EMU2n Speed Measurement Timer Counter Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0058_H

Value after reset: $0000\ 0000_H$

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

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

Table 24.70 EMU2nVMTCNT Register Contents

Bit Position	Bit Name	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24 to 0	DATA	Data 25-bit increment counter value is stored. Set the counter value.

The EMU2nVMTCNT counter is a 25-bit increment counter that is operated by the CCLK when the STR bit in the EMU2nVMTCTR register is set to 1. This register value can be changed regardless whether or not the counter is operating. Reading this register returns the current counter value.

When the EMU2nVMTCNT counter value is stored in the EMU2nVMTCAP register or the counter overflows, the EMU2nVMTCNT counter becomes $0000\ 0000_H$ and continues counting.

24.3.38 EMU2nVMTCAP — EMU2n Speed Measurement Timer Capture Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $005C_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.71 EMU2nVMTCAP Register Contents

Bit Position	Bit Name	Function
31 to 25	—	These bits are read as 0. The write value should be 0.
24 to 0	DATA	Data Data is stored.

This is a read-only register. The EMU2nVMTCNT counter value is stored in this register in response to the Z-phase pulse from R/D converter.

When the speed measurement timer is used, set the EMU2nANGCTR register to 02_H.

24.3.39 EMU2nVMTOF — EMU2n Speed Measurement Timer Overflow Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0060_H

Value after reset: $0000\ 0000_H$

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

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

Table 24.72 EMU2nVMTOF Register Contents

Bit Position	Bit Name	Function
31 to 1	—	These bits are read as 0. The write value should be 0.
0	OVF	Overflow Flag 0: Overflow has not occurred. 1: Overflow has occurred, or 1 has been written when this bit is 0.

This register is enabled when the OVFSW bit in the EMU2nVMTCTR register is set to 0.

OVF Bit

[Conditions to become 0]

- 0 is written to the OVF bit while the OVF bit is 1.

[Conditions to become 1]

- The EMU2nVMTCNT counter has overflowed ($01FF\ FFFF_H \rightarrow 0000\ 0000_H$) while the OVFSW bit in EMU2nVMTCTR is set to 0 (EMU2nVMTOF register is selected).
- 1 is written to the OVF bit while the OVF bit is 0.

24.3.40 EMU2nCTRINMD — EMU2n Input IP Control Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0080_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	INSTCTR1	INSTCTR0	CMUVW2	CMUVW1	CMUVW0	CMES	—	FREGIN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W

Table 24.73 EMU2nCTRINMD Register Contents

Bit Position	Bit Name	Function
15 to 8	—	These bits are read as 0. The write value should be 0.
7, 6	INSTCTR[1:0]	Selects the activation timing of input IP*1. 0 0: On completion of all A/D conversion (scan of the A/D converter scan group 4 ended) 0 1: On completion of A/D conversion of CH0 (conversion of the A/D converter virtual channel 0 completed) 1 0: On completion of A/D conversion of CH1 (conversion of the A/D converter virtual channel 1 completed) 1 1: On completion of A/D conversion of CH2 (conversion of the A/D converter virtual channel 2 completed)
5 to 3	CMUVW[2:0]	Selects the object of current measurement when the CMES bit is set to 1. 0 0 0: Measures currents of 3 phases (U, V, and W) 0 0 1: Measures currents of 2 phases (V and W) 0 1 0: Measures currents of 2 phases (U and W) 0 1 1: Measures currents of 2 phases (U and V) The other settings are prohibited.
2	CMES	Selects the object of current measurement. 0: 2 phases (V and W) 1: Object selected by the CMUVW[2:0] bits
1	—	This bit is read as 0. The write value should be 0.
0	FREGIN	Selects electrical angle to be used for input IP. 0: Uses User-set value 1: Uses electrical angle and resolver angle generated by angle generation IP

Note 1. Modify the INSTCTR[1:0] bit when A/D converter is stopped.

This register controls input IP calculation. For details on calculation defined with each bit setting, see **Section 24.4.5, Input IP**.

CAUTION

Following table describes the relations of connection between virtual channel registers (VCRn) and data registers (DRn) of A/D converter with ADm data register k of EMU2.

A/D Converter Side	EMU2 Side	Current Expected by EMU2
Virtual channel 0 (VCR0, DR0)	ADm data register 0 (EMU2nADm0)	V phase current
Virtual channel 1 (VCR1, DR1)	ADm data register 1 (EMU2nADm1)	W phase current
Virtual channel 2 (VCR2, DR2)	ADm data register 2 (EMU2nADm2)	U phase current

Set the internal registers of A/D converter so that the currents expected by EMU2 can be output from A/D converter virtual channels 0 to 2 as above.

24.3.41 EMU2nADmk — EMU2n ADm Data Register k (n = 0, 1) (m = 0, 1) (k = 0 to 2)

Access: Readable/writable in 16-bit units.

Address: EMU20AD00: <EMU20_base> + 0084_H, EMU20AD01: <EMU20_base> + 0088_H,
EMU20AD02: <EMU20_base> + $008C_H$
EMU21AD10: <EMU21_base> + 0084_H, EMU21AD11: <EMU21_base> + 0088_H,
EMU21AD12: <EMU21_base> + $008C_H$

Value after reset: 0000_H

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

Table 24.74 EMU2nADmk Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data A/D conversion value is stored. A/D conversion value can be set.

The conversion result of ADCCmDRk register is stored in this register. In addition, the conversion result can be set to this register by the program. If storing of an A/D conversion value and writing of a current value by the CPU occur at the same time, storing of an A/D conversion value takes priority. For details, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results**.

The conversion result is stored in bits 11 to 0 of the EMU2nADmk register regardless of the setting of the ADCCnADCR2.DFMT bit.

24.3.42 EMU2nADDOFSmk — EMU2n ADm Channel k Origin Correction Value Register (n = 0, 1) (m = 0, 1) (k = 0 to 2)

Access: Readable/writable in 16-bit units.

Address: EMU20ADDOFS00: <EMU20_base> + 0086_H, EMU20ADDOFS01: <EMU20_base> + $008A_H$, EMU20ADDOFS02: <EMU20_base> + $008E_H$
EMU21ADDOFS10: <EMU21_base> + 0086_H, EMU21ADDOFS11: <EMU21_base> + $008A_H$, EMU21ADDOFS12: <EMU21_base> + $008E_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.75 EMU2nADDOFSmk Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nADDOFSmk register setting, see **Section 24.4.5 (3) Calculating Motor Current**.

24.3.43 EMU2nADFIXmk — EMU2n A/Dm Channel k Conversion Value Register (n = 0, 1) (m = 0, 1) (k = 0 to 2)

Access: Readable in 16-bit units.

Address: EMU20ADFIX00: <EMU20_base> + 0090_H, EMU20ADFIX01: <EMU20_base> + 0094_H, EMU20ADFIX02: <EMU20_base> + 0098_H
EMU21ADFIX10: <EMU21_base> + 0090_H, EMU21ADFIX11: <EMU21_base> + 0094_H, EMU21ADFIX12: <EMU21_base> + 0098_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.76 EMU2nADFIXmk Register Contents

Bit Position	Bit Name	Function
15	S	Sign The sign bit is stored.
14 to 0	DATA	Data Data is stored.

For details on calculation with the EMU2nADFIXmk register setting, see **Section 24.4.5 (3) Calculating Motor Current**.

24.3.44 EMU2nTHTESFT — EMU2n Electrical Angle Software Input Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0092_H

Value after reset: 0000_H

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

Table 24.77 EMU2nTHTESFT Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

For details on calculation with the EMU2nTHTESFT register setting, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results**.

24.3.45 EMU2nEARD — EMU2n Electrical Angle Response Delay Correction Variable Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0096_H

Value after reset: 0000_H

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

Table 24.78 EMU2nEARD Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

For details on calculation with the EMU2nEARD register setting, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results**.

24.3.46 EMU2nTHTE — EMU2n Electrical Angle Retention Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $009A_H$

Value after reset: 0000_H

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

Table 24.79 EMU2nTHTE Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nTHTE register, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results** and **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.47 EMU2nTHTREFIXIN — EMU2n Resolver Angle Monitor Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $009C_H$

Value after reset: 0000_H

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

Table 24.80 EMU2nTHTREFIXIN Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nTHTREFIXIN register, see **Section 24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results**.

24.3.48 EMU2nSR2 — EMU2n dq-Axis Current Transformation Coefficient Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00A0_H$

Value after reset: $0000\ D106_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.81 EMU2nSR2 Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nSR2 register setting, see **Section 24.4.5 (4) dq-axis Current Transformation** and **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.49 EMU2nDIVLSB — EMU2n LSB Adjustment Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00A4_H$

Value after reset: $0001\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.82 EMU2nDIVLSB Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nDIVLSB register setting, see **Section 24.4.5 (3) Calculating Motor Current**.

24.3.50 EMU2nIUFIX — EMU2n U-phase Current Value Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $00A8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.83 EMU2nIUFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nIUFIX register, see **Section 24.4.5 (3) Calculating Motor Current**.

24.3.51 EMU2nIVFIX — EMU2n V-phase Current Value Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $00AC_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.84 EMU2nIVFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nIVFIX register, see **Section 24.4.5 (3) Calculating Motor Current**.

24.3.52 EMU2nIWFIX — EMU2n W-phase Current Value Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $00B0_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.85 EMU2nIVFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nIWFIX register, see **Section 24.4.5 (3) Calculating Motor Current** and **Section 24.4.5 (4) dq-axis Current Transformation**.

24.3.53 EMU2nIDFIX — EMU2n d-Axis Current Value Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $00B4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.86 EMU2nIDFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nIDFIX register, see **Section 24.4.5 (4) dq-axis Current Transformation**, **Section 24.4.6, PI Control IP**, and **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.54 EMU2nIQFIX — EMU2n q-axis Current Value Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $00B8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.87 EMU2nIQFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nIQFIX register, see **Section 24.4.5 (4) dq-axis Current Transformation**, **Section 24.4.6, PI Control IP**, and **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.55 EMU2nPICTR — EMU2n PI Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $00C0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	FSUMIQ	FSUMID
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 24.88 EMU2nPICTR Register Contents

Bit Position	Bit Name	Function
7 to 2	—	These bits are read as 0. The write value should be 0.
1	FSUMIQ	sum_iq Software Input Select Selects the value to be used for sum_iq. 0: User-set value 1: EMU calculation result
0	FSUMID	sum_id Software Input Select Selects the value to be used for sum_id. 0: User-set value 1: EMU calculation result

24.3.56 EMU2nIDIN — EMU2n d-Axis Current Target Value Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00C4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.89 EMU2nIDIN Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nIDIN register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.57 EMU2nIQIN — EMU2n q-axis Current Target Value Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00C8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.90 EMU2nIQIN Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nIQIN register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.58 EMU2nID — EMU2n ID Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00CC_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.91 EMU2nID Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nID register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.59 EMU2nIQ — EMU2n IQ Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00D0_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.92 EMU2nIQ Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nIQ setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.60 EMU2nGPD0 — EMU2n GPD0 Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00D4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.93 EMU2nGPD0 Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGPD0 register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.61 EMU2nGPQ0 — EMU2n GPQ0 Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00D8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.94 EMU2nGPQ0 Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGPQ0 register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.62 EMU2nGPD — EMU2n GPD Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00DC_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.95 EMU2nGPD Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGPD register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.63 EMU2nGPQ — EMU2n GPQ Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00E0_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.96 EMU2nGPQ Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGPQ register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.64 EMU2nGID — EMU2n GID Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00E4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.97 EMU2nGID Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGID register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.65 EMU2nGIQ — EMU2n GIQ Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00E8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.98 EMU2nGIQ Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGIQ register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.66 EMU2nGIDMAX — EMU2n GID_MAX Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00EC_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.99 EMU2nGIDMAX Register Contents

Bit Position	Bit Name	Function
31	—	This bit is read as 0. The write value should be 0.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGIDMAX register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.67 EMU2nGIQMAX — EMU2n GIQ_MAX Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00F0_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.100 EMU2nGIQMAX Register Contents

Bit Position	Bit Name	Function
31	—	This bit is read as 0. The write value should be 0.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGIQMAX register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.68 EMU2nVDMAX — EMU2n VD_MAX Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00F4_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.101 EMU2nVDMAX Register Contents

Bit Position	Bit Name	Function
31 to 29	—	These bits are read as 0. The write value should be 0.
28 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVDMAX register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.69 EMU2nVQMAX — EMU2n VQ_MAX Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $00F8_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.102 EMU2nVQMAX Register Contents

Bit Position	Bit Name	Function
31 to 29	—	These bits are read as 0. The write value should be 0.
28 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVQMAX register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.70 EMU2nSUMID — EMU2n SUM_ID Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0100_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.103 EMU2nSUMID Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nSUMID register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.71 EMU2nSUMIDM — EMU2n SUM_ID Monitor Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0104_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.104 EMU2nSUMIDM Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nSUMIDM register, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.72 EMU2nSUMIQ — EMU2n SUM_IQ Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0108_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.105 EMU2nSUMIQ Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nSUMIQ register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.73 EMU2nSUMIQM — EMU2n SUM_IQ Monitor Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $010C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.106 EMU2nSUMIQM Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nSUMIQM register, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.74 EMU2nVD — EMU2n d-Axis Voltage Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0110_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.107 EMU2nVD Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nVD register, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.75 EMU2nVQ — EMU2n q-Axis Voltage Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0114_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.108 EMU2nVQ Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nVQ register, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

24.3.76 EMU2nPWMCTR — EMU2n PWM IP Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0120_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PWMSEL	SHIPWM	FLININIP	SETRVRS	CARRMOU	CARRVAL	SETHARM	SETPWM
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.109 EMU2nPWMCTR Register Contents

Bit Position	Bit Name	Function
7	PWMSEL	PWM Operation Reference Value Select Selects how to generate the PWM operation reference value. 0: Generates the value based on carrier cycle setting. 1: Generates the value based on carrier cycle and dead time settings.
6	SHIPWM	Carrier Output/PWM Compare Value Shift Outputs setting values related to the carrier cycle and PWM compare which EMU2 outputs after shifting to the left by one bit (double). 0: Outputs setting values without shifting. 1: Outputs shifted setting values to the left by one bit.
5	FLININIP	Electrical Angle Select Selects the value to be used for electrical angle. 0: User-set value 1: EMU calculation result
4	SETRVRS	PWM Inversion 0: 50% duty cycle of PWM + voltage correction value 1: 50% duty cycle of PWM – voltage correction value
3	CARRMOU	Carrier Counter Crest Sets the behavior of IP at the carrier counter crest timing. 0: Does not reflect compare register and carrier cycle values, nor generate an interrupt at the crest of the carrier wave. 1: Reflects compare register and carrier cycle values, and generate an interrupt at the crest of the carrier wave.
2	CARRVAL	Carrier Counter Trough Sets the behavior of IP at the carrier counter trough timing. 0: Does not reflect compare register and carrier cycle values, nor generate an interrupt at the trough of the carrier wave. 1: Reflects compare register and carrier cycle values, and generate an interrupt at the trough of the carrier wave.
1	SETHARM	Third Harmonic Superposition Setting 0: Disables third harmonic superposition. 1: Enables third harmonic superposition.
0	SETPWM	PWM Setting Sets the value to be used for PWM wave generation. 0: User-set value 1: EMU calculation result

This register controls PWM IP calculation. For details on calculation defined with each bit, see **Section 24.4.7, PWM IP**.

24.3.77 EMU2nPWMDT — EMU2n PWM Data Transfer Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0122_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PWMDT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.110 EMU2nPWMDT Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	PWMDT	PWM Data Transfer When 1 is written to this bit while the SETPWM bit in the EMU2nPWMCTR register is 0, the carrier cycle (EMU2nCARR register) and duty cycle value (EMU2nUPWM, EMU2nVPWM, and EMU2nWPWM registers) are transferred to TSG3. Writing 1 to this bit while the SETPWM bit in the EMU2nPWMCTR register is 1 is invalid. After 1 is written to this bit, it is automatically cleared to 0. Writing 0 to this bit is invalid. This bit is always read as 0.

24.3.78 EMU2nVDCRCT — EMU2n d-Axis Voltage Correction Value Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0124_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.111 EMU2nVDCRCT Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVDCRCT register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

For details on the EMU2nVDCRCT register value reflection in the EMU internal circuit, see **Section 24.3.4, EMU2nREFCTR — EMU2n Register Value Reflection Control Register (n = 0, 1)**.

24.3.79 EMU2nVQCRCT — EMU2n q-Axis Voltage Correction Value Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0128_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.112 EMU2nVQCRCT Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVQCRCT register setting, see **Section 24.4.7 (2) d-Axis and q-Axis Voltage Correction**.

For details on the EMU2nVQCRCT register value reflection in the EMU internal circuit, see **Section 24.3.4, EMU2nREFCTR — EMU2n Register Value Reflection Control Register (n = 0, 1)**.

24.3.80 EMU2nSR23 — EMU2n Three-phase Voltage Transformation Coefficient Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $012C_H$

Value after reset: $0000\ D106_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.113 EMU2nSR23 Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nSR23 register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.81 EMU2nUVOFS — EMU2n U-phase Voltage Offset Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0130_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.114 EMU2nUVOFS Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nUVOFS register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.82 EMU2nWVOFS — EMU2n W-phase Voltage Offset Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0134_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.115 EMU2nWVOFS Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nWVOFS register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.83 EMU2nPHI — EMU2n d-Axis Reference Voltage Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0138_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.116 EMU2nPHI Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nPHI register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

For details on the EMU2nPHI register value reflection in the EMU internal circuit, see **Section 24.3.4, EMU2nREFCTR — EMU2n Register Value Reflection Control Register (n = 0, 1)**.

24.3.84 EMU2nGTHT — EMU2n Electrical Angle Adjustment Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $013C_H$

Value after reset: 0100_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.117 EMU2nGTHT Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nGTHT register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.85 EMU2nTHTFORESFT — EMU2n Predicted Electrical Angle Software Input Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $013E_H$

Value after reset: 0000_H

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

Table 24.118 EMU2nTHTFORESFT Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

For details on calculation with the EMU2nTHTFORESFT register setting, see **Section 24.4.7 (3) Transformation to 3-phase Voltage**.

24.3.86 EMU2nPWMK1 — EMU2n Digit Alignment Register 1 (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0140_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.119 EMU2nPWMK1 Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nPWMK1 register setting, see **Section 24.4.7 (4) Calculating Duty Cycle**.

24.3.87 EMU2nPWMK2 — EMU2n Digit Alignment Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0144_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.120 EMU2nPWMK2 Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nPWMK2 register setting, see **Section 24.4.7 (8) Setting PWM**.

24.3.88 EMU2nVOLV — EMU2n Input Voltage Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0146_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.121 EMU2nVOLV Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVOLV register setting, see **Section 24.4.7 (4) Calculating Duty Cycle** and **Section 24.4.7 (8) Setting PWM**.

24.3.89 EMU2nDTUL — EMU2n Duty Cycle Upper Limit Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + 0148_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.122 EMU2nDTUL Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nDTUL register setting, see **Section 24.4.7 (7) Selecting and Limiting Output Voltage of Each Phase**.

24.3.90 EMU2nDTLL — EMU2n Duty Cycle Lower Limit Register (n = 0, 1)

Access: Readable/writable in 32-bit units.

Address: <EMU2n_base> + $014C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

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

Table 24.123 EMU2nDTLL Register Contents

Bit Position	Bit Name	Function
31	S	Sign Set the sign bit.
30 to 0	DATA	Data Set data.

For details on calculation with the EMU2nDTLL register setting, see **Section 24.4.7 (7) Selecting and Limiting Output Voltage of Each Phase**.

24.3.91 EMU2nPWMUL — EMU2n PWM Upper Limit Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0150_H

Value after reset: 0000_H

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

Table 24.124 EMU2nPWMUL Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register sets the upper limit of PWM compare values.

24.3.92 EMU2nPWMLL — EMU2n PWM Lower Limit Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0152_H

Value after reset: 0000_H

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

Table 24.125 EMU2nPWMLL Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register sets the lower limit of PWM compare values. For details on PWM compare value calculation with the EMU2nPWMLL setting, see **Section 24.4.7, PWM IP**.

24.3.93 EMU2nDTT — EMU2n Dead Time Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0154_H

Value after reset: $0FFF_H$

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

Table 24.126 EMU2nDTT Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register sets the dead time. Set 0 or a value from 2 to (EMU2nCARR register value/2).

24.3.94 EMU2nCARR — EMU2n Carrier Cycle Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0156_H

Value after reset: $7FFF_H$

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

Table 24.127 EMU2nCARR Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register controls the carrier counter cycles for PWM waveform generation. Set a value so that (EMU2nCARR register + EMU2nDTT register) is equal to or less than $FFFF_H$.

24.3.95 EMU2nUPWM — EMU2n U-phase PWM Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0158_H

Value after reset: $47FF_H$

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

Table 24.128 EMU2nUPWM Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register is used to manually set the PWM compare values by the user . When the user-set values, not IP calculation results, are selected to be used for PWM compare values, set the value in this register. Do not set a value larger than (EMU2nCARR register value + EMU2nDTT register value).

24.3.96 EMU2nVPWM — EMU2n V-phase PWM Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $015A_H$

Value after reset: $47FF_H$

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

Table 24.129 EMU2nUPWM Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register is used to manually set the PWM compare values by the user. When the user-set values, not IP calculation results, are selected to be used for PWM compare values, set the value in this register. Do not set a value larger than (EMU2nCARR register value + EMU2nDTT register value).

24.3.97 EMU2nWPWM — EMU2n W-phase PWM Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $015C_H$

Value after reset: $47FF_H$

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

Table 24.130 EMU2nUPWM Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Set data.

This register is used to manually set the PWM compare values by the user. When the user-set values, not IP calculation results, are selected to be used for PWM compare values, set the value in this register. Do not set a value larger than (EMU2nCARR register value + EMU2nDTT register value).

24.3.98 EMU2nVUFIX — EMU2n U-phase Output Voltage For Duty Cycle Calculation Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0160_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.131 EMU2nVUFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nVUFIX register, see **Section 24.4.7 (7) Selecting and Limiting Output Voltage of Each Phase** and **Section 24.4.7 (8) Setting PWM**.

24.3.99 EMU2nVVFIX — EMU2n V-phase Output Voltage For Duty Cycle Calculation Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0164_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.132 EMU2nVVFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nVVFIX register, see **Section 24.4.7 (7) Selecting and Limiting Output Voltage of Each Phase** and **Section 24.4.7 (8) Setting PWM**.

24.3.100 EMU2nVWFIX — EMU2n W-phase Output Voltage For Duty Cycle Calculation Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + 0168_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.133 EMU2nVWFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The sign bit is stored.
30 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nVWFIX register, see **Section 24.4.7 (7) Selecting and Limiting Output Voltage of Each Phase** and **Section 24.4.7 (8) Setting PWM**.

24.3.101 EMU2nPWMUIP — EMU2n U-phase Compare Value Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $016C_H$

Value after reset: 0000_H

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

Table 24.134 EMU2nPWMUIP Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nPWMUIP register, see **Section 24.4.7 (8) Setting PWM**.

24.3.102 EMU2nPWMVIP — EMU2n V-phase Compare Value Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $016E_H$

Value after reset: 0000_H

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

Table 24.135 EMU2nPWMVIP Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nPWMVIP register, see **Section 24.4.7 (8) Setting PWM**.

24.3.103 EMU2nPWMWIP — EMU2n W-phase Compare Value Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + 0170_H

Value after reset: 0000_H

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

Table 24.136 EMU2nPWMWIP Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA	Data Data is stored.

For details on calculation that can be monitored with the EMU2nPWMWIP register, see **Section 24.4.7 (8) Setting PWM**.

24.3.104 EMU2nVUOFS — EMU2n U-phase Output Voltage Correction Amount Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0174_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.137 EMU2nVUOFS Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVUOFS register setting, see **Section 24.4.7 (6) Offset Addition**.

24.3.105 EMU2nVVOFS — EMU2n V-phase Output Voltage Correction Amount Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0176_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.138 EMU2nVVOFS Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVVOFS register setting, see **Section 24.4.7 (6) Offset Addition**.

24.3.106 EMU2nVWOFS — EMU2n W-phase Output Voltage Correction Amount Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0178_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.139 EMU2nVWOFS Register Contents

Bit Position	Bit Name	Function
15	S	Sign Set the sign bit.
14 to 0	DATA	Data Set data.

For details on calculation with the EMU2nVWOFS register setting, see **Section 24.4.7 (6) Offset Addition**.

24.3.107 EMU2nRECCTR — EMU2n Rectangle IP Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0180_H

Value after reset: 04_H

Bit	7	6	5	4	3	2	1	0
	FDRCT	—	—	—	—	FIPPOSI	SLCTCMP0	SETREC
Value after reset	0	0	0	0	0	1	0	0
R/W	R/W	R	R	R	R	R/W	R/W	R/W

Table 24.140 EMU2nRECCTR Register Contents

Bit Position	Bit Name	Function
7	FDRCT	Rotation Direction Select Selects the rotation direction. 0: Positive rotation 1: Negative rotation
6 to 3	—	These bits are read as 0. The write value should be 0.
2	FIPPOSI	Feedback Data Select Selects whether to use feedback data for rectangle IP calculation. 0: EMU calculation result 1: User-set value
1	SLCTCMP0	Angle Compare 0 Setting Selects the value to be used for angle compare 0. 0: EMU2nCMP0 register value 1: EMU calculation result
0	SETREC	Batch Rectangular Wave Setting Selects the value to be used to generate batch rectangular wave. 0: User-set value 1: EMU calculation result

This register is used to control rectangle IP calculation. For details, refer to **Section 24.4.8, Rectangle IP**.

24.3.108 EMU2nPTNN — EMU2n Batch Rectangle Output Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0184_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.141 EMU2nPTNN Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	W-Phase High-Side Level Setting on Angle Compare 0 Match Sets the high-side level of W-phase on W-phase compare 0 match. 0: Low level 1: High level
1	VPTN	V-Phase High-Side Level Setting on Angle Compare 0 Match Sets the high-side level of V-phase on V-phase compare 0 match. 0: Low level 1: High level
0	UPTN	U-Phase High-Side Level Setting on Angle Compare 0 Match Sets the high-side level of U-phase on U-phase compare 0 match. 0: Low level 1: High level

Set the high-side level of the rectangular wave to this register when the user-set value is used to generate rectangular waves. Setting the SETREC bit in the EMU2nRECCTR register to 0 enables the pulse of each phase set in this register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.109 EMU2nPTNAB — EMU2n Batch Rectangle Output Pattern AB Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0185_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	WPTNB	VPTNB	UPTNB	WPTNA	VPTNA	UPTNA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.142 EMU2nPTNAB Register Contents

Bit Position	Bit Name	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTNB	3-phase Level Setting (Pattern B, W Phase) Sets the W-phase level for pattern B. 0: Low level output 1: High level output
4	VPTNB	3-phase Level Setting (Pattern B, V Phase) Sets the V-phase level for pattern B. 0: Low level output 1: High level output
3	UPTNB	3-phase Level Setting (Pattern B, U Phase) Sets the U-phase level for pattern B. 0: Low level output 1: High level output
2	WPTNA	3-phase Level Setting (Pattern A, W Phase) Sets the W-phase level for pattern A. 0: Low level output 1: High level output
1	VPTNA	3-phase Level Setting (Pattern A, V Phase) Sets the V-phase level for pattern A. 0: Low level output 1: High level output
0	UPTNA	3-phase Level Setting (Pattern A, U Phase) Sets the U-phase level for pattern A. 0: Low level output 1: High level output

Set the high-side level of the rectangular wave to this register when the IP calculation is used to generate rectangular waves. Setting the SETREC bit in the EMU2nRECCTR register to 1 enables the pulse of each phase set in the EMU2nPTNAB register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time. For details on rectangular wave output, refer to **Section 24.4.8, Rectangle IP**.

24.3.110 EMU2nPTNCD — EMU2n Batch Rectangle Output Pattern CD Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0186_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	WPTND	VPTND	UPTND	WPTNC	VPTNC	UPTNC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.143 EMU2nPTNCD Register Contents

Bit Position	Bit Name	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTND	3-phase Level Setting (Pattern D, W Phase) Sets the W-phase level for pattern D. 0: Low level output 1: High level output
4	VPTND	3-phase Level Setting (Pattern D, V Phase) Sets the V-phase level for pattern D. 0: Low level output 1: High level output
3	UPTND	3-phase Level Setting (Pattern D, U Phase) Sets the U-phase level for pattern D. 0: Low level output 1: High level output
2	WPTNC	3-phase Level Setting (Pattern C, W Phase) Sets the W-phase level for pattern C. 0: Low level output 1: High level output
1	VPTNC	3-phase Level Setting (Pattern C, V Phase) Sets the V-phase level for pattern C. 0: Low level output 1: High level output
0	UPTNC	3-phase Level Setting (Pattern C, U Phase) Sets the U-phase level for pattern C. 0: Low level output 1: High level output

Set the high-side level of the rectangular wave to this register when the IP calculation is used to generate rectangular waves. Setting the SETREC bit in the EMU2nRECCTR register to 1 enables the pulse of each phase set in the EMU2nPTNCD register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time. For details on rectangular wave output, refer to **Section 24.4.8, Rectangle IP**.

24.3.111 EMU2nPTNEF — EMU2n Batch Rectangle Output Pattern EF Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0187_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	WPTNF	VPTNF	UPTNF	WPTNE	VPTNE	UPTNE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.144 EMU2nPTNEF Register Contents

Bit Position	Bit Name	Function
7, 6	—	These bits are read as 0. The write value should be 0.
5	WPTNF	3-phase Level Setting (Pattern F, W Phase) Sets the W-phase level for pattern F. 0: Low level output 1: High level output
4	VPTNF	3-phase Level Setting (Pattern F, V Phase) Sets the V-phase level for pattern F. 0: Low level output 1: High level output
3	UPTNF	3-phase Level Setting (Pattern F, U Phase) Sets the U-phase level for pattern F. 0: Low level output 1: High level output
2	WPTNE	3-phase Level Setting (Pattern E, W Phase) Sets the W-phase level for pattern E. 0: Low level output 1: High level output
1	VPTNE	3-phase Level Setting (Pattern E, V Phase) Sets the V-phase level for pattern E. 0: Low level output 1: High level output
0	UPTNE	3-phase Level Setting (Pattern E, U Phase) Sets the U-phase level for pattern E. 0: Low level output 1: High level output

Set the high-side level of the rectangular wave to this register when the IP calculation is used to generate rectangular waves. Setting the SETREC bit in the EMU2nPTNEF register to 1 enables the pulse of each phase set in the EMU2nPTNEF register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time. For details on rectangular wave output, refer to **Section 24.4.8, Rectangle IP**.

24.3.112 EMU2nCMP0 — EMU2n Compare Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + 0188_H

Value after reset: 0000_H

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

Table 24.145 EMU2nCMP0 Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

This register is used when angle compare 0 value is manually set by the user. It is used as an initial angle compare 0 value.

24.3.113 EMU2nCMP1 — EMU2n Compare Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $018A_H$

Value after reset: 0000_H

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

Table 24.146 EMU2nCMP1 Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set data.

This register is used when angle compare 1 value is manually set by the user. It is used as an initial angle compare 1 value.

24.3.114 EMU2nPHQSFT — EMU2n q-Axis Reference Voltage Phase Software Input Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $018C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	S					DATA										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.147 EMU2nPHQSFT Register Contents

Bit Position	Bit Name	Function
15 to 11	S	Sign Set the sign bit. Set 0's to bits 15 to 11 for a positive value, and 1's for a negative value.
10 to 0	DATA	Data Set data.

For details on the calculation using the EMU2nPHQSFT register value, refer to **Section 24.4.8 (2) Generating Switching Instruction**.

24.3.115 EMU2nPSWSFT — EMU2n Phase Differential Software Input Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $018E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.148 EMU2nPSWSFT Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data Set data.

For details on the calculation using the EMU2nPSWSFT register value, refer to **Section 24.4.8 (2) Generating Switching Instruction**. A value from 0 to 5 can be set.

24.3.116 EMU2nPSW — EMU2n Phase Differential Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + $018F_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DATA		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.149 EMU2nPSW Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2 to 0	DATA	Data Data is stored.

For details on the calculation that can be monitored by the EMU2nPSW register, refer to **Section 24.4.8 (2) Generating Switching Instruction**.

24.3.117 EMU2nIPCMP0 — EMU2n IP Compare Value 0 Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + 0190_H

Value after reset: 0000_H

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

Table 24.150 EMU2nIPSCMP0 Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Data is stored.

24.3.118 EMU2nIRECCTR — EMU2n Independent Rectangle IP Control Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + 0198_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RECMD
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.151 EMU2nIRECCTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are read as 0. The write value should be 0.
0	RECMD	Rectangular Wave Output Mode Switching 0: Batch rectangular wave output 1: Independent rectangular wave output

24.3.119 EMU2nIRPTN — EMU2n Independent Rectangle Output Pattern Update Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $019C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WINIPTN	VINIPTN	UINIPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.152 EMU2nIRPTN Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WINIPTN	W-Phase Pattern 0: Low level 1: High level
1	VINIPTN	V-Phase Pattern 0: Low level 1: High level
0	UINIPTN	U-Phase Pattern 0: Low level 1: High level

Set the high-side level of the rectangular wave to this register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.120 EMU2nIRCTRST — EMU2n Independent Rectangle IP Flag/Select Signal Reset Register (n = 0, 1)

Access: Readable/writable in 8-bit units.

Address: <EMU2n_base> + $019E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WINIT	VINIT	UINIT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 24.153 EMU2nIRCTRST Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WINIT	W-Phase Flag/Select Signal Reset The flag/select signals of W-phase buffers 0 to 2 become the value after reset by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
1	VINIT	V-Phase Flag/Select Signal Reset The flag/select signals of V-phase buffers 0 to 2 become the value after reset by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.
0	UINIT	U-Phase Flag/Select Signal Reset The flag/select signals of U-phase buffers 0 to 2 become the value after reset by writing 1 to this bit. After 1 is written, this bit automatically becomes 0. Writing 0 to this bit is disabled. This bit is always read as 0.

24.3.121 EMU2nIRUCPPN0 — EMU2n Independent Rectangle IP U-phase Compare/ Pattern Setting Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01A0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UPTN0	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.154 EMU2nIRUCPPN0 Register Contents

Bit Position	Bit Name	Function
15	UPTN0	U-Phase Output Pattern 0 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in this register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.122 EMU2nIRUCPPN1 — EMU2n Independent Rectangle IP U-phase Compare/ Pattern Setting Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01A2_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UPTN1	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.155 EMU2nIRUCPPN1 Register Contents

Bit Position	Bit Name	Function
15	UPTN1	U-Phase Output Pattern 1 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRUCPPN1 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.123 EMU2nIRUCPPN2 — EMU2n Independent Rectangle IP U-phase Compare/ Pattern Setting Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01A4_H$

Value after reset: 0000_H

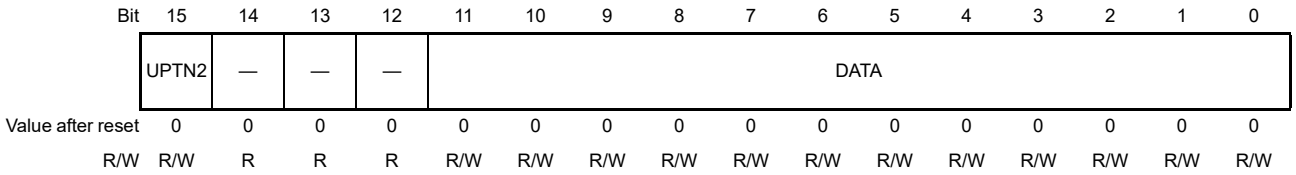


Table 24.156 EMU2nIRUCPPN2 Register Contents

Bit Position	Bit Name	Function
15	UPTN2	U-Phase Output Pattern 2 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRUCPPN2 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.124 EMU2nIRVCPPN0 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01A8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPTN0	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.157 EMU2nIRVCPPN0 Register Contents

Bit Position	Bit Name	Function
15	VPTN0	V-Phase Output Pattern 0 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRVCPPN0 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.125 EMU2nIRVCPPN1 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01AA_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPTN1	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.158 EMU2nIRVCPPN1 Register Contents

Bit Position	Bit Name	Function
15	VPTN1	V-Phase Output Pattern 1 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRVCPPN1 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.126 EMU2nIRVCPPN2 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01AC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VPTN2	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.159 EMU2nIRVCPPN2 Register Contents

Bit Position	Bit Name	Function
15	VPTN2	V-Phase Output Pattern 2 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRVCPPN2 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.127 EMU2nIRWCPPN0 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01B0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WPTN0	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.160 EMU2nIRWCPPN0 Register Contents

Bit Position	Bit Name	Function
15	WPTN0	W-Phase Output Pattern 0 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRWCPPN0 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.128 EMU2nIRWCPPN1 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01B2_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WPTN1	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.161 EMU2nIRWCPPN1 Register Contents

Bit Position	Bit Name	Function
15	WPTN1	W-Phase Output Pattern 1 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRWCPPN1 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.129 EMU2nIRWCPPN2 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <EMU2n_base> + $01B4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WPTN2	—	—	—	DATA											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 24.162 EMU2nIRWCPPN2 Register Contents

Bit Position	Bit Name	Function
15	WPTN2	W-Phase Output Pattern 2 0: Low level 1: High level
14 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data Set the compare value.

Set the high-side level of the rectangular wave to this register. Based on the values set in the EMU2nIRWCPPN2 register, TSG3 outputs the high-side signal and low-side signal including dead time.

24.3.130 EMU2nIRFLGM — EMU2n Independent Rectangle IP Flag Monitor Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $01B8_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WFLG2	WFLG1	WFLG0	VFLG2	VFLG1	VFLG0	UFLG2	UFLG1	UFLG0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.163 EMU2nIRFLGM Register Contents

Bit Position	Bit Name	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	WFLG2	W-Phase Flag 2 Indicates the flag status of W-phase buffer 2. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
7	WFLG1	W-Phase Flag 1 Indicates the flag status of W-phase buffer 1. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
6	WFLG0	W-Phase Flag 0 Indicates the flag status of W-phase buffer 0. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
5	VFLG2	V-Phase Flag 2 Indicates the flag status of V-phase buffer 2. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
4	VFLG1	V-Phase Flag 1 Indicates the flag status of V-phase buffer 1. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
3	VFLG0	V-Phase Flag 0 Indicates the flag status of V-phase buffer 0. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
2	UFLG2	U-Phase Flag 2 Indicates the flag status of U-phase buffer 2. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
1	UFLG1	U-Phase Flag 1 Indicates the flag status of U-phase buffer 1. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.
0	UFLG0	U-Phase Flag 0 Indicates the flag status of U-phase buffer 0. 0: Before the CPU write to the buffer, or after the compare match occurrence. 1: Between the CPU write to the buffer and compare match occurrence.

24.3.131 EMU2nIRSELM — EMU2n Independent Rectangle IP Select Signal Monitor Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $01BA_H$

Value after reset: 0049_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WSEL[2:0]			VSEL[2:0]			USEL[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.164 EMU2nIRSELM Register Contents

Bit Position	Bit Name	Function
15 to 9	—	These bits are read as 0. The write value should be 0.
8	WSEL2	W-Phase Select Signal*1 Indicates the select signal value of the W-phase buffers. 0 0 1: Buffer 0 0 1 0: Buffer 1 1 0 0: Buffer 2
7	WSEL1	
6	WSEL0	
5	VSEL2	V-Phase Select Signal*1 Indicates the select signal value of the V-phase buffers. 0 0 1: Buffer 0 0 1 0: Buffer 1 1 0 0: Buffer 2
4	VSEL1	
3	VSEL0	
2	USEL2	U-Phase Select Signal*1 Indicates the select signal value of the U-phase buffers. 0 0 1: Buffer 0 0 1 0: Buffer 1 1 0 0: Buffer 2
1	USEL1	
0	USEL0	

Note 1. The value changes as 001_B => 010_B => 100_B => 001_B => ….

24.3.132 EMU2nCBADmk — EMU2n ADm Data k Verification Buffer Register (n = 0, 1) (m = 0, 1) (k = 0 to 2)

Access: Readable in 16-bit units.

Address: EMU20CBAD00:<EMU20_base>+$01C0_H$, EMU20CBAD01:<EMU20_base>+$01C2_H$, EMU20CBAD02:<EMU20_base>+$01C4_H$
EMU21CBAD10:<EMU21_base>+$01C0_H$, EMU21CBAD11:<EMU21_base>+$01C2_H$, EMU21CBAD12:<EMU21_base>+$01C4_H$

Value after reset: 0000_H

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

Table 24.165 EMU2nCBADmk Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data The buffering result of the EMU2nADmk register is stored.

24.3.133 EMU2nCBTHTREFIXIN — EMU2n Resolver Angle Verification Buffer Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <EMU2n_base> + $01C6_H$

Value after reset: 0000_H

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

Table 24.166 EMU2nCBTHTREFIXIN Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are read as 0. The write value should be 0.
11 to 0	DATA	Data The buffering result of the EMU2nTHTREFIXIN register is stored.

24.3.134 EMU2nCBIDFIX — EMU2n d-Axis Current Value Verification Buffer Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $01C8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.167 EMU2nCBIDFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The buffering result of the EMU2nIDFIX register is stored.
30 to 0	DATA	Data The buffering result of the EMU2nIDFIX register is stored.

24.3.135 EMU2nCBIQFIX — EMU2n q-Axis Current Value Verification Buffer Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $01CC_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	S	DATA														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

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

Table 24.168 EMU2nCBIQFIX Register Contents

Bit Position	Bit Name	Function
31	S	Sign The buffering result of the EMU2nIQFIX register is stored.
30 to 0	DATA	Data The buffering result of the EMU2nIQFIX register is stored.

24.3.136 EMU2nCBPWMUIP — EMU2n U-phase Compare Value Verification Buffer Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $01D0_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.169 EMU2nCBPWMUIP Register Contents

Bit Position	Bit Name	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data The buffering result of the EMU2nPWMUIP register is stored.

24.3.137 EMU2nCBPWMVIP — EMU2n V-phase Compare Value Verification Buffer Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $01D4_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.170 EMU2nCBPWMVIP Register Contents

Bit Position	Bit Name	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data The buffering result of the EMU2nPWMVIP register is stored.

24.3.138 EMU2nCBPWMWIP — EMU2n W-phase Compare Value Verification Buffer Register (n = 0, 1)

Access: Readable in 32-bit units.

Address: <EMU2n_base> + $01D8_H$

Value after reset: $0000\ 0000_H$

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

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

Table 24.171 EMU2nCBPWMWIP Register Contents

Bit Position	Bit Name	Function
31 to 18	—	These bits are read as 0. The write value should be 0.
17 to 0	DATA	Data The buffering result of the EMU2nPWMWIP register is stored.

24.3.139 EMU2nCBBREC — EMU2n Batch Rectangle Pattern Verification Buffer Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + $01DC_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.172 EMU2nCBBREC Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	Batch Rectangle W-Phase Pattern Value The buffering result of the batch rectangle W-phase pattern value is stored.
1	VPTN	Batch Rectangle V-Phase Pattern Value The buffering result of the batch rectangle V-phase pattern value is stored.
0	UPTN	Batch Rectangle U-Phase Pattern Value The buffering result of the batch rectangle U-phase pattern value is stored.

24.3.140 EMU2nCBIREC — EMU2n Independent Rectangle Pattern Verification Buffer Register (n = 0, 1)

Access: Readable in 8-bit units.

Address: <EMU2n_base> + $01DD_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	WPTN	VPTN	UPTN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 24.173 EMU2nCBIREC Register Contents

Bit Position	Bit Name	Function
7 to 3	—	These bits are read as 0. The write value should be 0.
2	WPTN	Independent Rectangle W-Phase Pattern Value The buffering result of the independent rectangle W-phase pattern value is stored.
1	VPTN	Independent Rectangle V-Phase Pattern Value The buffering result of the independent rectangle V-phase pattern value is stored.
0	UPTN	Independent Rectangle U-Phase Pattern Value The buffering result of the independent rectangle U-phase pattern value is stored.

24.4 Operations

24.4.1 EMU2 Initialization

Figure 24.3 shows the initial setting flow for the EMU2.

Use the following procedure as well when restarting the EMU2 operation by the EMUST bit in the EMU2nCTR register after being disabled.

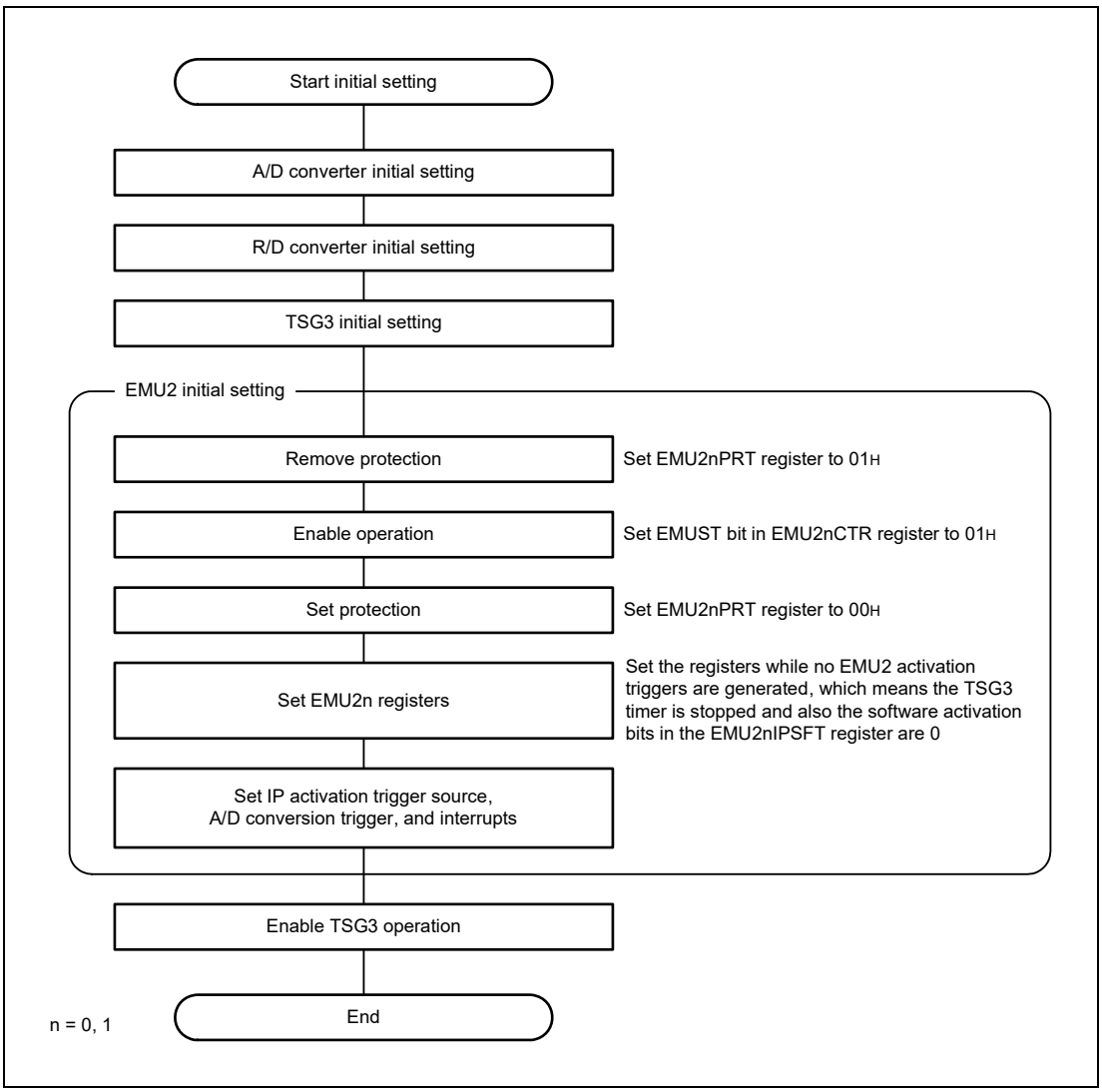


Figure 24.3 Initial Setting Flow for EMU2

24.4.2 A/D Conversion Trigger Setting

By setting bits CAMOUAD, CAVALAD, and CMPAD in the EMU2nADTRG register, the A/D conversion trigger for ADCCm can be generated at the crest/trough of the carrier, or on angle compare 0 match. **Figure 24.4** shows the configuration of the A/D conversion trigger generation function.

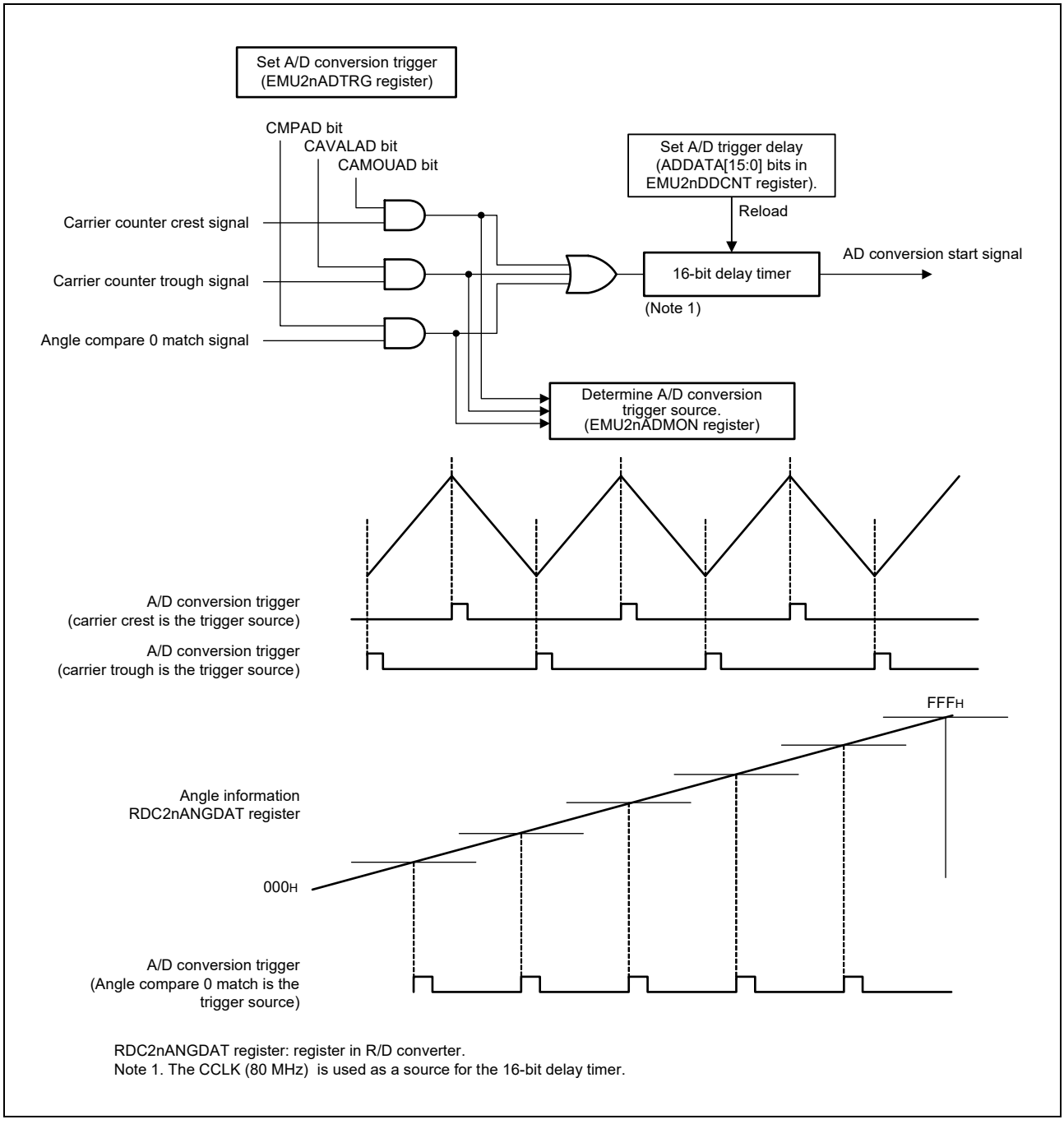


Figure 24.4 Configuration of A/D Conversion Trigger Generation Function

The A/D conversion trigger source can be determined by reading the EMU2nADMON register. Even if an A/D conversion start source occurs during A/D conversion, an A/D conversion trigger is ignored; however, the corresponding A/D conversion trigger source determination bit in the EMU2nADMON register becomes 1.

Each of the source determination bits in the EMU2nADMON register becomes 0 by writing 1 to the corresponding bit in the EMU2nADMONC register.

By setting a value to the ADDATA[15:0] bits in the EMU2nDDCNT register, generation of an A/D conversion trigger can be delayed for any length of time from occurrence of the A/D conversion start source. The counter for defining the delay is a 16-bit timer which starts decrementing on occurrence of the A/D conversion start source. Then, the A/D conversion trigger is generated when the counter reaches 0.*1

Note 1. If another A/D conversion start source occurs during decrementing, it is ignored.

24.4.3 Calculation Block

The calculation block has the calculator dedicated for motor control, which enables to control motor without hardly affecting the CPU processing speed.

The block consists of the angle generation IP, input IP, PI control IP, PWM IP, and rectangle IP.

Figure 24.5 shows the calculation block overview.

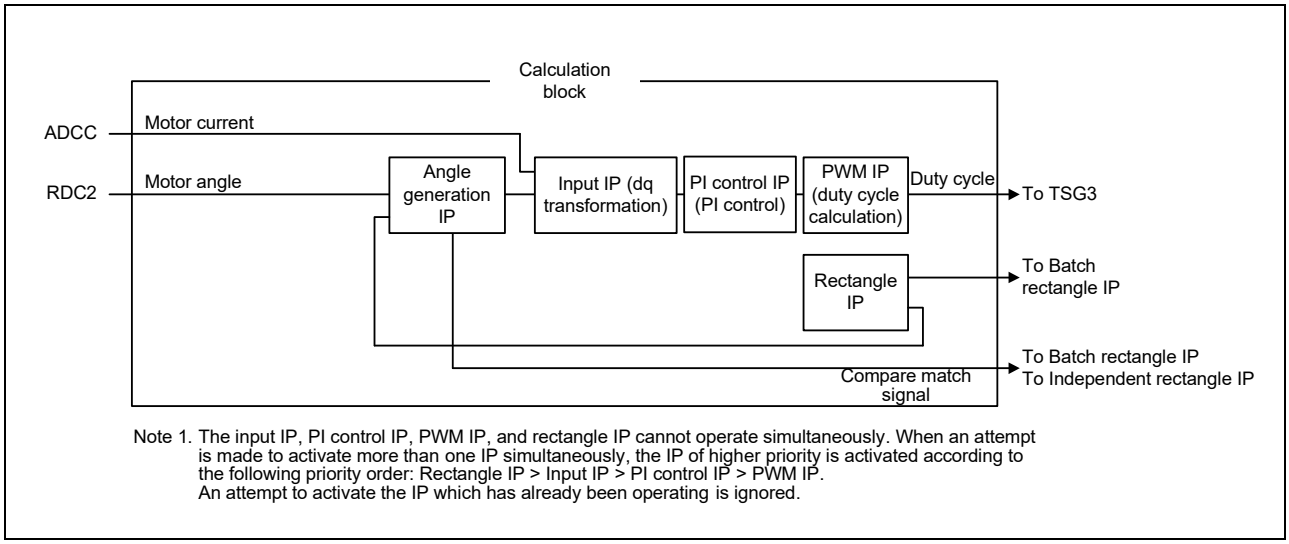


Figure 24.5 Calculation Block Overview

24.4.3.1 Representations of IP Module Calculations

1. Variables are suffixed with the information indicating whether it is signed or unsigned, and the valid bit length. _s and _u represent signed and unsigned respectively, and are followed by the bit length. For a signed variable, the most significant bit is the sign bit.

   Example:
   Ia_v_Lo_fix_s[31:0] represents data from 0th to 31st bits of variable Ia_v_Lo_fix, and the 31st bit is a sign bit.

2. An overflow occurs if a result of operation expression exceeds the specified number of bits.

   Example:
   A: $7FFF_H$ B: $7FFF_H$

   C_s[15:0] ← A_s[15:0] + B_s[15:0]

Note 1. If an overflow occurs as a result of calculation by the input IP, the INIPOF bit in the EMU2nOFMON register becomes 1. If an overflow occurs as a result of calculation by the PI control IP, the PIIPOF bit in the EMU2nOFMON register becomes 1. If an overflow occurs as a result of calculation by the PWM IP, the PWMIPOF bit in the EMU2nOFMON register becomes 1.

3. When an operation expression is immediately followed by division or right shift, bit extension is first performed and then division or right shift is performed before the result is stored.

   Example:
   C_s[15:0] ← (A_s[15:0] × B_s[15:0]) >> 10

   The result of A_s[15:0] × B_s[15:0] is first extended to be 32 bits long, then shifted to the right by 10 bits, and the 16 bits of the result are stored in C_s[15:0].

   (Similarly, when A, B, and C are all 32 bits long, the result of A_s[31:0] × B_s[31:0] is first extended to be 64 bits long, then shifted to the right by 10 bits, and the 32 bits of the result are stored.)

   D_s[15:0] ← (A_s[15:0] × B_s[15:0]) / C_s[15:0]

   The result of (A_s[15:0] × B_s[15:0]) is stored after being extended to be 32 bits long, then divided by C_s[15:0], and the result is stored in D_s[15:0].

4. An operation expression is bit-extended to match with the bit width of the storage location.

   Example:

   D_s[31:0] ← C_s[31:0] + A_s[15:0] × B_s[15:0]

   The result of (A_s[15:0] × B_s[15:0]) is extended to be 32 bits long, and then added to C_s[31:0] as it is.

CAUTION

If the result of adding the value to C_s[31:0] exceeds 32 bits, an overflow occurs.

Example:
C_s[15:0] ← A_s[11:0] + B_s[11:0]

The result of (A_s[11:0] + B_s[11:0]) is sign-extended to be 16 bits long, and then stored in C_s[15:0].

5. | Variable | means finding the absolute value.

6. Division is performed as follows.

The fractional portion is rounded down in division.

Example:
5/2 = 2
-5/2 = -2

(1) Division: 16 bits ← 32 bits/16 bits
0/0 = 0
Negative value/0 = 8000_H
Positive value/0 = $7FFF_H$

(2) Division: 32 bits ← 32 bits/16 bits
0/0 = 0
Negative value/0 = $8000\ 0000_H$
Positive value/0 = $7FFF\ FFFF_H$

(3) Division: 16 bits ← 32 bits/16 bits
0/0 = 0
If $7FFF_H \leq$ result, result ← $7FFF_H$
If $8000_H \geq$ result, result ← 8000_H

7. The variables represented in the lower case alphanumeric characters in this manual are basically set in the EMU2 registers, bits, and cache; they cannot be set by the user directly. The value after reset of the EMU2 internal register is 0, unless otherwise noted.

8. In the following tables using a double line as a border, the left side of the double-line border represents the object to which conditions are applied, and the right side represents the location in which the results are stored. Items in the tables with double-lined border in this section (see the following table) is interpreted as “When object A satisfies condition B, result E is stored in D, or when object A satisfies condition C, result F is stored in D.”

Example:

Object A to which Conditions are Applied	Location D in which Results are Stored
Condition B	Result E
Condition C	Result F

24.4.3.2 Registers to be Set before IP Activation

The registers listed in **Table 24.174** are not provided with buffering scheme. Therefore, modifying these registers during IP operation leads to undefined behavior. Be sure to set these registers before activating the IP. **Table 24.174** lists the registers to be set before activating IP.

Table 24.174 Registers to be Set before Activating IP

Register Name	Symbol
EMU2n PWM lower limit register	EMU2nPWMLL
EMU2n PWM upper limit register	EMU2nPWMUL
EMU2n batch rectangle output pattern AB register	EMU2nPTNAB
EMU2n batch rectangle output pattern CD register	EMU2nPTNCD
EMU2n batch rectangle output pattern EF register	EMU2nPTNEF
EMU2n compare judgment correction register 0	EMU2nCPJUD0
EMU2n compare judgment correction register 1	EMU2nCPJUD1
EMU2n electrical angle generation coefficient register	EMU2nPXR
EMU2n digit alignment register 1	EMU2nPWMK1
EMU2n digit alignment register 2	EMU2nPWMK2
EMU2n duty cycle upper limit register	EMU2nDTUL
EMU2n duty cycle lower limit register	EMU2nDTLL

24.4.3.3 Registers Updated upon IP Completion

Some of the calculation results output by the EMU2 are updated upon completion of each IP.

Table 24.175 lists the registers updated upon completion of the input IP, and **Figure 24.6** shows the register update timing using the IP completion signal.

Table 24.175 Registers Updated upon IP Completion

Register Name	Symbol
EMU2n resolver angle monitor register	EMU2nTHTREFIXIN
EMU2n A/Di channel 0 conversion value register	EMU2nADFIXm0
EMU2n A/Di channel 1 conversion value register	EMU2nADFIXm1
EMU2n A/Di channel 2 conversion value register	EMU2nADFIXm2
EMU2n U-phase current value register	EMU2nIUFIX
EMU2n V-phase current value register	EMU2nIVFIX
EMU2n W-phase current value register	EMU2nIWFIX
EMU2n d-axis current value register	EMU2nIDFIX
EMU2n q-axis current value register	EMU2nIQFIX

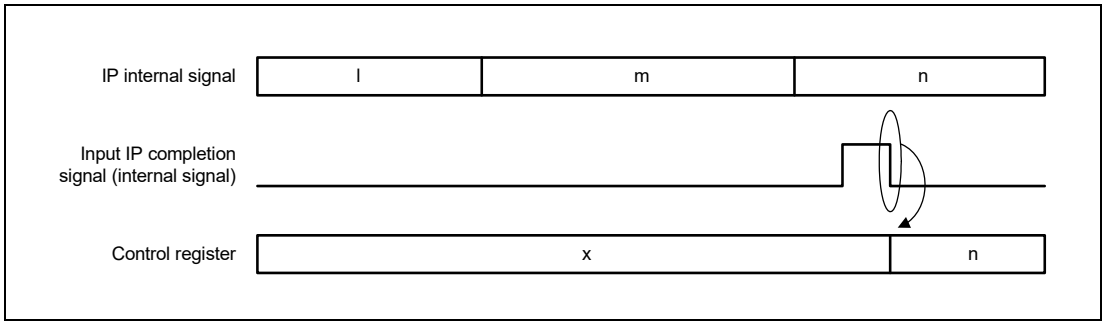


Figure 24.6 Register Update Timing Triggered by IP Completion Signal

24.4.4 Angle Generation IP

The angle generation IP is activated on each change in the angle data, generates the electrical angle, and checks for a compare match. If an activation request is made during operation, it is ignored.

Figure 24.7 shows the process flow of the angle generation IP. **Figure 24.8** shows an example of initial setting procedure for the angle generation IP.

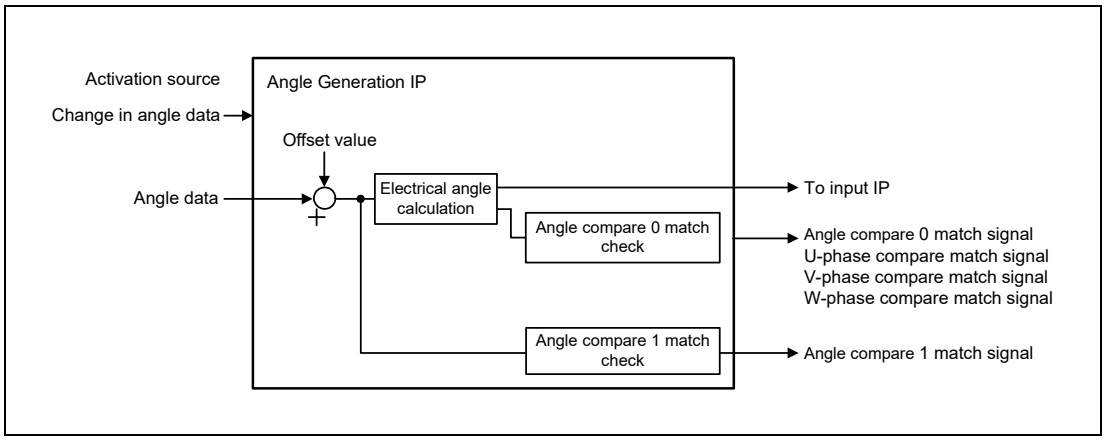


Figure 24.7 Process Flow of Angle Generation IP

(1) Initial Setting of Angle Generation IP

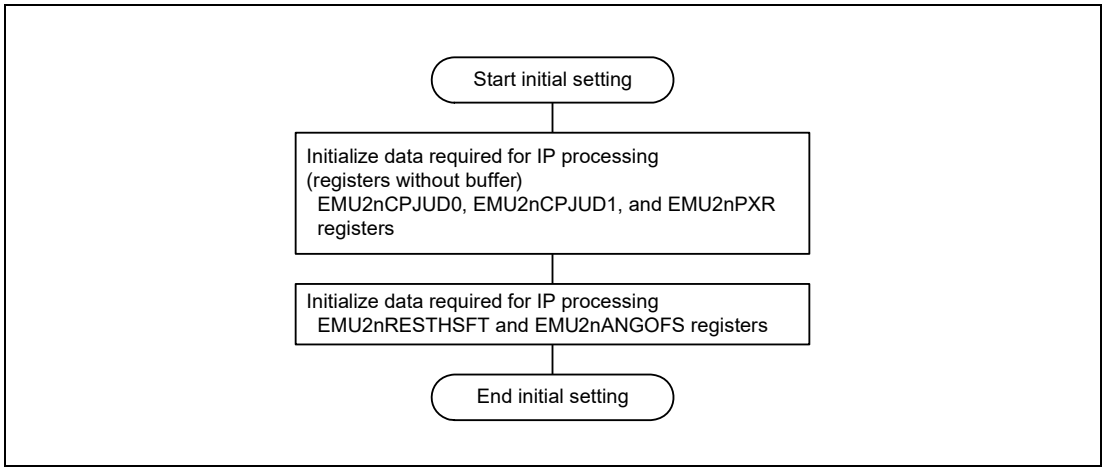


Figure 24.8 Example of Initial Setting Procedure for Angle Generation IP

(2) Obtaining Angle Data

When RDPHIEN bit = 1 and PHISEL bit = 0 in the EMU2nANGCTR register
θorg0_u[11:0] ← angle data

When the PHISEL bit = 1 in the EMU2nANGCTR register
θorg0_u[11:0] ← EMU2nRESTHSFT_u[11:0]

EMU2nRESTHETA_u[11:0] ← θorg0_u[11:0]

θres_fix_u[11:0] ← (θorg0_u[11:0] + EMU2nANGOFS_s[15:0]) & $0FFF_H$

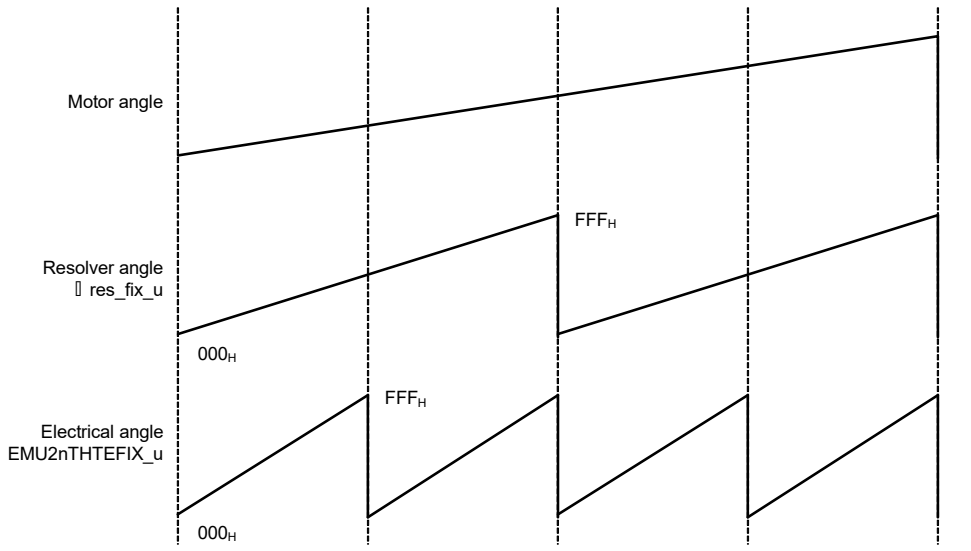
(3) Generating Electrical Angle

The electrical angle is generated as follows.

The EMU2nRESCNT register is incremented by one when the angle data from the resolver overflows, and decremented by one when the angle data from the resolver underflows.

Set the following value to the EMU2nRESRLD register: (number of resolver poles) - 1.

Set the following value to the EMU2nPXR register: ((number of motor pole pairs) / (number of resolver poles)) × 256



When the number of resolver poles is 2 and the number of motor pole pairs is 4, this is an example of an output of an electrical angle from a resolver angle in the following settings:
Set EMU2nRESRLD to 1_H, the number of resolver poles (2)−1.
Set EMU2nPXR to 0200_H, (number of motor pole pairs)/(number of resolver poles) × 256.

Figure 24.9 Example of Electrical Angle Output

Condition 1 (overflow): $F00_H$ < the previous value of θres_fix_u[11:0] and the current value of θres_fix_u[11:0] < $0FF_H$.

Condition 2 (underflow): $F00_H$ < the current value of θres_fix_u[11:0] and the previous value of θres_fix_u[11:0] < $0FF_H$.

When condition 1 is satisfied, bits EMU2nRESCNT_u[2:0] is incremented by 1, and

when condition 2 is satisfied, bits EMU2nRESCNT_u[2:0] is decremented by 1.

When EMU2nRESCNT_u[2:0] > EMU2nRESRLD_u[2:0]:

EMU2nRESCNT_u[2:0] ← 0

When EMU2nRESCNT_u[2:0] < 0:

(i.e. when condition 2 is satisfied and the result of “EMU2nRESCNT_u[2:0] - 1” is less than 0)

EMU2nRESCNT_u[2:0] ← EMU2nRESRLD_u[2:0]

EMU2nTHTEFIX_u[11:0] ← 0FFFH & ((EMU2nPXR_s[15:0] × ((EMU2nRESCNT_u[2:0] << 12) + θres_fix_u[11:0])) >>8)

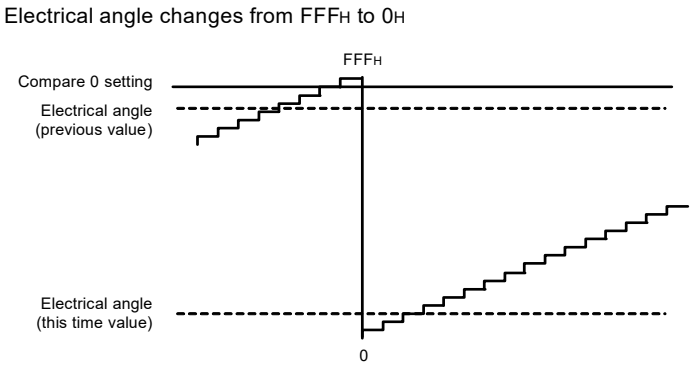
(4) Determining Angle Compare 0 Match

A compare 0 match is determined to have occurred when the following condition is satisfied.

For positive rotation:
electrical angle (previous value) < compare 0 setting ≤ electrical angle (this time value)

For negative rotation:
electrical angle (previous value) > compare 0 setting ≥ electrical angle (this time value)

However, the following conditions 1 and 2 are added to the preceding conditions in order to determine whether or not a change in the electrical angle from FFF_H to 0_H (or from 0_H to FFF_H) has occurred between the previous electrical angle and this electrical angle.



Condition 1: “FFF_H - EMU2nCPJUD0_u[7:0] < EMU2nTHTEFIX_u[11:0](previous value)”
AND “EMU2nTHTEFIX_u[11:0](this time value) < EMU2nCPJUD0_u[7:0]”

Condition 2: “FFF_H - EMU2nCPJUD0_u[7:0] < EMU2nTHTEFIX_u[11:0](this time value)”
AND “EMU2nTHTEFIX _u[11:0](previous value) < EMU2nCPJUD0_u[7:0]”

For positive rotation:
“condition 1 = not satisfied”
AND “condition 2 = not satisfied”
AND “EMU2nTHTEFIX_u[11:0](previous value) < compare 0 ≤ EMU2nTHTEFIX_u[11:0](this time value)”

For negative rotation:
“condition 1 = not satisfied”
AND “condition 2 = not satisfied”
AND “EMU2nTHTEFIX_u[11:0](previous value) > compare 0 ≥ EMU2nTHTEFIX _u[11:0](this time value)”

For positive rotation and electrical angle changing from FFF_H to 0_H:
“condition 1 = satisfied”
AND (“EMU2nTHTEFIX_u[11:0](previous value) < compare 0”
OR “compare 0 ≤ EMU2nTHTEFIX _u[11:0](this time value)”)

For negative rotation and electrical angle changing from 0_H to FFF_H:
“condition 2 = satisfied”
AND(“EMU2nTHTEFIX_u[11:0](previous value) > compare 0”
OR “compare 0 ≥ EMU2nTHTEFIX _u[11:0](this time value)”)

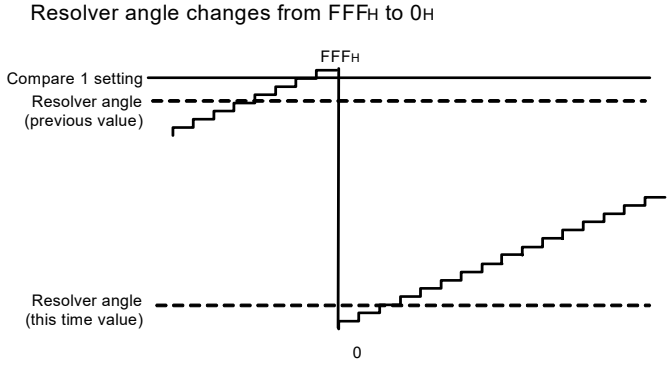
(5) Determining Angle Compare 1 Match

A compare 1 match is determined to have occurred when the following condition is satisfied.

For positive rotation:
resolver angle (previous value) < compare 1 setting ≤ resolver angle(this time value)

For negative rotation:
resolver angle (previous value) > compare 1 setting ≥ resolver angle(this time value)

However, the following conditions 1 and 2 are added to the preceding conditions in order to determine whether or not a change in the resolver angle from FFF_H to 0_H (or from 0_H to FFF_H) has occurred between the previous resolver angle and this resolver angle.



Condition 1: “FFF_H-EMU2nCPJUD1_u[7:0] < EMU2nRESTHETA_u[11:0](previous value)”
AND “EMU2nRESTHETA_u[11:0](this time value) < EMU2nCPJUD1_u[7:0]”

Condition 2: “FFF_H-EMU2nCPJUD1_u[7:0] < EMU2nRESTHETA_u[11:0](this time value)”
AND “EMU2nRESTHETA_u[11:0](previous value) < EMU2nCPJUD1_u[7:0]”

For positive rotation:
“condition 1 = not satisfied”
AND “condition 2 = not satisfied”
AND “EMU2nRESTHETA_u[11:0](previous value) < compare 1 ≤ EMU2nRESTHETA_u[11:0](this time value)”

For negative rotation:
“condition 1 = not satisfied”
AND “condition 2 = not satisfied”
AND “EMU2nRESTHETA_u[11:0](previous value) > compare 1 ≥ EMU2nRESTHETA_u[11:0](this time value)”

For positive rotation and resolver angle changing from FFF_H to 0_H:
“condition 1 = satisfied”
AND (“EMU2nRESTHETA_u[11:0](previous value) < compare 1”
OR “compare 1 ≤ EMU2nRESTHETA_u[11:0](this time value)”)

For negative rotation and resolver angle changing from 0_H to FFF_H:
“condition 2 = satisfied”
AND (“EMU2nRESTHETA_u[11:0](previous value) > compare 1”
OR “compare 1 ≥ EMU2nRESTHETA_u[11:0](this time value)”)

(6) U/V/W-Phase Angle Compare Match for Independent Rectangle

The angle compare values of the U/V/W phases, which are output by the independent rectangle block, are compared with the electrical angle. A compare match is determined in the same way as the angle compare 0 match.

24.4.5 Input IP

The input IP obtains the motor current value and performs dq transformation and other calculations. The A/D value indicating the motor current is automatically input to the input IP.

Figure 24.10 shows the process flow of the input IP.

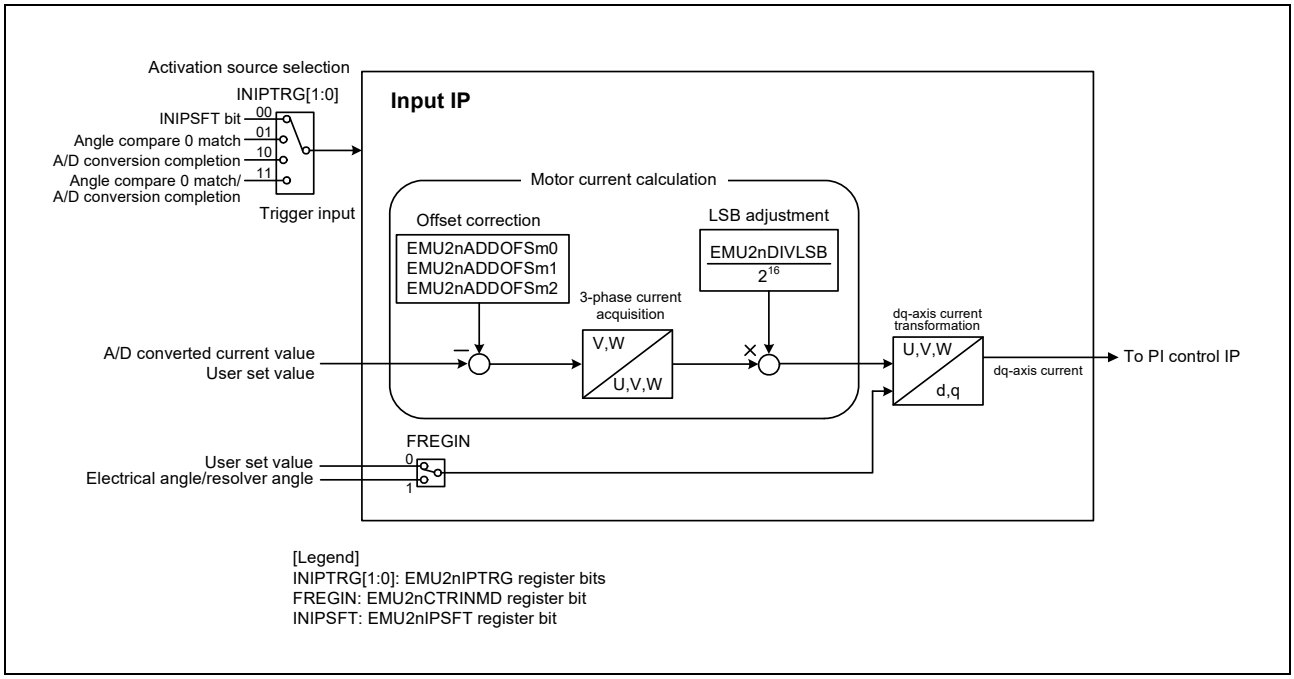


Figure 24.10 Process Flow of Input IP

After the activation source is selected using the INIPTRG[1:0] bits in the EMU2nIPTRG register, the input IP is activated upon occurrence of the selected source. If the software trigger is selected as the activation source, setting the INIPSFT bit in the EMU2nIPSFT register to 1 can activate the input IP.

When the input IP completes its processing, the INIF bit in the EMU2nINTSD register becomes 1. The INIF bit becomes 0 by writing 1 to the INIFC bit in the EMU2nINTSDC register.

An interrupt can be generated at the completion of the input IP.

Figure 24.11 shows an example of initial setting procedure for the input IP (when an interrupt is used).

(1) Initial Setting of Input IP

The input IP can be initialized according to **Figure 24.11** Example of Initial Setting Procedure for Input IP.

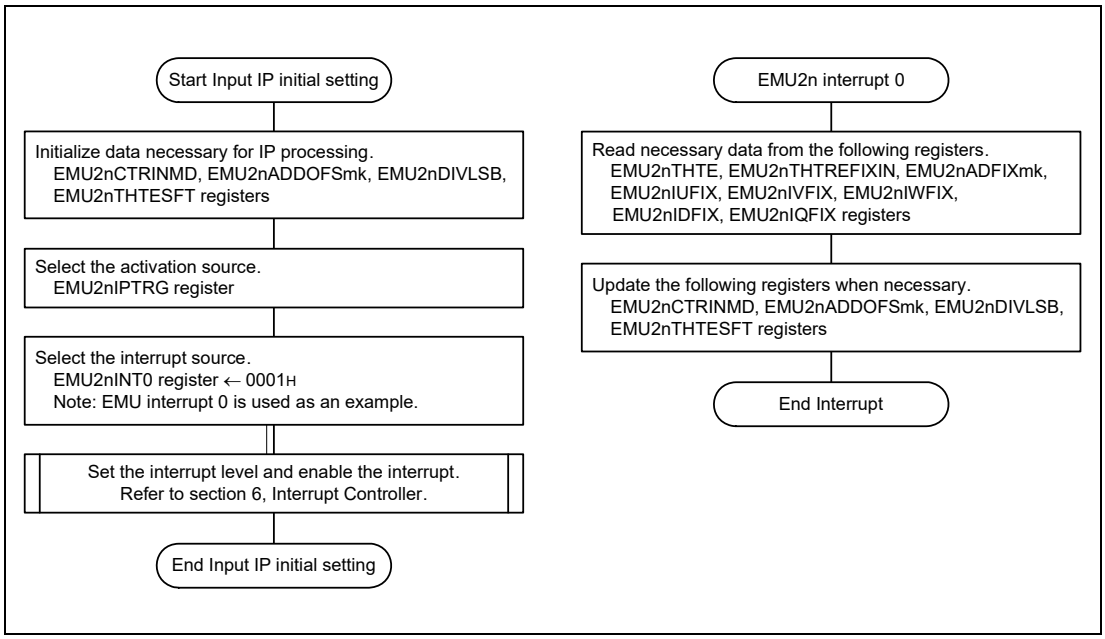


Figure 24.11 Example of Initial Setting Procedure for Input IP (when an interrupt is used)

The following describes the input IP processing in detail.

(2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results

Select the electrical angle source.

EMU2nTHTREFIXIN_u[11:0] ← θres_fix_in_u[11:0] ← θres_fix _u[11:0]

Note 1. θres_fix_u is generated by the angle generation IP.

Note 2. Transfer of θres_fix_in_u[11:0] ← θres_fix _u[11:0] is carried out after the R/D conversion trigger output delay time has elapsed, and transfer of EMU2nTHTREFIXIN_u[11:0] ← θres_fix_in_u[11:0] is carried out upon input IP completion.

EMU2nTHTE_u[11:0] ← EMU2nTHTEFIX_u[11:0]

Note 1. EMU2nTHTEFIX_u is generated by the angle generation IP.

Note 2. The above transfer is carried out after the R/D conversion trigger output delay time set in the EMU2nDDCNT register has elapsed.

FREGIN Bit in EMU2nCTRINMD Register	Electrical Angle θe_u[11:0]
0 (user-set value used)	EMU2nTHTESFT_u[11:0]
1 (electrical angle, resolver angle used)	EMU2nTHTE_u[11:0] + EMU2nEARD_u[11:0]

Note 1. Based on the electrical angle from the angle generation IP and correction value, the electrical angle to be used by the input IP is generated.

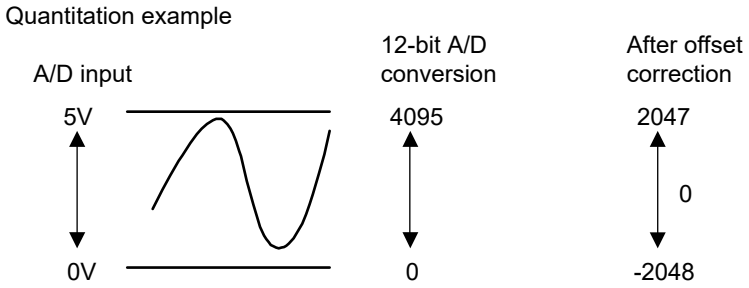
Transfer the values when the input IP is activated.

EMU2nADFIXm0_s[15:0] ← EMU2nADm0_u[11:0]

EMU2nADFIXm1_s[15:0] ← EMU2nADm1_u[11:0]

EMU2nADFIXm2_s[15:0] ← EMU2nADm2_u[11:0]

(3) Calculating Motor Current



ia_0ch_tmp1_s[15:0] ← EMU2nADFIXm0_s[15:0] - EMU2nADDOFSm0_s[15:0]

ia_1ch_tmp1_s[15:0] ← EMU2nADFIXm1_s[15:0] - EMU2nADDOFSm1_s[15:0]

ia_2ch_tmp1_s[15:0] ← EMU2nADFIXm2_s[15:0] - EMU2nADDOFSm2_s[15:0]

When the CMES bit in the EMU2nCTRINMD register = 0:

The U-phase current is generated based on the V- and W-phase currents.

EMU2nIVFIX_s[31:0] ← (ia_0ch_tmp1_s[15:0] × EMU2nDIVLSB_s[31:0]) >>16

EMU2nIWFIX_s[31:0] ← (ia_1ch_tmp1_s[15:0] × EMU2nDIVLSB_s[31:0]) >>16

EMU2nIUFIX_s[31:0] ← - (EMU2nIVFIX_s[31:0] + EMU2nIWFIX_s[31:0])

When the CMES bit in the EMU2nCTRINMD register = 1:

EMU2nIVFIX_s[31:0] ← (ia_0ch_tmp1_s[15:0] × EMU2nDIVLSB_s[31:0]) >>16

EMU2nIWFIX_s[31:0] ← (ia_1ch_tmp1_s[15:0] × EMU2nDIVLSB_s[31:0]) >>16

EMU2nIUFIX_s[31:0] ← (ia_2ch_tmp1_s[15:0] × EMU2nDIVLSB_s[31:0]) >>16

EMU2nIVFIX_s[31:0] ← - (EMU2nIWFIX_s[31:0] + EMU2nIUFIX_s[31:0]) (CMUVW[2:0] bits = 010_B)

EMU2nIWFIX_s[31:0] ← - (EMU2nIUFIX_s[31:0] + EMU2nIVFIX_s[31:0]) (CMUVW[2:0] bits = 100_B)

EMU2nIUFIX_s[31:0] ← - (EMU2nIWFIX_s[31:0] + EMU2nIVFIX_s[31:0]) (CMUVW[2:0] bits = 001_B)

(4) dq-axis Current Transformation

The transformation formula is as follows.

$$\begin{pmatrix} \text{EMU2nIDFIX\_s[31:0]} \\ \text{EMU2nIQFIX\_s[31:0]} \end{pmatrix} = \text{EMU2nSR2\_s[31:0]} \times \begin{pmatrix} \sin(\theta\text{e\_u}+ 90°) & -\sin(\theta\text{e\_u}+150°) & -\sin(\theta\text{e\_u}+ 30°) \\ -\sin(\theta\text{e\_u}+ 0°) & \sin(\theta\text{e\_u}+ 60°) & -\sin(\theta\text{e\_u}+120°) \end{pmatrix} \begin{pmatrix} \text{EMU2nIUFIX\_s[31:0]} \\ \text{EMU2nIVFIX\_s[31:0]} \\ \text{EMU2nIWFIX\_s[31:0]} \end{pmatrix}$$

The actual processes are as follows.

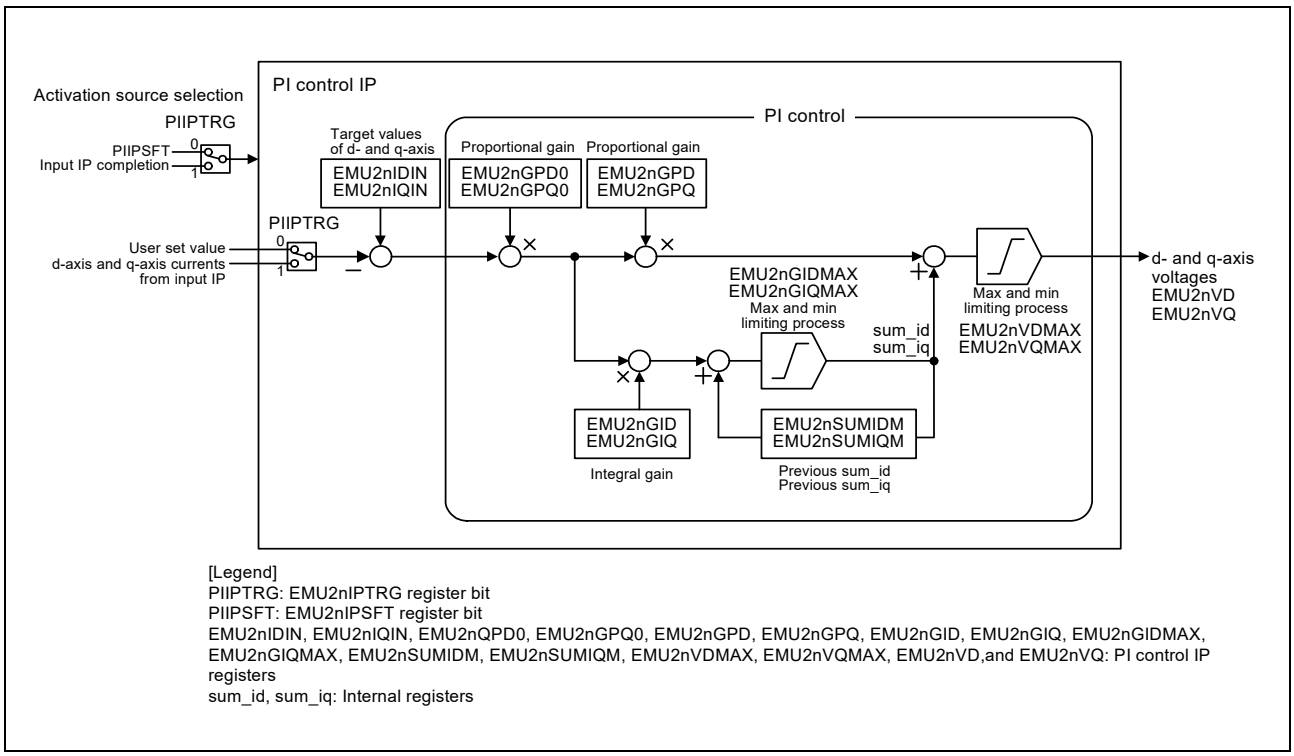
EMU2nIDFIX_s[31:0] ← (EMU2nSR2_s[31:0]×(
((cos(θe_u+ 0°)_s[16:0] × EMU2nIUFIX_s[31:0]) >> 15)
- ((cos(θe_u+60°)_s[16:0] × EMU2nIVFIX_s[31:0]) >> 15)
- ((sin(θe_u+30°)_s[16:0] × EMU2nIWFIX_s[31:0]) >> 15)
)) >> 16

EMU2nIQFIX_s[31:0]←(EMU2nSR2_s[31:0]×(
- ((sin(θe_u+ 0°)_s[16:0] × EMU2nIUFIX_s[31:0]) >>15)
+ ((sin(θe_u+60°)_s[16:0] × EMU2nIVFIX_s[31:0]) >>15)
- ((cos(θe_u+30°)_s[16:0] × EMU2nIWFIX_s[31:0]) >>15)
)) >> 16

The sine values used in the above calculations are provided by the sine data table in the EMU2 circuit. The values in the sine data table are results of multiplying the sine value by 8000_H.

24.4.6 PI Control IP

The PI control IP generates the Vd and Vq voltages (EMU2nVD register and EMU2nVQ register) from the d-axis and q-axis currents. The d-axis and q-axis currents can be supplied from the input IP. **Figure 24.12** shows the process flow of the PI control IP.



[Legend]
PIIPTRG: EMU2nIPTRG register bit
PIIPSFT: EMU2nIPSFT register bit
EMU2nIDIN, EMU2nIQIN, EMU2nQPD0, EMU2nGPQ0, EMU2nGPD, EMU2nGPQ, EMU2nGID, EMU2nGIQ, EMU2nGIDMAX, EMU2nGIQMAX, EMU2nSUMIDM, EMU2nSUMIQM, EMU2nVDMAX, EMU2nVQMAX, EMU2nVD,and EMU2nVQ: PI control IP registers
sum_id, sum_iq: Internal registers

Figure 24.12 Process Flow of PI Control IP

When the PIIPTRG bit in the EMU2nIPTRG register is set to 1, the PI control IP is activated upon input IP completion. When the PIIPTRG bit is set to 0, setting the PIIPSFT bit in the EMU2nIPSFT register to 1 can activate the PI control IP.

When the PI control IP completes its processing, the PIIF bit in the EMU2nINTSD register becomes 1. The PIIF bit becomes 0 by writing 1 to the PIIFC bit in the EMU2nINTSDC register.

An interrupt can be generated at the completion of the PI control IP.

(1) Initial Setting of PI control IP

The PI control IP can be initialized according to **Figure 24.13** Example of Initial Setting Procedure for PI Control IP (when an interrupt is used).

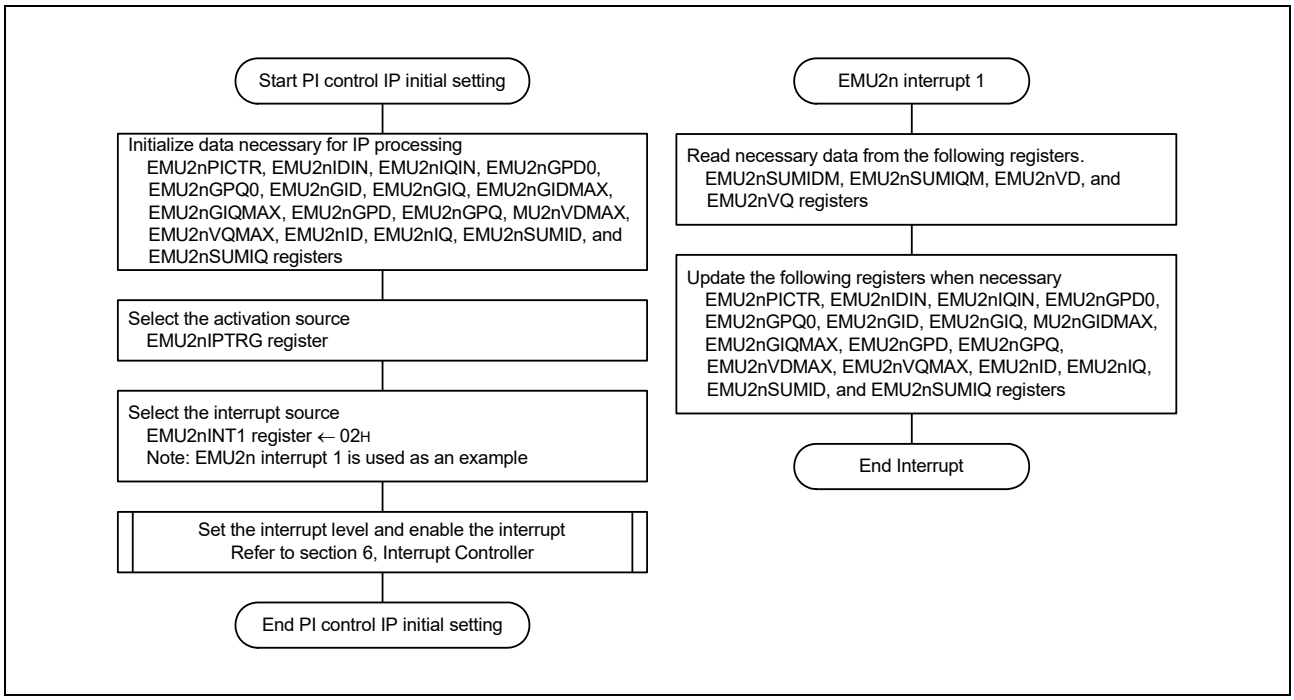


Figure 24.13 Example of Initial Setting Procedure for PI Control IP (when an interrupt is used)

(2) PI Control IP

[When the PIIPTRG bit in the EMU2nIPTRG register = 1]

The d- and q-axis currents supplied from the input IP are used.

id_s[31:0] ← EMU2nIDFIX_s[31:0]

iq_s[31:0] ← EMU2nIQFIX_s[31:0]

[When the PIIPTRG bit in the EMU2nIPTRG register = 0]

The d- and q-axis currents set in registers are used.

id_s[31:0] ← EMU2nID_s[31:0]

iq_s[31:0] ← EMU2nIQ_s[31:0]

The PI control based on the d-axis current is performed.

Set the proportional gain to registers EMU2nGPD0 and EMU2nGPD, and set the integral gain to the EMU2nGID register.

The integral value can be limited by the EMU2nGIDMAX register, and the generated d-axis voltage value can be limited by the EMU2nVDMAX register.

ids_a_s[31:0] ← EMU2nIDIN_s[31:0] - id_s[31:0]

ids_s[31:0] ← (ids_a_s[31:0] × EMU2nGPD0_s[31:0])>>16

[When the FSUMID bit in the EMU2nPICTR register = 0]

sum_id_s[31:0] ← EMU2nSUMID_s[31:0]

emvd_tmp_s[31:0] ← sum_id_s[31:0] + (ids_s[31:0] × EMU2nGPD_s[31:0]) >>16

[When the FSUMID bit in the EMU2nPICTR register = 1]

sum_id_s[31:0] ← EMU2nSUMIDM_s[31:0](previous value) + (ids_s[31:0] × EMU2nGID_s[31:0]) >>16

Condition	EMU2nSUMIDM_s[31:0]
sum_id_s[31:0] > EMU2nGIDMAX_u[31:0]	EMU2nGIDMAX_u[31:0]
sum_id_s[31:0] < -EMU2nGIDMAX_u[31:0]	-EMU2nGIDMAX_u[31:0]
Other than above	sum_id_s[31:0]

emvd_tmp_s[31:0] ← EMU2nSUMIDM_s[31:0] + (ids_s[31:0] × EMU2nGPD_s[31:0]) >>16

Condition	EMU2nVD_s[31:0]
emvd_tmp_s[31:0] > EMU2nVDMAX_u[31:0]	EMU2nVDMAX_u[31:0]
emvd_tmp_s[31:0]< -EMU2nVDMAX_u[31:0]	-EMU2nVDMAX_u[31:0]
Other than above	emvd_tmp_s[31:0]

The PI control based on the q-axis current is performed.

Set the proportional gain to registers EMU2nGPQ0 and EMU2nGPQ, and set the integral gain to the EMU2nGIQ register.

The integral value can be limited by the EMU2nGIQMAX register, and the generated q-axis voltage value can be limited by the EMU2nVQMAX register.

iqs_a_s[31:0] ← EMU2nIQIN_s[31:0] – iq_s[31:0]

iqs_s[31:0] ← (iqs_a_s[31:0] × EMU2nGPQ0_s[31:0])>>16

[When the FSUMIQ bit in the EMU2nPICTR register = 0]

sum_iq_s[31:0] ← EMU2nSUMIQ_s[31:0]

emvq_tmp_s[31:0] ← sum_iq_s[31:0] + (iqs_s[31:0] × EMU2nGPQ_s[31:0]) >>16

[When the FSUMIQ bit in the EMU2nPICTR register = 1]

sum_iq_s[31:0] ← EMU2nSUMIQM _s[31:0](previous value) + (iqs_s[31:0] × EMU2nGIQ_s[31:0]) >>16

Condition	EMU2nSUMIQM_s[31:0]
sum_iq_s[31:0] > EMU2nGIQMAX_u[31:0]	EMU2nGIQMAX_u[31:0]
sum_iq_s[31:0] < -EMU2nGIQMAX_u[31:0]	-EMU2nGIQMAX_u[31:0]
Other than above	sum_iq_s[31:0]

emvq_tmp_s[31:0] ← EMU2nSUMIQM_s[31:0] + (iqs_s[31:0] × EMU2nGPQ_s[31:0]) >>16

Condition	EMU2nVQ_s[31:0]
emvq_tmp_s[31:0] > EMU2nVQMAX_u[31:0]	EMU2nVQMAX_u[31:0]
emvq_tmp_s[31:0]< -EMU2nVQMAX_u[31:0]	-EMU2nVQMAX_u[31:0]
Other than above	emvq_tmp_s[31:0]

24.4.7 PWM IP

The PWM IP calculates the PWM duty cycle from the d-axis and q-axis voltages and electrical angle. The electrical angle can be supplied from the input IP. **Figure 24.14** shows the process flow of the PWM IP.

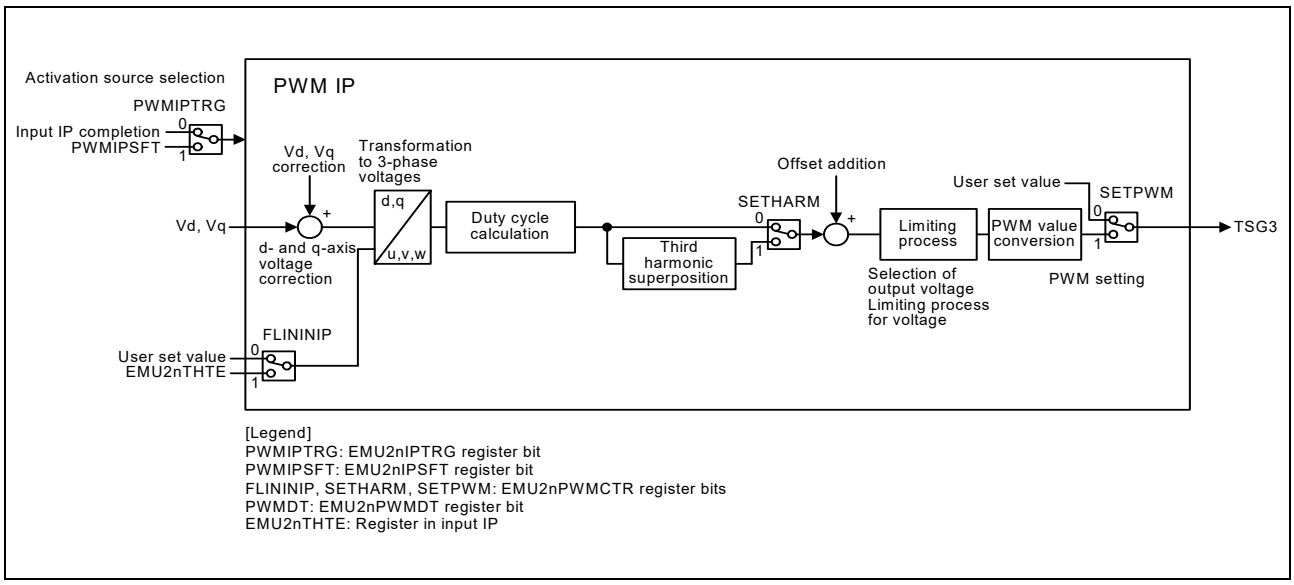


Figure 24.14 Process Flow of PWM IP

When the PWMIPTRG bit in the EMU2nIPTRG register is set to 1, the PWM IP is activated upon PI control IP completion. When the PWMIPTRG bit is set to 0, setting the PWMIPSFT bit in the EMU2nIPSFT register to 1 can activate the PWM IP.

When the PWM IP completes its processing, the PWMIF bit in the EMU2nINTSD register becomes 1. The PWMIF bit becomes 0 by writing 1 to the PWMFC bit in the EMU2nINTSDC register.

An interrupt can be generated at the completion of the PWM IP.

(1) Initial Setting of PWM IP

The PWM IP can be initialized according to **Figure 24.15** Example of Initial Setting Procedure for PWM IP (when an interrupt is used).

Once initial setting is completed, the values in the registers listed in **Table 24.37** Registers for Which Reflection can be Enabled or Disabled by FPWMREFPER, can be modified according to **Figure 24.16** Modifying Procedure for Simultaneous reflection registers (PWM IP).

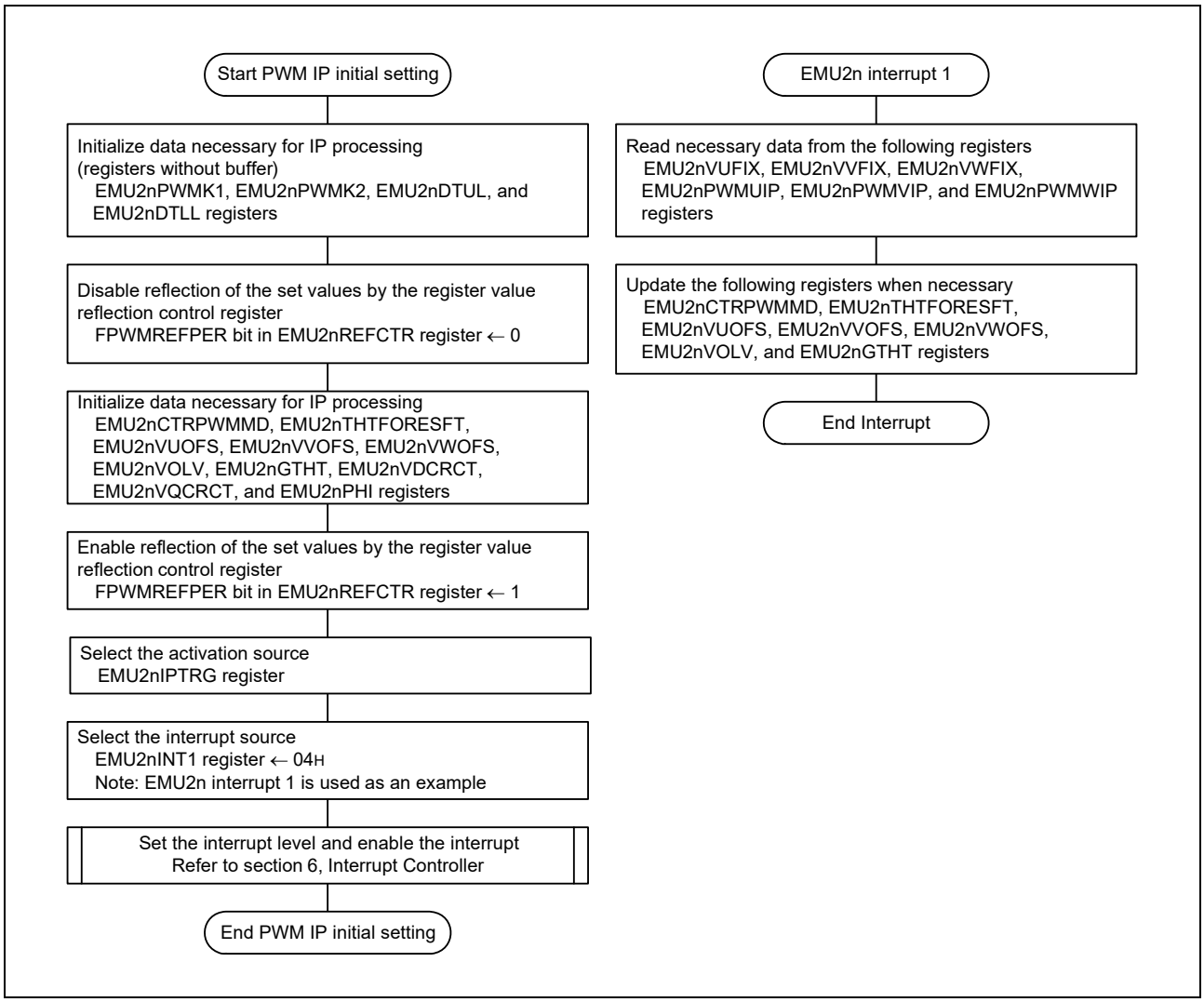


Figure 24.15 Example of Initial Setting Procedure for PWM IP (when an interrupt is used)

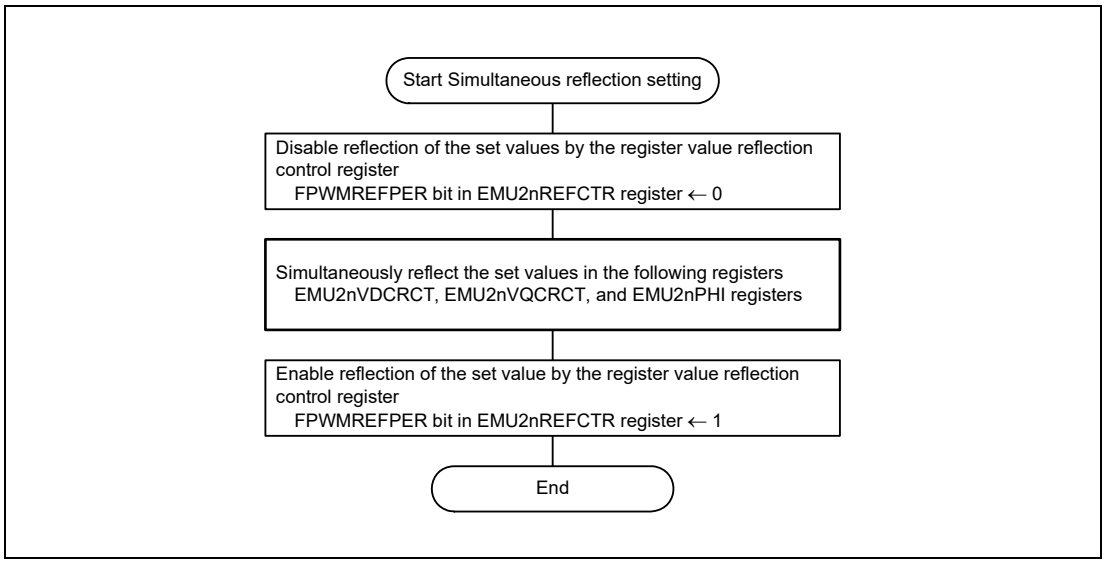
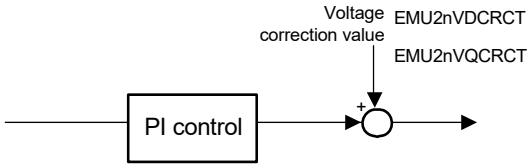


Figure 24.16 Modifying Procedure for Simultaneous Reflection Registers (PWM IP)

(2) d-Axis and q-Axis Voltage Correction



vr1_s[31:0] ← EMU2nVD_s[31:0] + EMU2nVDCRCT_s[31:0]

vr2_s[31:0] ← EMU2nVQ_s[31:0] + EMU2nVQCRCT_s[31:0]

(3) Transformation to 3-phase Voltage

When the FLININIP bit in the EMU2nPWMCTR register = 1

θe'_u[11:0] ← EMU2nTHTE_u[11:0]

When the FLININIP bit in the EMU2nPWMCTR register = 0

θe'_u[11:0] ← EMU2nTHTFORESFT_u[11:0]

θ'_u[11:0] ← ((θe'_u[11:0] + EMU2nPHI_s[15:0]) & $0FFF_H$ × EMU2nGTHT_s[15:0]) >> 8

The phase progresses by setting the value in the EMU2nPHI register.

The transformation formula is as follows.

$$\begin{pmatrix} \text{vul\_s[31:0]} \\ \text{vwl\_s[31:0]} \end{pmatrix} = (\text{EMU2nSR23\_s[31:0]} \times \begin{pmatrix} \cos(\theta\text{'\_u} + 0°) & -\sin(\theta\text{'\_u}+0°) \\ \cos(\theta\text{'\_u} + 120°) & -\sin(\theta\text{'\_u}+120°) \end{pmatrix} \begin{pmatrix} \text{vrl\_s[31:0]} \\ \text{vr2\_s[31:0]} \end{pmatrix} + \begin{pmatrix} \text{EMU2nUVOFS\_s[31:0]} \\ \text{EMU2nWVOFS\_s[31:0]} \end{pmatrix}$$

The actual processes are as follows.

vu1_s[31:0] ← (EMU2nSR23_s[31:0]× (
((cos(θ'_u+ 0°)_s[16:0] × vr1_s[31:0]) >> 15)
- ((sin(θ'_u+ 0°)_s[16:0] × vr2_s[31:0]) >> 15)
)) >> 16 + EMU2nUVOFS_s[31:0]

vw1_s[31:0]← (EMU2nSR23_s[31:0]× (
((cos(θ'_u+120°)_s[16:0] × vr1_s[31:0]) >> 15)
- ((sin (θ'_u+120°)_s[16:0] × vr2_s[31:0]) >> 15)
)) >> 16 + EMU2nWVOFS_s[31:0]

The sine values used in the above calculations are provided by the sine data table in the EMU2 circuit. The values in the sine data table are results of multiplying the sine value by 8000_H. Therefore, 15-bit shift is carried out in the above calculations.

(4) Calculating Duty Cycle

vu_s[31:0] ← ((vu1_s[31:0] × EMU2nPWMK1_s[31:0]) >>16)/EMU2nVOLV_s[15:0]

vw_s[31:0] ← ((vw1_s[31:0] × EMU2nPWMK1_s[31:0]) >>16)/EMU2nVOLV_s[15:0]

vv_s[31:0] ← -(vu_s[31:0] + vw_s[31:0])

(5) Superposing Third Harmonic

When the SETHARM bit in the EMU2nPWMCTR register = 1

vu2'_s[31:0] ← vu_s[31:0] - Third harmonic

vv2'_s[31:0] ← vv_s[31:0] - Third harmonic

vw2'_s[31:0] ← vw_s[31:0] - Third harmonic

When the SETHARM bit in the EMU2nPWMCTR register = 0

vu2'_s[31:0] ← vu_s[31:0]

vv2'_s[31:0] ← vv_s[31:0]

vw2'_s[31:0] ← vw_s[31:0]

(6) Offset Addition

vu2_s[31:0] ← vu2'_s[31:0] + EMU2nVUOFS_s[15:0]

vv2_s[31:0] ← vv2'_s[31:0] + EMU2nVVOFS_s[15:0]

vw2_s[31:0] ← vw2'_s[31:0] + EMU2nVWOFS_s[15:0]

(7) Selecting and Limiting Output Voltage of Each Phase

vu4'_s[31:0]← vu2_s[31:0], vv4'_s[31:0]← vv2_s[31:0], vw4'_s[31:0] ← vw2_s[31:0]

Condition	EMU2nVUFIX_s[31:0]
vu4'_s[31:0] > EMU2nDTUL_s[31:0]	EMU2nDTUL_s[31:0]
vu4'_s[31:0] < EMU2nDTLL_s[31:0]	EMU2nDTLL_s[31:0]
Other than above	vu4'_s[31:0]

Condition	EMU2nVVFIX_s[31:0]
vv4'_s[31:0] > EMU2nDTUL_s[31:0]	EMU2nDTUL_s[31:0]
vv4'_s[31:0] < EMU2nDTLL_s[31:0]	EMU2nDTLL_s[31:0]
Other than above	vv4'_s[31:0]

Condition	EMU2nVWFIX_s[31:0]
vw4'_s[31:0] > EMU2nDTUL_s[31:0]	EMU2nDTUL_s[31:0]
vw4'_s[31:0] < EMU2nDTLL_s[31:0]	EMU2nDTLL_s[31:0]
Other than above	vw4'_s[31:0]

Note 1. Due to the limiting process for the voltage value using registers EMU2nDTLL and EMU2nDTUL, the value needs to be the EMU2nPWMLL register value or smaller to output 0 % duty cycle waveform and pwm_max (EMU2nCARR value + EMU2nDTT value - EMU2nPWMUL value) value or greater to output 100% duty cycle waveform.
Since registers EMU2nVUFIX, EMU2nVVFIX, EMU2nVWFIX are calculated with (EMU2nCARR value)/2 and (EMU2nDTT value)/2, significant digits are lost if the register values of EMU2nCARR and EMU2nDTT are odd. To output 100% or 0% duty cycle waveform, set the EMU2nPWMLL and EMU2nPWMUL registers appropriately with consideration for the equations given in **Section 24.4.7 (8) Setting PWM**.

(8) Setting PWM

When the PWMSEL bit in the EMU2nPWMCTR register = 0

The reference value for the PWM operation is generated based on the carrier cycle setting.

pwm2_u[14:0]←EMU2nCARR_u[15:0]/2

When the PWMSEL bit in the EMU2nPWMCTR register = 1

The reference value for the PWM operation is generated based on the carrier cycle setting and dead time setting.

pwm2_u[14:0]←(EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2

pwm_max_u[15:0] ← EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0] – EMU2nPWMUL_u[15:0]

When the SETRVRS bit in the EMU2nPWMCTR register = 0

The operation is performed as follows:

PWM target value of each phase ← Voltage target value of each phase + 50% duty cycle of PWM

pwmu'_s[31:0] ← ((EMU2nVUFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >>16) + (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2

pwmw'_s[31:0] ← ((EMU2nVWFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >>16) + (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2

pwmv'_s[31:0] ← ((EMU2nVVFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >>16) + (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2

When the SETRVRS bit in the EMU2nPWMCTR register = 1

The operation is performed as follows:

PWM target value of each phase ← 50% duty cycle of PWM – Voltage target value of each phase

pwmu'_s[31:0] ← (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2 - (EMU2nVUFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >> 16

pwmw'_s[31:0] ← (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2 - (EMU2nVWFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >>16

pwmv'_s[31:0] ← (EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2 - (EMU2nVVFIX_s[31:0] × (pwm2_u[14:0] × EMU2nPWMK2_s[15:0])) >> 16

The PWM duty cycle is set based on the PWM target values pwmu', pwmv', and pwmw'.

The PWM duty cycle is 0% when pwmu', pwmv', and pwmw' are equal to or smaller than the EMU2nPWMLL register value, and 100% when equal to or larger than pwm_max value.

Condition	EMU2nPWMUIP_u[15:0]
pwmu'_s[31:0] ≥ pwm_max_u[15:0]	EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0]
pwmu'_s[31:0] ≤ EMU2nPWMLL_u[15:0]	0
EMU2nVOLV_s[15:0] = 0	(EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2
Other than above	pwmv'_u[15:0]

Condition	EMU2nPWMVIP_u[15:0]
pwmv'_s[31:0] ≥ pwm_max_u[15:0]	EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0]
pwmv'_s[31:0] ≤ EMU2nPWMLL_u[15:0]	0
EMU2nVOLV_s[15:0] = 0	(EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2
Other than above	pwmv'_u[15:0]

Condition	EMU2nPWMWIP_u[15:0]
pwmw'_s[31:0] ≥ pwm_max_u[15:0]	EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0]
pwmw'_s[31:0] ≤ EMU2nPWMLL_u[15:0]	0
EMU2nVOLV_s[15:0] = 0	(EMU2nCARR_u[15:0] + EMU2nDTT_u[15:0])/2
Other than above	pwmw'_u[15:0]

The EMU2nCARR register value is buffered in the PWM IP at the PWM IP activation. The buffered EMU2nCARR register value, and register values of the EMU2nPWMUIP, EMU2nPWMVIP, and EMU2nPWMWIP are simultaneously transferred to the TSG3.

Transfer to the TSG3 is performed as follows.

When the SHIPWM bit in the EMU2nPWMCTR register = 0

TSG3 register ← each register value

TSG3iCMP0E[17:0] ← EMU2nCARR[15:0]

TSG3iCMP2E[17:0], TSG3iCMP1E[17:0] ← EMU2nPWMUIP[15:0]

TSG3iCMP6E[17:0], TSG3iCMP5E[17:0] ← EMU2nPWMVIP[15:0]

TSG3iCMP10E[17:0], TSG3iCMP9E[17:0] ← EMU2nPWMWIP[15:0]

When the SHIPWM bit in the EMU2nPWMCTR register = 1

TSG3 register ← each register value << 1 (1-bit left shift)

TSG3iCMP0E[17:0] ← EMU2nCARR[15:0] << 1

TSG3iCMP2E[17:0], TSG3iCMP1E[17:0] ← EMU2nPWMUIP[15:0] << 1

TSG3iCMP6E[17:0], TSG3iCMP5E[17:0] ← EMU2nPWMVIP[15:0] << 1

TSG3iCMP10E[17:0], TSG3iCMP9E[17:0] ← EMU2nPWMWIP[15:0] << 1

Note 1. When the PWM data is transferred by writing 1 to the PWMDT bit in the EMU2nPWMDT register (software activation), the same left shift as preceding is performed on each output pin if the SHIPWM bit is set to 1.

24.4.8 Rectangle IP

The rectangle IP calculates the rectangular wave output level and compare setting. **Figure 24.17** shows the process flow of the rectangle IP.

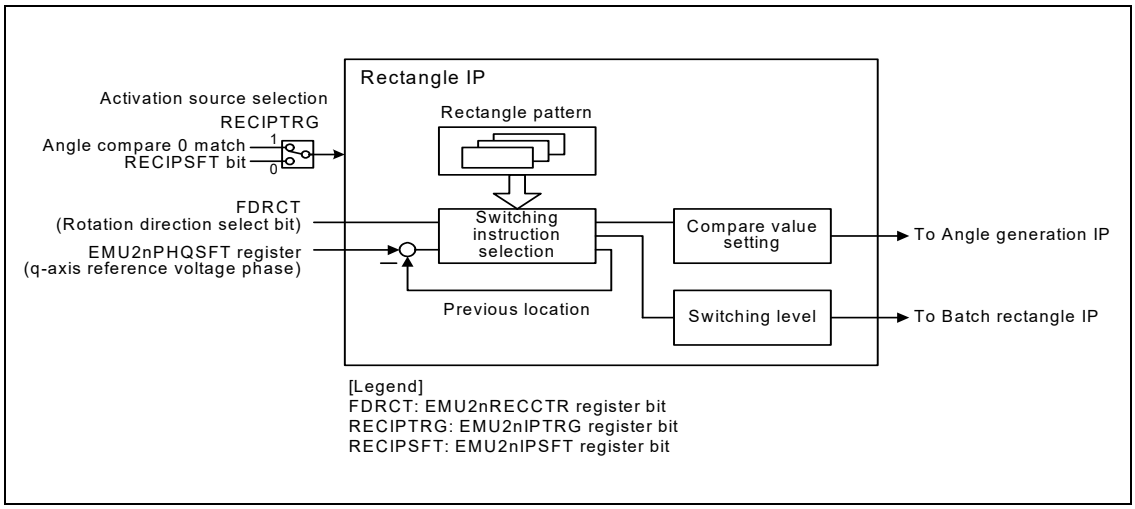


Figure 24.17 Process Flow of Rectangle IP

When the RECIPTRG bit in the EMU2nIPTRG register is set to 1, the rectangle IP is activated upon angle compare 0 match. When the RECIPTRG bit is set to 0, setting the RECIPSFT bit in the EMU2nIPSFT register to 1 can activate the rectangle IP.

When the rectangle IP completes its processing, the RECIF bit in the EMU2nINTSD register becomes 1. The RECIF bit becomes 0 by writing 1 to the RECIFC bit in the EMU2nINTSDC register.

An interrupt can be generated at the completion of the rectangle IP.

Figure 24.18 shows an example of initial setting procedure for the rectangle IP (when an interrupt is used).

(1) Initial Setting of Rectangle IP

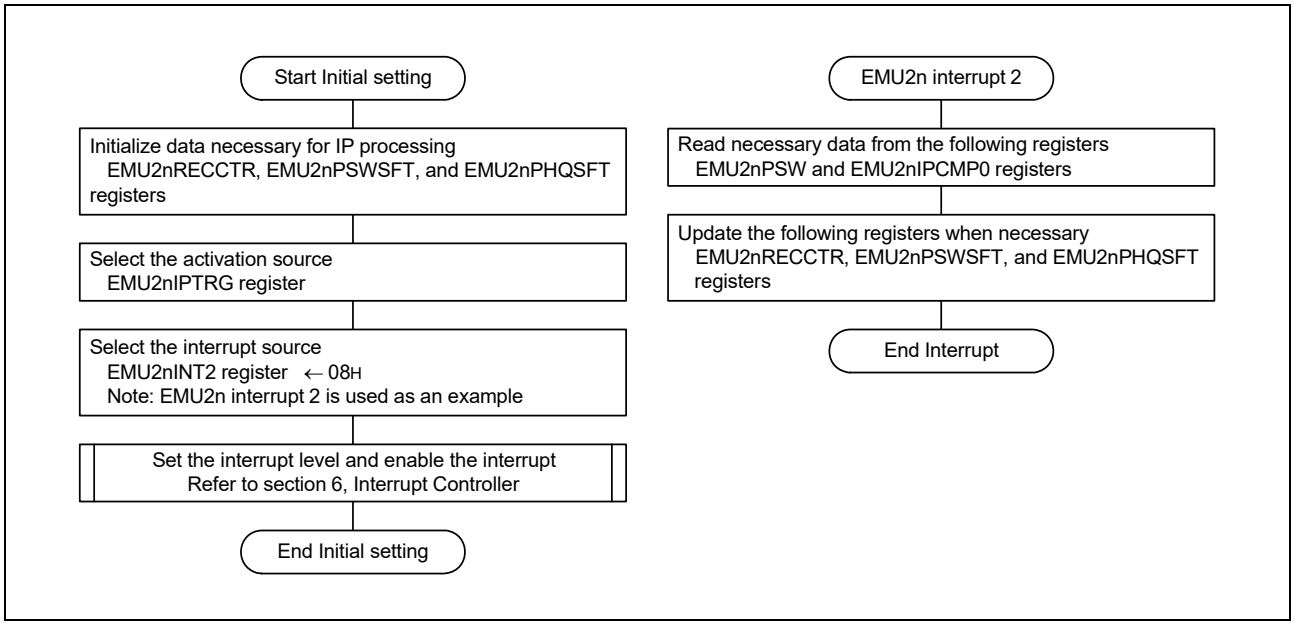


Figure 24.18 Example of Initial Setting Procedure for the Rectangle IP (when an Interrupt is Used)

(2) Generating Switching Instruction

Selection of Feedback Data FIPPOSI in EMU2nRECCTR Register	psw_old_u[2:0]
0 (EMU2 calculation result used)	EMU2nPSW_u[2:0]
1 (user-set value used)	EMU2nPSWSFT_u[2:0]

Rotation Direction FDRCT in EMU2nRECCTR register	psw_old_u[2:0]	EMU2nPSW_u[2:0]	eθ'sw_u[11:0]	Next Switching
0 (positive rotation)	0	1	60° - ϕq_s[11:0]	Side A of EMU2nPTNAB
	1	2	120° - ϕq_s[11:0]	Side B of EMU2nPTNAB
	2	3	180° - ϕq_s[11:0]	Side C of EMU2nPTNCD
	3	4	240° - ϕq_s[11:0]	Side D of EMU2nPTNCD
	4	5	300° - ϕq_s[11:0]	Side E of EMU2nPTNEF
	5	0	360° - ϕq_s[11:0]	Side F of EMU2nPTNEF
1 (negative rotation)	0	5	180° + ϕq_s[11:0]	Side E of EMU2nPTNEF
	1	0	240° + ϕq_s[11:0]	Side F of EMU2nPTNEF
	2	1	300° + ϕq_s[11:0]	Side A of EMU2nPTNAB
	3	2	360° + ϕq_s[11:0]	Side B of EMU2nPTNAB
	4	3	60° + ϕq_s[11:0]	Side C of EMU2nPTNCD
	5	4	120° + ϕq_s[11:0]	Side D of EMU2nPTNCD

Note 1. Although θ is calculated in terms of degrees in the table, it is actually handled as a value from 0 to FFF_H.
$60° = 2AA_H$, $120° = 555_H$, $180° = 800_H$, $240° = AAA_H$, $300° = D55_H$, $360°(0°) = 000_H$
The calculation result is ANDed with $0FFF_H$ in order for eθ'sw to be unsigned 12-bit data.

(3) Setting Compare 0 Value

EMU2nIPCM0_u[11:0] ← eθ'sw_u[11:0]

24.4.9 Rectangular Wave Generation Block

The rectangular wave generation block consists of the batch rectangle IP and independent rectangle IP. The rectangular wave generated in this block is output via the TSG3. The RECMD bit in the EMU2nIRECCTR register is used to select which IP to use.

The batch rectangle IP updates the output waveforms of the U/V/W phases simultaneously based on the rectangular wave output level and compare value calculated by the rectangle IP in the calculation block. The independent rectangle IP updates the output waveforms of the U/V/W phases at different timings based on the compare values and output levels set for each of U/V/W phases.

24.4.10 Batch Rectangle IP

The rectangular wave output is generated by switching the rectangular wave output levels at the timing when the electrical angle*1 generated from the input angle data matches with target angle 0. The upper 12 bits of the angle value*5 in the RDC2nANGDAT register of the R/D converter is used as the angle data.

Target angle 0 is set in the compare 0 register, which is the internal compare register*2. The compare 0 register can be set either by the EMU2nCMP0 register or by rectangle IP*4 depending on the SLCTCMP0 bit setting in the EMU2nRECCTR register. The rectangular wave output level is determined by the EMU2nPTNN register when the SETREC bit in the EMU2nRECCTR register is set to 0. When the SETREC bit is set to 1, the rectangle IP automatically sets the value from registers EMU2nPTNAB, EMU2nPTNCD, and EMU2nPTNEF.

Target angle 1 is set in the compare 1 register, which is the internal compare register*3. The compare 1 register can be set by the EMU2nCMP1 register.

Note 1. Be sure to set the electrical angle generation coefficient value to the EMU2nPXR register. Refer to **24.4.4 (3) Generating Electrical Angle**.

Note 2. It is referred to as angle compare 0 match that the electrical angle matches with or passes the value set in the compare 0 register.

Note 3. It is referred to as angle compare 1 match that the resolver angle matches with or passes the value set in the compare 1 register.

Note 4. Even when the SLCTCMP0 bit in the EMU2nRECCTR register is set to 1, the compare 0 register is loaded with the EMU2nCMP0 register value for the first comparison after the EMU2 is activated.

Note 5. When the DATSEL[3:0] bits in the RDC2nCONSEL register are set to 0000_B (12-bit angle data), bit 11 to bit 0 in the RDC2nANGDAT register are used as the angle data.

Figure 24.19 shows generation of the angle data, compare 0, and compare 1, and **Figure 24.20** shows the overview of rectangular wave output.

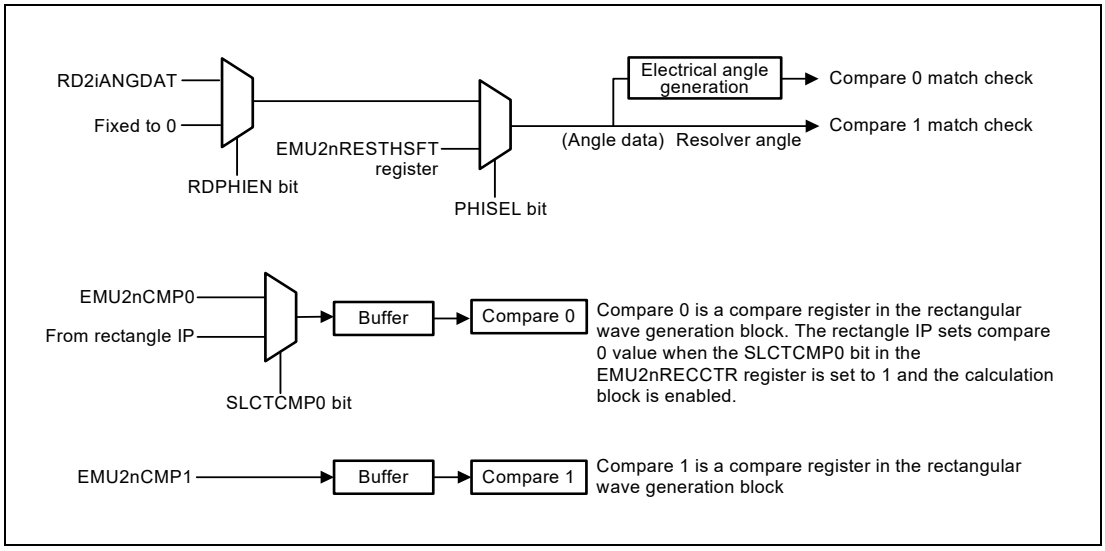


Figure 24.19 Generation of Angle Data, Compare 0, and Compare 1

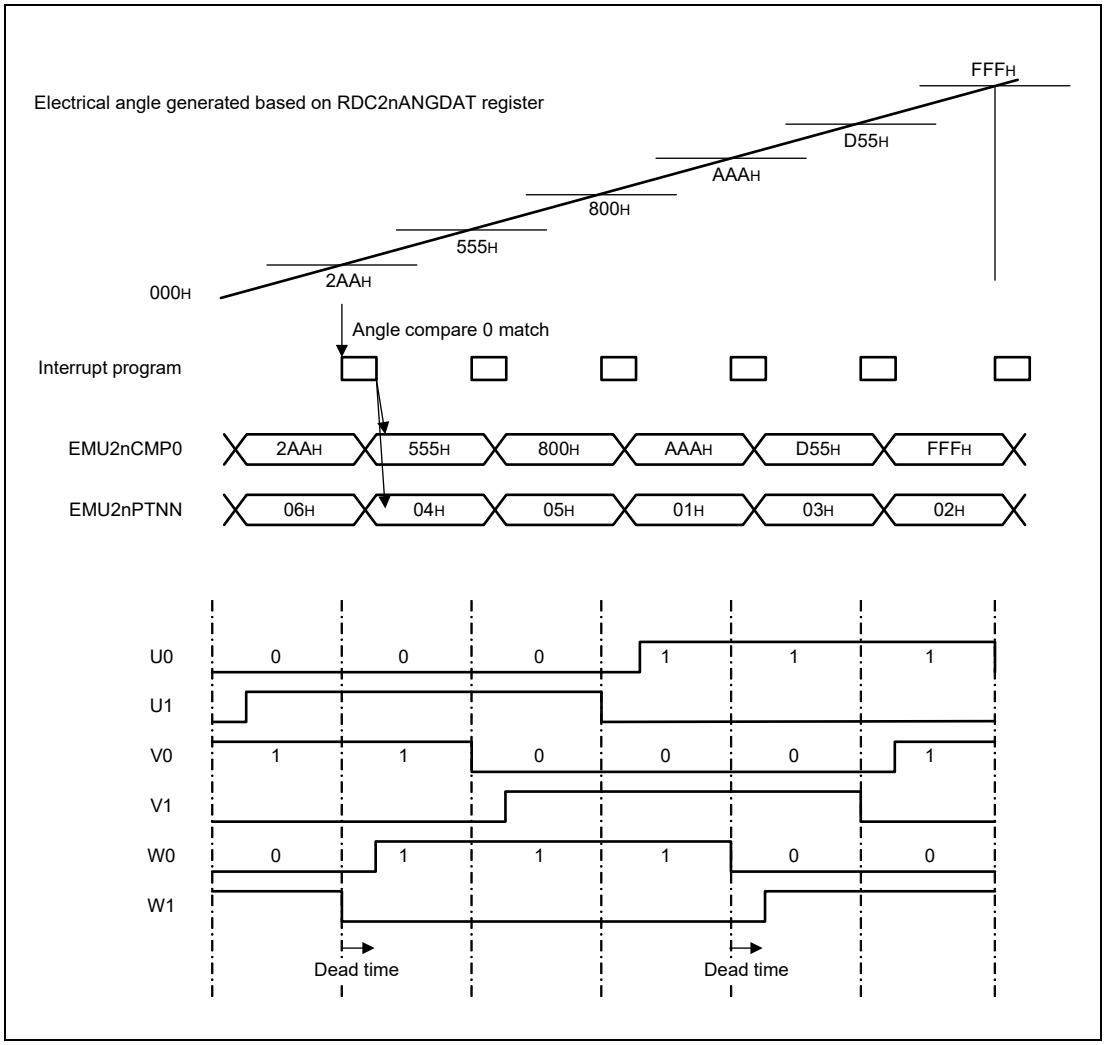


Figure 24.20 Overview of Rectangular Wave Output

24.4.11 Independent Rectangle IP

The independent rectangle IP can generate flexible rectangular wave output patterns by updating the patterns when the electrical angle matches with the compare values specified for each of U/V/W phases.

Setting the RECMD bit in the EMU2nIRECCTR register to 1 (independent rectangular wave output) allows the output pattern of the independent rectangle IP to be output to the TSG3.

When a U-phase, V-phase, or W-phase compare match occurs, the independent rectangle output pattern of the corresponding phase is updated. Then, the IRECWIF, IRECVIF, or IRECUIF bit in the EMU2nINTSD register of the corresponding phase becomes 1. These bits become 0 by writing 1 to the IRECWIFC, IRECVIFC, and IRECUIFC bits in the EMU2nINTSDC register, respectively.

By writing to the UINIPTN, VINIPTN, and WINIPTN bits in the EMU2nIRPTN register, the output patterns of U, V, and W phases can be modified. When these bits are read, the last written data is returned.

The output levels of the U, V, and W phases can also be changed when the electrical angle generated from the input angle data matches with each target angle of U-, V-, and W-phase. Three compare values and three output pattern values can be set for each of U/V/W phases (EMU2nIRUCPPN0 to EMU2nIRUCPPN2, EMU2nIRVCPPN0 to EMU2nIRVCPPN2, and EMU2nIRWCPPN0 to EMU2nIRWCPPN2 registers), and the compare value is switched each time a compare match occurs. Switching can be done completely independent of other phases.

Only when the electrical angle matches the compare value 2 (EMU2nIRUCPPN2, EMU2nIRVCPPN2, or EMU2nIRWCPPN2 registers), the interrupt source flag (the IRECUIF, IRECVIF, or IRECWIF bit in the EMU2nINTSD register) becomes 1, and an interrupt request is generated if the interrupt is enabled by the interrupt source select register (EMU2nINT0 to EMU2nINT4).

Angle compare match is detected by the angle generation IP. The angle generation IP compares the electrical angle converted from the resolver angle with the compare value specified by the selector to generate the compare match signal. The independent rectangle IP outputs the compare values to be compared for each phase to the angle generation IP, and the compare match signal is input from the angle generation IP. **Figure 24.21** shows a block diagram of the independent rectangular wave output function.

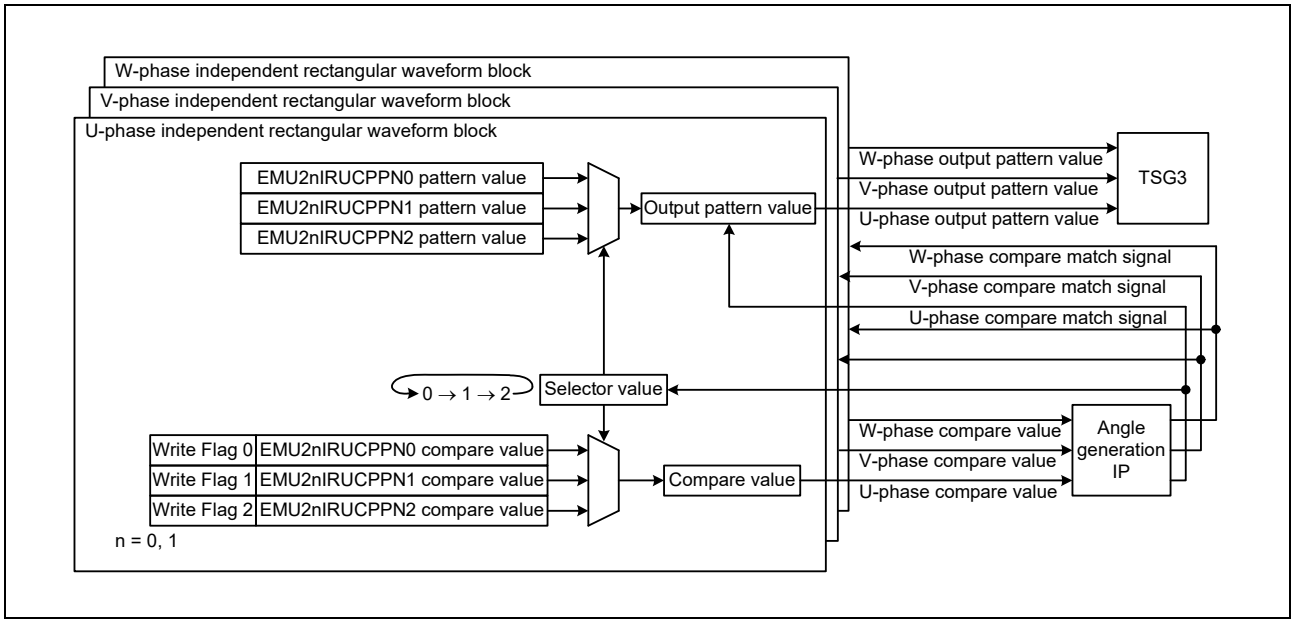


Figure 24.21 Block Diagram of Independent Rectangular Wave Output Function

Each phase has three compare/pattern setting registers (0 to 2). The selector value specifies which one of the compare/pattern setting registers to use. The selector value becomes 0 after reset and is incremented by one each time a compare match occurs. When a compare match occurs while the selector value is 2, the corresponding interrupt source determination flag becomes 1 and the selector value returns to 0. The current selector value can be monitored by the EMU2nIRSELM register. The selector value becomes 0 by writing 1 to the corresponding independent rectangle IP flag/select signal reset register (EMU2nIRCTRST).

The write flags 0 to 2 in the EMU2nIRFLGM register are used to prevent repeated comparison of the same compare value. The write flag becomes 1 when a value is written to the corresponding compare/pattern setting register, and becomes 0 when the corresponding compare match occurs. While the write flag is 0, the output pattern is not updated because the corresponding compare match does not occur. All the write flags become 0 in the compare/pattern setting registers 0 to 2 by writing 1 to the corresponding flag/select signal reset bit in the independent rectangle IP flag/select signal reset register (EMU2nIRCTRST).

Note 1. The selector value is incremented only when a compare match occurs. If the corresponding write flag for the compare/pattern setting register currently specified by the selector value is 0, a compare match no longer occurs. Do not leave any write flag to be 0 for the compare/pattern setting registers lying between the register that is currently specified by the selector and the one that is written by the CPU.

Note 2. Do not set the same compare value to the different compare/pattern setting registers for the same phase.

Note 3. Several cycles are required for switching the output in the same phase. If the interval between the output switching angles (compare values) is too short, the output switching may be delayed.

24.4.12 3-phase PWM Waveform Output Control

When the SETPWM bit in the EMU2nPWMCTR register is set to 1, the result of the PWM IP calculation is reflected in the duty cycle of the 3-phase PWM waveform, which is output from the TSG3. The PWM IP calculates the PWM duty cycle based on the values of the EMU2nCARR register which contains the carrier cycle, the EMU2nDTT register, and other control registers containing data necessary for the PWM IP. These values are set before PWM IP activation. For details of the PWM IP operation, refer to **Section 24.4.7, PWM IP**. **Figure 24.22** shows an example of the control flow of the 3-phase PWM waveform output.

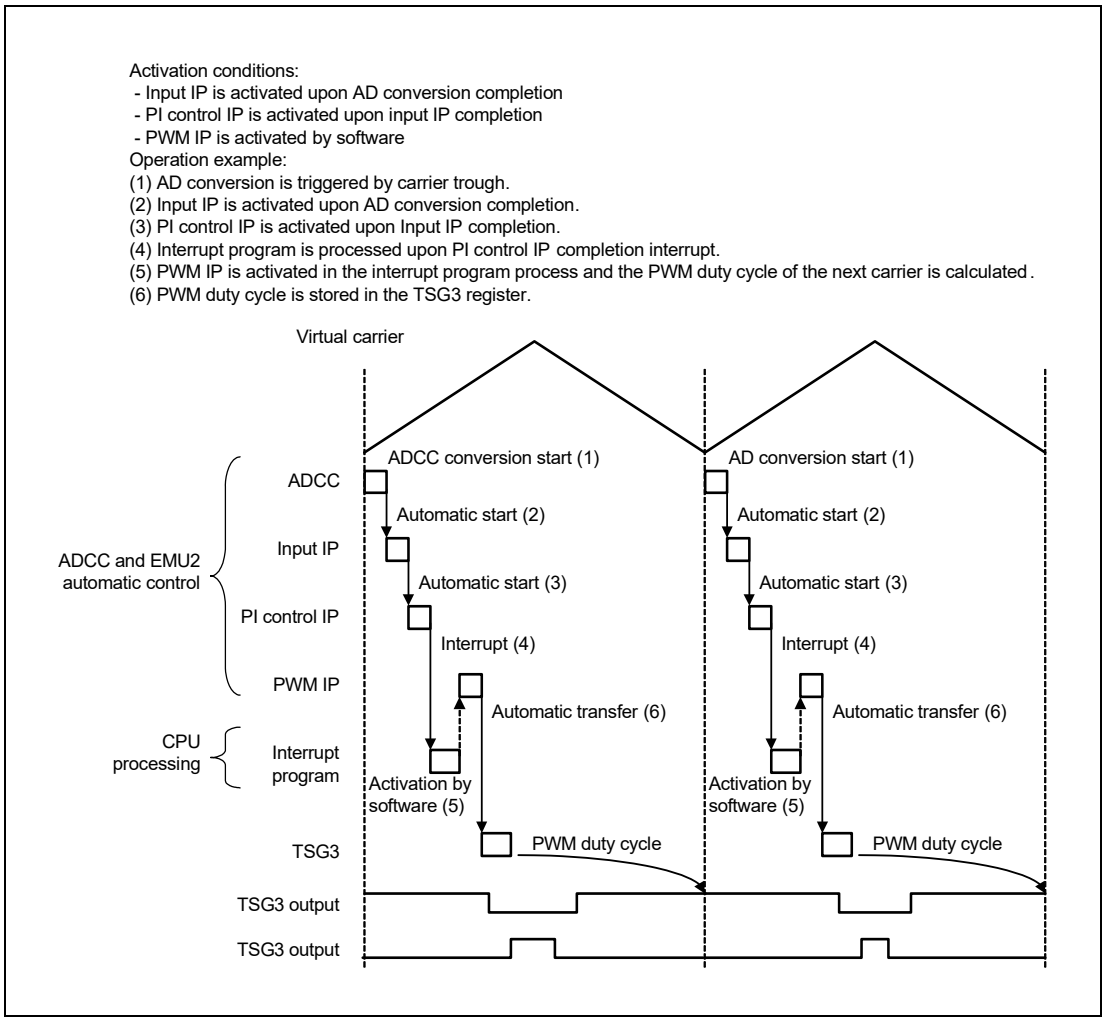


Figure 24.22 Example of Control Flow of 3-phase PWM Waveform Output

24.4.13 Batch Rectangular Wave Output Control

Setting the SETREC bit in the EMU2nRECCTR register to 1 allows the output pattern calculated by the rectangle IP to be reflected in the rectangular waveform output. When the SLCTCMP0 bit in the EMU2nRECCTR register is set to 1, the result of the rectangle IP is set to the compare 0 register, which is used as the rectangular wave switching timing.

For details of the batch rectangle IP operations, refer to **Section 24.4.8, Rectangle IP**. **Figure 24.23** shows an example of the control flow of the batch rectangular wave output, and **Figure 24.24** shows an example of angle generation.

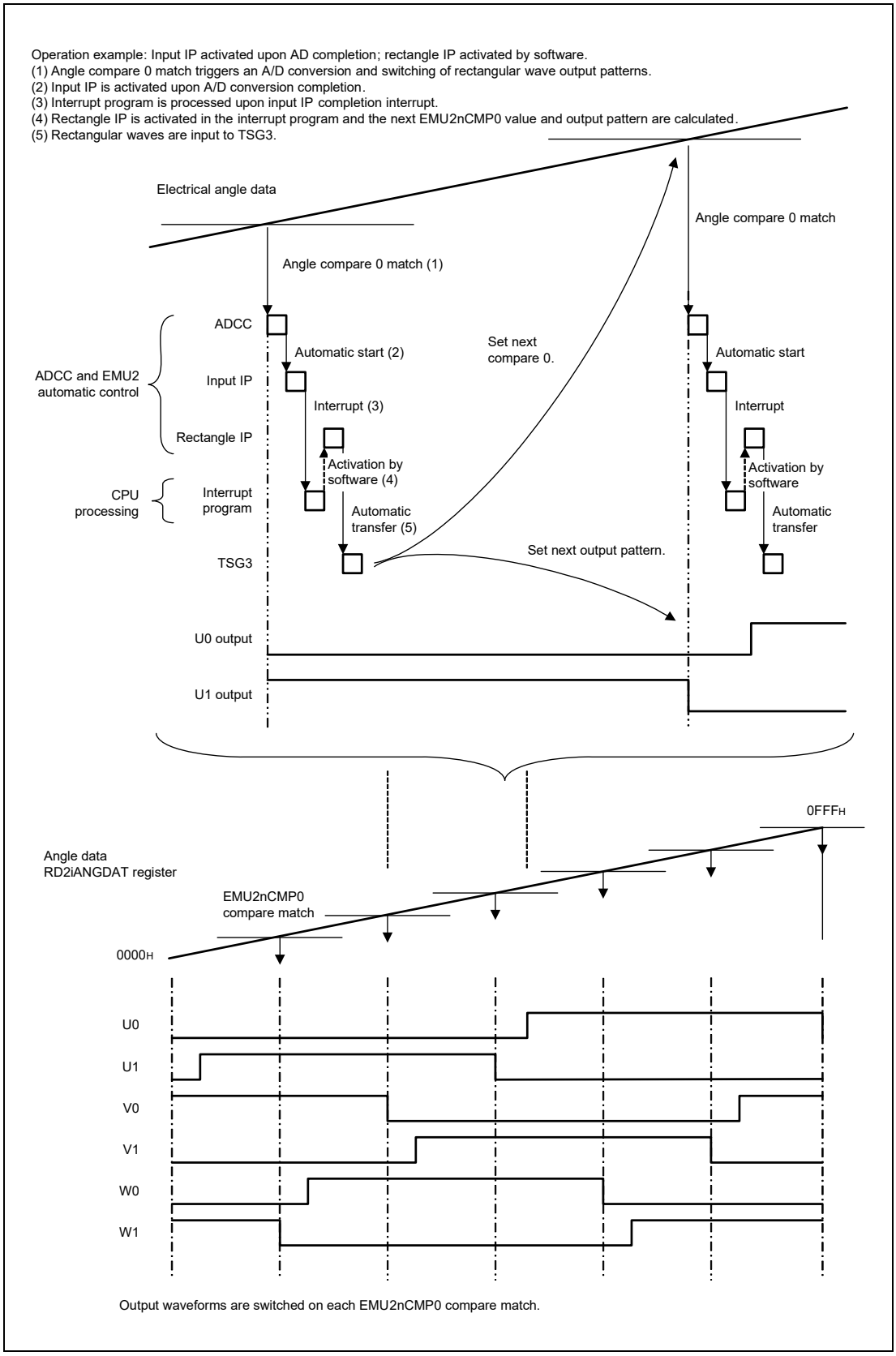


Figure 24.23 Example of Control Flow of Batch Rectangular Wave Output

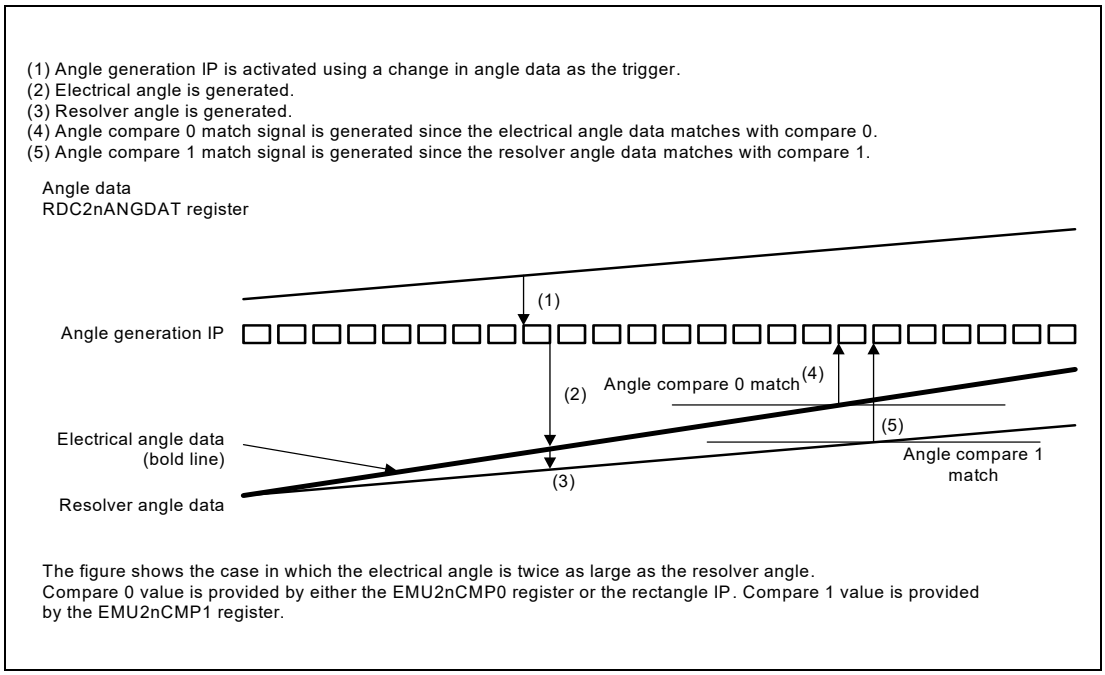


Figure 24.24 Example of Angle Generation

24.4.14 Speed Measurement Timer

The speed measurement timer measures the cycle of the Z-phase signal output from the R/D converter. Setting the STR bit in the EMU2nVMTCTR register starts the timer. When the rising edge of the Z-phase signal is detected, the counter value is captured to the EMU2nVMTCAP register, and the counter is initialized and continues counting from 0. The accurate Z-phase signal cycle can be captured to the EMU2nVMTCAP register after the second rising edges of the Z-phase signal.

Table 24.176 Speed Measurement Timer Specifications

Item	Specifications
Counter source	CCLK
Counter start and stop	Controlled with the STR bit in the EMU2nVMTCTR register
Counter operation	• 25-bit counting up • EMU2nVMTCNT counter is initialized to 0000 0000_H after the counter value is captured to the EMU2nVMTCAP register on the capture signal, or when an overflow occurs. • Continues counting after initialization.
Reading counter	Reading the EMU2nVMTCNT counter returns the 25-bit counter value.
Writing to counter	The EMU2nVMTCNT counter can be written, irrespective of whether the timer is operating or not.
Overflow output	Either the OVF flag becomes 1, or an interrupt is generated
Cycle measurement signal (capture signal)	Z-phase signal from the RDC2

24.5 Verification Function

The verification function is used for fault detection. It includes two functions: one is buffering function for verification, in which input/output values of calculation are transferred to the buffer register, and the other is equivalence check function, in which channels 0 and 1 simultaneously carry out the same process and the results are compared to each other after the process is completed. **Table 24.177** lists the verification-related registers and verification methods.

Table 24.177 Verification-Related Registers and Verification Methods

Register and Value (Transfer Source)	Verification Buffer Register (Transfer Destination)	Verification Method
EMU2n ADm data register k (EMU2nADmk)	EMU2n ADm data k verification buffer register (EMU2nCBADmk)	Software
EMU2n resolver angle monitor register (EMU2nTHTREFIXIN)	EMU2n resolver angle verification buffer register (EMU2nCBTHTREFIXIN)	Software
EMU2n d-axis current value register (EMU2nIDFIX)	EMU2n d-axis current value verification buffer register (EMU2nCBIDFIX)	Software
EMU2n q-axis current value register (EMU2nIQFIX)	EMU2n q-axis current value verification buffer register (EMU2nCBIQFIX)	Software
EMU2n U-phase compare value register (EMU2nPWMUIP)	EMU2n U-phase compare value verification buffer register (EMU2nCBPWMUIP)	• Software • Equivalence check function
EMU2n V-phase compare value register (EMU2nPWMVIP)	EMU2n V-phase compare value verification buffer register (EMU2nCBPWMVIP)	• Software • Equivalence check function
EMU2n W-phase compare value register (EMU2nPWMWIP)	EMU2n W-phase compare value verification buffer register (EMU2nCBPWMWIP)	• Software • Equivalence check function
U/V/W-phase batch rectangle pattern value	EMU2n batch rectangle pattern verification buffer register (EMU2nCBBREC)	Equivalence check function
U/V/W-phase independent rectangle pattern value	EMU2n independent rectangle pattern verification buffer register (EMU2nCBIREC)	Equivalence check function

24.5.1 Buffering Function for Verification

Using the buffering function, the target registers and the values can be transferred to the buffer registers for verification at a timing specified by the EMU2nCBTIM register.

Buffering can be enabled in either of the following two ways. When the CBEN1 bit in the EMU2nCBCTR0 register is set to 0, buffering is enabled only once each time 1 is written to the CBEN0 bit. Here, the CBMON bit in the EMU2nCBCTR0 register becomes 1 when 1 is written to the CBEN0 bit and becomes 0 when buffering is completed.

When the CBEN1 bit is set to 1, buffering is always enabled. Here, the CBMON bit always becomes 1 while the CBEN1 bit is set to 1.

When the CBUFINTj bit in the EMU2nINTj register (j = 0 to 4) is set to 1 (interrupt enabled), an verification buffering completion interrupt is generated upon buffering completion.

24.5.2 Equivalence Check Function

Using the equivalence check function, channels 0 and 1 can be simultaneously activated and the calculation results of the two channels can be automatically compared to each other after they are transferred to the verification buffer registers. If the results of channels 0 and 1 are different from each other, an equivalence check error interrupt is generated.

After the trigger signal is selected by the DATTRG bit in the EMU2nSMLCTR register and the SMLEN bit in the EMU2nSMLCTR register is set to 1, channels 0 and 1 are simultaneously activated when the selected trigger signal is generated with the CBMON bit in the EMU2nCBCTR0 register being 0 in both channels. The values to be compared upon buffering completion are selected by the EMU2nCBCTR1 register.

When the SMLINTj bit in the EMU2nINTj register is set to 1 (interrupt enabled), an equivalence check error interrupt is generated if the values in the verification buffers are different between channels 0 and 1.

The equivalence check function can compare the PWM compare values, batch rectangle patterns, and independent rectangle patterns.

24.5.3 Verification Methods

Two verification methods are provided: one compares the output value with the calculation result by software, and the other automatically compares the values to each other by hardware using the equivalence check function control register.

24.5.3.1 Verification by Software

The buffering function and the verification buffering completion interrupt are used for verification by software.

When the verification buffering completion interrupt is generated, d-axis and q-axis currents are calculated in the interrupt routine based on the motor current value (EMU2nCBADmk register) and angle value (EMU2nCBTHTREFIXIN register), which are input to the input IP. These calculation results are compared to the d-axis and q-axis current values (EMU2nCBIDFIX and EMU2nCBIQFIX register), which are output from the input IP. In addition, the values output from the PWM IP can be verified by performing the PWM IP calculation by software and comparing the calculation results with the U/V/W-phase compare values (EMU2nCBPWMUIP, EMU2nCBPWMVIP, and EMU2nCBPWMWIP register).

24.5.3.2 Verification of Agreement of Equivalence Check Results

Buffering function, equivalence check function, and equivalence check error interrupts are used for verification by hardware.

Equivalence check function has the following three modes according to the setting of the EMU2nSMLCTR.DATTRG bit.

Operating Mode	Bit EMU20SMLCTR.DATTRG	Bit EMU21SMLCTR.DATTRG	Operation
A	0	0	Single-motor control with one of two independent channels (channels 0 and 1) or two-motor control by using both channels (with equivalence check function off).
B	0	1	Single-motor control by channel 0 (with equivalence check function on). The same input values are fed to channel 1 for checking to detect failures in matching with the results of buffering.
C	1	0	Single-motor control by channel 1 (with equivalence check function on). The same input values are fed to channel 0 for checking to detect failures in matching with the results of buffering.

When equivalence check function is on, the input to the main channel of the following items external to the EMU2 are simultaneously and automatically applied to the channel used in checking, so setting of input data for use in checking is not required.

- The carrier peak and trough triggers from the TSG3
- The current value from the A/D convertor
- The resolver angle from the R/D convertor
- Access to registers by the CPU

Take particular care in case of access to registers by the CPU since the above inputs are applied to the channel used in checking immediately after a setting is written to the DATTRG bit.

The table below summarizes the behaviors in each operating mode in response to access to registers by the CPU. In operating modes B and C, writing to registers of the main channel is simultaneously reflected in the register of the channel used in checking. Attempted writing to registers of the channel used in checking is ignored.

Operating Mode	Write Access to ch0	Read Access from ch0	Write Access to ch1	Read Access from ch1
A	Writing to ch0	Reading from ch0	Writing to ch1	Reading from ch1
B	Writing to ch0 is simultaneously reflected to ch1	Reading from ch0	Ignored	Reading from ch1
C	Ignored	Reading from ch0	Writing to ch1 is simultaneously reflected to ch0	Reading from ch1

Care must thus be taken regarding the order when making the initial settings of the registers that contain a DATTRG bit.

In the procedure for initial settings of the EMU2 shown in **Figure 24.3**, make the same settings in the registers for both channels and then set the DATTRG bits. Detailed examples of the settings are given below.

The examples include settings for retaining PWM comparison values in the buffer and for equivalence checking to verify matches between the main channel and channel used in checking. The issuing of an interrupt in case of a non-match with the result of buffering can also be selected.

- Example of the Order of Settings for Operating Mode B:

First, complete initial settings including the following except the EMU2nSMLCTR.DATTRG bit.

Channel 0	
EMU20CBCTR0 = 0x02;	Enable buffering.
EMU20CBCTR1 = 0x01;	Enable comparison of the results of PWM compare values.
EMU20CBTIM = 0x0004;	Select “PWM IP is complete” for buffer timing.
Channel 1	
EMU21CBCTR0 = 0x02;	Enable buffering.
EMU21CBCTR1 = 0x01;	Enable comparison of the results of PWM compare values.
EMU21CBTIM = 0x0004;	Select “PWM IP is complete” for buffer timing.
EMU20SMLCTR = 0x01;	Write 0x01 (0 to the DATTRG bit) to the main channel first.
EMU21SMLCTR = 0x03;	Write 0x03 (1 to the DATTRG bit) to the channel for use in checking.

- Example of the Incorrect Order of Settings for Operating Mode B:

EMU21SMLCTR = 0x03;	Writing 0x03 (1 to the DATTRG bit) to the channel for use in checking first.
EMU20SMLCTR = 0x01;	Since writing to the register of the main channel is reflected in that of the channel for use in checking, 0x01 (0 to the DATTRG bit) will also be written to the channel for use in checking. As a result, the DATTRG bits of both channels will be cleared to 0 and equivalent check function will be turned off. Never write to the registers in this order.

- Example of the Order of Settings for Operating Mode C:

First, complete initial settings including the following except the EMU2nSMLCTR.DATTRG bit.

Channel 0	
EMU20CBCTR0 = 0x02;	Enable buffering.
EMU20CBCTR1 = 0x01;	Enable comparison of the results of PWM compare values.
EMU20CBTIM = 0x0004;	Select “PWM IP is complete” for buffer timing.
Channel 1	
EMU21CBCTR0 = 0x02;	Enable buffering.
EMU21CBCTR1 = 0x01;	Enable comparison of the results of PWM compare values.
EMU21CBTIM = 0x0004;	Select “PWM IP is complete” for buffer timing.
EMU21SMLCTR = 0x01;	Write 0x01 (0 to the DATTRG bit) to the main channel first.
EMU20SMLCTR = 0x03;	Write 0x03 (1 to the DATTRG bit) to the channel for use in checking.

- Example of the Incorrect Order of Settings for Operating Mode C:

EMU20SMLCTR = 0x03;	Write 0x03 (1 to the DATTRG bit) to the channel for use in checking first.
EMU21SMLCTR = 0x01;	Since writing to the register of the main channel is reflected in that of the channel for use in checking, 0x01 (0 to the DATTRG bit) will also be written to the channel for use in checking. As a result, the DATTRG bits of both channels will be cleared to 0 and equivalent check function will be turned off. Never write to the registers in this order.

24.6 Notes

24.6.1 Compare Match during Rectangular Wave Output

Once an angle compare 0 value or angle compare 1 value is used for a compare match, the same angle compare value cannot be used for a comparison. Therefore, a compare match is cancelled until another compare value is set. This prevention function of continuous compare match occurrence is applied to angle compare 0 match and angle compare 1 match used for IP activation (except rectangle IP) and for switching of rectangular wave active levels.

Note 1. A compare match trigger for rectangle IP activation is ignored while the rectangle IP is operating.

Figure 24.25 outlines the prevention against continuous compare match occurrence.

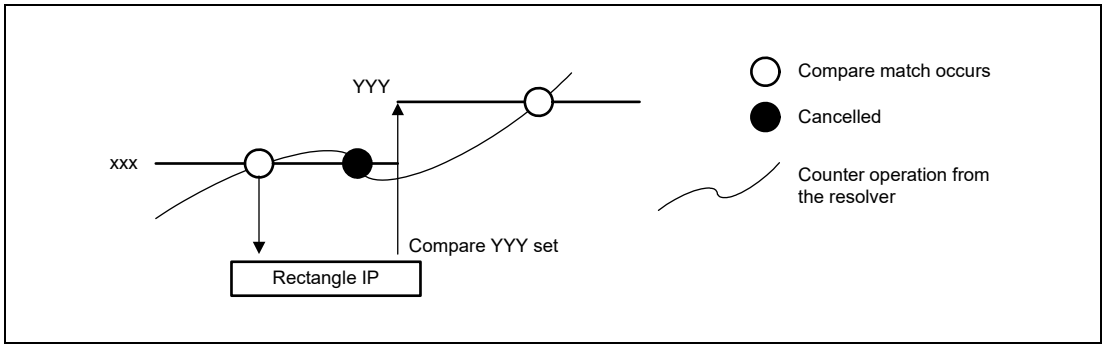


Figure 24.25 Prevention against Continuous Compare Match Occurrence

24.7 EMU2 Application Example

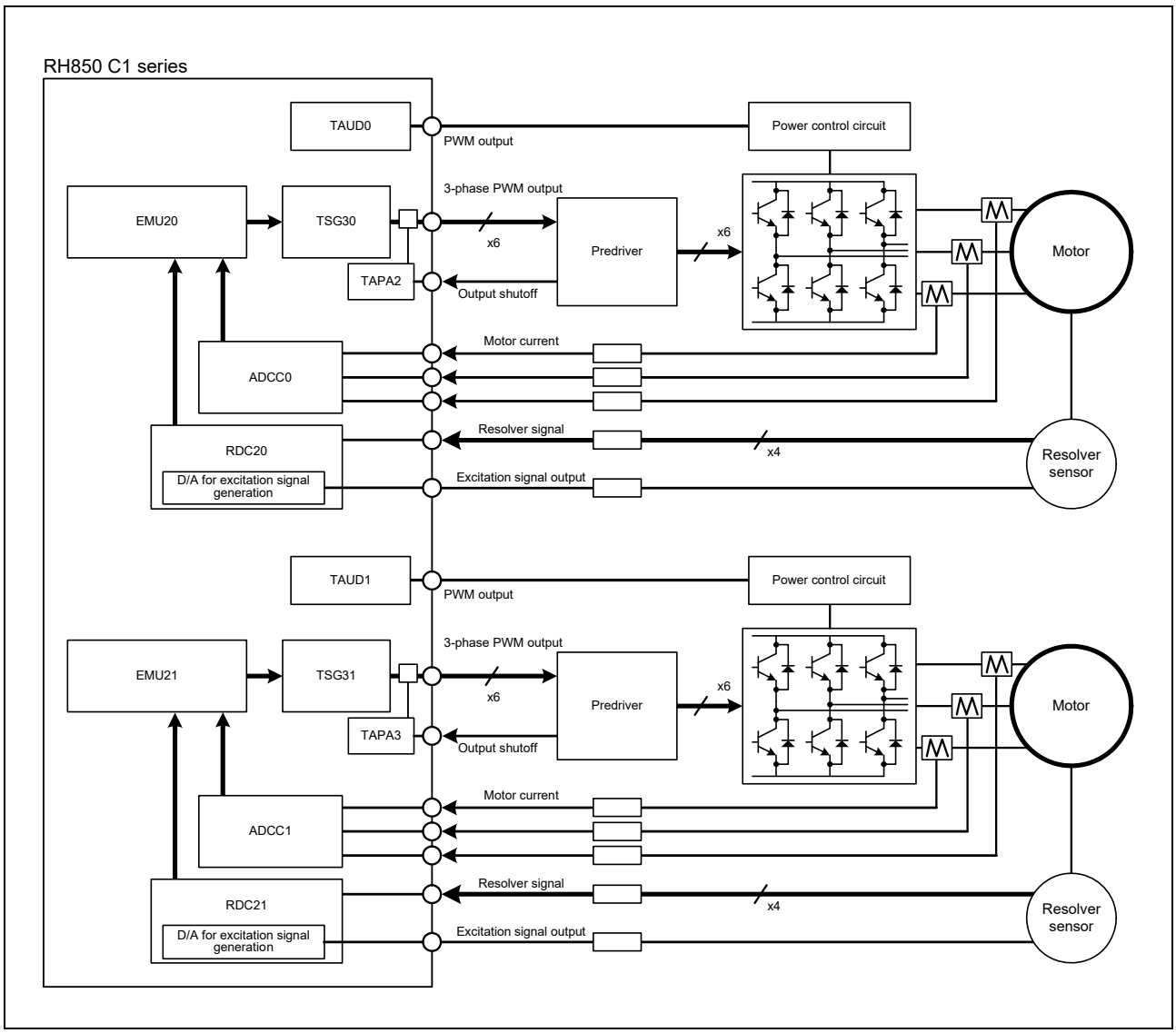


Figure 24.26 Connecting Two Motors

Section 25 R/D Converter (RDC2)

The R/D (resolver/digital) converter 2 converts analog values (angle information) that are output corresponding to the rotor angle of the resolver into digital values with a maximum length of 16 bits.

25.1 Features of RH850/C1x RDC2

25.1.1 Number of Units

This microcontroller has the following number of units of RDC2.

Table 25.1 Number of Units

Product	RH850/C1H	RH850/C1M
Number of units	2	1
Name	RDC20, RDC21	RDC20

Table 25.2 Index

Index	Meaning
n	Throughout this section, the individual RDC2 units are identified by the index “n”; for example, RDC2nCONSEL for the RDC2n conversion condition select register.

25.1.2 Register Base Address

RDC2 base addresses are listed in the table below.

RDC2 register addresses are given as offsets from the base addresses in general.

Table 25.3 Register Base Address

Register Base Address Name	Register Base Address
<RCD20_base>	FFED $D000_H$
<RDC21_base>	FFED $E000_H$

25.1.3 Clock Supply

The RDC2 clock supply is shown in the following table.

Table 25.4 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RDC2	PCLK	CLKC_LSB (unmodulated low-speed peripheral clock)

25.1.4 Interrupt Requests

RDC2 interrupt requests are listed in the following table.

Table 25.5 Interrupt Source

Interrupt Name	Interrupt Number	DMA Trigger Number	DTS Trigger Number
RDC20			
RDC20 Z phase interrupt	66	86	68
RDC20 RDC error interrupt	67	—	—
RDC20 compare match interrupt 0	68	87	69
RDC20 compare match interrupt 1	69	88	70
RDC20 compare match interrupt 2	70	89	71
RDC20 excitation timer (ET) interrupt	71	—	—
RDC20 excitation timer (ET) DMA request	—	90	72
RDC21			
RDC21 Z phase interrupt	72	91	73
RDC21 RDC error interrupt	73	—	—
RDC21 compare match interrupt 0	74	92	74
RDC21 compare match interrupt 1	75	93	75
RDC21 compare match interrupt 2	76	94	76
RDC21 excitation timer (ET) interrupt	77	—	—
RDC21 excitation timer (ET) DMA request	—	95	77

25.1.5 Reset Source

RDC2 reset sources are listed in the following table.

Table 25.6 Reset Source

Unit Name	Reset Source
RDC2	All reset sources

25.1.6 External Input/Output Signals

External input/output signals of RDC2 are listed in the following table.

Table 25.7 External Input/Output Signals

Unit Signal	Outline	Alternative Port Pin Signal	Product Supported	
			C1H	C1M
			252	144
Common to RDC20 and RDC21				
RVDD	RDC2 analog power supply (5V)	RVCC	√	√
RVSS	RDC2 analog ground (0V)	RVSS	√	√
RDC20				
RDC20SINMNT*1	Input amplifier monitor output	RDC20SINMNT	√	√
RDC20S4	Resolver signal input	RDC20S4	√	√
RDC20S2	Resolver signal input	RDC20S2	√	√
RDC20S1	Resolver signal input	RDC20S1	√	√
RDC20S3	Resolver signal input	RDC20S3	√	√
RDC20COSMNT*1	Input amplifier monitor output	RDC20COSMNT	√	√
RDC20RSO	Input/output for input/output excitation signal	RDC20RSO	√	√
RDC20COM	Input/output of common voltage for input/output excitation signal	RDC20COM	√	√
RDC21				
RDC21SINMNT*1	Input amplifier monitor output	RDC21SINMNT	√	—
RDC21S4	Resolver signal input	RDC21S4	√	—
RDC21S2	Resolver signal input	RDC21S2	√	—
RDC21S1	Resolver signal input	RDC21S1	√	—
RDC21S3	Resolver signal input	RDC21S3	√	—
RDC21COSMNT*1	Input amplifier monitor output	RDC21COSMNT	√	—
RDC21RSO	Input/output for input/output excitation signal	RDC21RSO	√	—
RDC21COM	Input/output of common voltage for input/output excitation signal	RDC21COM	√	—

Note 1. Input amplifier monitor pins (RDC2nSINMNT and RDC2nCOSMNT) are also function as the A/D converter analog input pins.
These pins must not be used for input amplifier monitor output pins and for A/D conversion pins simultaneously.

25.2 Overview

25.2.1 Functional Overview

In addition to the function of converting the analog angle signal from the resolver into a digital angle signal, RDC2 provides excitation signal output function, error detection function, and self-diagnosis function.

Table 25.8 lists the specifications for RDC2.

Table 25.8 RDC2 Specifications

Item	Function	Description
Excitation signal	Excitation signal output function (RDC2nRSO, RDC2nCOM)	• Generates the excitation signal with the voltage buffer. • Output sine wave voltage signal generated by 7-bit D/A from RDC2nRSO pin. • Voltage amplitude (Vpp) is 2 V, which can be changed with the register setting. • Output common voltage, which is RVDD divided by 2, from the RDC2nCOM pin.
	Amplitude Automatic Adjustment Function	• Automatically adjusts the amplitude of the input monitor signal (RDC2nSINMNT, RDC2nCOSMNT) into an appropriate value. • Targets to be adjusted: Input gain resistance and excitation signal output amplitudes
Tracking loop	Excitation signal source selection function	Selects the excitation signal (RDC2nRSO/RDC2nCOM) generated in RDC2 or that input externally.
	Required sensor selection function	Selects VR resolver or DC resolver.
	Excitation component extraction function	Uses the excitation components extracted from the resolver signal for R/D conversion.
	PI compensator bandwidth setting function	Selects from six bandwidths (five fixed and one auto-adjusted)
	Forced gain control function	Improves the tracking performance when the resolver angle deviates significantly from the R/D converted angle.
	Maximum angular velocity setting function	Sets a maximum angular velocity (resolution) in the range from 960,000 rpm (10 bits) to 15,000 rpm (16 bits).
	Monitor function	Reads angle information (°), angular velocity information (rpm), and control deviation values (%) directly from a register.
	Angle compare function	When the angle that is set in the angle compare registers 0 to 2 and the R/D converted angle match, a compare match interrupt request is generated.
	Encoder pulse output function	Outputs A-, B-phase, and Z-phase signals (4096 Edge/Revolution).
Error detection	Error detection function	Detects resolver signal error, resolver signal disconnect error, or R/D conversion error.
Self-diagnosis	Built-in self-test function	Angle conversion BIST (0°, 45°, 270°) Error detection BIST (resolver signal error, resolver signal disconnect error, R/D conversion error)
Others	Excitation timer (ET) function	• Measures period of excitation signal • Generates event signals (AD trigger, DMA request)
Interrupts		• Compare match interrupts 0 to 2 • Z-phase interrupt • RDC error interrupt • Excitation timer interrupt

25.2.2 Block Diagram

Figure 25.1 is a block diagram of the entire RDC2 module.

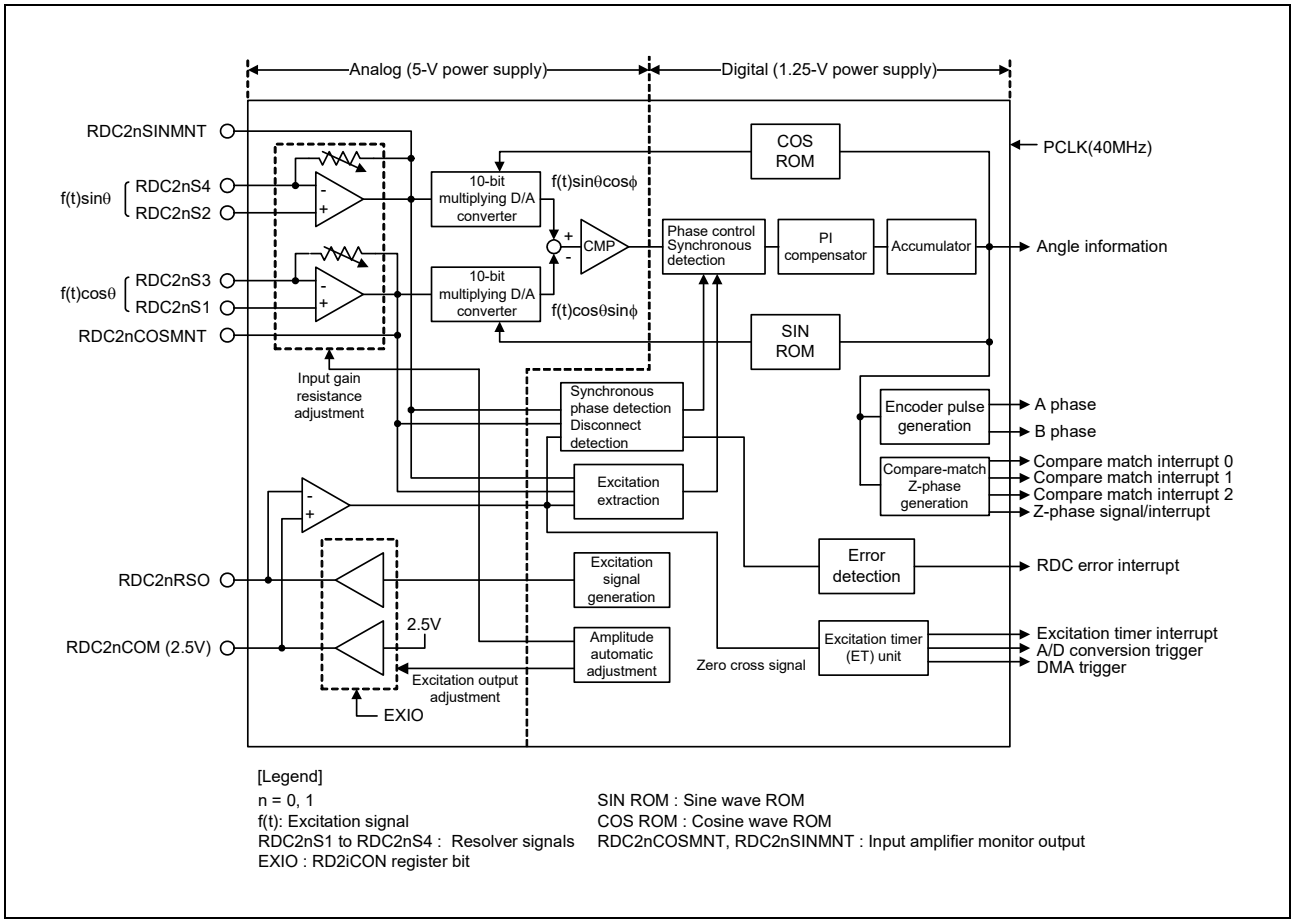


Figure 25.1 Entire RDC2 Module Block Diagram

25.2.3 Operating Principle

The following describes the operating principles of the RDC2 module.

This R/D converter module uses the tracking method to convert analog resolver signals to digital signals (R/D conversion).

When the excitation signal f(t) is input to the excitation coil, f(t) • sinθ and f(t) • cosθ are output from the resolver according to the angle (θ) of the resolver rotor. These signals are input to the RDC2nS2-RDC2nS4, RDC2nS1-RDC2nS3 pins, respectively.

These signals are amplified and then input to the multiplying D/A converter. At the same time, cosϕ (or sinϕ) is generated by passing the accumulator output through COS ROM (or SIN ROM) and is fed back to the corresponding D/A converter. Then, the subtraction is performed on the outputs from both D/A converters.

$f(t) \cdot (\sin\theta \cdot \cos\phi - \cos\theta \cdot \sin\phi)$

$= f(t) \cdot \sin(\theta - \phi)$

$\approx f(t) \cdot (\theta - \phi)$

The result is converted to 1-bit digital value by the comparator (CMP) and then passed to the digital block.

The excitation component f(t) is removed in the synchronous detection circuit to obtain the control deviation $\varepsilon = \theta - \phi$.

The negative feedback control over the entire analog and digital circuits provides feedback so that the control deviation becomes zero. When $\theta = \phi$, the analog angle information from the resolver has been converted to digital angle ϕ (16-bit wide). The following describes the PI compensator and accumulator.

Figure 25.2 shows a PI compensator and an accumulator.

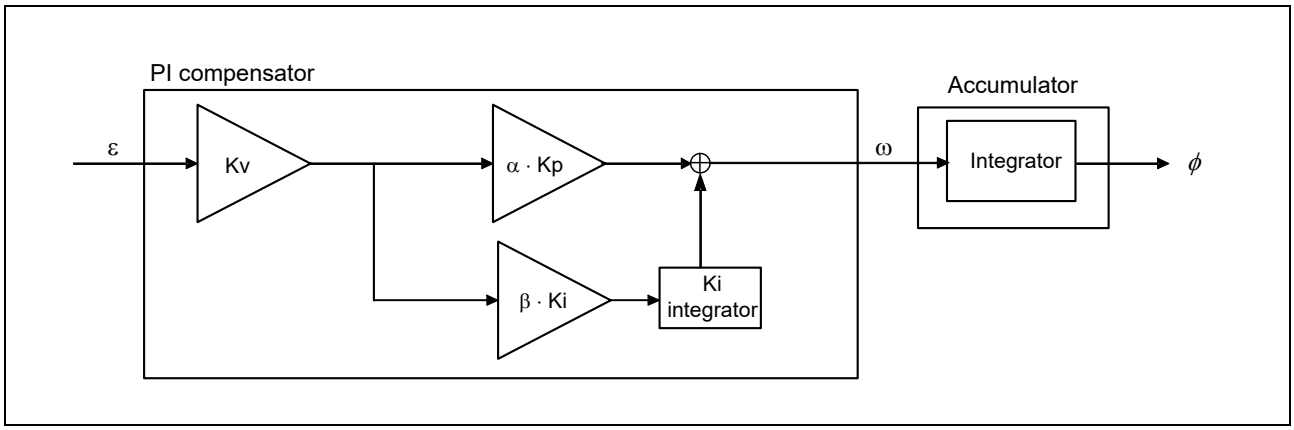


Figure 25.2 PI Compensator and Accumulator

The PI compensator converts the control deviation according to the following formula, and passes the result to the accumulator circuit

$\omega = (\alpha \bullet Kp + (\beta \bullet Ki)/(s \bullet T)) \bullet Kv \bullet \varepsilon$

Kv, Kp, Ki: control coefficients that can be set in a register

α, β: fixed values

s: Laplace variable

T: Integration time constant

ω: PI compensator output, angular velocity information

The accumulator circuit calculates the angle ϕ based on the angular velocity information ω.

25.3 Register

25.3.1 List of Registers

Table 25.9 lists the registers for the RDC2.

For <RDC2n_base>, see **Section 25.1.2, Register Base Address**.

Table 25.9 List of Registers

Module Name	Register Name	Symbol	Address
RDC2n	RDC2n conversion condition select register	RDC2nCONSEL	<RDC2n_base> + 0000_H
RDC2n	RDC2n control gain select register	RDC2nCGSEL	<RDC2n_base> + 0002_H
RDC2n	RDC2n maximum angular velocity setting register	RDC2nMAXV	<RDC2n_base> + 0006_H
RDC2n	RDC2n angle data register	RDC2nANGDAT	<RDC2n_base> + 0008_H
RDC2n	RDC2n reset register	RDC2nRST	<RDC2n_base> + $000C_H$
RDC2n	RDC2n monitor pin setting register	RDC2nMNTC	<RDC2n_base> + $001A_H$
RDC2n	RDC2n data storage register	RDC2nDATSTR	<RDC2n_base> + $001C_H$
RDC2n	RDC2n angle compare register 0	RDC2nCMP0	<RDC2n_base> + 0020_H
RDC2n	RDC2n angle compare register 1	RDC2nCMP1	<RDC2n_base> + 0022_H
RDC2n	RDC2n angle compare register 2	RDC2nCMP2	<RDC2n_base> + 0024_H
RDC2n	RDC2n initializing register	RDC2nINIT	<RDC2n_base> + 0026_H
RDC2n	RDC2n output control register	RDC2nOUTC	<RDC2n_base> + 0028_H
RDC2n	RDC2n control register	RDC2nCON	<RDC2n_base> + $002C_H$
RDC2n	RDC2n BIST setting register	RDC2nBISTC	<RDC2n_base> + 0030_H
RDC2n	RDC2n error detection function enable register	RDC2nERDEN	<RDC2n_base> + 0032_H
RDC2n	RDC2n input gain resistance value register	RDC2nINGR	<RDC2n_base> + 0036_H
RDC2n	RDC2n loop gain setting register	RDC2nLPGAIN	<RDC2n_base> + 0038_H
RDC2n	RDC2n excitation amplitude automatic adjustment circuit setting register	RDC2nEXAAT	<RDC2n_base> + $003A_H$
RDC2n	RDC2n error detection register	RDC2nERDET	<RDC2n_base> + $003E_H$
RDC2n	RDC2n compare match interrupt register	RDC2nCMINT	<RDC2n_base> + 0044_H
RDC2n	RDC2n encoder pulse register	RDC2nENCP	<RDC2n_base> + 0046_H
RDC2n	RDC2n analog circuit stop register	RDC2nANSTP	<RDC2n_base> + 0048_H
RDC2n	RDC2n ET event generation counter register	RDC2nETECNT	<RDC2n_base> + $004C_H$
RDC2n	RDC2n ET control register	RDC2nETCON	<RDC2n_base> + $004E_H$
RDC2n	RDC2n ET compare register	RDC2nETCMP	<RDC2n_base> + 0050_H
RDC2n	RDC2n ET capture register	RDC2nETCAP	<RDC2n_base> + 0052_H
RDC2n	RDC2n ET reload register	RDC2nETRLD	<RDC2n_base> + 0054_H
RDC2n	RDC2n ET period measurement counter register	RDC2nETPMCNT	<RDC2n_base> + 0056_H
RDC2n	RDC2n excitation amplitude integral square-sum monitor register	RDC2nEXSQR	<RDC2n_base> + 0058_H

25.3.2 RDC2nCONSEL — RDC2n Conversion Condition Select Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0000_H

Value After Reset: $2C00_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DEVCK[2:0]			EDPS[1:0]		DVW[1:0]		DATSEL[3:0]				—	EXFS[2:0]		
Value after reset	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 25.10 RDC2nCONSEL Register Contents

Bit Position	Bit Name	Function
15	Reserved	This bit is read as 0. The write value should be 0.
14 to 12	DEVCK[2:0]	Control Deviation Determination Clock Select*3 0 0 0: 50-µs clock period 0 0 1: 100-µs clock period 0 1 0: 200-µs clock period 0 1 1: 25-µs clock period 1 0 0: 400-µs clock period 1 0 1: 800-µs clock period The other settings are prohibited.
11 to 10	EDPS[1:0]	R/D Conversion Error Determination Time Select*4*5 0 0: 95.8 msec 0 1: 147.0 msec 1 0: 4.92 msec 1 1: 7.37 msec
9 to 8	DVW[1:0]	Control Deviation Weight *1*2 0 0: × 1 0 1: × 3 1 0: × 5 1 1: × 7
7 to 4	DATSEL[3:0]	RDC Data Select Outputs RDC data to the RDC2nDATSTR register. For details, see **Table 25.11**. Setting a value other than those in **Table 25.11** is prohibited.
3	Reserved	This bit is read as 0. The write value should be 0.
2 to 0	EXFS[2:0]	Excitation Signal Frequency Select 0 0 0: 10 kHz 0 0 1: 5 kHz 0 1 0: 20 kHz 0 1 1: 40 kHz 1 0 0: 15 kHz The other settings are prohibited.

Note 1. It is recommended to set the DVW[1:0] bits to the value after reset. To modify the bandwidth setting, select from six PI compensator settings using the LPGS[2:0] bits in the RDC2nLPGAIN register.

Note 2. Do not set the DVW[1:0] bits in the RDC2nCONSEL register, bits KPS[1:0], KPF, and HKVS[3:0] in the RDC2nCGSEL register to values listed in **Table 25.13** and **Table 25.14**. In case the combinations listed in these tables are used, the P component of the PI compensator overflows, resulting in unexpected value if the Kv gain is large.

Note 3. Set the control deviation determination clock period to be longer than the used excitation signal period. For example, when a 10-kHz (100-µs period) excitation signal is used, set the control deviation determination clock period to 100, 200, 300 or 800 µs. Do not select 50 or 25 µs.

Note 4. Do not set EDPS[1:0] to a shorter determination time than the current setting during R/D converter is operating. However, changing from 11b to 10b is allowed if the ERDEN bit in the RDC2nERDEN register is 0 after a reset. After changing the bits, set the ERDEN bit to 1.

Note 5. Set the EDPS[1:0] bits to 10_B or 11_B when the CVEDS bit in the RDC2nCON register is set to 0 (R/D conversion error detection circuits (supporting the resolver high-speed rotation) selected).

EDPS[1:0] Bits

Set the time to determine R/D conversion errors. If a deviation between the input angle and R/D converted angle remains large for more than 50% of the specified determination time, it is considered as the R/D conversion error and bits RDCE, ERR, and ERRHD in the RDC2nERDET register are set to 1.

DATSEL[3:0] Bits

Select data to be read from the RDC2nDASTR register.

Table 25.11 RDC Data Selection

DATSEL[3:0]	Output Data	Output to
0000	12-bit angle data	Bits 11 to 0 in the RDC2nDATSTR register
0100	16-bit angle data	Bits 15 to 0 in the RDC2nDATSTR register
0101	Compare match interrupt 2 signal	Bit 10 in the RDC2nDATSTR register
	Compare match interrupt 1 signal	Bit 9 in the RDC2nDATSTR register
	Compare match interrupt 0 signal	Bit 8 in the RDC2nDATSTR register
	Control deviation value	Bits 7 to 0 in the RDC2nDATSTR register
1001	Angular velocity data	Bits 15 to 0 in the RDC2nDATSTR register

25.3.3 RDC2nCGSEL — RDC2n Control Gain Select Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0002_H

Value After Reset: 0000_H *1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KVMS[1:0]		KPS[1:0]		LKVS[3:0]				HKVS[3:0]				KPF	KIS[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.12 RDC2nCGSEL Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	KVMS[1:0]	Kv Gain Method Select 0 0: 12-level AGC method 0 1: 7-level AGC method 1 0: 2-level AGC method 1 1: Fixed Kv method
13, 12	KPS[1:0]	Kp Gain Select*2 0 0: × 1 0 1: × 0.25 1 0: × 0.5 1 1: × 2
11 to 8	LKVS[3:0]	Low Kv Gain Select Set the gain in the lower Kv side when the fixed Kv method or 2-level AGC method is selected. 0 0 0 0: × 1 0 0 0 1: × 0.0625 0 0 1 0: × 0.125 0 0 1 1: × 0.25 0 1 0 0: × 0.5 0 1 0 1: × 1 0 1 1 0: × 2 0 1 1 1: × 4 1 0 0 0: × 8 1 0 0 1: × 16 1 0 1 0: × 32 1 0 1 1: × 64 1 1 0 0: × 128 Settings other than above are prohibited.
7 to 4	HKVS[3:0]	High Kv Gain Select*2 Set the gain in the higher Kv side when the fixed Kv method or 2-level AGC method is selected. 0 0 0 0: × 32 0 0 0 1: × 0.0625 0 0 1 0: × 0.125 0 0 1 1: × 0.25 0 1 0 0: × 0.5 0 1 0 1: × 1 0 1 1 0: × 2 0 1 1 1: × 4 1 0 0 0: × 8 1 0 0 1: × 16 1 0 1 0: × 32 1 0 1 1: × 64 1 1 0 0: × 128 Settings other than above are prohibited.
3	KPF	Kp Gain Quadruple*2 0: Kp gain is not quadrupled. 1: Kp gain is quadrupled.

Table 25.12 RDC2nCGSEL Register Contents (2/2)

Bit Position	Bit Name	Function
2 to 0	KIS[2:0]	Ki Gain Select Sets the Ki gain in the PI compensator. 0 0 0: × 1 0 0 1: × 0.125 0 1 0: × 0.25 0 1 1: × 0.5 1 0 0: × 2 1 0 1: × 4 1 1 0: × 8 1 1 1: × 16

Note 1. It is recommended to set the RDC2nCGSEL register to the value after reset. To change the bandwidth setting, select one of the six PI compensator settings using the LPGS[2:0] bits in the RDC2nLPGAIN register.

Note 2. Do not set the DVW[1:0] bits in the RDC2nCONSEL register, bits KPS[1:0], KPF, and HKVS[3:0] in the RDC2nCGSEL register to values listed in **Table 25.13** and **Table 25.14**. In case the combinations listed in these tables are used, the P component of the PI compensator overflows, resulting in unexpected value if the Kv gain is large.

Table 25.13 Prohibited Combinations 1 (When KVMS[1:0] = 00_B or 01_B)

DVW[1:0]	KPS[1:0]	KPF
00 (× 1)	11 (× 2)	1 (× 4)
01 (× 3)	11 (× 2)	1 (× 4)
01 (× 3)	00 (× 1)	1 (× 4)
10 (× 5)	11 (× 2)	1 (× 4)
10 (× 5)	11 (× 2)	0 (× 1)
10 (× 5)	00 (× 1)	1 (× 4)
10 (× 5)	10 (× 0.5)	1 (× 4)
11 (× 7)	11 (× 2)	1 (× 4)
11 (× 7)	11 (× 2)	0 (× 1)
11 (× 7)	00 (× 1)	1 (× 4)
11 (× 7)	10 (× 0.5)	1 (× 4)

Table 25.14 Prohibited Combinations 2 (When KVMS[1:0] = 10_B or 11_B) (1/2)

DVW[1:0]	KPS[1:0]	KPF	HKVS[3:0]
00 (× 1)	11 (× 2)	1 (× 4)	1100 (× 128)
01 (× 3)	11 (× 2)	1 (× 4)	1100 (× 128)
01 (× 3)	11 (× 2)	1 (× 4)	1011 (× 64)
01 (× 3)	00 (× 1)	1 (× 4)	1100 (× 128)
10 (× 5)	11 (× 2)	1 (× 4)	1100 (× 128)
10 (× 5)	11 (× 2)	1 (× 4)	1011 (× 64)
10 (× 5)	11 (× 2)	1 (× 4)	1010 (× 32)
10 (× 5)	11 (× 2)	0 (× 1)	1100 (× 128)
10 (× 5)	00 (× 1)	1 (× 4)	1100 (× 128)
10 (× 5)	00 (× 1)	1 (× 4)	1011 (× 64)
10 (× 5)	10 (× 0.5)	1 (× 4)	1100 (× 128)
11 (× 7)	11 (× 2)	1 (× 4)	1100 (× 128)
11 (× 7)	11 (× 2)	1 (× 4)	1011 (× 64)
11 (× 7)	11 (× 2)	1 (× 4)	1010 (× 32)
11 (× 7)	11 (× 2)	0 (× 1)	1100 (× 128)

Table 25.14 Prohibited Combinations 2 (When KVMS[1:0] = 10_B or 11_B) (2/2)

DVW[1:0]	KPS[1:0]	KPF	HKVS[3:0]
11 (× 7)	00 (× 1)	1 (× 4)	1100 (× 128)
11 (× 7)	00 (× 1)	1 (× 4)	1011 (× 64)
11 (× 7)	10 (× 0.5)	1 (× 4)	1100 (× 128)

KVMS[1:0] Bits

Select the Kv gain method in the PI compensator. When the AGC (Auto Gain Control) method is selected, the Kv gain is automatically selected according to the amount of control deviation. It is recommended to use the value after reset (12-level AGC method).

- 12-level AGC method (value after reset)

Table 25.15 lists the control deviation amount and the value of the selected Kv gain for the 12-level AGC. The control deviation amount indicates a bias of control deviation ε (High or Low) within the determination clock period. With regard to the resolver angle signal θ and the R/D converter output angle signal ϕ, the High and the Low appear in equal proportions if θ and ϕ are equal. In such a case, the control deviation amount is ±0%. If ϕ is completely behind θ, the ε always becomes High. In this case, the control deviation amount is +100%. If ϕ is completely ahead of θ, ε always becomes Low. In this case, the control deviation amount is -100%. The determination clock period can be selected by the DEVCK[2:0] bits in the RDC2nCONSEL register.

Table 25.15 Control Deviation Amount and Kv Gain for 12-Level AGC

Control Deviation Amount (Absolute Value)	Kv Gain
After Reset	× 128
76.8% to 100%	× 64
64.0% to 76.8%	× 32
57.6% to 64.0%	× 16
51.2% to 57.6%	× 8
44.8% to 51.2%	× 4
38.4% to 44.8%	× 2
32.0% to 38.4%	× 1
25.6% to 32.0%	× 0.5
19.2% to 25.6%	× 0.25
12.8% to 19.2%	× 0.125
0.0% to 12.8%	× 0.0625

- 7-level AGC method

Table 25.16 lists the control deviation amount and the value of the selected Kv gain for the 7-level AGC.

Table 25.16 Control Deviation Amount and Kv Gain for 7-Level AGC

Control Deviation Amount (Absolute Value)	Kv Gain
After Reset	× 128
76.8% to 100%	× 64
51.2% to 76.8%	× 16
38.4% to 51.2%	× 4
25.6% to 38.4%	× 1
12.8% to 25.6%	× 0.25
0.0% to 12.8%	× 0.0625

- 2-level AGC method

Table 25.17 lists the control deviation amount and the value of the selected Kv gain for the 2-level AGC. The low and high gains can be set using bits LKVS[3:0] and HKVS[3:0] in the RDC2nCGSEL register.

Table 25.17 Control Deviation Amount and Kv Gain for 2-Level AGC

Control Deviation Amount (Absolute Value)	Kv Gain
76.8% to 100%	Transition to the high Kv gain (the value selected in the HKVS[3:0] bits)
25.6% to 76.8%	Kv gain maintained (no transition)
0.0% to 25.6%	Transition to the low Kv gain (the value selected in the LKVS[3:0] bits)

Note 1. If the Kv gain falls below × 128 after a reset, the Kv returns to × 128 on any of the following conditions:
1. KIRST bit in the RDC2nRST register is set to 1 (Ki reset)
2. Recovery from the excitation signal error status.

- Fixed Kv method

If the fixed Kv method is selected, the module operates with the low Kv gain (the value selected in the LKVS [3:0] bits in the RDC2nCGSEL register), irrespective of the control deviation amount. The module, however, transitions to the high Kv gain (the value selected in the HKVS [3:0] bits) only if the following three conditions are all satisfied; an error (R/D conversion error or resolver signal disconnect error) is detected, then a recovery is made from the error, and the control deviation amount falls within 76.8% to 100%. After that, the module transitions to the low Kv gain side when the absolute value of the control deviation amount falls below 25.6%.

- Digital tracking servo frequency response characteristics by Kv, Kp, Ki gains and by deviation weighting

Table 25.18 lists the digital tracking servo frequency response characteristics by Kv, Kp, Ki gain settings and by deviation weighting settings. The values can be set with bits LKVS [3:0], HKVS [3:0], KPS [1:0], and KIS [2:0] in the RDC2nCGSEL register and the DVW [1:0] bits in the RDC2nCONSEL register.

Table 25.18 Kv, Kp, Ki Gain Settings and Frequency Characteristics

Kv	Kp	Ki	Deviation Weighting (DVW[1:0] in the RDC2nCONSEL register)	Bandwidth (-3 dB)
0.5	1	0.5	1	800 Hz equivalent
0.125	1	1	7	1500 Hz equivalent
0.125	1	0.5	5	1000 Hz equivalent
0.0625	1	0.25	5	500 Hz equivalent
0.0625	2	0.125	1	200 Hz equivalent

Note 1. Settings other than above are prohibited.

25.3.4 RDC2nMAXV — RDC2n Maximum Angular Velocity Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0006_H

Value After Reset: 8140_H

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

Table 25.19 RDC2nMAXV Register Contents

Bit Position	Bit Name	Function
15	Reserved	This bit is read as 1. The write value should be 1.
14 to 11	Reserved	These bits are read as 0. The write value should be 0.
10 to 8	MAXV[2:0]	Maximum angular velocity select Set the maximum angular velocity. See **Table 25.20**. Do not set values other than those listed in **Table 25.20**.
7	Reserved	This bit is read as 0. The write value should be 0.
6	Reserved	This bit is read as 1. The write value should be 1.
5 to 0	Reserved	These bits are read as 0. The write value should be 0.

MAXV[2:0] Bits

Table 25.20 lists the selected maximum angular velocity and R/D conversion resolution.

Table 25.20 Maximum Angular Velocity Select Bit Settings

MAXV[2:0]	Maximum Angular Velocity (rpm)	Resolution (bits)
000_B	120,000	13
001_B	240,000	12
010_B	480,000	11
011_B	960,000	10
100_B	15,000	16
101_B	60,000	14

25.3.5 RDC2nANGDAT — RDC2n Angle Data Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <RDC2n_base> + 0008_H

Value After Reset: $XXXX_H$

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

Table 25.21 RDC2nANGDAT Register Contents

Bit Position	Bit Name	Function
15 to 0	ANG[15:0]	Angle Data 16-bit angle data is stored If the read value is n, the angle is $360/2^{16} \times n$ (°).

25.3.6 RDC2nRST — RDC2n Reset Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $000C_H$

Value After Reset: 0100_H

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

Table 25.22 RDC2nRST Register Contents

Bit Position	Bit Name	Function
15 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	Reserved	This bit is read as 1. The write value should be 0.
7 to 2	Reserved	These bits are read as 0. The write value should be 0.
1	KIRST	Ki Reset The values of the Ki integrator and accumulator integrator become 0 by writing 1 to this bit. This bit becomes 0 after 2 clock cycles have elapsed since setting this bit to 1.
0	Reserved	This bit is read as 0. The write value should be 0.

25.3.7 RDC2nMNTC — RDC2n Monitor Pin Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $001A_H$

Value After Reset: $00x0_H$

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

Table 25.23 RDC2nMNTC Register Contents

Bit Position	Bit Name	Function
15 to 13	Reserved	These bits are read as 0. The write value should be 0.
12	MNTC	Monitor Pin Control 0: Leaves RDC2nSINMNT and RDC2nCOSMNT pins open. 1: Outputs from RDC2nSINMNT and RDC2nCOSMNT pins.
11 to 0	Reserved	These bits are read as 0. The write value should be 0.

25.3.8 RDC2nDATSTR — RDC2n Data Storage Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <RDC2n_base> + $001C_H$

Value After Reset: $XXXX_H$

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

Table 25.24 RDC2nDATSTR Register Contents

Bit Position	Bit Name	Function
15 to 0	DATA[15:0]	Data Data specified by the DATSEL[3:0] bits in the RDC2nCONSEL register is stored.

Note 1. Undefined value is read from a bit that is not specified by the DATSEL[3:0] bits in the RDC2nCONSEL register.

25.3.9 RDC2nCMP0 — RDC2n Angle Compare Register 0 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0020_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.25 RDC2nCMP0 Register Contents

Bit Position	Bit Name	Function
15 to 0	CMP0[15:0]	Angle Compare Value Set the angle compare value with 16-bit width. If the angle is θ, then the set value n is $2^{16}/360 \times \theta$.

25.3.10 RDC2nCMP1 — RDC2n Angle Compare Register 1 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0022_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.26 RDC2nCMP1 Register Contents

Bit Position	Bit Name	Function
15 to 0	CMP1[15:0]	Angle Compare Value Set the angle compare value with 16-bit width. If the angle is θ, then the set value n is $2^{16}/360 \times \theta$.

25.3.11 RDC2nCMP2 — RDC2n Angle Compare Register 2 (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0024_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMP2[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.27 RDC2nCMP2 Register Contents

Bit Position	Bit Name	Function
15 to 0	CMP2[15:0]	Angle Compare Value Set the angle compare value with 16-bit width. If the angle is θ, then the set value n is $2^{16}/360 \times \theta$.

25.3.12 RDC2nINIT — RDC2n Initializing Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0026_H

Value After Reset: $1XXX_H$

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

Table 25.28 RDC2nINIT Register Contents

Bit Position	Bit Name	Function
15	INIT	Initializes RDC2n internal circuit when 1 is written. After becoming 1 and the initialization is completed, this bit becomes 0.
14 to 0	—	These bits are read as 0. When writing, write the value after a reset.

25.3.13 RDC2nOUTC — RDC2n Output Control Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0028_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	CINTEN	ABEN	—	ZEN	EINTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W

Table 25.29 RDC2nOUT Register Contents

Bit Position	Bit Name	Function
15 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	CINTEN	Compare Match Interrupt Enable 0: Disables the interrupt 1: Enables the interrupt
3	ABEN	A Phase and B Phase Output Enable 0: Disables output 1: Enables output
2	Reserved	This bit is read as 0. The write value should be 0.
1	ZEN	Z Phase Output and Z Phase Signal Interrupt Enable 0: Disables output 1: Enables output
0	EINTEN	RDC Error Interrupt Enable 0: Disables the interrupt 1: Enables the interrupt

25.3.14 RDC2nCON — RDC2n Control Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $002C_H$

Value After Reset: 4200_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FGCON	BWCS	CVEDS	—	BDVTH	REDTH	SENS	EXIO	—	—	—	—	—	—	—	—
Value after reset	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.30 RDC2nCON Register Contents

Bit Position	Bit Name	Function
15	FGCON	Forced Gain Control*1 0: Uses the forced gain control function. 1: Does not use the forced gain control function.
14	BWCS	Bandwidth Setting Method Select*2 Selects the PI compensator setting method. 0: Using registers RDC2nCONSEL and RDC2nCGSEL. 1: Using the LPGS[2:0] bits in the RDC2nLPGAIN register. (See the description in the RDC2nLPGAIN register for the set value.)
13	CVEDS	R/D Conversion Error Detection Circuit Select*3 0: Selects a circuit supporting the resolver high-speed rotation of the R/D conversion error detection signal. 1: Selects a circuit NOT supporting the resolver high-speed rotation of the R/D conversion error detection signal.
12	Reserved	This bit is read as 0. The write value should be 0.
11	BDVTH	Disconnect Detection Analog Voltage Threshold Sets an analog voltage threshold for disconnect detection. 0: Depends on the SENS and EXIO bits: If SENS = 0 and EXIO = 1, RDC2nCOM + 0.35 × RVDD. Otherwise, RDC2nCOM + 0.08 × RVDD. 1: The value is always RDC2nCOM + 0.35 × RVDD
10	REDTH	Resolver Signal Error Detection Threshold Sets a threshold for resolver signal error detection. 0: 0.10 × (RVDD±5%) [Vp-p] 1: 0.14 × (RVDD±5%) [Vp-p]
9	SENS	Sensor Select Selects a sensor to be used. 0: DC resolver*4 1: VR resolver
8	EXIO	Excitation Signal Input/output Switching*3 Switches input and output of excitation signal. 0: Excitation signal input 1: Excitation signal output
7 to 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. Change the FGCON bit setting while the resolver is stopped (angular velocity is 1 rpm or below). After changing the FGCON bit, write 1 to the KIRST bit in the RDC2nRST register.

Note 2. It is recommended to set the BWCS bit to 1, which is the value after reset.

Note 3. Set the CVEDS bit to 1 when the EXFS[2:0] bits in the RDC2nCONSEL register to 011B (excitation frequency: 40 KHz).
Input the excitation signal whose frequency is below 22 kHz when the CVEDS bit is set to 0 and the EXIO bit is set to 0.

Note 4. Set the EXIO bit to 1 when the SENS bit is set to 0.

25.3.15 RDC2nBISTC — RDC2n BIST Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0030_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BEXE	—	—	—	—	—	—	BSTF	BRLT[3:0]				BCON[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 25.31 RDC2nBISTC Register Contents

Bit Position	Bit Name	Function
15	BEXE	BIST Execution 0: Stops BIST. 1: Enables BIST execution.
14 to 9	Reserved	These bits are read as 0. The write value should be 0.
8	BSTF	BIST Indicates the BIST execution status. 0: BIST is stopped. 1: BIST is in execution.
7 to 4	BRLT[3:0]	BIST Result Storage BIST execution result is stored. See **Table 25.32**.
3 to 0	BCON[3:0]	BIST Setting Select BIST to be executed. See **Table 25.32**. Do not set values other than those in the table.

BEXE Bit

Starts BIST when the BEXE bit is set to 1 and the BCON[3:0] bits are set to other than 0000_B.
Stops BIST when either the BEXE bit is set to 0 or BCON[3:0] bits are set to 0000_B.

Table 25.32 BIST Settings and Results (BCON[3:0] and BRLT[3:0] Contents) (1/2)

BCON[3:0]				BIST to be Executed	BRLT[3:0]				BIST Execution Result
0	0	0	0	BEXE bit is disabled.	0	0	0	0	No tested
0	1	0	1	Angle conversion BIST1: Target angle = 0°	0	1	0	1	OK
					1	1	1	1	NO
0	1	1	0	Angle conversion BIST2: Target angle = 45°	0	1	1	0	OK
					1	1	1	1	NO
0	1	1	1	Angle conversion BIST3: Target angle = 270°	0	1	1	1	OK
					1	1	1	1	NO
1	0	0	1	Error detection BIST: Resolver signal error detection BIST	1	0	0	1	OK
					1	1	1	1	NO
1	0	1	0	Error detection BIST: Resolver signal disconnect error detection BIST (cosine side)	1	0	1	0	OK
					1	1	1	1	NO

Table 25.32 BIST Settings and Results (BCON[3:0] and BRLT[3:0] Contents) (2/2)

BCON[3:0]				BIST to be Executed	BRLT[3:0]				BIST Execution Result
1	0	1	1	Error detection BIST: Resolver signal disconnect error detection BIST (sine side)	1	0	1	1	OK
					1	1	1	1	NO
1	1	0	0	Error detection BIST: R/D conversion error detection BIST	1	1	0	0	OK
					1	1	1	1	NO

BSTF Bit

The BSTF bit becomes 0 when the BEXE bit is set to 0 or BCON[3:0] bits are set to 0000_B. Otherwise, BSTF bit becomes 1.

25.3.16 RDC2nERDEN — RDC2n Error Detection Function Enable Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0032_H

Value After Reset: 0000_H

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

Table 25.33 RDC2nERDEN Register Contents

Bit Position	Bit Name	Function
15 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	ERDEN	Error Detection Function Enable The error detection function is enabled after 26 ms have elapsed since setting this bit to 1.
3 to 0	Reserved	These bits are read as 0. The write value should be 0.

ERDEN Bit

When the error detection function is enabled, the RDC error interrupt request is generated if an error occurs. To generate an interrupt request, set the EINTEN bit in the RDC2nOUTC register to 1. The error detection function is disabled after a reset. Once 1 is written to the ERDEN bit, the error detection function cannot be disabled even if 0 is written to this bit. By setting the EINTEN bit to 0, the RDC error interrupt can be disabled.

25.3.17 RDC2nINGR — RDC2n Input Gain Resistance Value Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <RDC2n_base> + 0038_H

Value After Reset: $00XX_H$

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

Table 25.34 RDC2nINGR Register Contents

Bit Position	Bit Name	Function
15 to 5	Reserved	These bits are read as 0. The write value should be 0.
4	IGRT	Input Gain Resistance Tuning Indicates whether or not the input gain resistance value is tuned at shipment. 0: Tuned 1: Not tuned
3 to 0	IGRM[3:0]	Input Gain Resistance Value Monitor Indicate the input gain resistance value tuned at shipment. 0 0 0 0: Input gain resistance value TYP -40% 0 0 0 1: Input gain resistance value TYP -30% 0 0 1 0: Input gain resistance value TYP -20% 0 0 1 1: Input gain resistance value TYP -10% 0 1 0 0: Input gain resistance value TYP ±0% 0 1 0 1: Input gain resistance value TYP +10% 0 1 1 0: Input gain resistance value TYP +20% 0 1 1 1: Input gain resistance value TYP +30% 1 x x x: Input gain resistance value TYP +40%

25.3.18 RDC2nLPGAIN — RDC2n Loop Gain Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0038_H

Value After Reset: 0000_H

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

Table 25.35 RDC2nLPGAIN Register Contents

Bit Position	Bit Name	Function
15 to 8	Reserved	These bits are read as 0. The write value should be 0.
7 to 5	LPGS[2:0]	Loop Gain Select*1 By setting the BWCS bit in the RDC2nCON register to 1, the LPGS[2:0] bits are enabled and the PI compensator can be set using the LPGS[2:0] bits. See **Table 25.36**.
4 to 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. It is recommended to set the LPGS[2:0] bits to 111_B.

Table 25.36 PI Compensator Settings Using LPGS[2:0]

	LPGS[2:0]			PI Compensator Settings
BWCS Bit	b7	b6	b5	Bandwidth
0	x	x	x	Set by registers RDC2nCONSEL and RDC2nCGSEL.
1	0	0	0	800 Hz equivalent
1	0	1	1	1500 Hz equivalent
1	1	0	0	1000 Hz equivalent
1	1	0	1	500 Hz equivalent
1	1	1	0	200 Hz equivalent
1	1	1	1	Auto-adjusted

Note 1. Settings other than above are prohibited.

25.3.19 RDC2nEXAAT — RDC2n Excitation Amplitude Automatic Adjustment Circuit Setting Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $003A_H$

Value After Reset: $08X4_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EAAOD	IRSS1	EXOS	—	EAATSP	—	—	—	EXOC[1:0]		—	IRSS0	IRSC[3:0]			
Value after reset	0	0	0	0	1	0	0	0	1	0	—	0	0	1	0	0
R/W	R/W	R/W	R/W	R	R/W	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Table 25.37 RDC2nEXAAT Register Contents (1/2)

Bit Position	Bit Name	Function
15	EAAOD	Excitation Amplitude Automatic Adjustment Priority Select 0: Adjustment is performed on excitation output first, and then on input gain resistance. 1: Adjustment is performed on input gain resistance first, and then on excitation output.
14	IRSS1	Input Gain Resistance Value Select 1 Selects the input gain resistance value. See **Table 25.38**. 0: A value set by IRSC[3:0] or tuned at shipment 1: Input gain resistance automatic adjusted value
13	EXOS	Excitation Output Amplitude Select Selects excitation output amplitude that is output from the excitation amplitude automatic adjustment circuit. 0: A value set by EXOC[1:0] bit 1: Excitation amplitude automatic adjusted value
12	Reserved	This bit is read as 0. The write value should be 0.
11	EAATSP	Amplitude Automatic Adjustment Stop Stops excitation amplitude adjustment by amplitude automatic adjustment circuit. 0: Performs automatic adjustment of input gain resistance value and excitation output amplitude. 1: Stops automatic adjustment of input gain resistance value and excitation output amplitude.
10 to 8	Reserved	These bits are read as 0. The write value should be 0.
7, 6	EXOC[1:0]	Excitation Output Value Setting Sets an excitation output value with the EXOS bit being 0. 0 0: Excitation output value TYP -40% 0 1: Excitation output value TYP -20% 1 0: Excitation output value TYP ±0% 1 1: Excitation output value TYP +20%
5	Reserved	The read value is undefined. The write value should be 0.
4	IRSS0	Input Gain Resistance Select 0 Selects input gain resistance value in combination with the IRSS1 bit. See **Table 25.38**.

Table 25.37 RDC2nEXAAT Register Contents (2/2)

Bit Position	Bit Name	Function
3 to 0	IRSC[3:0]	Input Gain Resistance Setting Set an input gain resistance value. 0 0 0 0: Input gain resistance value TYP -40% 0 0 0 1: Input gain resistance value TYP -30% 0 0 1 0: Input gain resistance value TYP -20% 0 0 1 1: Input gain resistance value TYP -10% 0 1 0 0: Input gain resistance value TYP ±0% 0 1 0 1: Input gain resistance value TYP +10% 0 1 1 0: Input gain resistance value TYP +20% 0 1 1 1: Input gain resistance value TYP +30% 1 x x x: Input gain resistance value TYP +40% Settings other than above are prohibited.

IRSS0 and IRSS1 Bits

Select a method to set an input gain resistance value. When setting the input gain resistance value using the IRSC[3:0] bits in the RDC2nEXAAT register, directly set the value to the IRSC[3:0] bits if the IGRT bit in the RDC2nINGR register is 1. If the IGRT bit is 0, read the tuned value from the IGRM[3:0] bits in the RDC2nINGR register and set increase or decrease based on the read value.

Table 25.38 Input Gain Resistance Value

IRSS1 Bit in the RDC2nEXAAT register	IRSS0 Bit in the RDC2nEXAAT register	IGRT Bit in the RDC2nINGR register	Input Gain Resistance Value to be Selected
0	0	0	Value tuned at shipment
0	1	0 or 1	Value set by IRSC[3:0] in the RDC2nEXAAT register
1	0 or 1	0 or 1	Input gain resistance automatic adjusted value

25.3.20 RDC2nERDET — RDC2n Error Detection Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $003E_H$

Value After Reset: 7000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ERRST	EID[2:0]			—	—	ERR	ERHD	RDCE	RESE	CBRE	SBRE	—	—	—	—
Value after reset	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.39 RDC2nERDET Register Contents

Bit Position	Bit Name	Function
15	ERRST	Error Signal Reset The following bits become 0 by writing 1 to this bit: Bits EID[2:0], ERHD, RDCE, RESE, CBRE, and SBRE in the RDC2nERDET register. This bit becomes 0 after 2 clock cycles have elapsed since setting this bit to 1.
14 to 12	EID[2:0]	Error Identification Indicate an error type detected by the R/D converter.*1 See **Table 25.40**.
11, 10	Reserved	These bits are read as 0. The write value should be 0.
9	ERR	Error Indicates the status of R/D conversion error, resolver signal error, and resolver signal disconnect error. 0: An error has not been detected.*2 1: An error has been detected*3
8	ERHD	Error Retain Indicates the status of the R/D conversion error, resolver signal error, and resolver signal disconnect error. 0: An error has not been detected.*1 1: An error has been detected*3
7	RDCE	R/D Conversion Error 0: An error has not been detected.*1 1: An error has been detected
6	RESE	Resolver Signal Error 0: An error has not been detected.*1 1: An error has been detected
5	CBRE	Resolver Signal Disconnect Error (cosine side) 0: An error has not been detected.*1 1: An error has been detected
4	SBRE	Resolver Signal Disconnect Error (sine side) 0: An error has not been detected.*1 1: An error has been detected
3 to 0	Reserved	These bits are read as 0. The write value should be 0.

Note 1. These bits become 0 by writing 1 to the ERRST bit in the RDC2nERDET register. They, however, do not become 0 if 1 is written to the ERRST bit during error occurrence.

Note 2. This bit becomes 0 on recovery from an error.

Note 3. This bit does not become 1 if the EINTEN bit of the RDC2nOUTC register is 0.

Table 25.40 Errors Indicated by EID[2:0]

EID[2:0]			Detected Error*1
0	0	0	No error has been detected.
0	0	1	Resolver signal error
0	1	0	Resolver signal disconnect error (cosine side)
0	1	1	Resolver signal disconnect error (sine side)
1	0	0	R/D conversion error
1	1	1	Error detection function is disabled (for 26 ms from the time the ERDEN bit in the RDC2nERDEN register is set to 1 after release from the reset state).

Note 1. If two or more types of errors are detected, error code with higher priority is stored according to the following priority order:
Resolver signal disconnect error (cosine side) > resolver signal disconnect error (sine side) > resolver signal error > R/D conversion error.

25.3.21 RDC2nCMINT — RDC2n Compare Match Interrupt Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0044_H

Value After Reset: $XXXX_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRS	INTCLR2	INTCLR1	INTCLR0	—	INTFLG2	INTFLG1	INTFLG0	—	—	—	—	—	—	—	—
Value after reset	0	—	—	—	0	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.41 RDC2nCMINT Register Contents

Bit Position	Bit Name	Function
15	IRS	Compare Match Interrupt Request Signal Select 0: Compare match signal 1: Signal latching compare match signal
14	INTCLR2	Compare Match Interrupt 2 Clear Writing 1 to this bit clears the INTFLG2 bit to 0. This bit becomes 0 after 2 clock cycles have elapsed since setting this bit to 1.
13	INTCLR1	Compare Match Interrupt 1 Clear Writing 1 to this bit clears the INTFLG1 bit to 0. This bit becomes 0 after 2 clock cycles have elapsed since setting this bit to 1.
12	INTCLR0	Compare Match Interrupt 0 Clear Writing 1 to this bit clears the INTFLG0 bit to 0. This bit becomes 0 after 2 clock cycles have elapsed since setting this bit to 1.
11	Reserved	This bit is read as 0. The write value should be 0.
10	INTFLG2	Compare Match Interrupt 2 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.
9	INTFLG1	Compare Match Interrupt 1 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.
8	INTFLG0	Compare Match Interrupt 0 Flag 0: Interrupt request has not been generated. 1: Interrupt request has been generated.
7 to 0	Reserved	The read value is undefined. The write value should be 0.

IRS Bit

Selects the compare match interrupt request signal to the interrupt controller.

When this bit is set to 0, the interrupt request signal becomes high while the angle value matches the value set in the RDC2nCMPj register (j = 0 to 2), and becomes low while these values do not match.

When this bit is set to 1, the interrupt request signal becomes high when the angle value matches the value set in the RDC2nCMPj register. Once the signal becomes high, it remains high even if these values do not match anymore. It becomes low when 1 is written to the INTCLRj bit while the angle value does not match the RDC2nCMPj register value.

As a result, the INTFLGj bit is ready to be notified as the interrupt request signal. To avoid this, clear the INTFLGj bit by INTCLRj bit after interrupt handling.

Note the followings when modifying the settings of this bit.

- Make sure INTFLGj = 0 before setting this bit to 1. If INTFLGj = 1, clear the flag while phi and the compare values of RDC2nCMP0, RDC2nCMP1, RDC2nCMP2) are not matched (INTCLRj = 1).

- Do not modify the settings during operation.

INTCLR0, INTCLR1, and INTCLR2 Bits

The INTCLR0, INTCLR1, and INTCLR2 bits clear INTFLG0, INTFLG1, and INTFLG2 bits, respectively. These bits also clear interrupt signal when IRS = 1.

While the angle value matches the value set in the RDC2nCMPj register, the INTFLG0, INTFLG1, and INTFLG2 bits do not become 0 even if 1 is written.

Write 1 again while the values do not match the RDC2nCMPj register value.

INTFLG0, INTFLG1, and INTFLG2 Bits

These bits become 1 when the angle value matches the value set in the RDC2nCMPj register and becomes 0 when 1 is written to the INTCLRj bit while the angle value does not match the RDC2nCMPj register value.

25.3.22 RDC2nENCP — RDC2n Encoder Pulse Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0046_H

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ENCPA	ENCPB	ENCPZ	—	—	—	—	HYSS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 25.42 RDC2nENCP Register Contents

Bit Position	Bit Name	Function
15 to 8	Reserved	These bits are read as 0. The write value should be 0.
7	ENCPA	Encoder A-phase Pulse The encoder A-phase pulse value is stored. 0: Low 1: High
6	ENCPB	Encoder B-phase Pulse The encoder B-phase pulse value is stored. 0: Low 1: High
5	ENCPZ	Encoder Z-phase Pulse The encoder Z-phase pulse value is stored. 0: Low 1: High
4 to 1	Reserved	These bits are read as 0. The write value should be 0.
0	HYSS	Hysteresis Select 0: With hysteresis 1: Without hysteresis

HYSS Bit

When this bit is set to 0, ±1 of hysteresis is added for generation of the encoder pulse signal and compare match signal. This hysteresis circuit supports only the setting of 12-bit resolution (MAXV[2:0] = 001_B). When another resolution is used, set the HYSS bit to 1 (without hysteresis).

25.3.23 RDC2nANSTP — RDC2n Analog Circuit Stop Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0048_H

Value After Reset: 8000_H

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

Table 25.43 RDC2nANSTP Register Contents

Bit Position	Bit Name	Function
15	ANSTP	R/D Converter Stop Register Stops analog circuits. 0: Analog circuits operate. 1: Analog circuits are stopped.
14 to 0	Reserved	These bits are read as 0. The write value should be 0.

ANSTP Bit

When this bit is set to 1, all the analog circuits are stopped and the analog pins (RDC2nSINMNT, RDC2nCOSMNT, RDC2nS1, RDC2nS2, RDC2nS3, RDC2nS4, RDC2nRSO, RDC2nCOM) become high impedance.

25.3.24 RDC2nETECNT — RDC2nET Event Generation Counter Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <RDC2n_base> + $004C_H$

Value After Reset: 0000_H

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

Table 25.44 RDC2nETECNT Register Contents

Bit Position	Bit Name	Function
15 to 0	CNT[15:0]	Event Generation Counter Value The event generation counter value is stored.

25.3.25 RDC2nETCON — RDC2nET Control Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + $004E_H$

Value After Reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ZCSTRG	—	—	—	—	—	—	—	—	—	CMPEN	IREN	DREN	ADTEN	ZCES	CNTEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.45 RDC2nETCON Register Contents

Bit Position	Bit Name	Function
15	ZCSTRG	Zero-cross Signal Software Trigger Writing 1 to this bit generates a zero-cross signal trigger for the excitation timer. This bit is always read as 0.
14 to 6	Reserved	These bits are read as 0. The write value should be 0.
5	CMPEN	Compare Match Function Enable 0: Disables the compare match function. 1: Enables the compare match function.
4	IREN	Excitation Timer Interrupt Request Enable 0: Disables the interrupt. 1: Enables the interrupt.
3	DREN	DMA Request Enable 0: Disables DMA request. 1: Enables DMA request.
2	ADTEN	A/D Conversion Start Trigger Enable 0: Disables the A/D conversion start trigger. 1: Enables the A/D conversion start trigger.
1	ZCES	Zero-cross Signal Edge Select Selects the edge to be detected. 0: Rising edge 1: Falling edge
0	CNTEN	Count Operation Enable Enables operation of the period measurement timer and event generation timer. 0: Period measurement timer and event generation timer are stopped.*1 1: Period measurement timer and event generation timer operate.

Note 1. The value of the reload register (RDC2nETRLD) is loaded to the event generation counter when it is stopped.

25.3.26 RDC2nETCMP — RDC2nET Compare Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0050_H

Value After Reset: $FFFF_H$

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

Table 25.46 RDC2nETCMP Register Contents

Bit Position	Bit Name	Function
15 to 0	CMP[15:0]	Period Measurement Timer Compare Value Set a value to be compared.

When a period measurement counter value matches this register value, the period measurement counter value is captured to the RDC2nETCAP register and the counter value becomes to 0000_H.

An excitation timer interrupt request is generated in the next cycle after the match between the values of the period measurement counter and the RDC2nETCMP register.

25.3.27 RDC2nETCAP — RDC2nET Capture Register (n = 0, 1)

Access: Readable in 16-bit units.

Address: <RDC2n_base> + 0052_H

Value After Reset: 0000_H

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

Table 25.47 RDC2nETCAP Register Contents

Bit Position	Bit Name	Function
15 to 0	CAP[15:0]	Period Measurement Timer Capture Data The period measurement timer value is stored when a zero-cross signal is detected or when a compare match occurs.

When a zero-cross signal is detected, the period measurement counter value is stored in this register.

In addition, when the period measurement counter value matches the RDC2nETCMP register value with the CMPEN bit in the RDC2nETCON register set to 1, the period measurement counter value is stored in the RDC2nETCAP register.

25.3.28 RDC2nETRLD — RDC2nET Reload Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0054_H

Value After Reset: 0000_H

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

Table 25.48 RDC2nETRLD Register Contents

Bit Position	Bit Name	Function
15 to 0	RLD[15:0]	Event Generation Counter Reload Value Set the value to be reloaded to the event generation counter when the zero-cross signal edge is detected. Set the value of the counter to be 0002_H or more. (Setting 0000_H and 0001_H are prohibited.)

25.3.29 RDC2nETPMCNT — RDC2nET Period Measurement Counter Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0056_H

Value After Reset: 0000_H

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

Table 25.49 RDC2nETPMCNT Register Contents

Bit Position	Bit Name	Function
15 to 0	CNT[15:0]	Period Measurement Counter Value The value of the period measurement counter is stored.

When the counter reaches $FFFF_H$, the counter stops operation.

25.3.30 RDC2nEXSQR — RDC2n Excitation Amplitude Integral Square-Sum Monitor Register (n = 0, 1)

Access: Readable/writable in 16-bit units.

Address: <RDC2n_base> + 0058_H

Value After Reset: $XXXX_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ESQULL	ESQOUL	RLT[13:0]													
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 25.50 RDC2nEXSQR Register Contents

Bit Position	Bit Name	Function
15	ESQULL	Excitation Amplitude Integral Square-sum Value Fall Below Lower Limit Determine Excitation amplitude integral square-sum value fall below lower limit determine signal is stored. 0: Excitation amplitude integral square-sum value is equal to or larger than the lower limit. 1: Excitation amplitude integral square-sum value is smaller than the lower limit.
14	ESQOUL	Excitation Amplitude Integral Square-sum Value Exceed Upper Limit Determine Excitation amplitude integral square-sum value exceed upper limit determine signal is stored. 0: Excitation amplitude integral square-sum value is equal to or smaller than the upper limit. 1: Excitation amplitude integral square-sum value is larger than the upper limit.
13 to 0	RLT[13:0]	Excitation Amplitude Integral Square-sum Value Upper 14 bits of the 18-bit excitation amplitude integral square-sum value is stored. The value is updated every 1 ms.

This register monitors excitation amplitude integral square-sum values of the excitation amplitude automatic adjustment circuit. The excitation amplitude integral square-sum value (formula below) is obtained by squaring each input signal in the SINE and COSINE sides, adding them, and integrating the result for a certain period.

$$\int_0^t \{(f(t)\sin\theta)^2 + (f(t)\cos\theta)^2\}\,dt = \int_0^t (f(t))^2\{\sin^2\theta + \cos^2\theta\}\,dt = \int_0^t (f(t))^2\,dt$$

25.4 Functional Description

25.4.1 Tracking Loop

25.4.1.1 PI Compensator Bandwidth Setting Function

This RDC can be set from six different bandwidths (five fixed and one auto-adjusted) using the registers. In addition, coefficients in the PI compensator can be configured in detail by setting the BWCS bit in the RDC2nCON register to 0. For further details, refer to the sections on registers RDC2nCONSEL and RDC2nCGSEL.

25.4.1.2 Forced Gain Control Function

The forced gain control function is designed to improve the tracking performance when the resolver angle deviates significantly from the R/D converted angle in situations such as after a reset.

The forced gain control function is executed when any of the following conditions is met:

[Conditions for the execution of forced gain control function]

1. After a reset
2. At the recovery from a resolver signal error
3. At the recovery from a resolver signal disconnect error
4. When BIST is started, or stopped
5. When 1 is written to the KIRST bit in the RDC2nRST register

When the forced gain control function is executed, the Kv gain transitions to the maximum value, and the coefficients in the PI compensator and the settings of the excitation extraction circuit are forcibly set to the values listed in **Table 25.51**.

The forced gain control function is executed for approximately 5 ms. If one of the conditions is met again during the execution, the Kv gain becomes the maximum value and the process is started over from that point. The execution period is further extended by approximately 5 ms.

When the FGCON bit in the RDC2nCON register is set to 1 (does not to use the forced gain control function), the forced gain control function will not be executed even if the forced gain control function execution condition is met, with the exception of after a reset.

Since the Kv gain becomes the maximum value during the execution, the R/D converted angle fluctuates significantly for approximately 1 ms even after the resolver angle and the R/D converted angle have matched.

25.4.1.3 Excitation Signal Source Selection Function

Without the excitation signal output from the RDC2nRSO pin and common voltage output from the RDC2nCOM pin, the R/D conversion can be performed using externally generated excitation signal input to pins RDC2nRSO and RDC2nCOM. When an externally input signal is used, set the EXIO bit in the RDC2nCON register to 0.

25.4.1.4 Required Sensor Selection Function

The DC resolver signal (E·sinθ, E·cosθ) which does not contain excitation component can also be used by setting the SENS bit in the RDC2nCON register to 0, instead of the resolver signal (f(t) • sinθ, f(t) • cosθ) which contains excitation components. When the DC resolver signal is used, the excitation component extraction function is disabled.

25.4.1.5 Excitation Component Extraction Function

The excitation signals (RDC2nRSO, RDC2nCOM) and resolver signals (RDC2nS1 to RDC2nS4) are analog signals input to the RDC. If a phase difference between the excitation component (sine wave component) in the excitation signal line and that in the resolver signal line can cause an error in the angle conversion result in proportion to the phase difference. The phase difference between the resolver signal and excitation signal can be reduced by using the excitation component contained in the resolver signal line for an angle conversion.

When using an external excitation signal by setting the EXIO bit in the RDC2nCON register to 0, extracted excitation components cannot be used if the difference between the electrical angle of the resolver and the RDC converted angle is large (such as during power-up or occurrence of error). Therefore, when using an external excitation signal, be sure to enter the external excitation signal into the RDC2nRSO and RDC2nCOM pins.

The excitation component extraction function automatically performs the following process; the excitation signal (RDC2nRSO, RDC2nCOM) is used when the difference between the resolver electrical angle and the R/D converted angle is large, and the extracted excitation component is used when the difference is small. The exact adjustment of the phase difference between excitation and resolver signals is not required because of this function.

25.4.1.6 Maximum Angular Velocity Setting Function

This function can set the maximum angular velocity (resolution) that is capable of tracking operation by using the MAXV [2:0] bits in the RDC2nMAXV register. Discrete values do not occur in the selected resolution.

25.4.1.7 Compare Match Interrupt

When the angle set in the RDC2nCMPj register (j = 0 to 2) and the R/D converted angle match, a compare match interrupt request signal is generated. The bit width that is compared for a match is set using the MAXV [2:0] bits (maximum angular velocity selection) in the RDC2nMAXV register.

The compare match interrupt request signal can be selected from either a compare match signal or a signal latching a compare match signal by using the IRS bit in the RDC2nCMINT register. When the IRS bit is set to 0, the compare match interrupt request signal becomes high when the R/D converted angle and the angle set in the RDC2nCMPj register match, and becomes low when the values do not match.

When the IRS bit is set to 1, the compare match interrupt request signal becomes high when the R/D converted angle and the angle set in the RDC2nCMPj register match, and also the INTFLGj flag in the RDC2nCMINT register becomes 1. In this case, the request signal retains the high level even when the values do not match. When 1 is written to the INTCLRj bit in the RDC2nCMINT register while the R/D converted angle and the angle set in the RDC2nCMPj register do not match, the request signal becomes low and the INTFLGj flag becomes 0. If 1 is written to the INTCj bit while the R/D converted angle and the angle set in the RDC2nCMPj register match, the request signal remains high and the INTFLGj flag does not become 0.

Turning hysteresis on by setting the HYSS bit in the RDC2nENCP register to 0 prevents chattering of the output of the compare match interrupt signal and Z output signal when the angle output near the target bit for comparison is not stable.

This hysteresis circuit can only be used when 12-bit resolution is selected (MAXV[2:0] = 001_B).

Figure 25.3 and **Figure 25.4** show the timing charts for the compare match interrupt request signal when hysteresis is off and on, respectively.

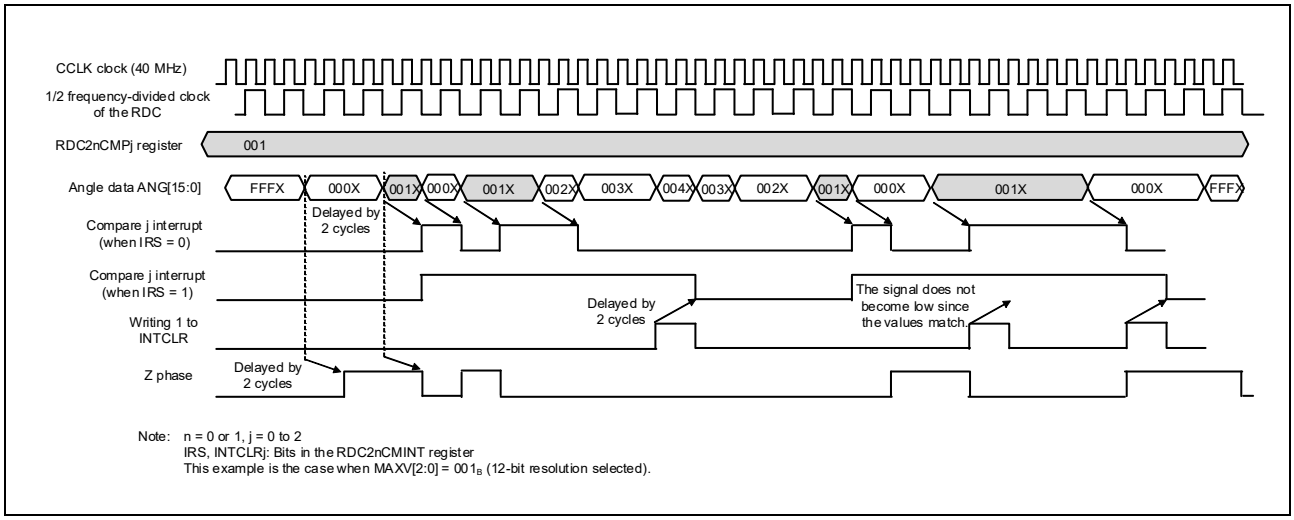


Figure 25.3 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is Off

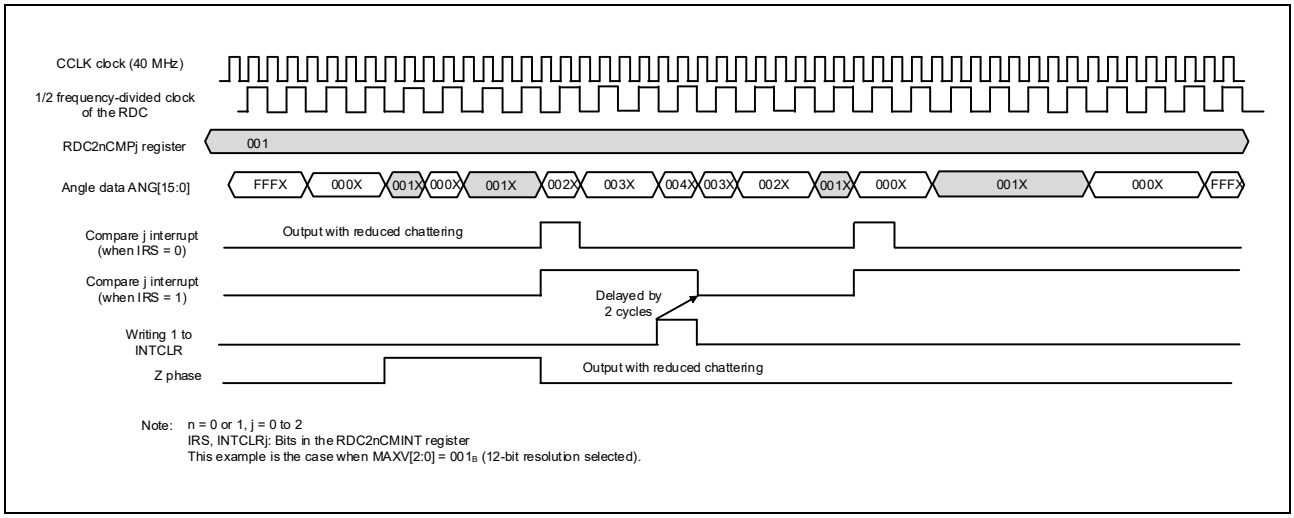


Figure 25.4 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is On

25.4.1.8 Encoder Pulse Output Function

This function enables to output encoder pulse signals (A, B, Z phases).

The Z-phase interrupt signal becomes high while the R/D converted angle is 0º. The bit width compared for a match is set using the MAXV [2:0] bits in the RDC2nMAXV register as it is for the compare match interrupt request signal. The encoder pulse signal is output when the corresponding bit in the RDC2nOUTC register is set to 1 (enables output).

Whether the encoder pulse signals are to be output through the hysteresis circuit or without going through it can be selected using the HYSS bit in the RDC2nENCP register. This hysteresis circuit can only be used when 12-bit resolution is selected (MAXV[2:0] = 001_B).

Figure 25.5 and **Figure 25.6** show the waveforms of encoder phase pulse operation when hysteresis is on and off, respectively.

Figure 25.3 and **Figure 25.4** for the z output waveform when the angle output near the target bit for comparison is not stable.

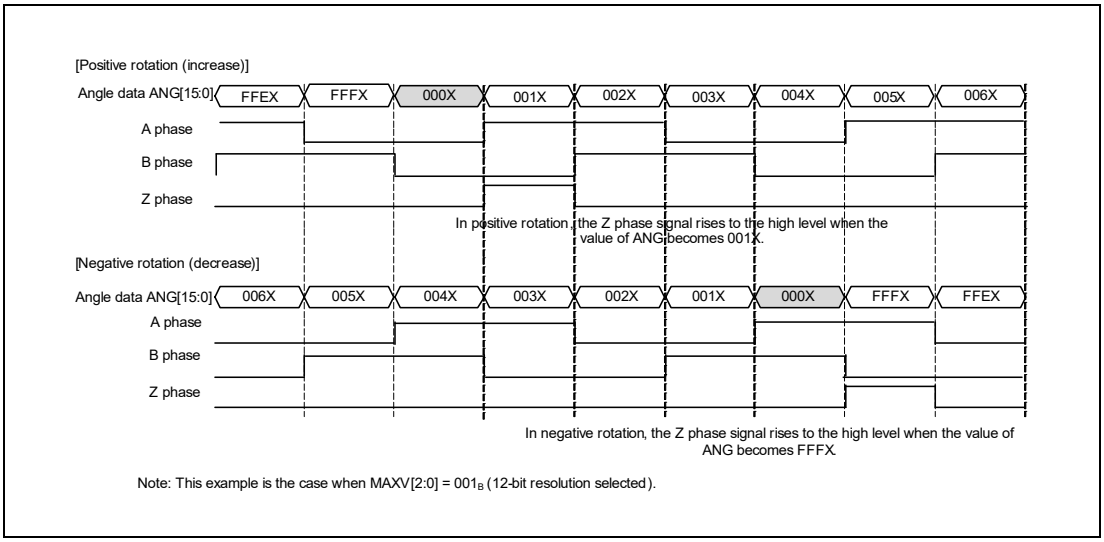


Figure 25.5 Waveform of Encoder Phase Pulse Operation when Hysteresis is On

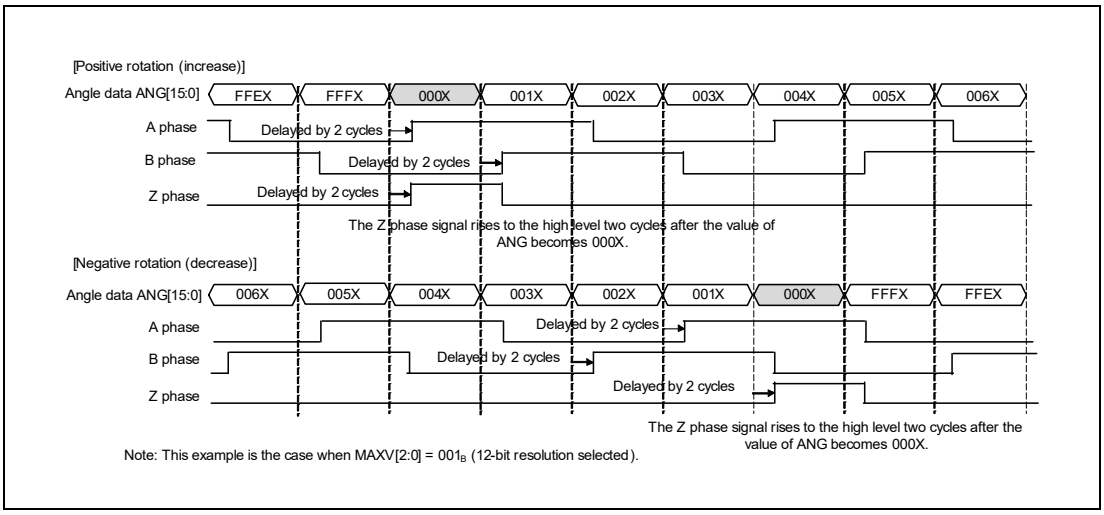


Figure 25.6 Waveform of Encoder Phase Pulse Operation when Hysteresis is Off

25.4.1.9 Monitor Function

This function enables to read angular velocity information and the control deviation value directly from the register. When reading angular velocity information, set the DATSEL [3:0] bits in the RDC2nCONSEL register to 1001_B and read bits b15 to b0 in the RDC2nDATSTR register. Negative numbers are represented in 2's complement.

Table 25.51 lists the relationship between the RDC2nDATSTR register and angular velocity information. If, for example, an angular velocity value is read as $CFE0_H$, its binary representation is as listed in the table. By calculating from this value represented in 2's complement, the original value, which has 1's in b13, b12, and b5 is obtained. The sum of the angular velocities of the applicable bits is equal to -440.597 rpm.

Table 25.51 Relationship between RDC2nDATSTR Register and Angular Velocity

Bit	Angular Velocity	Read Result (Example)	
		H'CFE0 (2's Complement)	Original Value
15	0: Positive, 1: Negative	1	Negative
14	585,938 rpm	1	0
13	292,969 rpm	0	1
12	146,484 rpm	0	1
11	73,242 rpm	1	0
10	36,621 rpm	1	0
9	18,311 rpm	1	0
8	9,155 rpm	1	0
7	4,578 rpm	1	0
6	2,289 rpm	1	0
5	1,144 rpm	1	1
4	572 rpm	0	0
3	286 rpm	0	0
2	143 rpm	0	0
1	72 rpm	0	0
0	36 rpm	0	0

When reading the control deviation, set the DATSEL [3:0] bits in the RDC2nCONSEL register to 0101_B and read bits b7 to b0 in the RDC2nDATSTR register. Negative numbers are represented in 2's complement.

Table 25.52 lists the relationship between the RDC2nDATSTR register and the control deviation (%). If, for example, a control deviation value is read as 93_H, its binary representation is as listed in the table. By calculating from this value represented in 2's complement, the original value, which has 1's in b6, b5, b3, b2, and b0 is obtained. The sum of the deviations of the applicable bits is equal to -85.16%.

Table 25.52 Relationship between RDC2nDATSTR Register and Control Deviation (%)

Bit	Control Deviation (%)	Read Result (Example)	
		H'93 (2's Complement)	Original Value
7	Sign (0: positive, 1: negative)	1	Negative
6	50%	0	1
5	25%	0	1
4	12.5%	1	0
3	6.25%	0	1
2	3.13%	0	1
1	1.56%	1	0
0	0.78%	1	1

25.4.2 Excitation Signal Output

25.4.2.1 Excitation Signal Output (RDC2nRSO, RDC2nCOM) Function

Sine wave voltage signal generated by 7-bit D/A can be output from the RDC2nRSO pin. RVDD/2 (2.5V) common voltage can be output from the RDC2nCOM pin. Setting the EXIO bit in the RDiCON register to 0 (excitation signal input) disables the output and the RDC2nRSO and RDC2nCOM pins become input. The amplitude of the sine wave signal that is output from the RDC2nRSO pin is set in the EXOC[1:0] bits in the RDC2nEXAAT register. The amplitude of the standard value is 0.4 × RVDD [Vp-p].

25.4.2.2 Amplitude Automatic Adjustment Function

In order to obtain an appropriate R/D conversion accuracy, the resolver signal input amplitude (monitor signal amplitude) needs to be controlled within a range of 0.4 × RVDD to 0.6 × RVDD [Vp-p]. The amplitude automatic adjustment function automatically adjusts the excitation signal output amplitude and the input gain resistance so that the monitor signal amplitude fits within the approximate range of 0.4 × RVDD to 0.6 × RVDD [Vp-p].

The automatic adjustment is performed on the excitation signal output amplitude and the input gain resistance. Priority can be specified by setting the EAAOD bit in the RDC2nEXAAT register. When the EAATSP bit in the RDC2nEXAAT register is set to 0, the automatic adjustment can be performed continuously. By setting the EAATSP bit to 1, the automatic adjustment can be stopped and the adjusted value at that time is retained. Setting the EAATSP bit to 0 restarts the automatic adjustment process.

The function monitors the monitor amplitude with a simplified ADC (4-bit resolution), determines the size of amplitude based on the integral square-sum of the excitation amplitude for a period of 1 ms, and adjusts the amplitude. The determination threshold is approximately 3 Vp-p on the high side, and approximately 2 Vp-p on the low side.

It is also possible to use the excitation signal output settings and the input gain resistance settings that are specified in the register, without using the results of automatic adjustments.

25.4.3 Error Detection

25.4.3.1 Error Detection Function

This function monitors and detects errors in resolver signal and in R/D conversion operations. If any of the errors listed in **Table 25.53** is detected, an RDC error interrupt request is generated, and the corresponding bit in the RDC2nERDET register becomes 1.

The following table lists factors that lead to error detection.

Table 25.53 Detected Errors

Item	Detected Factor
Resolver signal error	• Excitation signal line disconnection (RDC2nRSO, RDC2nCOM) (including contact failure) • Excitation signal down (excitation signal output circuit down, short circuit between lines) • Short circuit between signal lines (RDC2nS1 and RDC2nS3, RDC2nS2 and RDC2nS4) • Resolver coil layer short
Resolver signal disconnect error	Resolver signal disconnection (RDC2nS1 to RDC2nS4) (including contact failure)
R/D conversion error	Excessive control deviation (ε) of tracking control loop (negative feedback control system)

25.4.3.2 Resolver Signal Error Detection Function

This function detects disturbance in the resolver signal balance caused by an error in excitation signals input to the resolver. When a resolver signal error is detected, an RDC error interrupt request signal becomes high. A resolver signal error is detected if the monitor outputs (RDC2nSINMNT, RDC2nCOSMNT) fall below the threshold for approximately 220 us.

25.4.3.3 Resolver Signal Disconnect Error Detection Function

This function detects disconnection (including contact failure) of the resolver signals (RDC2nS1 to RDC2nS4). When a resolver signal disconnect error is detected, the RDC error interrupt request signal becomes high. A resolver signal disconnect error is detected if the DC level fluctuations in the monitor output (RDC2nSINMNT, RDC2nCOSMNT) rise above the threshold. For a description of the relationship between register values and threshold values, see **Section 35.5.3, Error Detect Characteristics**.

25.4.3.4 R/D Conversion Error Detection Function

This function monitors the control deviation in R/D conversion loop, and detects operation errors in the R/D conversion function. When an R/D conversion error is detected, the RDC error interrupt request signal becomes high.

A control deviation (ε) is considered excessive if the control deviation rises above or falls below a configured threshold level. For a description of the relationship between register values and threshold values, see **Section 35.5.3, Error Detect Characteristics**. An R/D conversion error is detected if the control deviation stay excessive for more than 50% of the error determination time set in the EDPS[1:0] bits in the RDC2nCONSEL register.

The R/D conversion error detection circuit includes a circuit supporting high-speed rotation of a resolver and a circuit not supporting it. These circuits can be selected by using the CVEDS bit in the RDC2nCON register.

25.4.4 Self-Diagnosis

25.4.4.1 Built-in Self-Test Function

In order to check the validity of a given operation, the Built-in Self-Test (BIST) function generates an intended signal input that is simulated internally by setting a BIST instruction in the RDC2nBISTC register, and monitors the signal that is output in response to the simulated signal input. **Table 25.54** lists test items.

Each output during the execution of BIST operates in response to the simulated signal.

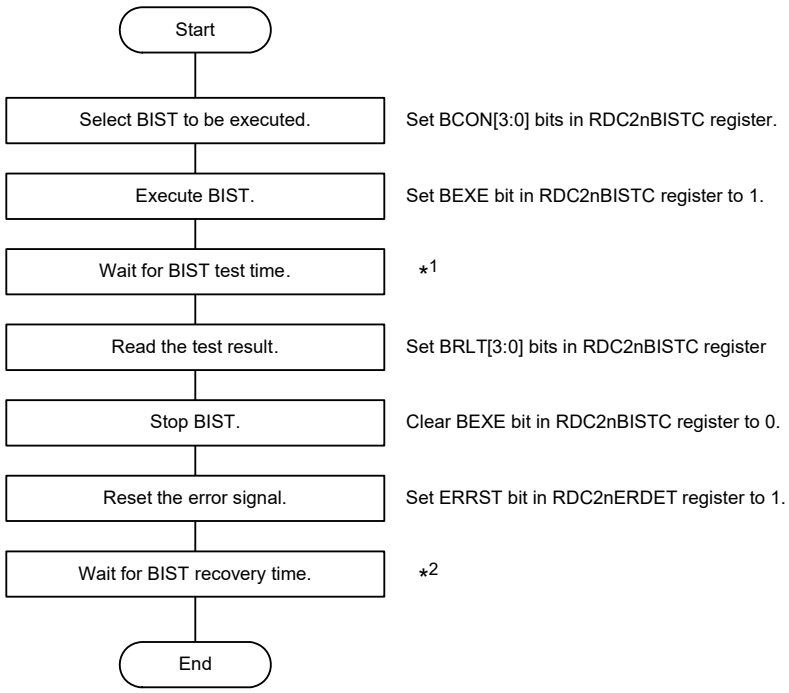
Perform the following settings to execute BIST.

(1) Enable the forced gain control function during BIST operation. (Set the FGCON bit in the RDC2nCON register to 0.)

(2) When the BIST is completed, set the ERRST bit in the RDC2nERDET register to 1 to reset the error signal.

Table 25.54 BIST Instructions

BIST	Test Items
Angle conversion BIST	Self-test for the R/D conversion function The following electrical angles can be set as a resolver signal input: • Target angle 0° • Target angle 45° • Target angle 270°
Error detection BIST	Self-test for the error detection function • Resolver signal error detection BIST • Resolver signal disconnect error detection BIST • R/D conversion error detection BIST



Note: n = 0, 1

Note 1. For BIST test time, see **Section 35.5.1, RDC Conversion Performance**.

Note 2. For BIST recovery time, see **Section 35.5.1, RDC Conversion Performance**.

Figure 25.7 Flowchart of Built-In Self-Test (BIST) Process

25.4.5 Excitation Timer (ET) Function

The excitation timer comprises two 16-bit timers: a period measurement timer and an event generation timer. The operating clock of the timers is PCLK (40 MHz). **Figure 25.8** shows a block diagram of the excitation timer.

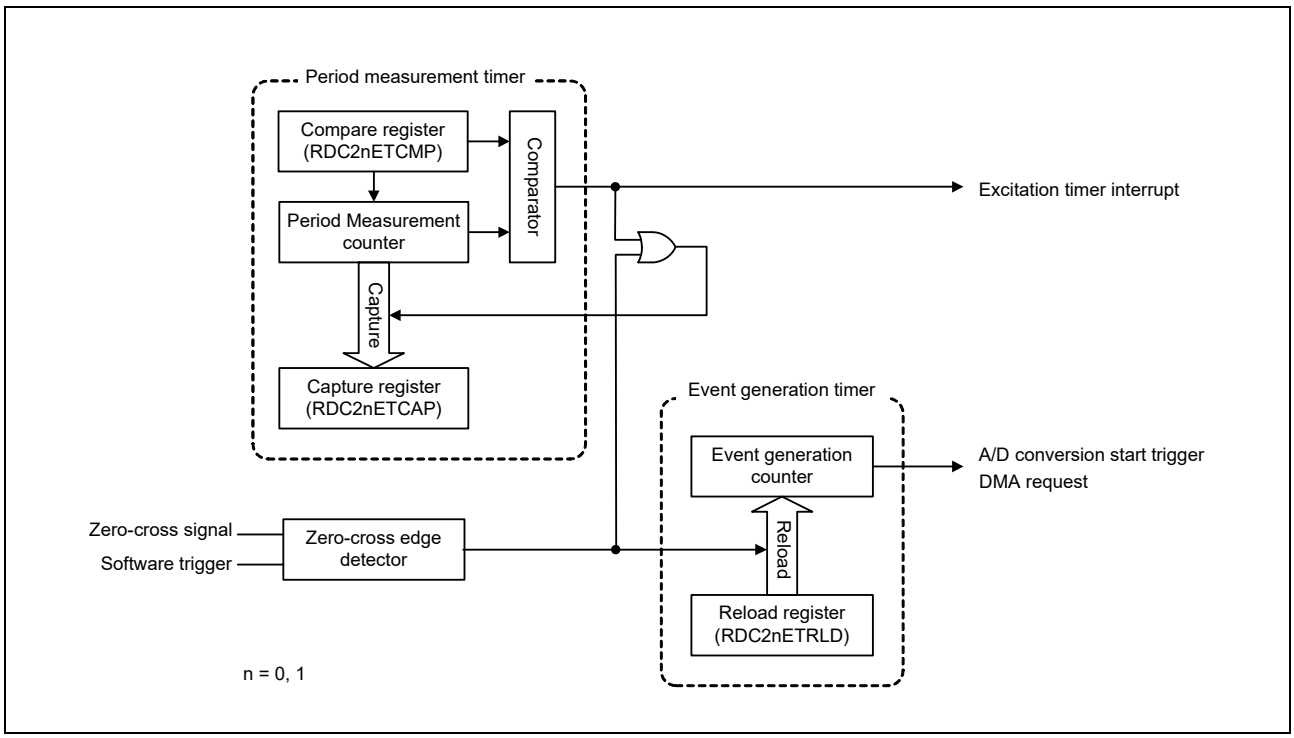


Figure 25.8 Block Diagram of Excitation Timer

25.4.5.1 Period Measurement Timer

The period measurement timer measures the cycle of excitation signal (zero-cross signal). When an edge (selectable from rise edge and fall edge) of the zero-cross signal is detected, the value of the period measurement counter is captured and stored in the RDC2nETCAP register. By reading the RDC2nETCAP register, the cycle of excitation signal can be obtained.

The cycle of excitation signal can be calculated from the following formula: (RDC2nETCAP register value + 1) × PCLK cycle (25 ns).

When the IREN bit in the RDC2nETCON register is set to 1 (enabled the interrupt), an excitation timer interrupt request is generated if the value set in the RDC2nETCMP register matches the period measurement counter value. The excitation signal cycle error can be detected by setting a value of longer duration than the excitation signal cycle to the RDC2nETCMP register.

When a zero-cross signal trigger is generated by writing 1 to the ZCSTRG bit in the RDC2nETCON register, the period measurement timer operates in the same way as the zero-cross signal edge detection.

Figure 25.9 shows an example of period measurement timer operation.

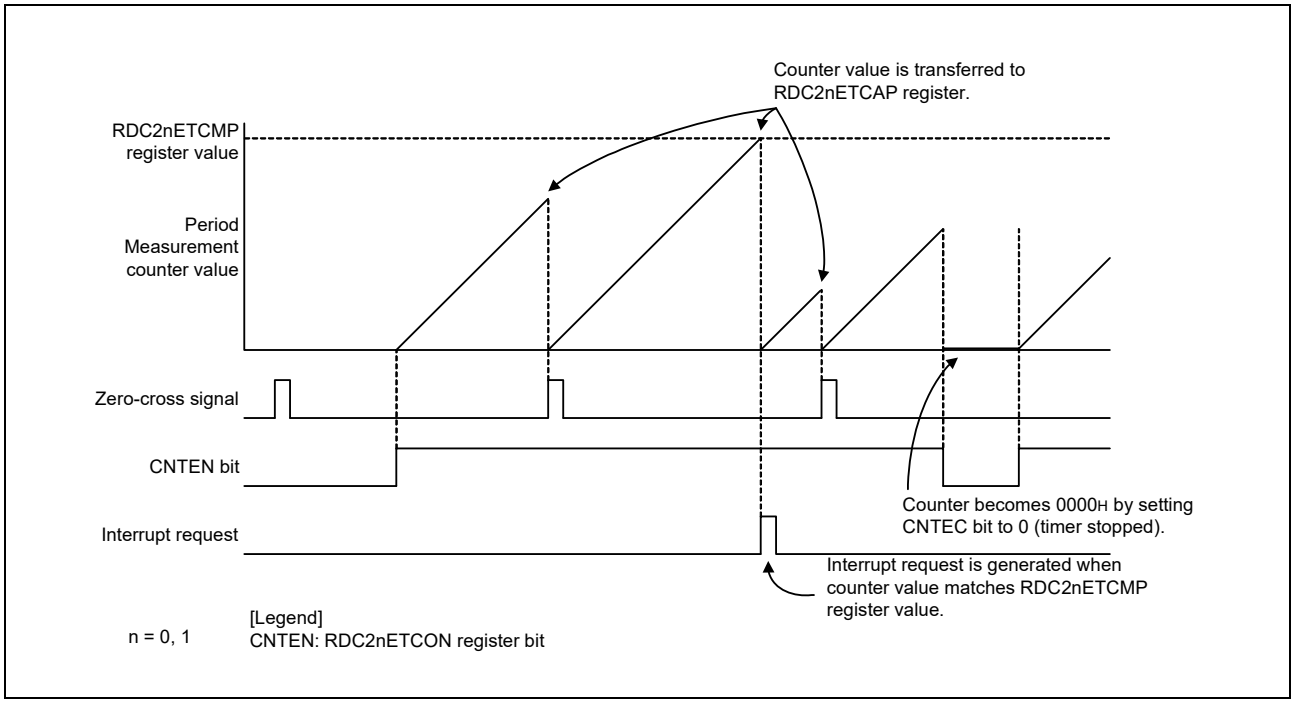


Figure 25.9 Example of Period Measurement Timer Operation

25.4.5.2 Event Generation Timer

The event generation timer can generate a trigger signal (A/D conversion trigger, DMA request) after the time set in the RDC2nETRLD register has elapsed since the occurrence of an edge of the zero-cross signal. When a zero-cross signal trigger is generated by writing 1 to the ZCSTRG bit in the RDC2nETCON register, the event generation timer operates in the same way as the zero-cross signal edge detection.

Figure 25.10 shows an example of event generation timer operation.

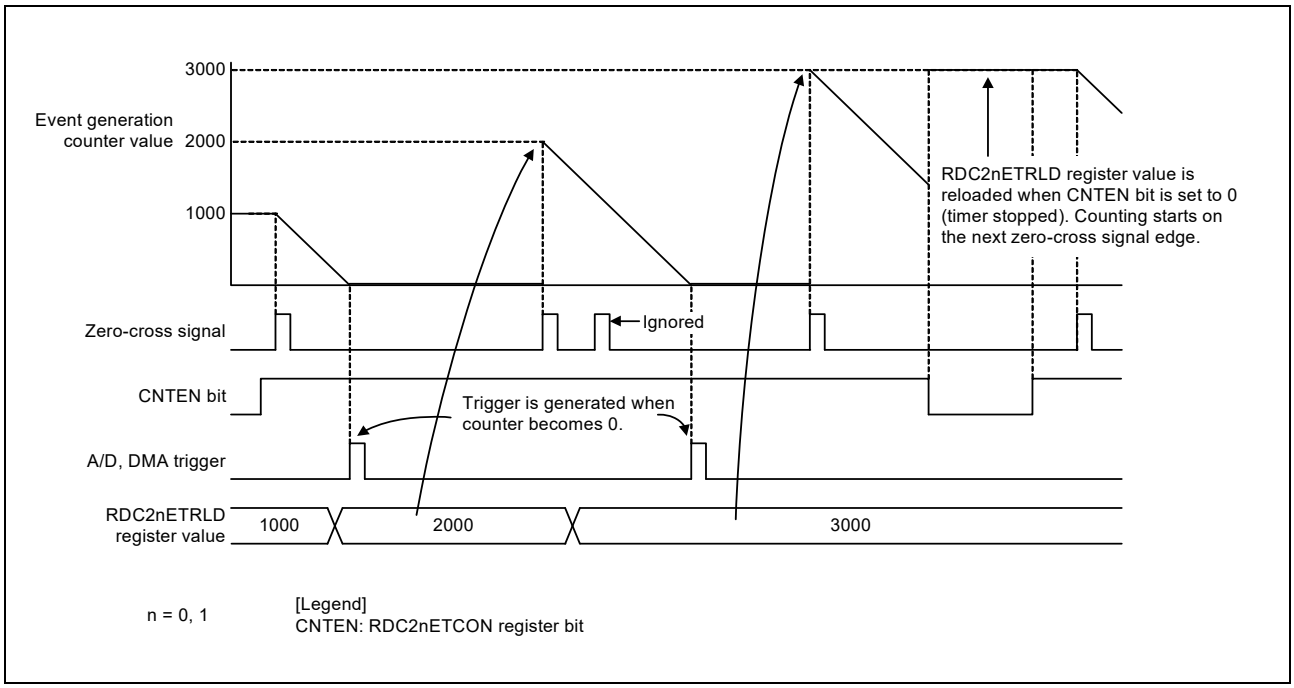


Figure 25.10 Example of Event Generation Timer Operation

25.4.5.3 Excitation Zero-Cross Signal

The zero-cross signal indicates zero-cross timing for differential excitation signal input. The zero-cross signal is input to the excitation timer circuit to be used for detecting the excitation vertex.

Figure 25.11 shows the relationship between the differential excitation signal input waveforms and zero-cross signal.

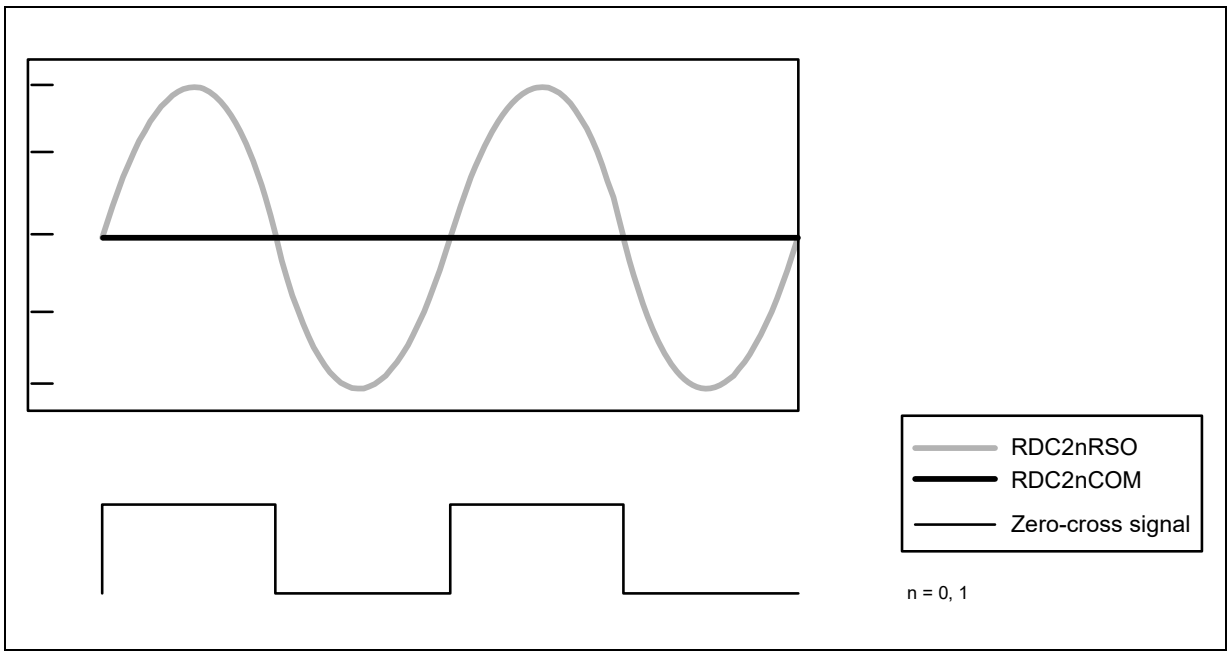
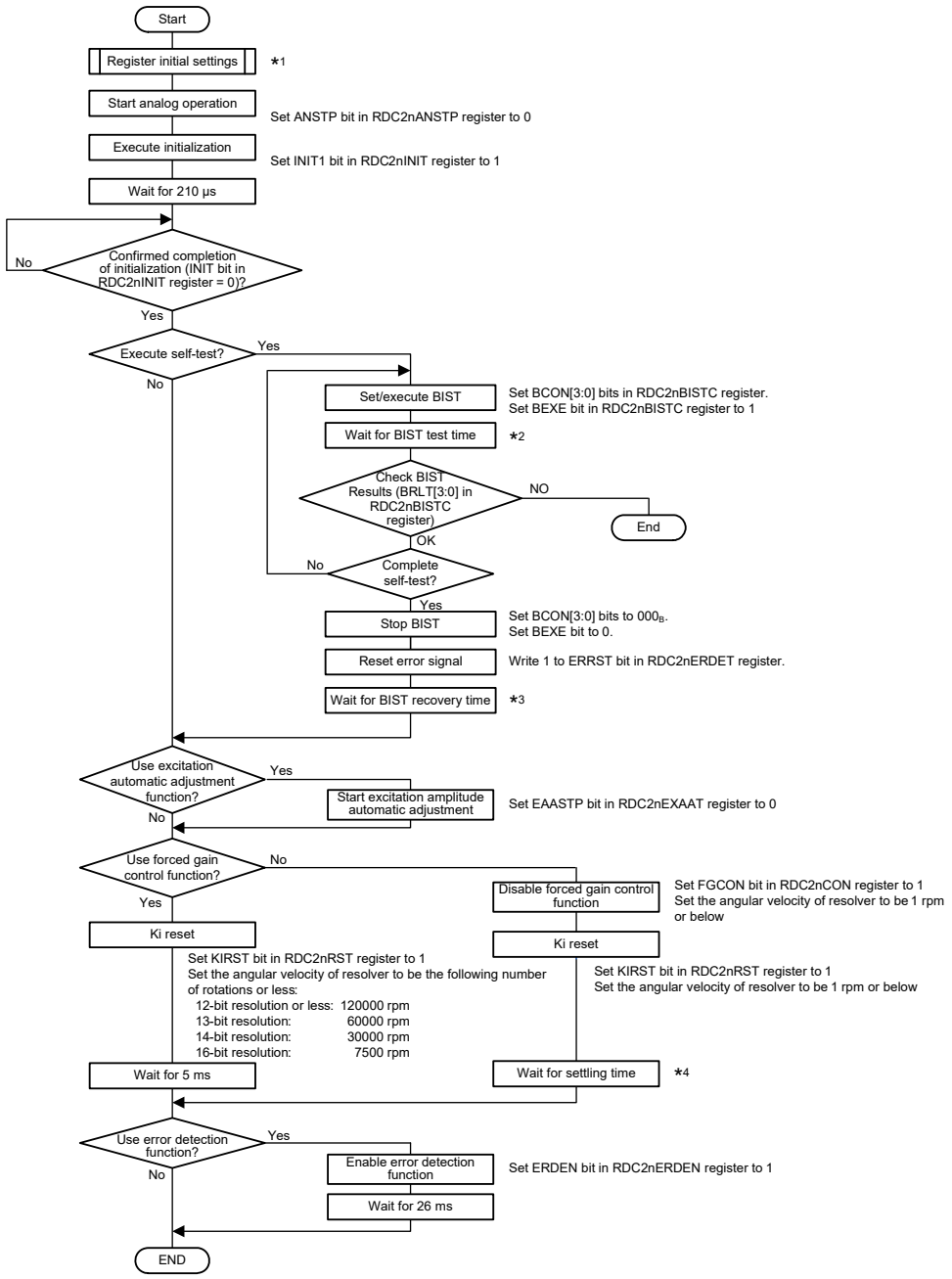


Figure 25.11 Relationship between Differential Excitation Signal Input (RDC2nRSO, RDC2nCOM) and Zero-Cross Signal

25.5 Initial Operation Procedure

Figure 25.12 shows the RDC2 initial operation flow and **Figure 25.13** shows the register initial setting flow.



Note: n = 0, 1

Note 1. See **Figure 25.13, Register Initial Setting Flow**.

Note 2. For BIST test time, see **Section 35.5.1, RDC Conversion Performance**.

Note 3. For BIST recovery time, see **Section 35.5.1, RDC Conversion Performance**.

Note 4. For the settling time, see **Section 35.5.1, RDC Conversion Performance**.

Figure 25.12 RDC2 Initial Operation Flow

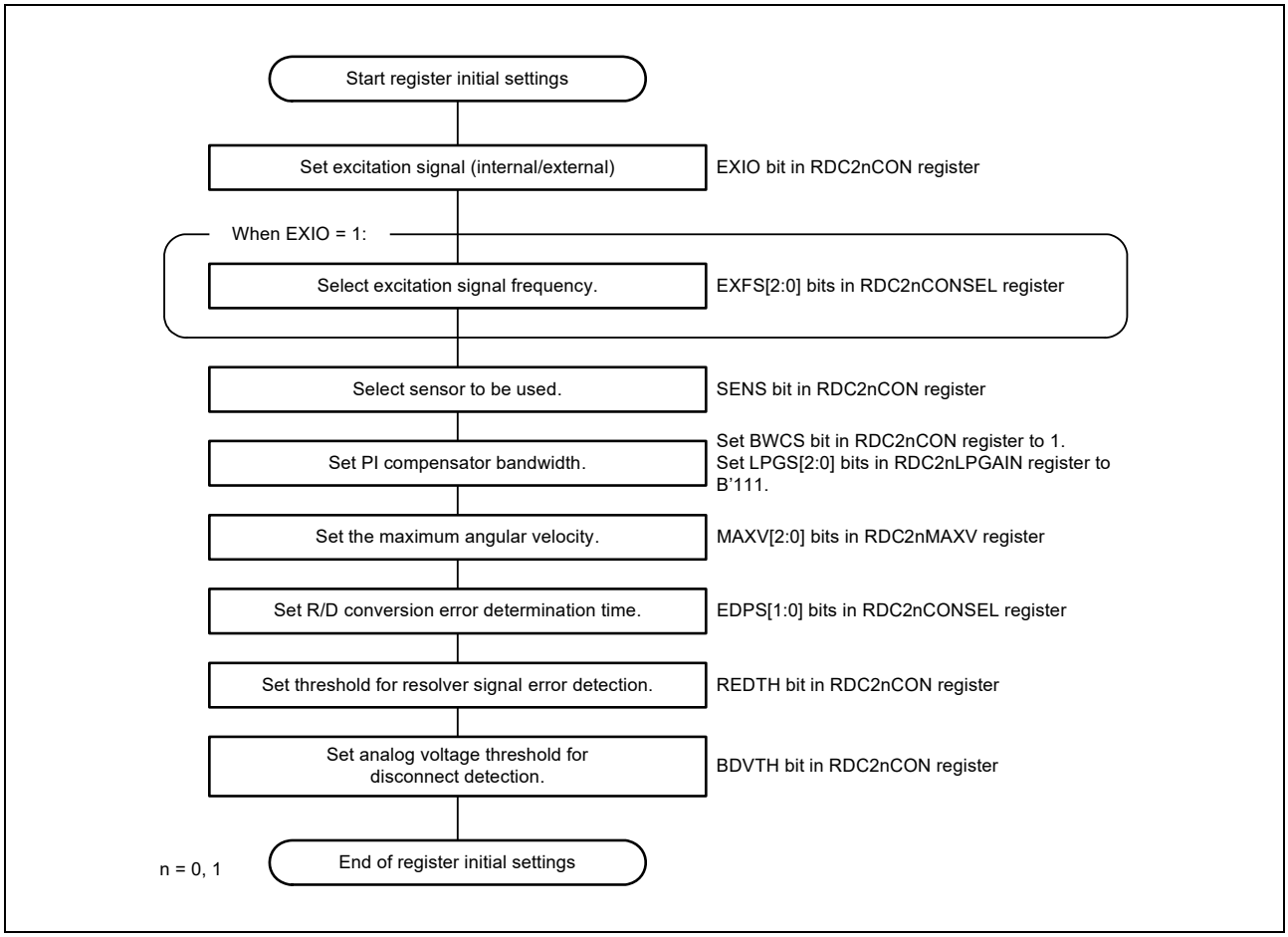


Figure 25.13 Register Initial Setting Flow

25.6 Resolver Interface Circuit

The following shows specific interface circuits as reference examples. When determining constants such as a resistance value and adding functions such as an input/output protection circuit, a careful decision must be made for each system and conduct adequate evaluation.

25.6.1 Resolver Signal Input (Differential Input) Circuit

Figure 25.14 shows a VR resolver signal input circuit and an equivalent circuit of monitor output.

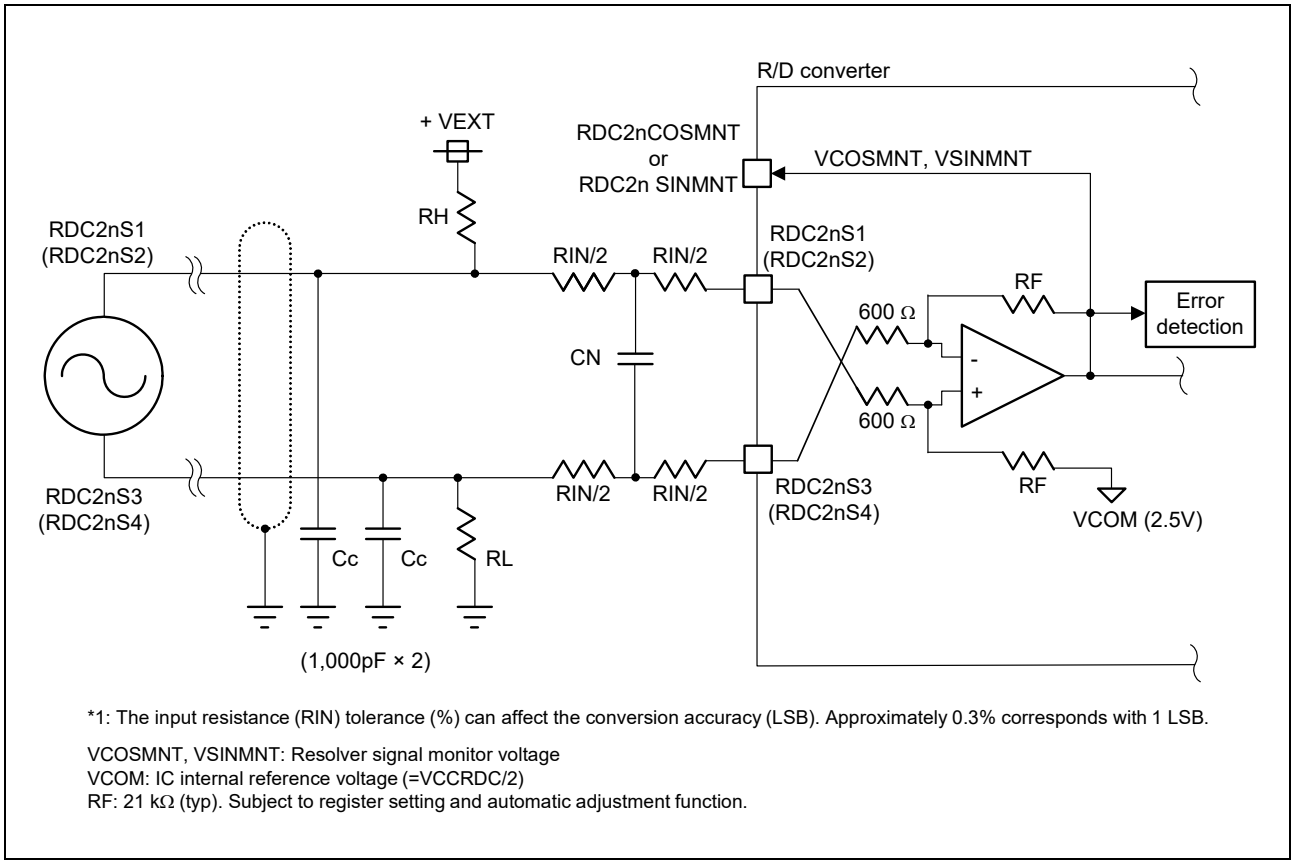


Figure 25.14 VR Resolver Signal Input Circuit and Equivalent Circuit of Monitor Output

- RIN: Resolver signal level

VCOSMNT or VSINMNT = (VIN) × (RF/ (RIN + 600 Ω) ≈ 2 to 3 [Vp-p]

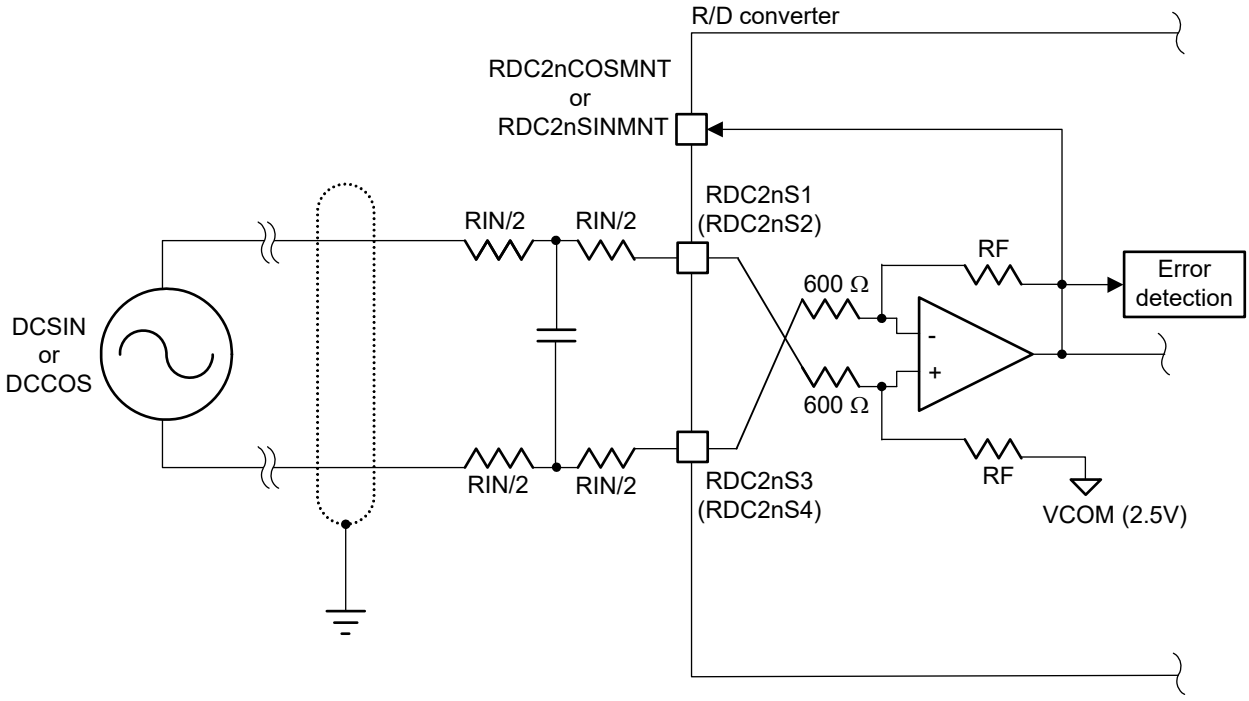
(where VIN denotes the signal output voltage between resolver pins [Vp-p], RIN ≥ 2 [kΩ])

- RH and RL: Determine a resistance value in an 80% to 100% range for the following calculated values:

(1) RH ≈ {(RVDD – VCOM) / (22.0×10^{-6})} –RIN, where VCOM = RVDD/2[V]

(2) RL ≈ {(VCOM) / (22.0×10^{-6})} –RIN, where VCOM = RVDD/2[V]

Figure 25.15 shows an equivalent circuit when DC resolver signal input is used.



Note 1: The input resistance (RIN) tolerance (%) can affect the conversion accuracy (LSB). Approximately 0.3% corresponds with 1 LSB. Differences in input signal conditions and peripheral circuits can affect the R/D conversion accuracy and the error detection function.

Figure 25.15 Equivalent Circuit when DC Resolver Signal Input is Used

25.6.2 Excitation Voltage Booster Amplifier Circuit

25.6.2.1 Excitation Voltage Booster Amplifier Circuit (Single Power Supply)

Figure 25.16 shows an equivalent circuit of the excitation voltage booster amplifier circuit with single power supply.

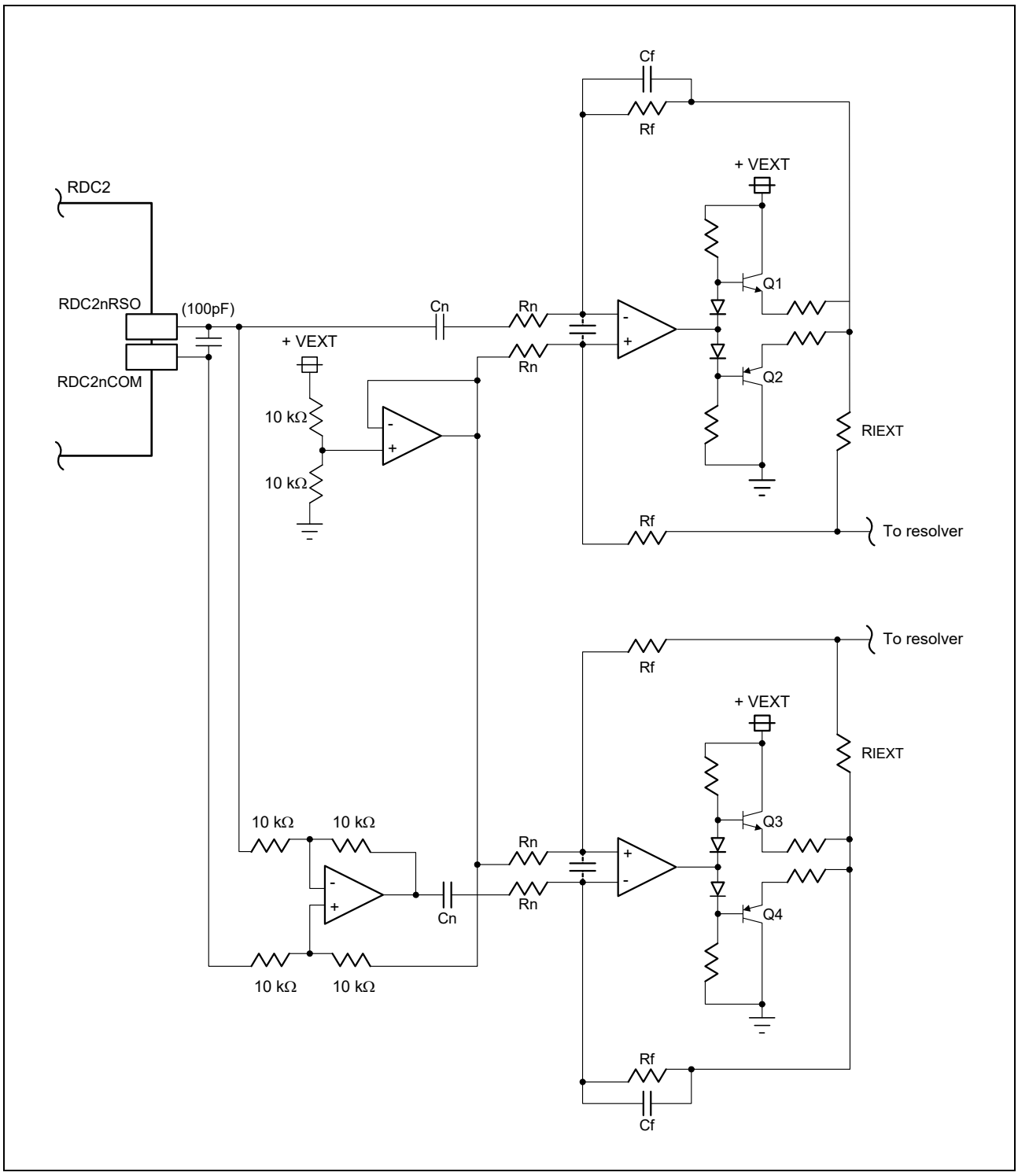


Figure 25.16 Equivalent Circuit Example of Excitation Voltage Booster Amplifier Circuit (Single Power Supply)

25.6.2.2 Excitation Voltage Booster Amplifier Circuit (Dual Power Supply)

Figure 25.17 shows an equivalent circuit of the excitation voltage booster amplifier circuit with dual power supply.

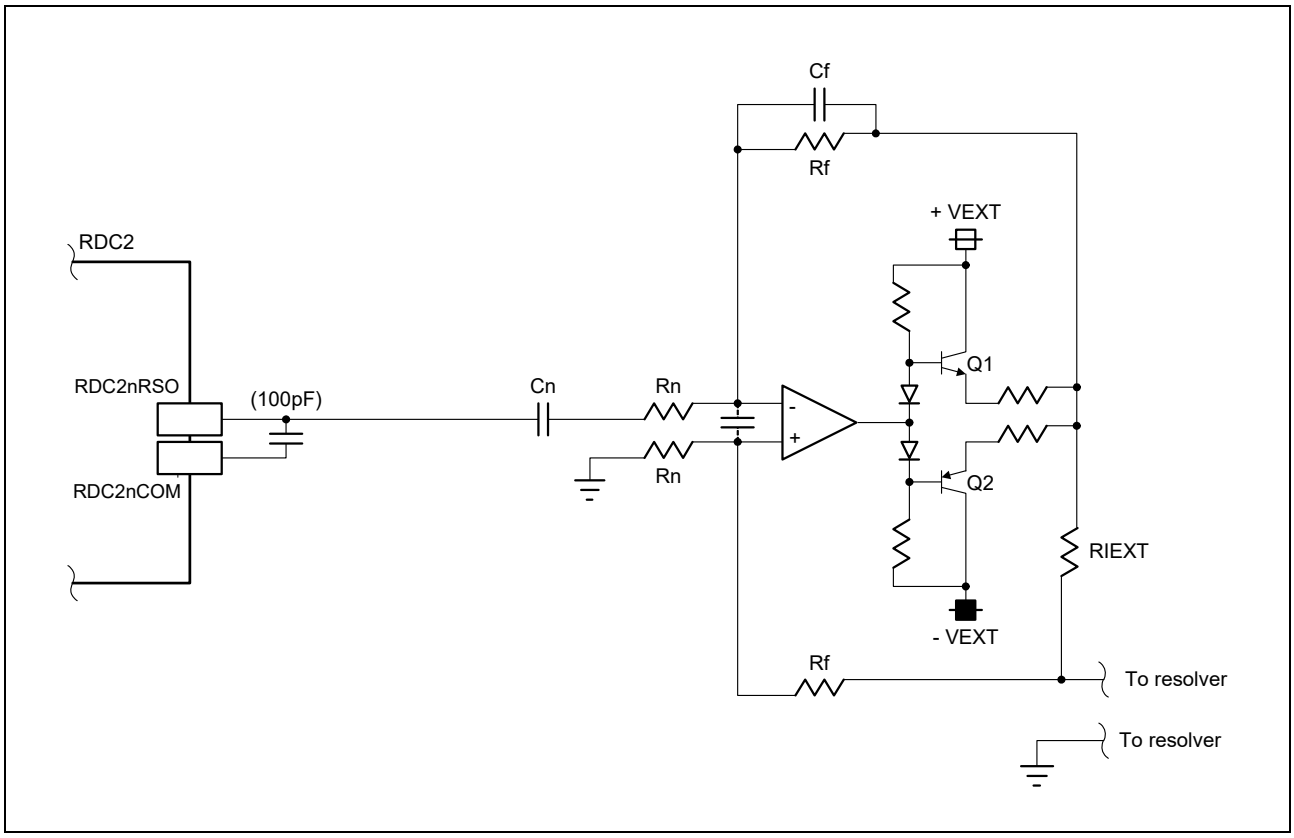


Figure 25.17 Equivalent Circuit Example of Excitation Voltage Booster Amplifier Circuit (Dual Power Supply)

25.6.2.3 Method for Setting Constants of Excitation Voltage Booster Amplifier Circuit

Both of the excitation voltage booster amplifier circuits shown as reference use the current control type. This type is effective in preventing secondary failure caused by short-circuit accident between excitation lines. Also, it is expected to improve the S/N ratio of resolver signal by boosting voltage.

Step (1): Calculate the excitation current by setting the excitation voltage based on the voltage of external power supply.

$$VREF = IREF \times ZRO$$

Step (2): Calculate the circuit constants based on the excitation current.

$$IREF = (VRSO \times Rf) / RIEXT \times Rn)$$

[Legend]

+VEXT, - VEXT:	External power supplies (for the excitation voltage booster amplifier circuit)
IREF:	Excitation current of Resolver
RIEXT:	Resistor for setting excitation current of Resolver
VREF:	Excitation voltage of Resolver
ZRO:	Input impedance of Resolver (specification value)
VRSO:	RDC2nRSO pin output voltage (= 2 Vp-p)

<Settings conditions>

- $RIEXT \leq (ZRO/10)$ [Ω]
- $Rf \geq 50$ kΩ, $Cn \times Rn \geq 5 \times 10^{-4}$ [s], $Cf \times Rf \leq 5 \times 10^{-6}$ [s]
- Use the same power supply for an operational amplifier as that for the transistor buffer.

25.6.3 Resolver Excitation Signal External Input Method

25.6.3.1 Resolver Excitation Signal Input Circuit (Single Power Supply)

Figure 25.18 shows an equivalent circuit when the resolver excitation signal is input from an external source (single power supply). **Table 25.55** lists additional resistor values (reference values) of the resolver excitation signal external input circuit with a single power supply.

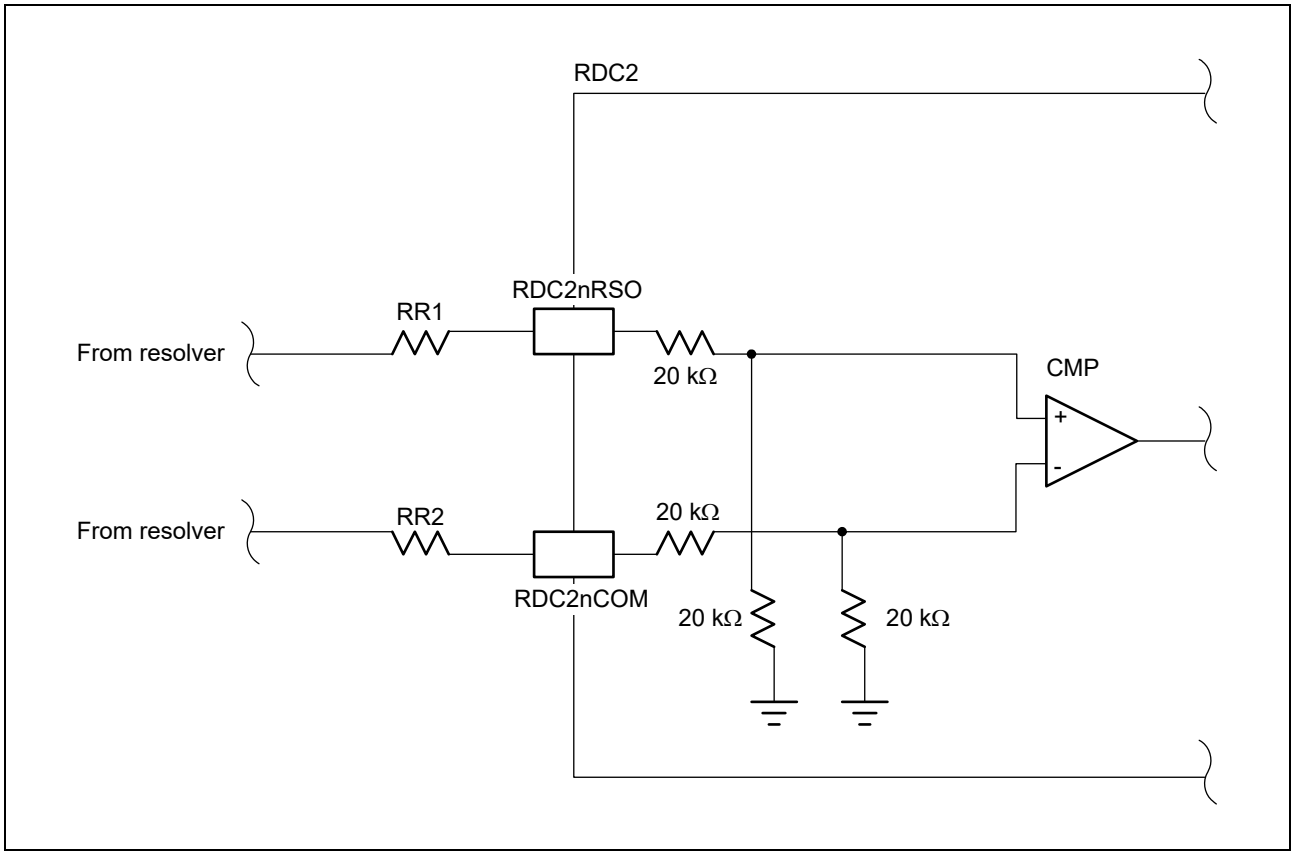


Figure 25.18 Resolver Excitation Signal External Input Circuit (Excitation Circuit is Single Power Supply)

Table 25.55 Additional Resistor Value for Resolver Excitation Signal External Input Circuit (Reference Values, Single Power Supply)

+ VEXT	RR1, RR2
5-V system	0 kΩ
12-V system	47 kΩ
24-V system	120 kΩ

25.6.3.2 Resolver Excitation Signal Input Circuit (Dual Power Supply)

Figure 25.19 shows an equivalent circuit when the resolver excitation signal is input from an external source (dual power supply). Adjust Rn1 and Rn2 so that the voltage on the RDC2nCOM pin does not exceed 5V.

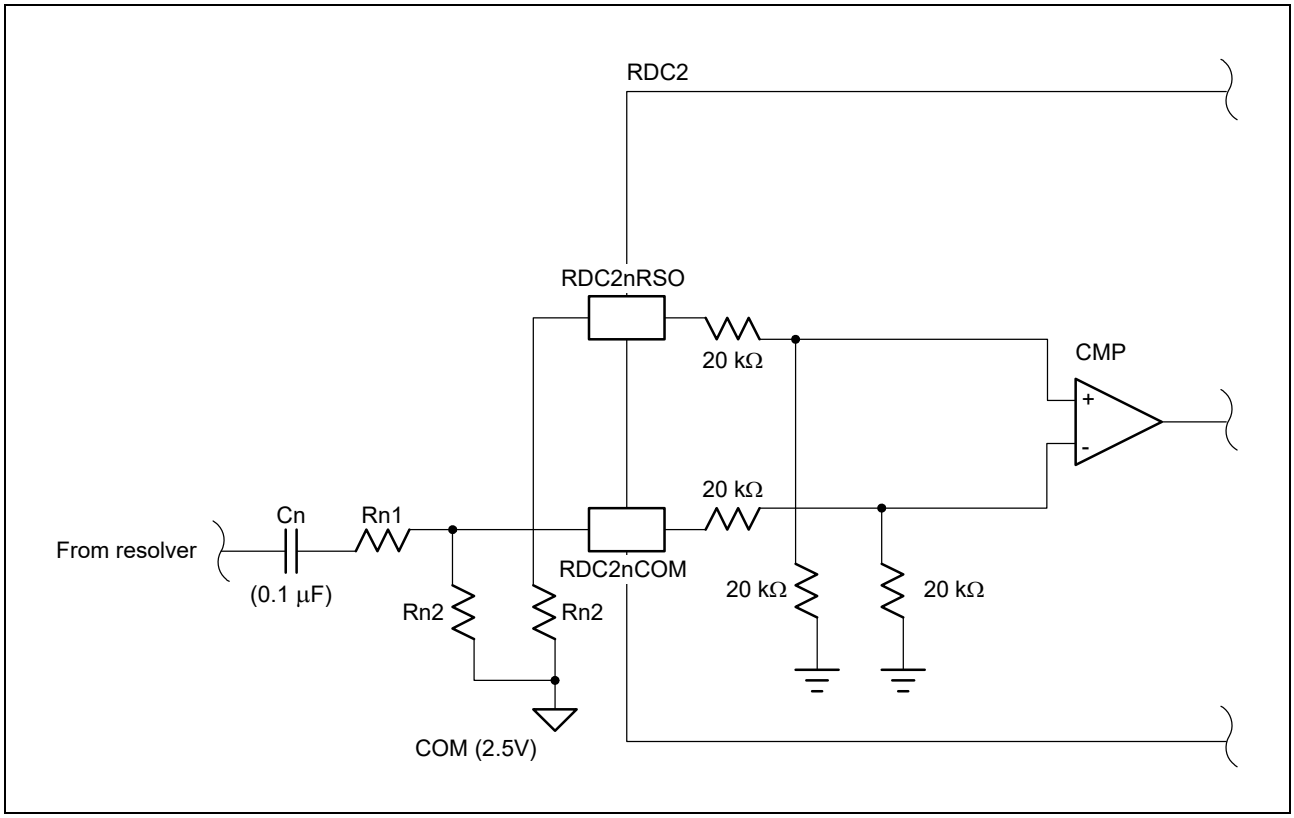


Figure 25.19 Resolver Excitation Signal External Input Circuit (Excitation Circuit is Dual Power Supply)

25.7 Usage Notes

When the resolver is used as a motor sensor, the resolver signal is affected by various types of noises depending on the drive control configuration of the motor. To perform R/D conversion successfully, sufficient S/N ratio of the resolver signal is required.

This RDC is a highly responsive R/D conversion module. Although considerations have been made for noise immunity, the RDC by itself is not designed to accommodate all noise environmental factors. Appropriate peripheral circuits need to be considered depending on the environment in which the module is deployed. The following describes the specific countermeasures for noise in [Countermeasure I] to [Countermeasure VIII] as reference.

25.7.1 Countermeasures for Magnetic Disturbance Noise

If the leakage flux of the motor passes through the resolver, the resolver signal behaves as if the angle changes, resulting in malfunction.

[Countermeasure I]

When installing the motor and resolver, use the configuration and material that block the magnetic loop passing through the resolver (magnetic shield effect) to minimize the leakage flux of the motor that passes through the resolver.

[Countermeasure II]

If the leakage flux of the motor that passes through the resolver cannot be completely avoided, raise the resolver excitation voltage (current) to improve the S/N ratio of the original signals.

25.7.2 Countermeasures for Electric Disturbance Noise

The electric disturbance (spike noise, and so on) caused by the PWM drive of the motor is extremely large, and affects all power systems, such as the excitation signal lines of resolver and power supply line, via various paths.

[Countermeasure III]

Insert a common mode/normal mode filter into the resolver excitation lines to remove the spike noise components. Generally, the low-impedance excitation line hardly has noise and the countermeasure is less needed.

[Countermeasure IV]

Insert a common mode/normal mode filter into the resolver signal lines (RDC2nS1-RDC2nS3, RDC2nS2-RDC2nS4) to remove the spike noise components. Select the time constant that takes effect only on noise and leaves the original resolver signal waveform undistorted. In addition, ensure that the electric noise waveform of the RDC2nS1 to RDC2nS4 pins viewed from RVSS (ground) are in phase.

If an error due to electric external disturbance noise persists after this countermeasure is taken, keeping the resolver signal level low can be effective.

[Countermeasure V]

If necessary, insert a bypass capacitor to the power supply (RVDD) line.

25.7.3 Other General Measures

[Countermeasure VI]

Use a shielded twisted pair cable for resolver wiring. Shielded terminals must be treated collectively on the circuit side (grounded to GND). The cable must be routed separately from the motor cable.

[Countermeasure VII]

Enhance the GND system for low impedance to reduce common impedance noise and to provide shielding effect. Another possible measure is to fix the potential of the motor driver radiator and motor case to the control-system ground potential.

[Countermeasure VIII]

Physically separate the motor driver from the sensor circuit and cover each of them with a shield case.

Section 26 A/D Converter (ADCC)

This section contains a generic description of the A/D Converter (ADCC).

The first part describes all RH850/C1x specific properties such as the number of units, register base address, etc. The subsequent parts describe the functions and registers of ADCC.

26.1 Features of RH850/C1x ADCC

26.1.1 Number of Units

This LSI has the following number of units of ADCC.

Table 26.1 Number of Units

Product	RH850/C1x
Number of units	2
Name	ADCCn (n = 0, 1)

Table 26.2 Index

Index	Meaning
n	Throughout this section, the individual ADCC units are identified by the index “n” (n = 0, 1), for example, ADCCnVCRj for the virtual channel register j.
j	Throughout this section, the number of data registers and virtual channels of ADCC is indicated by the index “j” (j = 0 to 35), for example, ADCCnDRj for the data register j.
k	Throughout this section, the number of each T&H channel of ADCC is indicated by the index “k” (k = 0 to 5).
p	Throughout this section, the physical channel group is identified by the index “p” (ADCC0: p = 0 to 3, ADCC1: p = 0 to 6).
q	Throughout this section, the physical subchannel group identified by the index “q” (ADCC0: q = 0 to 3, ADCC1: q = 0 to 2).
x	Throughout this section, the scan group is identified by the index “x” (x = 0 to 4).
y	Throughout this section, the number of A/D timers is indicated by the index “y” (y = 3 to 4).

26.1.2 Register Base Address

ADCC base addresses are listed in the following table.

ADCC register addresses are given as offsets from the individual base address.

Table 26.3 Register Base Address

Base Address Name	Base Address
<ADCC0_base>	FFF2 0000_H
<ADCC1_base>	FFF2 1000_H

26.1.3 Clock Supply

ADCC clock is listed in following table.

Table 26.4 Clock Supply

Unit Name	Unit Clock Name	Supply Clock Name
ADCCn	PCLK	CLK_LSB (low-speed peripheral clock)
	clkad	CLKC_LSB (unmodulated low-speed peripheral clock)

26.1.4 Interrupt Requests

ADCC interrupt requests are listed in the following table. Scan group x end interrupt of ADCCn is expressed as ADInx.

Table 26.5 Interrupt Requests

Interrupt Name (Outline)		Interrupt Number	DMA Trigger Number	DTS Trigger Number
ADCC0				
ADI00	ADCC0 scan group 0 end interrupt	172	64	0
ADI01	ADCC0 scan group 1 end interrupt	173	65	1
ADI02	ADCC0 scan group 2 end interrupt	174	66	2
ADI03	ADCC0 scan group 3 end interrupt	175	67	3
ADI04	ADCC0 scan group 4 end interrupt	176	68	4
ADE0	ADCC0 A/D error interrupt	184	—	—
ADCC1				
ADI10	ADCC1 scan group 0 end interrupt	177	69	5
ADI11	ADCC1 scan group 1 end interrupt	178	70	6
ADI12	ADCC1 scan group 2 end interrupt	179	71	7
ADI13	ADCC1 scan group 3 end interrupt	180	72	8
ADI14	ADCC1 scan group 4 end interrupt	181	73	9
ADE1	ADCC1 A/D error interrupt	185	—	—

26.1.5 Reset Sources

ADCC reset sources are listed in the following table. ADCC is initialized by these reset sources.

Table 26.6 Reset Source

Unit	Reset Source
ADCCn	Any reset source

26.1.6 External Input/Output Signals

External input/output signals of ADCC and pin assignment of track and hold (T&H) circuit are listed in the following table.

External output pins are not assigned for ADCC.

Table 26.7 External Input/Output Signals (1/2)

Unit Signal Name	Outline	Alternative Port Signal	Assignment of T & H circuit	Product	
				C1H 252 pin	C1M 144 pin
AVcc0	ADCC0 power supply pin	A0VCC	—	√	√
AVss0	ADCC0 ground pin	A0VSS	—	√	√
AVcc1	ADCC1 power supply pin	A1VCC	—	√	√
AVss1	ADCC1 ground pin	A1VSS	—	√	√
AVREFH0	ADCC0 reference voltage pin	A0VREFH	—	√	√
AVREFH1	ADCC1 reference voltage pin	A1VREFH	—	√	√
AN000	Analog input pin 000	ADCC0I00	—	√	√
AN001	Analog input pin 001	ADCC0I01	—	√	—
AN002	Analog input pin 002	ADCC0I02	—	√	—
AN003	Analog input pin 003	ADCC0I03	—	√	√
AN010	Analog input pin 010	ADCC0I10	—	√	—
AN011	Analog input pin 011	ADCC0I11	—	√	—
AN012	Analog input pin 012	ADCC0I12	—	√	—
AN013	Analog input pin 013	ADCC0I13	—	√	—
AN020	Analog input pin 020	ADCC0I20	—	√	√
AN021	Analog input pin 021	ADCC0I21	—	√	√
AN022	Analog input pin 022	ADCC0I22	T&H circuit 4	√	√
AN023	Analog input pin 023	ADCC0I23	T&H circuit 5	√	√
AN030	Analog input pin 030	ADCC0I30	T&H circuit 0	√	√
AN031	Analog input pin 031	ADCC0I31	T&H circuit 1	√	√
AN032	Analog input pin 032	ADCC0I32	T&H circuit 2	√	√
AN033	Analog input pin 033	ADCC0I33	T&H circuit 3	√	√
AN100	Analog input pin 100	ADCC1I00	—	√	√
AN101	Analog input pin 101	ADCC1I01	—	√	√
AN102	Analog input pin 102	ADCC1I02	—	√	√
AN110	Analog input pin 110	ADCC1I10	—	√	√
AN111	Analog input pin 111	ADCC1I11	—	√	√
AN112	Analog input pin 112	ADCC1I12	—	√	—
AN120	Analog input pin 120	ADCC1I20	—	√	√
AN121	Analog input pin 121	ADCC1I21	—	√	—
AN122	Analog input pin 122	ADCC1I22	—	√	—
AN130	Analog input pin 130	ADCC1I30	T&H circuit 0	√	√
AN131	Analog input pin 131	ADCC1I31	T&H circuit 1	√	√
AN132	Analog input pin 132	ADCC1I32	T&H circuit 2	√	√
AN140	Analog input pin 140	ADCC1I40	T&H circuit 3	√	√
AN141	Analog input pin 141	ADCC1I41	T&H circuit 4	√	√
AN142	Analog input pin 142	ADCC1I42	T&H circuit 5	√	√

Table 26.7 External Input/Output Signals (2/2)

Unit Signal Name	Outline	Alternative Port Signal	Assignment of T & H circuit	Product	
				C1H 252 pin	C1M 144 pin
AN150	Analog input pin 150	ADCC1I50	—	√	—
AN151	Analog input pin 151	ADCC1I51	—	√	—
AN152	Analog input pin 152	ADCC1I52	—	√	—
AN160	Analog input pin 160	ADCC1I60	—	√	√
AN161	Analog input pin 161	ADCC1I61	—	√	√
AN162	Analog input pin 162	ADCC1I62	—	√	√

26.1.7 Rule for Naming Analog Input Pins

An analog input pin name is represented by a unit signal name or alternative port signal name. An analog input pin is called a physical channel. The name of an analog input pin is represented by a physical channel group number and physical subchannel number.

A rule for naming the unit signal name and alternative port signal name of an analog input pin is as follows:

Unit signal name: AN + Unit number + Physical channel group + Physical subchannel

Alternative port signal name: ADCC + Unit number + I + Physical channel group + Physical subchannel

For example, AN161 (ADCC1I61) for ADCC1, physical channel group 6, and physical subchannel 1.

26.2 Overview

26.2.1 Functional Overview

ADCC has the following features.

Item	Outline
Resolution	12 bits
A/D converter	Successive approximation method
Conversion speed	1.0μs per channel
Number of virtual channels (virtual ch)	ADCC0: 36 channels (virtual ch0 to 35) ADCC1: 36 channels (virtual ch0 to 35)
Number of scan groups (SG)	ADCC0: 5 groups (SG0 to 4) ADCC1: 5 groups (SG0 to 4)
A/D conversion mode	Each ADCC has four A/D conversion modes: • Normal A/D conversion • A/D conversion with a hold value • Self-diagnosis for A/D conversion circuits • Addition A/D conversion
Scan mode	Each ADCC has two scan modes: • Multicycle scan mode: Specified number of scans are executed. • Continuous scan mode: Scans are repeatedly executed without limit.
A/D conversion start trigger	Each ADCC has three A/D conversion start triggers: • Hardware trigger (HW trigger) • Software trigger (SW trigger) • A/D timer trigger (only supported by SG3 and SG4) **NOTE** When you use the simultaneous track and hold function, A/D timer trigger cannot be used.
Priority of scan group processing	A processing for a scan group can interrupt an ongoing processing for another scan group. The priority is as follows: High Low SG4 > SG3 > SG2 > SG1 > SG0
Suspend function	Each ADCC has three suspend methods: • Synchronous suspend • Asynchronous suspend • Synchronous and asynchronous combination suspend Make sure that you select asynchronous suspend when using the simultaneous track and hold function.
Interrupts and DMA and DTS transfers function	Each ADCC can make the following interrupts: • Scan group × end interrupt • A/D error interrupt DMA and DTS transfers can be started by a scan group x end interrupt signal.
Function for transferring results of A/D conversion to EMU	Each ADCC can output the following signals to the EMU. • A/D conversion end signal (virtual ch0 to 2) • A/D converted data (virtual ch0 to 2) • SG4 scan end signal
Self-diagnosis function	Each ADCC has the following self-diagnosis functions: • Self-diagnosis for A/D conversion circuits • Pin-level self-diagnosis • Self-diagnosis for wiring-break detection

26.2.2 Block Diagram

Figure 26.1 shows the ADCC0 block diagram and **Figure 26.2** shows the ADCC1 block diagram.

(1) Structure of ADCC0

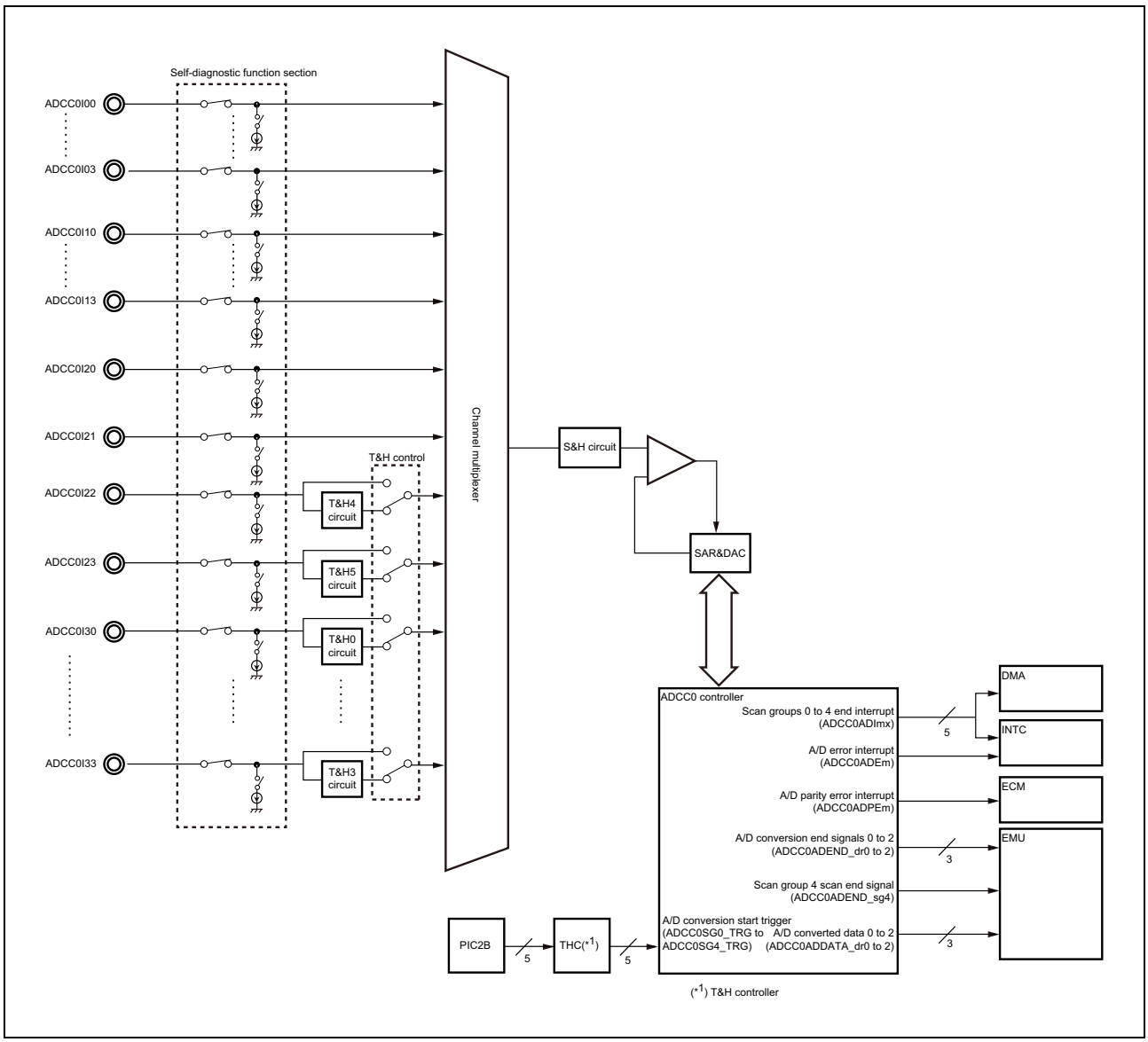


Figure 26.1 ADCC0 Block Diagram

(2) Structure of ADCC1

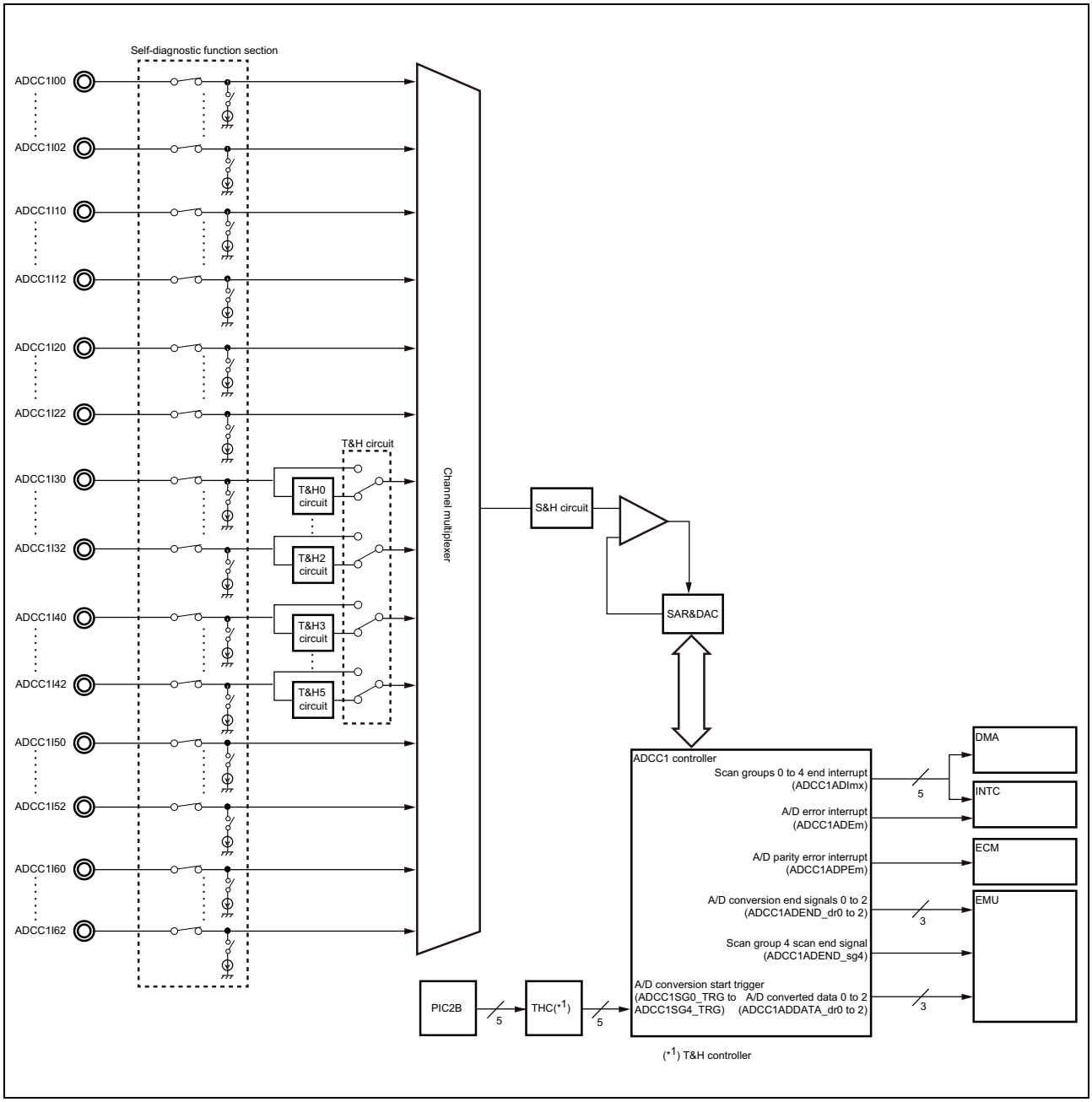


Figure 26.2 ADCC1 Block Diagram

(3) ADCC functional block diagram

Figure 26.3 shows a functional block diagram of the ADCC.

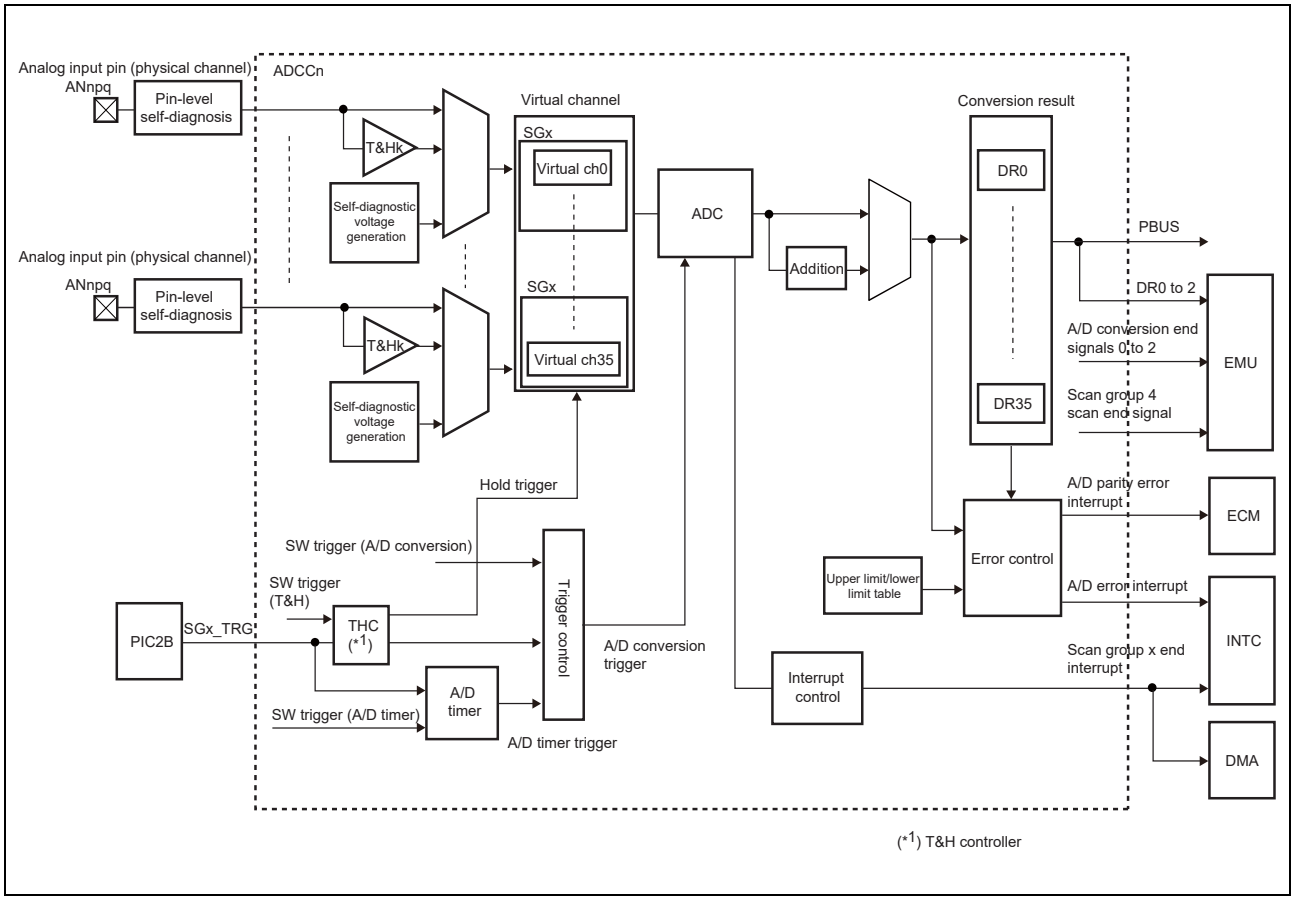


Figure 26.3 ADCC Functional Block Diagram

26.2.3 Virtual Channel (Virtual ch)

A virtual channel is an analog input pathway which can select an analog input pin (physical channel), T&Hk circuit output, or A/D conversion circuit self-diagnosis output. Each ADCC has 36 virtual channels. The result of A/D conversion is stored in the data register whose number is the same as the virtual channel.

The following diagram is an image of virtual channels.

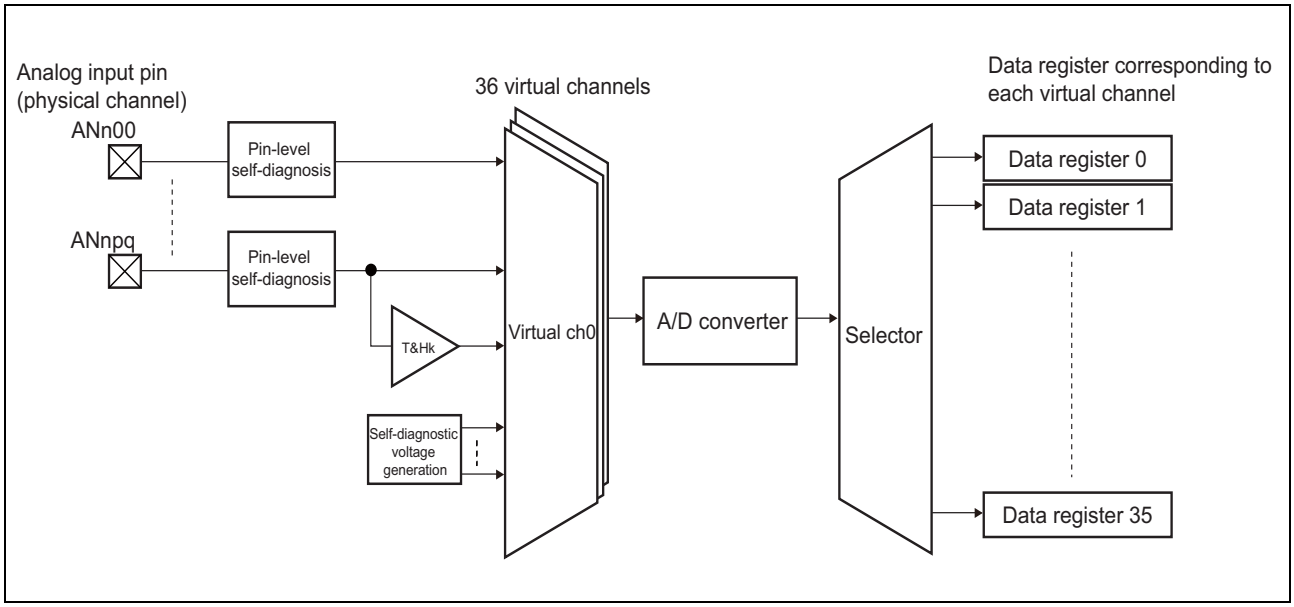


Figure 26.4 Image Diagram of Virtual Channels

26.2.4 Scan Group (SG)

A scan group is a group of multiple virtual channels.

Each ADCC has five scan groups. The priority of A/D conversion is as follows:
SG4 > SG3 > SG2 > SG1 > SG0

SGx can group sequential virtual channels. Specify the grouping settings with a start pointer (ADCCnSGVCSPx register) and an end pointer (ADCCnSGVCEPx register). Disable A/D conversion trigger input for unused scan groups.

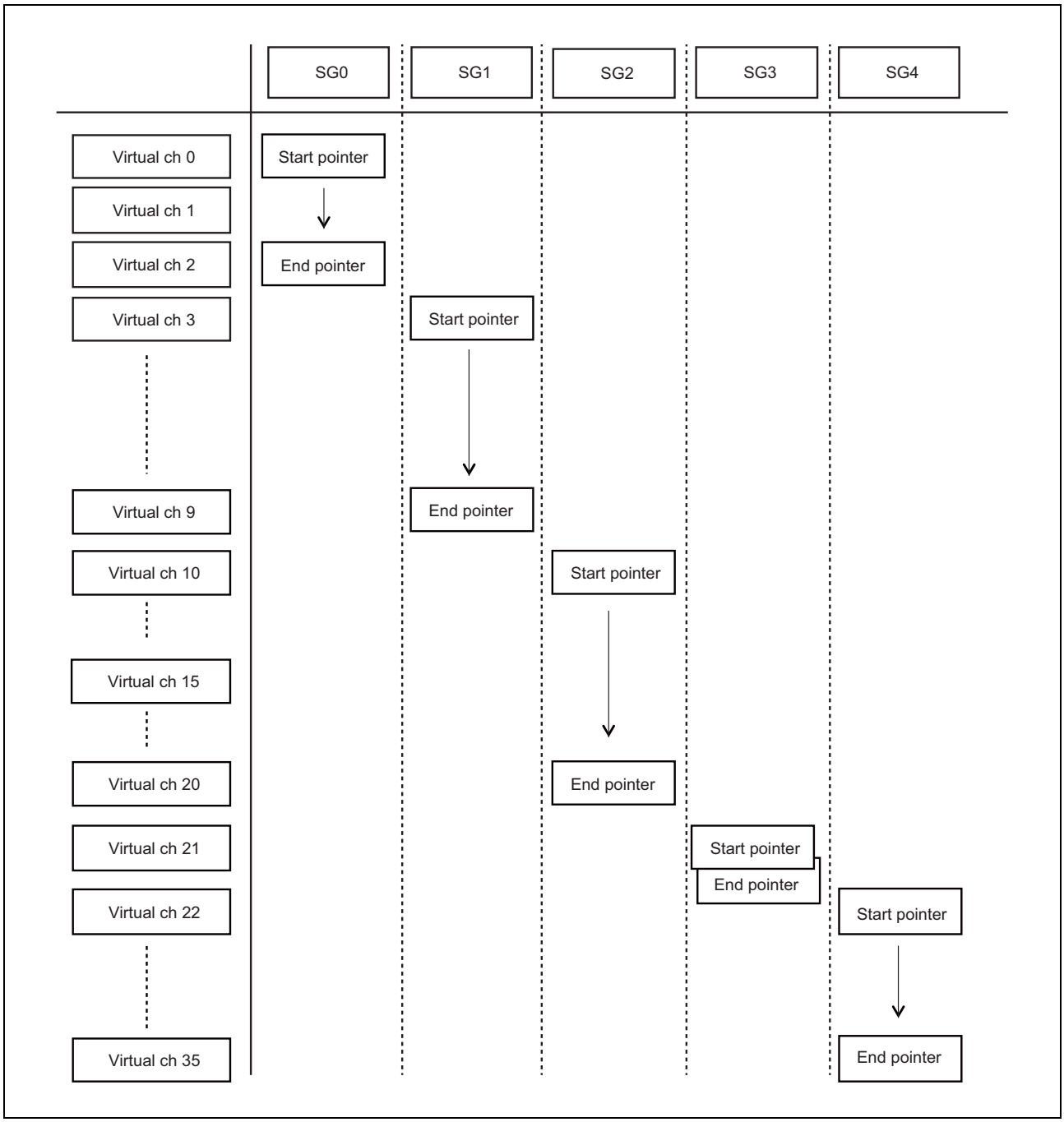


Figure 26.5 Example of SG assignment

26.3 Register

26.3.1 List of Registers

ADCC registers are listed in the following table.

For <ADCC0_base> and <ADCC1_base>, see **Section 26.1.2, Register Base Address**.

Table 26.8 List of Registers (1/2)

Module Name	Register Name	Symbol	Address
ADCCn	A/D synchronization start control register	ADCC0ADSYNSTCR	<ADCC0_base> + 300_H
ADCCn	A/D timer synchronization start control register	ADCC0ADTSYNSTCR	<ADCC0_base> + 304_H
ADCCn	Virtual channel register j	ADCCnVCRj	<ADCCn_base> + j × 4_H
ADCCn	Data register j	ADCCnDRj	<ADCCn_base> + 100_H + j × 2_H
ADCCn	Data supplementary information register j	ADCCnDIRj	<ADCCn_base> + 200_H + j × 4_H
ADCCn	A/D halt register	ADCCnADHALTR	<ADCCn_base> + 380_H
ADCCn	A/D control register 1	ADCCnADCR1	<ADCCn_base> + 384_H
ADCCn	A/D control register 2	ADCCnADCR2	<ADCCn_base> + 398_H
ADCCn	T&H sampling start control register	ADCCnTHSMPSTCR	<ADCCn_base> + 400_H
ADCCn	T&H stop control register	ADCCnTHSTPCR	<ADCCn_base> + 404_H
ADCCn	T&H control register	ADCCnTHCR	<ADCCn_base> + 408_H
ADCCn	T&H group A hold start control register	ADCCnTHAHLDSTCR	<ADCCn_base> + 410_H
ADCCn	T&H group B hold start control register	ADCCnTHBHLDSTCR	<ADCCn_base> + 414_H
ADCCn	T&H group A control register	ADCCnTHACR	<ADCCn_base> + 420_H
ADCCn	T&H group B control register	ADCCnTHBCR	<ADCCn_base> + 424_H
ADCCn	T&H enable register	ADCCnTHER	<ADCCn_base> + 430_H
ADCCn	T&H group select register	ADCCnTHGSR	<ADCCn_base> + 434_H
ADCCn	Safety control register	ADCCnSFTCR	<ADCCn_base> + $3C0_H$
ADCCn	Pin level self-diagnostic control register	ADCCnTDCR	<ADCCn_base> + $3C4_H$
ADCCn	Wiring-break detection control register	ADCCnODCR	<ADCCn_base> + $3C8_H$
ADCCn	Upper-limit/lower-limit table register 0	ADCCnULLMTBR0	<ADCCn_base> + $3CC_H$
ADCCn	Upper-limit/lower-limit table register 1	ADCCnULLMTBR1	<ADCCn_base> + $3D0_H$
ADCCn	Upper-limit/lower-limit table register 2	ADCCnULLMTBR2	<ADCCn_base> + $3D4_H$
ADCCn	Error clear register	ADCCnECR	<ADCCn_base> + $3D8_H$
ADCCn	Upper-limit/lower-limit error register	ADCCnULER	<ADCCn_base> + $3DC_H$
ADCCn	Overwrite error register	ADCCnOWER	<ADCCn_base> + $3E0_H$
ADCCn	Parity error register	ADCCnPER	<ADCCn_base> + $3E4_H$
ADCCn	ID error register	ADCCnIDER	<ADCCn_base> + $3E8_H$
ADCCn	Scan group x start control register	ADCCnSGSTCRx	<ADCCn_base> + x × 80_H + 480_H
ADCCn	A/D timer y start control register	ADCCnADTSTCRy	<ADCCn_base> + y × 80_H + 488_H
ADCCn	A/D timer y end control register	ADCCnADTENDCRy	<ADCCn_base> + y × 80_H + $48C_H$
ADCCn	Scan group x control register	ADCCnSGCRx	<ADCCn_base> + x × 80_H + 490_H
ADCCn	Scan group x start virtual channel pointer	ADCCnSGVCSPx	<ADCCn_base> + x × 80_H + 494_H
ADCCn	Scan group x end virtual channel pointer	ADCCnSGVCEPx	<ADCCn_base> + x × 80_H + 498_H
ADCCn	Scan group x multicycle register	ADCCnSGMCYCRx	<ADCCn_base> + x × 80_H + $49C_H$
ADCCn	Scan group x status register	ADCCnSGSRx	<ADCCn_base> + x × 80_H + $4A4_H$

Table 26.8 List of Registers (2/2)

Module Name	Register Name	Symbol	Address
ADCCn	A/D timer initial phase register y	ADCCnADTIPRy	<ADCCn_base> + y × 80_H + $4A8_H$
ADCCn	A/D timer cycle register y	ADCCnADTPRRy	<ADCCn_base> + y × 80_H + $4AC_H$
ADCCn	Scan group x upper-limit/lower-limit table select register	ADCCnULLMSRx	<ADCCn_base> + x × 80_H + $4B0_H$

26.3.2 ADCC0ADSYNSTCR — A/D Synchronization Start Control Register

This is a register that controls simultaneous start of A/D conversion for each scan group of ADCC0 and ADCC1. Only ADCC0 has this register.

Access: This register can be written in 8-bit units.

Address: <ADCC0_base> + 300_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.9 ADCC0ADSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADSTART	This bit simultaneously starts A/D conversion for each scan group of ADCC0 and ADCC1. 0: Nothing occurs. (A written value of 0 is ignored.) 1: A/D conversion starts. Enable the SG synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of the SG for which you want to simultaneously start A/D conversion.

26.3.3 ADCC0ADTSYNSTCR — A/D Timer Synchronization Start Control Register

This is a register that controls simultaneous start of count operation for each A/D timer of ADCC0 and ADCC1. Only ADCC0 has this register.

Access: This register can be written in 8-bit units.

Address: <ADCC0_base> + 304_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.10 ADCC0ADTSYNSTCR Register Contents

Bit Position	Bit Position	Bit Position
7 to 1	Reserved	When written, write the value after reset.
0	ADTSTART	This bit simultaneously starts count operation for each A/D timer of ADCC0 and ADCC1. 0: Nothing occurs. (A written value of 0 is ignored.) 1: A/D timer count starts. Enable the A/D timer synchronization start enable (ADCCnSGCRx.ADTSTARTE bit) of the A/D timer for which you want to simultaneously start count operation.

26.3.4 ADCCnVCRj — Virtual Channel Register j

This is a register for specifying the virtual channel settings.

Access: This register can be read or written in 16-bit units.

Address: <ADCCn_base> + j × 4_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNVCLS[2:0]			—	—	—	—	—	ADIE	—	GCTRL[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.11 ADCCnVCRj Register Contents

Bit Position	Bit Name	Function
15 to 13	CNVCLS[2:0]	These bits set the conversion mode: 0_H: Normal A/D conversion 1_H: Hold value A/D conversion 3_H: Self-diagnosis for A/D conversion circuits 4_H: Addition A/D conversion Other than above: Setting prohibited
12 to 8	Reserved	When written, write the value after reset.
7	ADIE	This bit sets whether to output a scan group x end interrupt signal (ADInx). 0: Output prohibited 1: Output allowed
6	Reserved	When written, write the value after reset.
5 to 0	GCTRL[5:0]	These bits specify the A/D conversion settings for each conversion mode. • Normal A/D conversion (CNVCLS[2:0] = 0_H) GCTRL[5:2]: Physical channel group GCTRL[1:0]: Physical subchannel • Hold value A/D conversion (CNVCLS[2:0] = 1_H) 00_H: Hold value of T&H0 is A/D converted. 01_H: Hold value of T&H1 is A/D converted. 02_H: Hold value of T&H2 is A/D converted. 03_H: Hold value of T&H3 is A/D converted. 04_H: Hold value of T&H4 is A/D converted. 05_H: Hold value of T&H5 is A/D converted. Other than above and the settings in **Section 26.7.2, Notes during Injection Current Application**, and **Section 26.7.3, Notes when Using Simultaneous Track-and-Hold**. • Self-diagnosis for A/D conversion circuits (CNVCLS[2:0] = 3_H) 00_H: AVREFH × 0 04_H: AVREFH × 1/4 08_H: AVREFH × 1/2 $0C_H$: AVREFH × 3/4 10_H: AVREFH × 1 Other than above: Setting prohibited. • Addition A/D conversion (CNVCLS[2:0] = 4_H) GCTRL[5:2]: Physical channel group GCTRL[1:0]: Physical subchannel

26.3.5 ADCCnDRj — Data Register j

This is a register that stores A/D conversion result data. This register can be read in 32-bit units (ADCCnDRj and ADCCnDRj + 1) because the ADCCnDRj register and the ADCCnDRj + 1 register are aligned successively.

When the read and clear enable (ADCCnSFTCR.RDCLRE bit) is 1 (enabled), be sure to read this register in 32-bit units (ADCCnDRj_j + 1). For the structure of the register, see **Figure 26.6, Data Register Alignment for 32-bit Read**.

Access: This register can be read in 16-bit units.

Address: <ADCCn_base> $+100_H + j \times 2_H$

Value after reset: 0000_H

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

Table 26.12 ADCCnDRj Register Contents

Bit Position	Bit Name	Function
15 to 0	DRj[15:0]	Data register These bits store A/D conversion result data.

CAUTION

Note that ADCCnDIRj.WFLG and ADCCnDIR(j+1).WFLG are cleared by reading ADCCnDRj in 16-bit units.

The ADCCnADCR2.DFMT bit can be used for changing the format to the signed fixed-point format or signed integer format. The following tables show data arrangement for each format. Convert twice and Convert four times in the table are data formats for addition A/D conversion.

For signed fixed-point format (ADCCnADCR2.DFMT = 0)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S	A/D conversion result												0	0	0
Convert twice	S	A/D conversion result (twice additional value)													0	0
Convert four times	S	A/D conversion result (four times additional value)														0

Position of decimal point

For signed integer format (ADCCnADCR2.DFMT = 1)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S	S	S	S	A/D conversion result											
Convert twice	S	S	S	A/D conversion result (twice additional value)												
Convert four times	S	S	A/D conversion result (four times additional value)													

Position of decimal point

S	: Sign bit (always 0)
0	: Zero extension

When you want to read the data register j in 32-bit units, read ADCCnDRj_j + 1. The data alignment of ADCCnDRj_j + 1 is as follows.

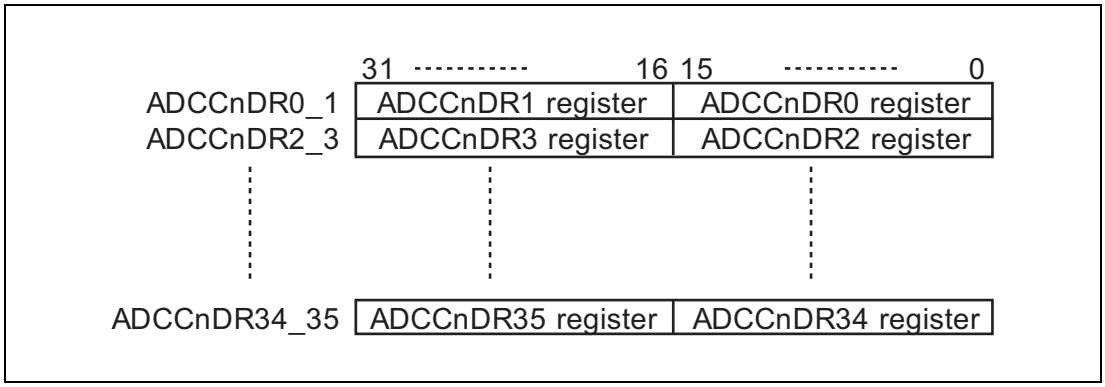


Figure 26.6 Data Register Alignment for 32-bit Read

26.3.6 ADCCnDIRj — Data Supplementary information Register j

This is a register that stores supplementary information of data register (ADCCnDRj register) and A/D converted value. This register must always be read as 32-bit data.

When the read and clear setting (ADCCnSFTCR.RDCLRE bit) is 1, if the ADCCnDRj or ADCCnDIRj register is read, the ADCCnDRj and ADCCnDIRj registers are all cleared to 0. The ADCCnDIRj.WFLG bit is cleared when the ADCCnDRj or ADCCnDIRj register is read regardless of the read and clear setting (ADCCnSFTCR.RDCLRE bit).

Access: This register can be read in 32-bit units.

Address: <ADCCn_base> $+200_H + j \times 4_H$

Value after reset: $0000\ 0000_H$

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

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

Table 26.13 ADCCnDIRj Register Contents

Bit Position	Bit Name	Function
31 to 26	Reserved	When written, write the value after reset.
25	WFLG	Write Flag Setting condition: An A/D converted value is stored in ADCCnDRj. Clearing condition: The ADCCnDRj or ADCCnDIRj register is read.
24	PRTY	Parity Parity bit (even parity) for the ADCCnDRj bit.
23 to 16	Reserved	When written, write the value after reset.
15 to 0	DRj[15:0]	Data register These bits store the same A/D conversion result data as the ADCCnDRj register.

26.3.7 ADCCnADHALTR — A/D Halt Register

This is a register for terminating each ADCC.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 380_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HALT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.14 ADCCnADHALTR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HALT	This bit forcibly terminates all the scan groups, A/D conversions, and A/D timers. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Termination is performed.

26.3.8 ADCCnADCR1 — A/D Control Register 1

This is a register for setting ADCC common control (suspend method).

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 384_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SUSMTD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 26.15 ADCCnADCR1 Register Contents

Bit Position	Bit Position	Bit Position
7 to 2	Reserved	When written, write the value after reset.
1, 0	SUSMTD[1:0]	These bits select the suspend method. 0: Synchronous suspend 1: Synchronous and asynchronous combination suspend 2: Asynchronous suspend 3: Setting prohibited

26.3.9 ADCCnADCR2 — A/D Control Register 2

This is a register for setting ADCC common control (data format and addition count for addition A/D conversion).

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 398_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	DFMT	—	—	—	ADDNT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W

Table 26.16 ADCCnADCR2 Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When written, write the value after reset.
4	DFMT	This bit is the data format setting for the ADCCnDRj and ADCCnDIRj register. 0: Signed fixed-point format 1: Signed integer format For details of data format, see **Section 26.3.5, ADCCnDRj — Data Register j**.
3 to 1	Reserved	When written, write the value after reset.
0	ADDNT	This bit selects the addition count for addition A/D conversion. 0: Add twice 1: Add four times

26.3.10 ADCCnTHSMPSTCR — T&H Sampling Start Control Register

This is a register that controls start of all T&H sampling.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 400_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SMPST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.17 ADCCnTHSMPSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	SMPST	This bit starts all T&H sampling. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts sampling

26.3.11 ADCCnTHSTPCR — T&H Stop Control Register

This is a register that controls stop of all T&H.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 404_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	THSTP
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.18 ADCCnTHSTPCR Register Contents

Bit Position	Bit Position	Bit Position
7 to 1	Reserved	When written, write the value after reset.
0	THSTP	This bit stops all T&H. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Stops all T&H.

CAUTION

Use the ADCCnTHSTPCR.THSTP bit after stopping all SGs by using the ADCCnADHALTR register.

After stopping an SG by setting the ADCCnTHSTPCR.THSTP bit, set ADCCnTHER.THkE to all 0s to prevent re-sampling operations due to the auto-sampling function (ADCCnTHCR.ASMPMSK bit).

26.3.12 ADCCnTHCR — T&H Control Register

This is a register that controls T&H sampling.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 408_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ASMPMSK
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 26.19 ADCCnTHCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ASMPMSK	This bit selects whether to perform auto sampling when A/D conversion of the relevant hold value is completed. 0: Automatic sampling is performed. 1: Automatic sampling is not performed.

26.3.13 ADCCnTHAHLDSTCR — T&H Group A Hold Start Control Register

This is a register that controls the start of hold for T&H group A.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 410_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HLDST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.20 ADCCnTHAHLDSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HLDST	This bit starts hold for T&H group A. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts hold.

26.3.14 ADCCnTHBHLDSTCR — T&H Group B Hold Start Control Register

This is a register that controls the start of hold for T&H group B.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + 414_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HLDST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.21 ADCCnTHBHLDSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	HLDST	This bit starts hold for T&H group B. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts hold.

26.3.15 ADCCnTHACR — T&H Group A Control Register

This is a register that controls T&H group A.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 420_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	HLDCTE	HLDTE	—	—	SGS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 26.22 ADCCnTHACR Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When written, write the value after reset.
5	HLDCTE	This bit selects whether hold control is enabled or disabled. 0: Disabled. 1: Hold control is enabled.
4	HLDTE	This bit selects whether H/W trigger signals are enabled or disabled. 0: Disabled. 1: The H/W trigger signal of the scan group selected in the ADCCnTHACR.SGS[1:0] bits is enabled.
3, 2	Reserved	When written, write the value after reset.
1, 0	SGS[1:0]	These bits select the scan group for T&H group A. 0: SG1 1: SG2 2: SG3 3: SG4

CAUTION

Set both the ADCCnTHACR.HLDCTE and ADCCnTHACR.HLDTE bits to 1 when using the simultaneous track and hold function with a H/W trigger. Set the ADCCnTHACR.HLDCTE bit to 1 and the ADCCnTHACR.HLDTE bit to 0 when using the simultaneous track and hold function with a S/W trigger.

Do not set the ADCCnTHACR.SGS[1:0] bits and the ADCCnTHBCR.SGS[1:0] bits to the same scan group when both T&H group A and B are used. Set this register in an A/D conversion stop state.

When the ADCCnTHACR.HLDCTE bit is set to 1, set the ADCCnSGCRx.TRGMD1 bit and the ADCCnSGCRx.TRGMD0 bit in SGx to be selected by using ADCCnTHACR.SGS[1:0] bits to 0 and 1 respectively.

26.3.16 ADCCnTHBCR — T&H Group B Control Register

This is a register that controls T&H group B.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 424_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	HLDCTE	HLDTE	—	—	SGS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 26.23 ADCCnTHBCR Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When written, write the value after reset.
5	HLDCTE	This bit selects whether hold control is enabled or disabled. 0: Disabled. 1: Hold control is enabled.
4	HLDTE	This bit selects whether H/W trigger signals are enabled or disabled. 0: Disabled. 1: The H/W trigger signal of the scan group selected in the ADCCnTHBCR.SGS[1:0] bits is enabled.
3, 2	Reserved	When written, write the value after reset.
1, 0	SGS[1:0]	These bits select the scan group for T&H group B. 0: SG1 1: SG2 2: SG3 3: SG4

CAUTION

Set both the ADCCnTHBCR.HLDCTE and ADCCnTHBCR.HLDTE bits to 1 when using the simultaneous track and hold function with a H/W trigger. Set the ADCCnTHBCR.HLDCTE bit to 1 and the ADCCnTHBCR.HLDTE bit to 0 when using the simultaneous track and hold function with a S/W trigger.

Do not set the ADCCnTHACR.SGS[1:0] bits and the ADCCnTHBCR.SGS[1:0] bits to the same scan group when both T&H group A and B are used. Set this register in an A/D conversion stop state.

When the ADCCnTHBCR.HLDCTE bit is set to 1, set the ADCCnSGCRx.TRGMD1 bit and the ADCCnSGCRx.TRGMD0 bit in SGx to be selected by using ADCCnTHBCR.SGS[1:0] bits to 0 and 1 respectively.

26.3.17 ADCCnTHER — T&H Enable Register

This is a register that controls whether to enable or disable of each T&H circuit.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + 430_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	TH5E	TH4E	TH3E	TH2E	TH1E	TH0E
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.24 ADCCnTHER Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When written, write the value after reset.
5 to 0	THkE	These bits set whether to enable or disable track and hold operation for T&Hk circuits. 0: Disabled 1: Enabled

26.3.18 ADCCnTHGSR — T&H Group Select Register

This is a register that selects the T&H group for each T&H.

Access: This register can be read or written in 16-bit units.

Address: <ADCCn_base> + 434_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TH5GS	—	TH4GS	—	TH3GS	—	TH2GS	—	TH1GS	—	TH0GS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W

Table 26.25 ADCCnTHGSR Register Contents

Bit Position	Bit Name	Function
15 to 11, 9, 7, 5, 3, 1	Reserved	When written, write the value after reset.
10, 8, 6, 4, 2, 0	THkGS	These bits selects the T&H group for T&Hk. 0: Selects T&H group A. 1: Selects T&H group B.

26.3.19 ADCCnSFTCR — Safety Control Register

This is a register for safety control.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + $3C0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	RDCLRE	ULEIE	OWEIE	PEIE	IDEIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 26.26 ADCCnSFTCR Register Contents

Bit Position	Bit Name	Function
7 to 5	Reserved	When written, write the value after reset.
4	RDCLRE	This bit sets whether to perform or prohibit read and clear operations. This bit sets whether to clear the ADCCnDRj and ADCCnDIRj registers to 0 (all 0) when reading the ADCCnDRj or ADCCnDIRj register. 0: Not read and cleared. 1: Read and cleared
3	ULEIE	This bit sets whether to allow or prohibit upper-limit/lower-limit error interrupts. 0: Prohibited 1: Allowed
2	OWEIE	This bit sets whether to allow or prohibit overwrite error interrupts. 0: Prohibited 1: Allowed
1	PEIE	This bit sets whether to allow or prohibit parity errors. 0: Prohibited 1: Allowed
0	IDEIE	This bit sets whether to allow or prohibit ID error interrupts. 0: Prohibited 1: Allowed

26.3.20 ADCCnTDCR — Pin Level Self-diagnostic Control Register

This is a register that controls the pin level self-diagnosis.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + $3C4_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	TDE	—	—	—	—	—	TDLV[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W

Table 26.27 ADCCnTDCR Register Contents

Bit Position	Bit Name	Function
7	TDE	This bit sets whether to perform or prohibit pin level self-diagnosis. 0: Pin level self-diagnosis is not performed. 1: Pin level self-diagnosis is performed. When TDE is set to 1, all analog pins are disconnected from the input buffer.
6 to 2	Reserved	When written, write the value after reset.
1, 0	TDLV[1:0]	These bits set the diagnosis voltage to be applied when pin level self-diagnosis is performed. 0: AVSS is applied to even numbers of physical subchannels, and AVCC is applied to odd numbers of physical subchannels. 1: AVCC is applied to even numbers of physical subchannels, and AVSS is applied to odd numbers of physical subchannels. 2: AVSS is applied to even numbers of physical subchannels, and 1/2 × AVCC is applied to odd numbers of physical subchannels. 3: 1/2 × AVCC is applied to even numbers of physical subchannels, and AVSS is applied to odd numbers of physical subchannels.

26.3.21 ADCCnODCR — Wiring-break Detection Control Register

This is a register that controls wiring-break detection.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + $3C8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ODE	—	ODPW[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.28 ADCCnODCR Register Contents

Bit Position	Bit Name	Function
7	ODE	This bit sets whether to perform or prohibit wiring-break detection. 0: Wiring-break detection is not performed. 1: Wiring-break detection is performed. When ODE is set to 1, wiring-break detection is enabled for all analog pins of ADCCn.
6	Reserved	When written, write the value after reset.
5 to 0	ODPW[5:0]	These bits set detection pulse width to be generated when wiring-break detection is performed. 04_H: 1 clock 05_H: 2 clocks : 13_H: 16 clocks 14_H: 17 clocks Note that ODPW[5:0] must be larger than 03_H and less than 15_H.

26.3.22 ADCCnULLMTBR0 to 2 — Upper Limit/Lower Limit Table Register 0 to 2

These registers are for setting the upper-limit and lower-limit values of an A/D converted value.

Access: This register can be read or written in 32-bit units.

Address: ADCCnULLMTBR0: <ADCCn_base> + $3CC_H$
ADCCnULLMTBR1: <ADCCn_base> + $3D0_H$
ADCCnULLMTBR2: <ADCCn_base> + $3D4_H$

Value after reset: $7FFE\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ULMTB[15:0]															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

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

Table 26.29 ADCCnULLMTBR Register Contents

Bit Position	Bit Name	Function
31 to 16	ULMTB[15:0]	These bits set the upper-limit of an A/D converted value. These bits should be set with the signed fixed-point format. Note that ULMTB[15] and ULMTB[0] are always fixed to 0.
15 to 0	LLMTB[15:0]	These bits set the lower-limit of an A/D converted value. These bits should be set with the signed fixed-point format. Note that LLMTB[15] and LLMTB[0] are always fixed to 0.

26.3.23 ADCCnECR — Error Clear Register

This is a register that controls error clear.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + $3D8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ULEC	OWEC	PEC	IDEC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 26.30 ADCCnECR Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When written, write the value after reset.
3	ULEC	This bit clears upper-limit and lower-limit errors. 0: Not cleared. 1: Cleared.
2	OWEC	This bit clears overwrite errors. 0: Not cleared. 1: Cleared.
1	PEC	This bit clears parity errors. 0: Not cleared. 1: Cleared.
0	IDEC	This bit clears ID errors. 0: Not cleared. 1: Cleared.

26.3.24 ADCCnULER — Upper Limit/Lower Limit Error Register

This is a register that indicates an upper limit or lower limit error. This register can be cleared by writing 1 to the upper-limit and lower-limit error clear (ADCCnECR.ULEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3DC_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ULE	—	ULECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.31 ADCCnULER Register Contents

Bit Position	Bit Name	Function
7	ULE	This bit indicates whether an upper-limit or lower-limit error exists. 0: An upper-limit or lower-limit error does not exist. 1: An upper-limit or lower-limit error exists.
6	Reserved	When written, write the value after reset.
5 to 0	ULECAP[5:0]	These bits indicate the number of the virtual channel at the time when an upper-limit or lower-limit error occurred.

CAUTION

This register is retained until 1 is written to the upper-limit and lower-limit error clear (ADCCnECR.ULEC bit). While the value of this register is retained, if an upper-limit and lower-limit error occurs, the new error information is discarded.

26.3.25 ADCCnOWER — Overwrite Error Register

This is a register that indicates an overwrite error. This register can be cleared by writing 1 to the overwrite error clear (ADCCnECR.OWEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	OWE	—	OWECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.32 ADCCnOWER Register Contents

Bit Position	Bit Name	Function
7	OWE	This bit indicates whether an overwrite error exists. 0: An overwrite error does not exist. 1: An overwrite error exists.
6	Reserved	When written, write the value after reset.
5 to 0	OWECAP[5:0]	These bits indicate the number of the virtual channel at the time when an overwrite error occurred.

CAUTION

This register is retained until 1 is written to the overwrite error clear (ADCCnECR.OWEC bit). While the value of this register is retained, if an overwrite error occurs, the new error information is discarded.

26.3.26 ADCCnPER — Parity Error Register

This is a register that indicates a parity error. This register can be cleared by writing 1 to the parity error clear (ADCCnECR.PEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E4_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PE	—	PECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.33 ADCCnPER Register Contents

Bit Position	Bit Name	Function
7	PE	This bit indicates whether a parity error exists. 0: A parity error does not exist. 1: A parity error exists.
6	Reserved	When written, write the value after reset.
5 to 0	PECAP[5:0]	These bits indicate the number of the virtual channel at the time when a parity error occurred.

CAUTION

This register is retained until 1 is written to the parity error clear (ADCCnECR.PEC bit). While the value of this register is retained, if a parity error occurs, the new error information is discarded.

26.3.27 ADCCnIDER — ID Error Register

This is a register that indicates an ID error. This register can be cleared by writing 1 to the ID error clear (ADCCnECR.IDEC bit).

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + $3E8_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	IDE	—	IDECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.34 ADCCnIDER Register Contents

Bit Position	Bit Name	Function
7	IDE	This bit indicates whether an ID error exists. 0: An ID error does not exist. 1: An ID error exists
6	Reserved	When written, write the value after reset.
5 to 0	IDECAP[5:0]	These bits indicate the number of the virtual channel at the time when an ID error occurred.

CAUTION

This register is retained until 1 is written to the ID error clear (ADCCnECR.IDEC bit). While the value of this register is retained, if an ID error occurs, the new error information is discarded.

26.3.28 ADCCnSGSTCRx — Scan Group x Start Control Register

This is a register that controls the start of A/D conversion for SGx.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 480_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SGST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.35 ADCCnSGSTCRx Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	SGST	This bit starts A/D conversion for SGx. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Start A/D conversion when the A/D conversion start scan group status (ADCCnSGSRx.SGACT) is 0.

26.3.29 ADCCnADTSTCRy — A/D Timer y Start Control Register

This is a register that controls the start of A/D timer y.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + y × 80_H + 488_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.36 ADCCnADTSTCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADTST	This bit starts A/D timer y. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Starts the A/D timer. Start the A/D timer when the A/D timer status (ADCCnSGSRx.ADTACT) is 0.

26.3.30 ADCCnADTENDCRy — A/D Timer y End Control Register

This is a register that controls the end of A/D timer y.

Access: This register can be written in 8-bit units.

Address: <ADCCn_base> + y × 80_H + $48C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTEND
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.37 ADCCnADTENDCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When written, write the value after reset.
0	ADTEND	This bit ends the A/D timer. 0: Nothing occurs. (A written value of 0 is ignored.) 1: Ends the A/D timer.

26.3.31 ADCCnSGCRx — Scan Group x Control Register

This is a register that controls SGx.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 490_H

Value after reset: 00_H

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	ADSTARTE	SCANMD	ADIE	—	—	—	TRGMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R/W

Table 26.38 ADCCnSGCRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7	Reserved	When written, write the value after reset.
6	ADSTARTE	This bit sets whether the synchronization start signal of SG is enabled or disabled. 0: Disabled 1: Enabled
5	SCANMD	This bit sets scan mode. 0: Multicycle scan mode 1: Continuous scan mode
4	ADIE	This bit sets whether the output of the scan group x end interrupt signal ADInx is allowed or prohibited. 0: Output prohibited. 1: Output allowed.
3 to 1	Reserved	When written, write the value after reset.
0	TRGMD0	This bit sets whether A/D conversion start trigger input to scan group x is enabled or disabled. 0: Disabled 1: Enabled

CAUTION

An A/D conversion start trigger input to scan group x that occurs while A/D conversion is being performed for scan group x is ignored.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	ADTSTARTE	ADSTARTE	SCANMD	ADIE	—	—	TRGMD1	TRGMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Table 26.39 ADCCnSGCRx Register Contents (x = 3 or 4)

Bit Position	Bit Name	Function
7	ADTSTARTE	This bit sets whether the synchronization start signal of A/D timers is enabled or disabled. 0: Disabled 1: Enabled
6	ADSTARTE	This bit sets whether the synchronization start signal of SG is enabled or disabled. 0: Disabled 1: Enabled
5	SCANMD	This bit sets scan mode. 0: Multicycle scan mode 1: Continuous scan mode
4	ADIE	This bit sets whether the output of the scan group x end interrupt signal ADInx is allowed or prohibited. 0: Output prohibited. 1: Output allowed.
3, 2	Reserved	When written, write the value after reset.
1	TRGMD1	This bit sets whether trigger input to A/D timers is enabled or disabled. 0: Disabled 1: Enabled
0	TRGMD0	This bit sets whether A/D conversion start trigger input to scan group x is enabled or disabled. 0: Disabled 1: Enabled

26.3.32 ADCCnSGVCSPx — Scan Group x Start Virtual Channel Pointer

This is a register that specifies the start pointer of a virtual channel.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 494_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	VCSP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.40 ADCCnSGVCSPx Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When written, write the value after reset.
5 to 0	VCSP[5:0]	These bits set the start virtual channel number of SGx.

CAUTIONS

1. ADCCnSGVCSPx must be equal or less than ADCCnSGVCEPx.
2. Do not make a setting greater than the number of virtual channels that are included.

26.3.33 ADCCnSGVCEPx — Scan Group x End Virtual Channel Pointer

This is a register that specifies the end pointer of a virtual channel.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + 498_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	VCEP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.41 ADCCnSGVCEPx Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	When written, write the value after reset.
5 to 0	VCEP[5:0]	These bits set the end virtual channel number of SGx.

CAUTIONS

1. ADCCnSGVCSPx must be equal or less than ADCCnSGVCEPx.
2. Do not make a setting greater than the number of virtual channels that are included.

26.3.34 ADCCnSGMCYCRx — Scan Group x Multicycle Register

This is a register that specifies the number of times for A/D conversion in multicycle scan mode.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $49C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	MCYC[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.42 ADCCnSGMCYCRx Register Contents

Bit Position	Bit Name	Function
7 to 0	MCYC[7:0]	These bits set the frequency of A/D conversion in multicycle scan mode. Number of times for A/D conversion = MCYC[7:0] + 1

26.3.35 ADCCnSGSRx — Scan Group x Status Register

This is a register that indicates the A/D conversion operation status of SGx.

Access: This register can be read in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $4A4_H$

Value after reset: 00_H

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.43 ADCCnSGSRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7 to 2	Reserved	When written, write the value after reset.
1	SGACT	This bit indicates the A/D conversion operation status of SGx. 0: A/D conversion for SGx is in idle state. 1: A/D conversion for SGx is running.
0	Reserved	When written, write the value after reset.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADTACT	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.44 ADCCnSGSRx Register Contents (x = 3 or 4)

Bit Position	Bit Name	Function
7 to 3	Reserved	When written, write the value after reset.
2	ADTACT	This bit indicates the operation status of an A/D timer. 0: A/D timer x is in idle state. 1: A/D timer x is running.
1	SGACT	This bit indicates the A/D conversion operation status of SGx. 0: A/D conversion for SGx is in idle state. 1: A/D conversion for SGx is running.
0	Reserved	When written, write the value after reset.

26.3.36 ADCCnADTIPRy — A/D Timer Initial Phase Register y

This is a register that sets the initial phase (initial value of the counter) of A/D timer y.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + y × 80_H + $4A8_H$

Value after reset: 0000 0000_H

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

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

Table 26.45 ADCCnADTIPRy Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	When written, write the value after reset.
20 to 0	ADTIP[20:0]	These bits set the initial phase (initial value of the counter) of A/D timer y.

CAUTION

Set this register before starting an A/D conversion and A/D timer.

26.3.37 ADCCnADTPRRy — A/D Timer Cycle Register y

This is a register that sets the cycle of A/D timer y.

Access: This register can be read or written in 32-bit units.

Address: <ADCCn_base> + y × 80_H + $4AC_H$

Value after reset: 001F FFFF$_H$

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

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

Table 26.46 ADCCnADTPRRy Register Contents

Bit Position	Bit Name	Function
31 to 21	Reserved	When written, write the value after reset.
20 to 0	ADTPR[20:0]	These bits set the cycle of A/D timer y.

CAUTION

Set this register before starting an A/D conversion and A/D timer.

26.3.38 ADCCnULLMSRx — Scan Group x Upper Limit/Lower Limit Table Select Register

This is a register that selects the upper-limit/lower-limit table of SGx.

Access: This register can be read or written in 8-bit units.

Address: <ADCCn_base> + x × 80_H + $4B0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ULS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 26.47 ADCCnULLMSRx Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When written, write the value after reset.
1, 0	ULS[1:0]	These bits select the upper-limit/lower-limit table. 0: Neither upper limit nor lower limit is checked. 1: Upper limit and lower limit are checked in ADCCnULLMTBR0. 2: Upper limit and lower limit are checked in ADCCnULLMTBR1. 3: Upper limit and lower limit are checked in ADCCnULLMTBR2.

26.4 Function

26.4.1 Method of A/D Conversion

A/D conversion is performed for each scan group. Trigger signals for A/D conversion are prepared for the number of scan groups. When a trigger signal (SGx_TRG) is input, A/D conversion is performed for the signals of virtual channels assigned to the scan group in ascending order. A/D conversion for each virtual channel is completed or A/D conversion for all virtual channels assigned to the scan group is completed, an A/D completion interrupt (ADInx) occurs.

The multiscan mode repeats A/D conversion for the specified number of times when a trigger signal is input. The continuous scan mode repeats A/D conversion unlimitedly when a trigger signal is input.

The signed fixed-point format or signed integer format can be selected as the data format for A/D conversion. For details about the bit alignment in the data format, see **Section 26.3.5, ADCCnDRj — Data Register j**.

The following figure shows an operation example of conversion.

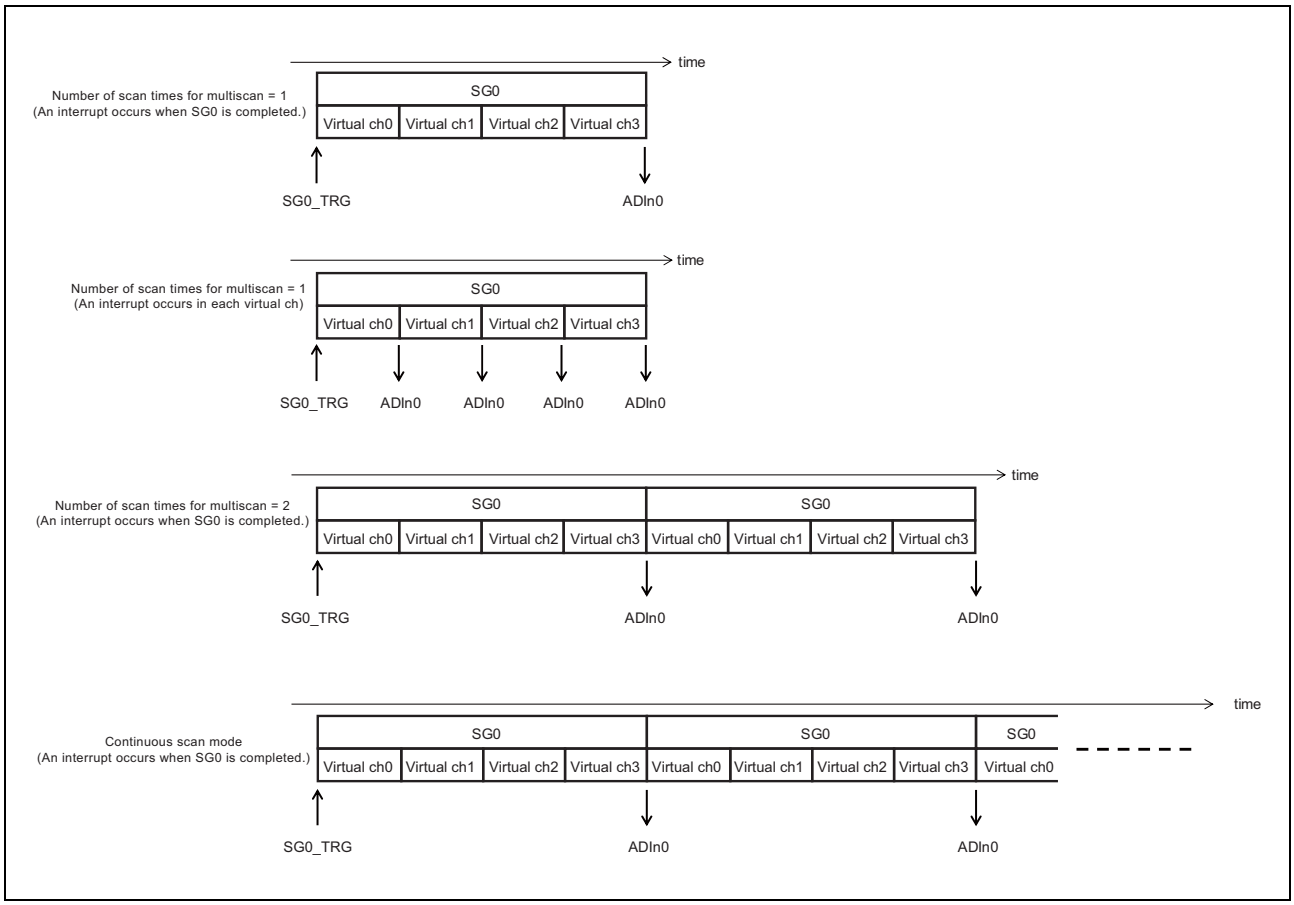


Figure 26.7 Operation Example of A/D Conversion

26.4.2 A/D Conversion Function

26.4.2.1 Normal A/D Conversion Function

The normal A/D conversion function performs A/D conversion for the analog signals of a physical channel.

26.4.2.2 Simultaneous track and hold function

This function holds multiple analog signals and performs A/D conversion for them. Analog signals to be simultaneously held can be assigned to two groups, T&H groups A and B, and can be held in a different timing for each group.

A physical channel that has a T&H circuit is assigned to a virtual channel and scan group, and the scan group is assigned to a T&H group. Note that SG0 cannot be assigned to a T&H group.

The method of T&H hold has the automatic sampling mode. This mode starts sampling automatically when A/D conversion for held analog signals is completed. If you do not specify the automatic sampling mode, you need to perform sampling by software.

The following figure is an operation example of the simultaneous track and hold function.

As shown in **Figure 26.8**, if a trigger signal (hold trigger A) for SG2 is input during A/D conversion, the virtual channel for which A/D conversion is currently being performed is forcibly stopped. After the virtual channel is stopped and analog signals become stable, the analog signals are held by the T&H circuit. After the hold operation is completed, A/D conversion resumes according to the priority of the scan groups.

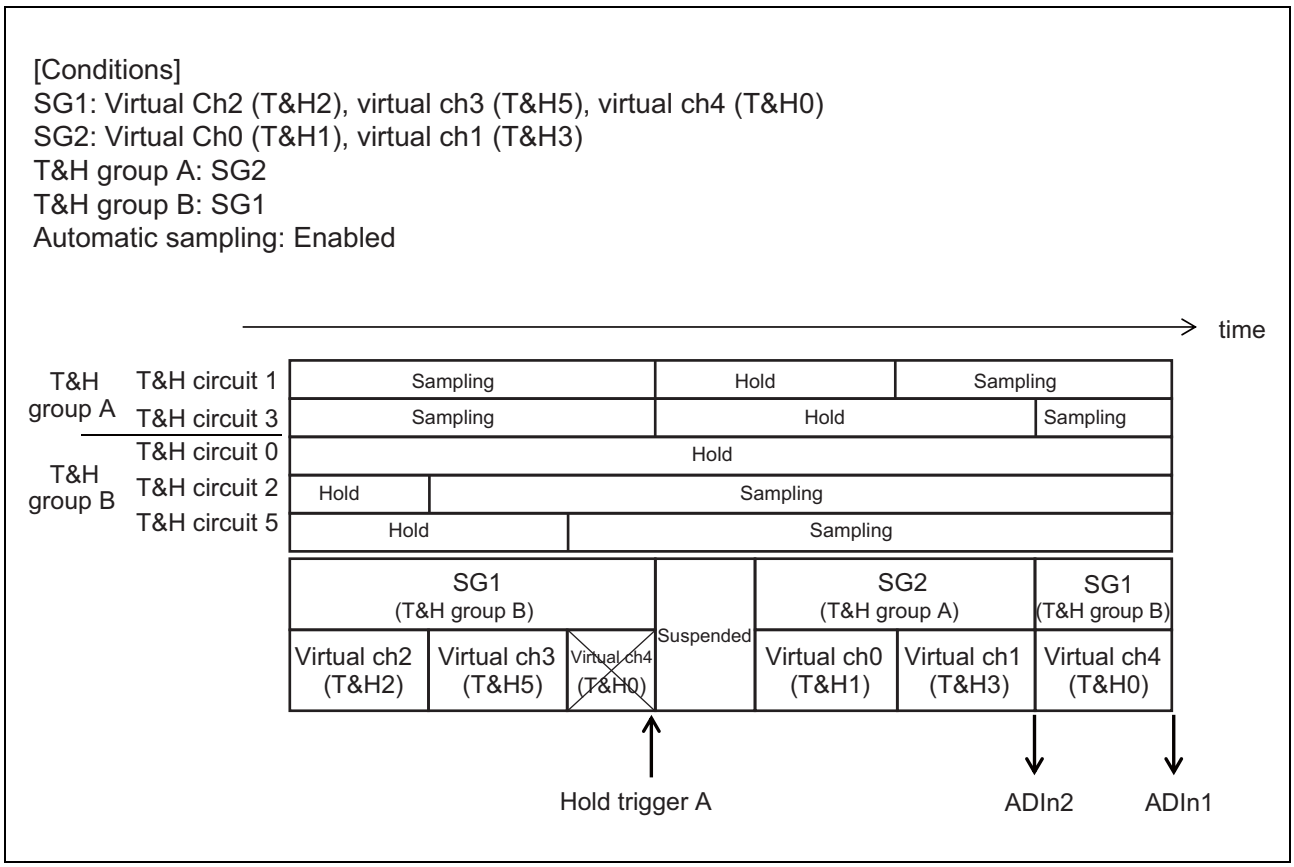


Figure 26.8 Operation Example of Simultaneous Track and Hold (High-Priority SG Hold Trigger Input)

For example, if the A/D conversion that is forcibly stopped is for SG3 as shown in **Figure 26.9**, A/D conversion for SG3 is first performed and then A/D conversion for SG2 is performed because SG3 has a higher priority.

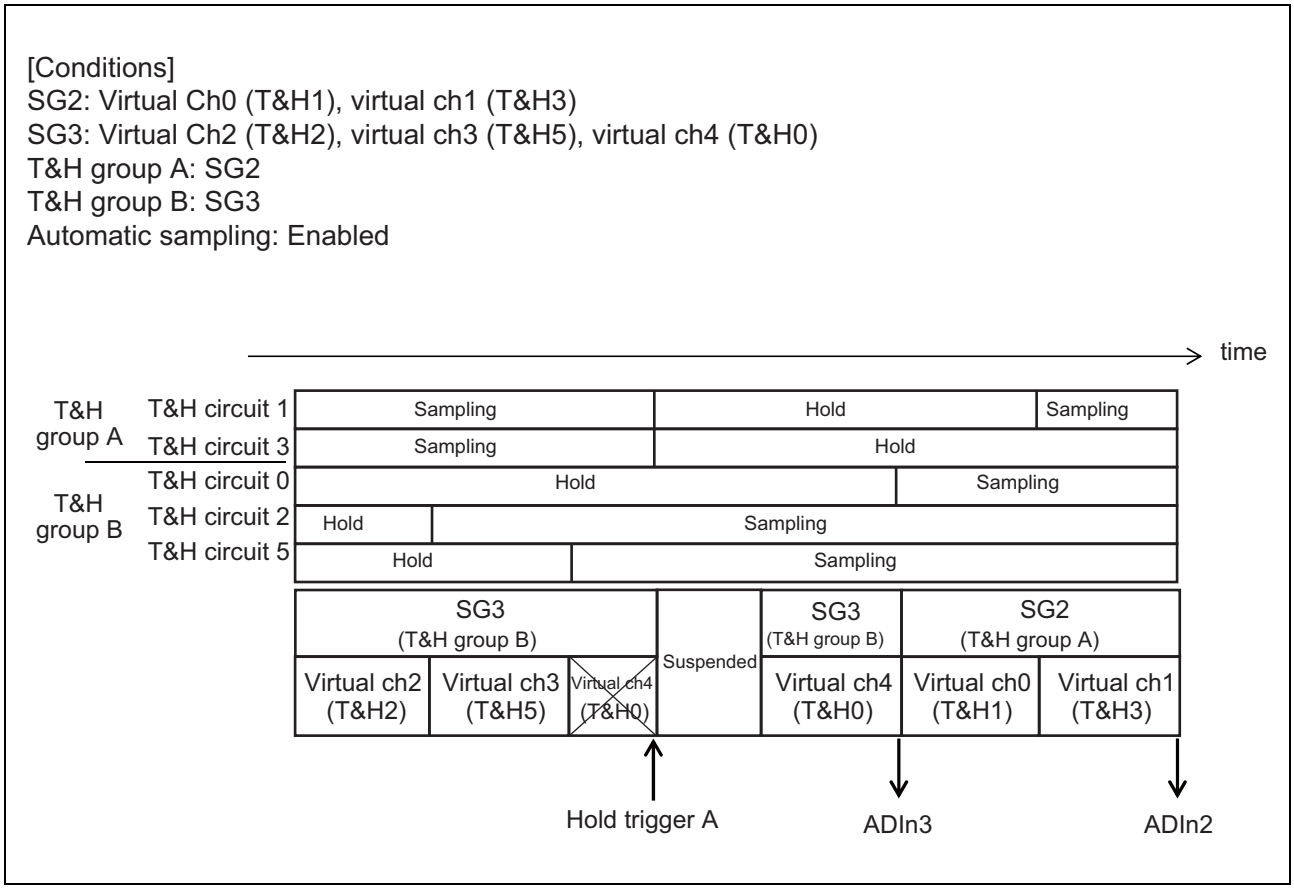


Figure 26.9 Operation Example of Simultaneous Track and Hold (Low-Priority SG Hold Trigger Input)

Also, when another trigger signal (hold trigger B) is input while A/D conversion is being forcibly stopped as shown in **Figure 26.9**, the stop time becomes longer than that in the case in which one trigger signal is input because of the waiting time until analog signals assigned to SG1 (T&H group B) are held.

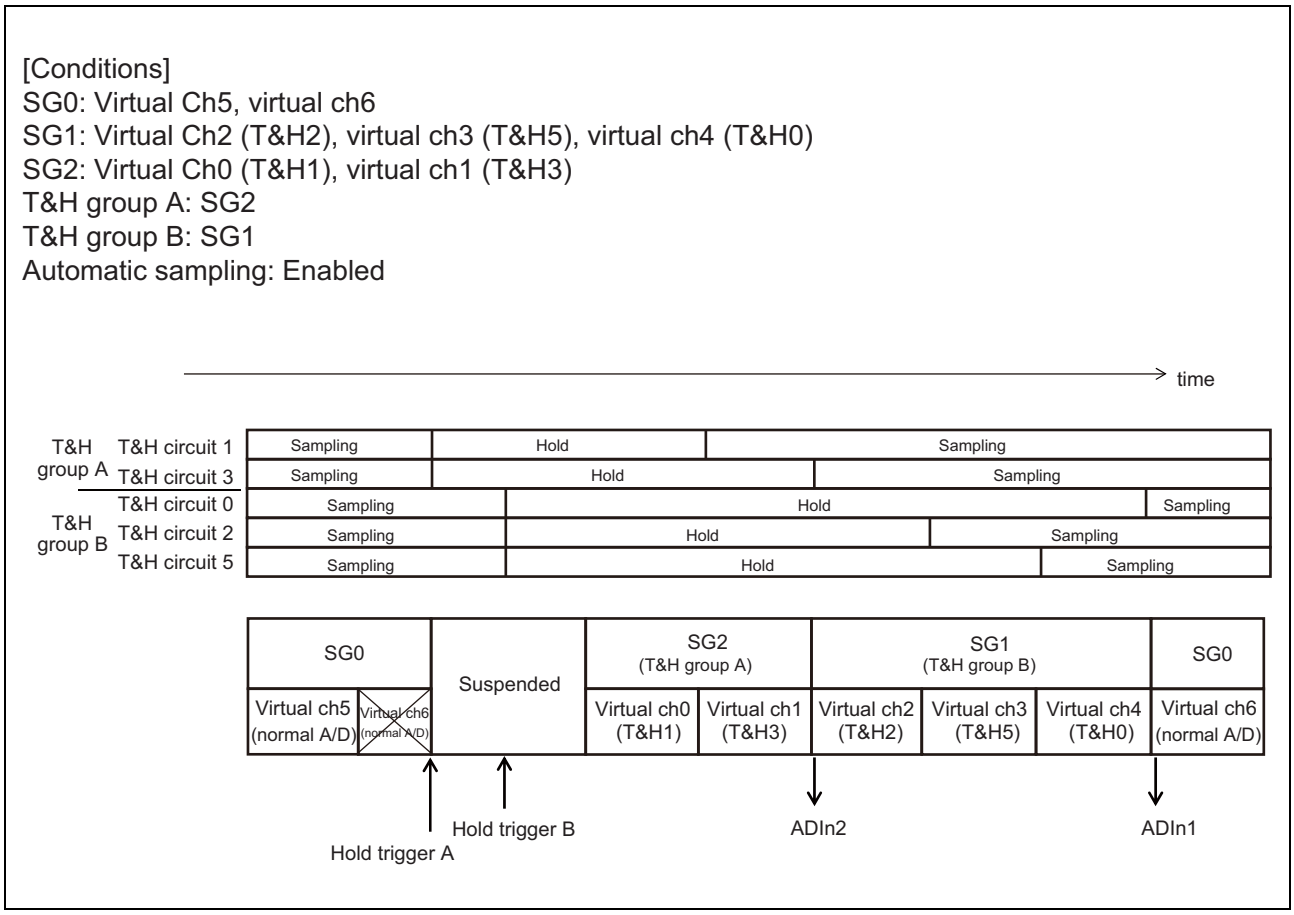


Figure 26.10 Operation Example of Simultaneous Track and Hold (Continuous Hold Trigger Input)

26.4.2.3 Addition A/D Conversion Function

This function performs A/D conversion two or four times in a row for analog signals of a physical channel and stores the added value in a data register. The addition count (two or four times) is common to all virtual channels.

26.4.2.4 Multicycle Scan Mode

This mode repeats A/D conversion for the specified number of times (1 to 256 times) for virtual channels assigned to a target SGx when a trigger is input.

26.4.2.5 Continuous Scan Mode

This mode repeats A/D conversion unlimitedly for virtual channels assigned to a target SGx when a trigger is input.

If you want to stop the continuous scan mode, stop A/D conversion according to the procedure for stopping A/D conversion. For details about the A/D conversion stop procedure, see **Section 26.5.3, Procedure for Stopping A/D Conversion**. You can stop all SGx and A/D timers by using the A/D halt register (ADCCnADHALTR register).

If a trigger for a lower-priority scan group is input to a scan group that is set to the continuous scan mode, the trigger is not accepted. Therefore, it is assumed that the continuous scan mode is set for SG0, which has the lowest priority

26.4.3 Trigger function

26.4.3.1 Input Selection of Triggers for Scan Groups

The following triggers can be selected as an A/D conversion start trigger for each scan group. Disable trigger input for unused scan groups.

Table 26.48 List of Trigger Supports

	HW Trigger		SW Trigger		
	SGx Trigger (SGx_TRG)	A/D Timer Trigger (SGy_TRG)	A/D Conversion Trigger	Hold Trigger	A/D Timer Trigger
SG0	Y	N	Y	N	N
SG1, SG2	Y	N	Y	Y	N
SG3, SG4	Y	Y	Y	Y	Y

Note: Y: Supported, N: Not supported

26.4.3.2 Starting Scan Groups by HW triggers

HW triggers include SGx triggers and A/D timer triggers.

(1) Starting Scan Groups by SGx Triggers

A/D conversion can be started by starting SGx by a SGx_TRG signal. The simultaneous track and hold function holds analog signals by a hold trigger and then performs A/D conversion for them.

A SGx_TRG signal is input from PIC2B. PIC2B controls trigger signals from each unit or pin with masks and outputs them as SGx_TRG signals. For details about trigger factors, see **Section 23.3.3.1, ADCC Trigger Select Function**.

(2) Starting Scan Groups by A/D Timer Triggers

A SGy_TRG signal starts an A/D timer (free run) and an A/D timer trigger signal is output for each underflow of the counter as shown in **Figure 26.11**. This A/D timer trigger signal allows A/D conversion for SGy to start at regular intervals. However, When you use the simultaneous track and hold function, A/D timer trigger cannot be used.

An A/D timer is counted by clkad.

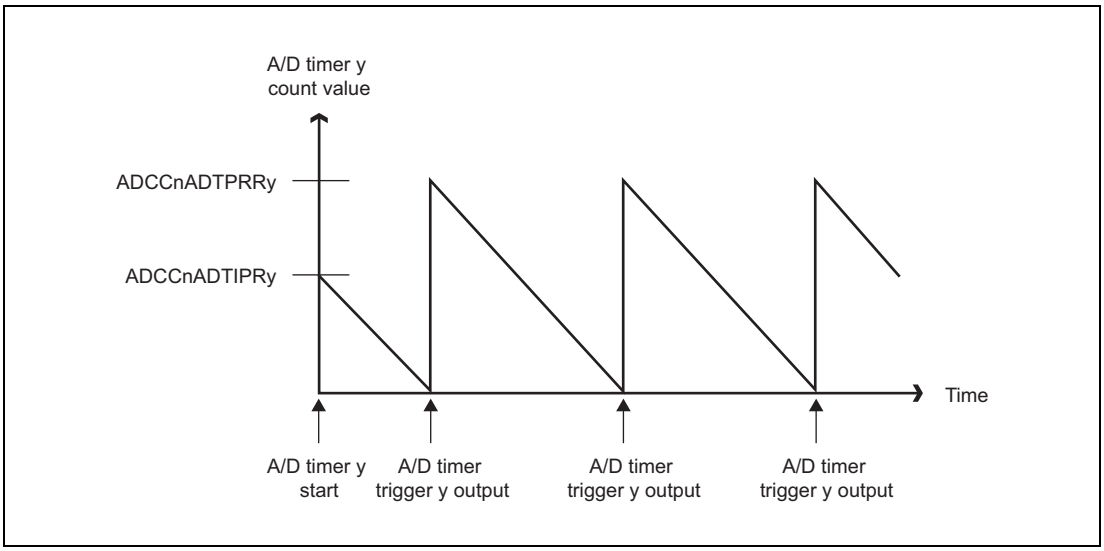


Figure 26.11 Operation Example of an A/D Timer

CAUTION

If you set the A/D timer initial phase register y (ADCCnADTIPRy register) to 0, an A/D timer trigger is output simultaneously with the start of an A/D timer. Also, if you set the A/D timer cycle register (ADCCnADTPRRy register) to 0, an A/D timer trigger is output for each clock.

26.4.3.3 Starting Scan Groups by SW triggers

SW triggers include A/D conversion triggers, hold triggers, and A/D timer triggers. Note that before enabling each trigger, you should confirm that the target SG trigger is disabled and the target scan group is stopped.

(1) A/D Conversion Trigger

This trigger function can simultaneously start multiple SGx in ADCC0 and ADCC1 and start A/D conversion as shown in **Figure 26.12, Functional Diagram of the SW Trigger Function**. Also, this trigger function can start A/D conversion for SGx individually.

(2) A/D Timer Trigger

This trigger function can simultaneously start multiple A/D timer y in ADCC0 and ADCC1 as shown in **Figure 26.12, Functional Diagram of the SW Trigger Function**. Also, this trigger function can start A/D timer y individually.

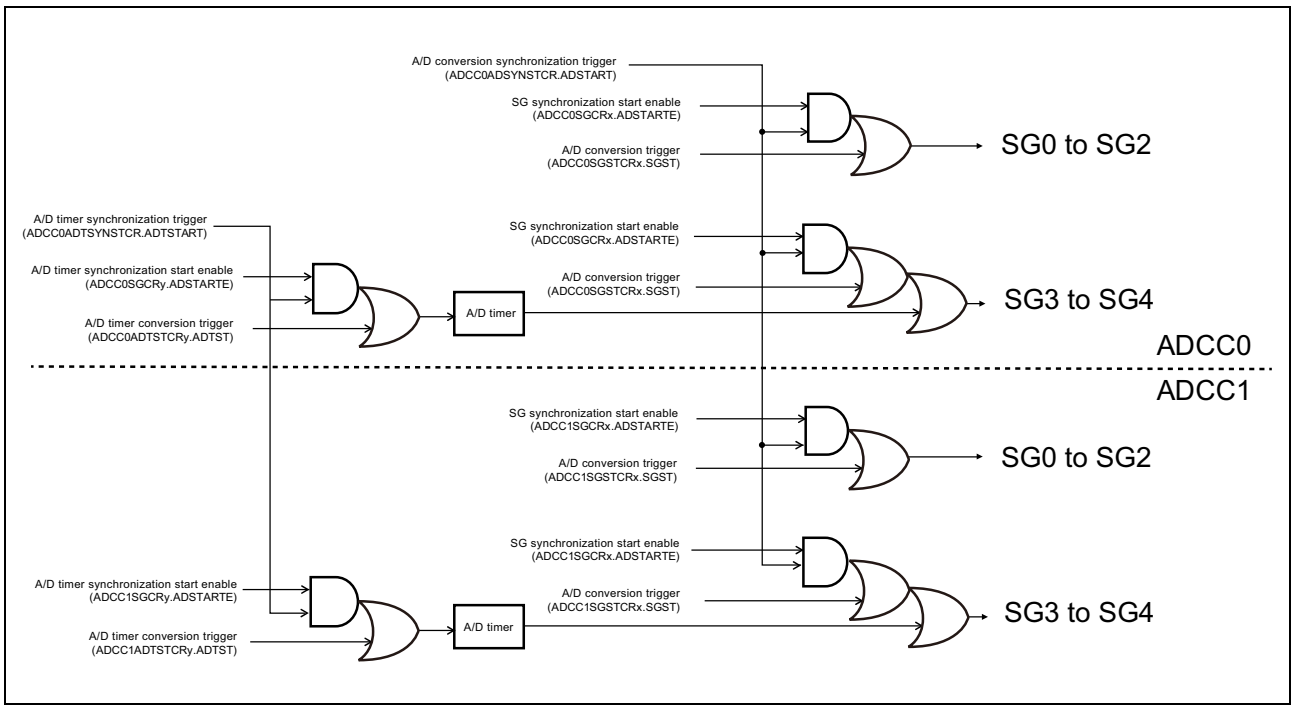


Figure 26.12 Functional Diagram of the SW Trigger Function

(3) Hold Trigger

This trigger function can start hold at any timing as shown in **Figure 26.13, Functional Diagram of the Hold Trigger Function**. Make sure that you perform 30 clkad or more samplings before starting hold.

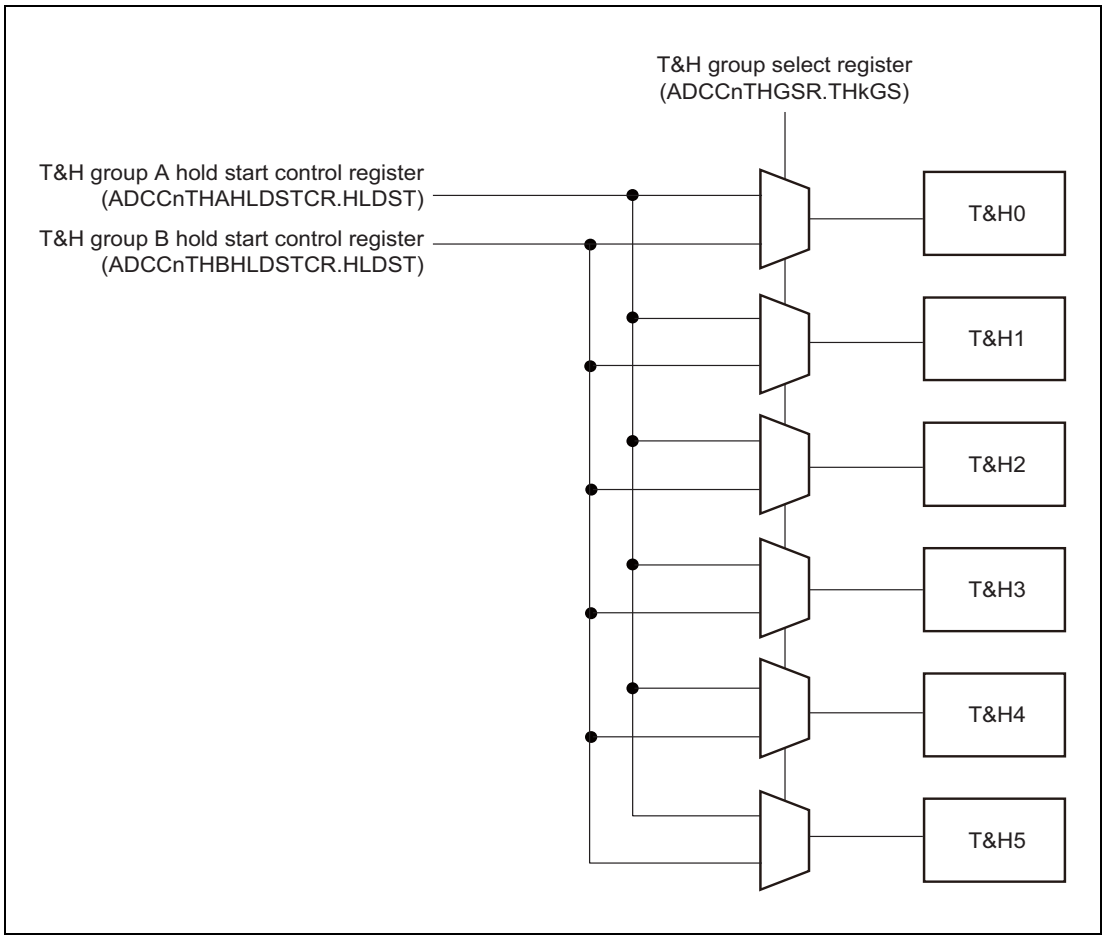


Figure 26.13 Functional Diagram of the Hold Trigger Function

26.4.4 Suspend Function

When a higher-priority scan group request is received during the processing of a lower-priority scan group, the suspend function suspends the lower-priority A/D conversion and performs the higher-priority A/D conversion. There are three types of suspend operation.

26.4.4.1 Synchronous Suspend Operation

If an A/D conversion trigger for a higher-priority scan group than the scan group for which A/D conversion is being performed occurs, after the conversion of the virtual channel for which A/D conversion is being performed is completed, A/D conversion for the higher-priority scan group is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion resumes from the suspended virtual channel.

The following figures show operation examples.

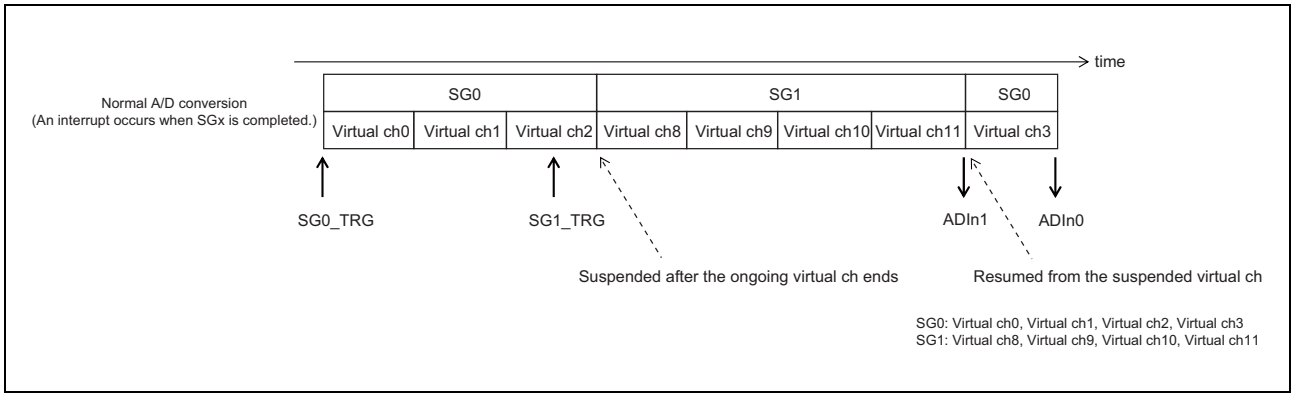


Figure 26.14 Operation Example of Synchronous Suspend (Normal A/D Conversion)

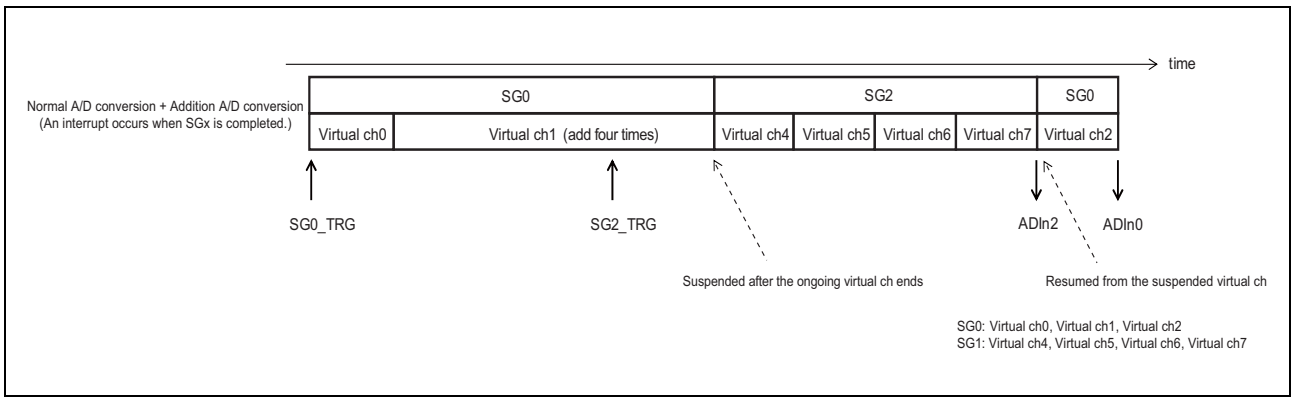


Figure 26.15 Operation Example of Synchronous Suspend (Normal A/D Conversion + Addition A/D Conversion)

26.4.4.2 Asynchronous Suspend Operation

If an A/D conversion trigger for a higher-priority scan group than the scan group for which A/D conversion is being performed occurs, the conversion for the virtual channel for which A/D conversion is being performed is immediately suspended, and A/D conversion for the higher-priority scan group is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion for the suspended virtual channel starts from the beginning.

The following figures show operation examples.

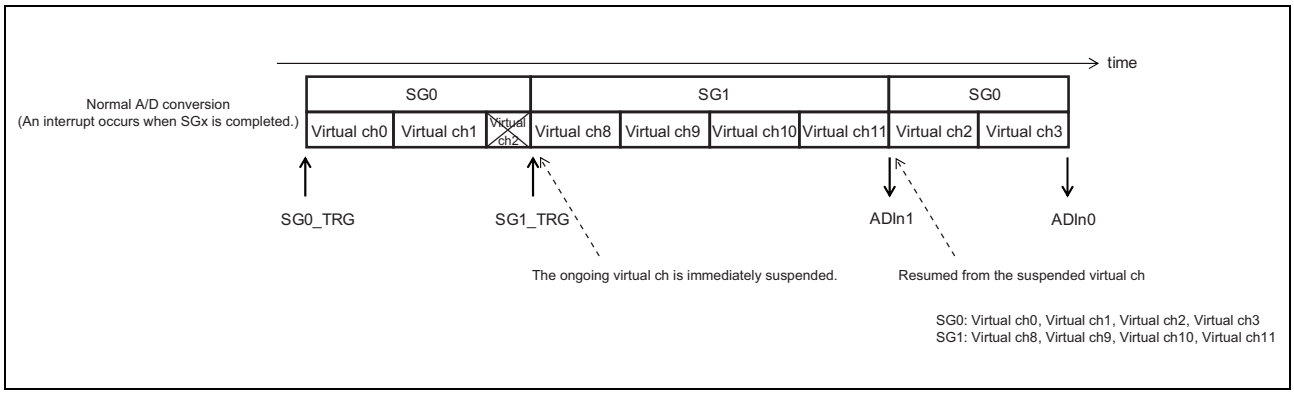


Figure 26.16 Operation Example of Asynchronous Suspend (Normal A/D Conversion)

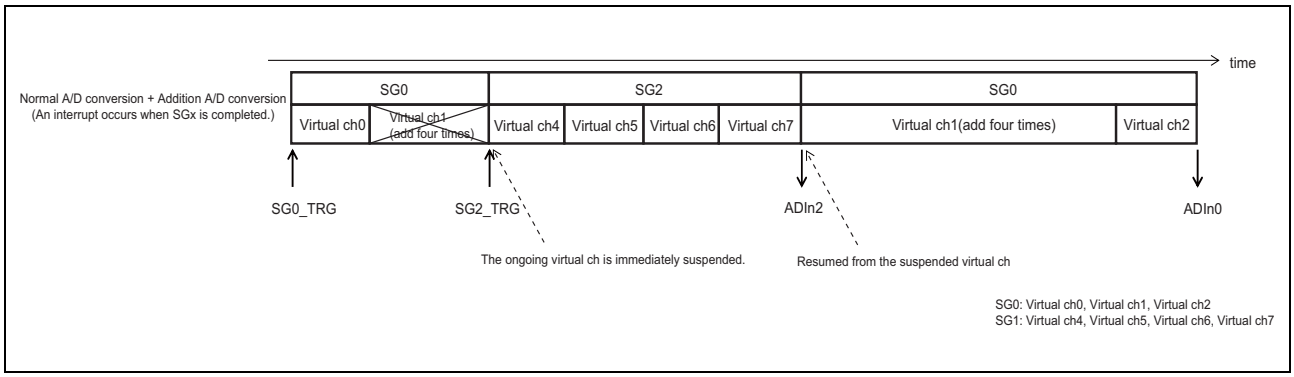


Figure 26.17 Operation Example of Asynchronous Suspend (Normal A/D Conversion + Addition A/D Conversion)

26.4.4.3 Synchronous and Asynchronous Combination Suspend Operation

If an A/D conversion trigger for a higher-priority scan group occurs during A/D conversion for SG0, asynchronous suspend operation is performed. If an A/D conversion trigger for a higher-priority scan group occurs during A/D conversion for a scan group other than SG0, synchronous suspend operation is performed. After the A/D conversion for the higher-priority scan group is completed, A/D conversion for the suspended virtual channel starts from the beginning.

The following figure shows an operation example.

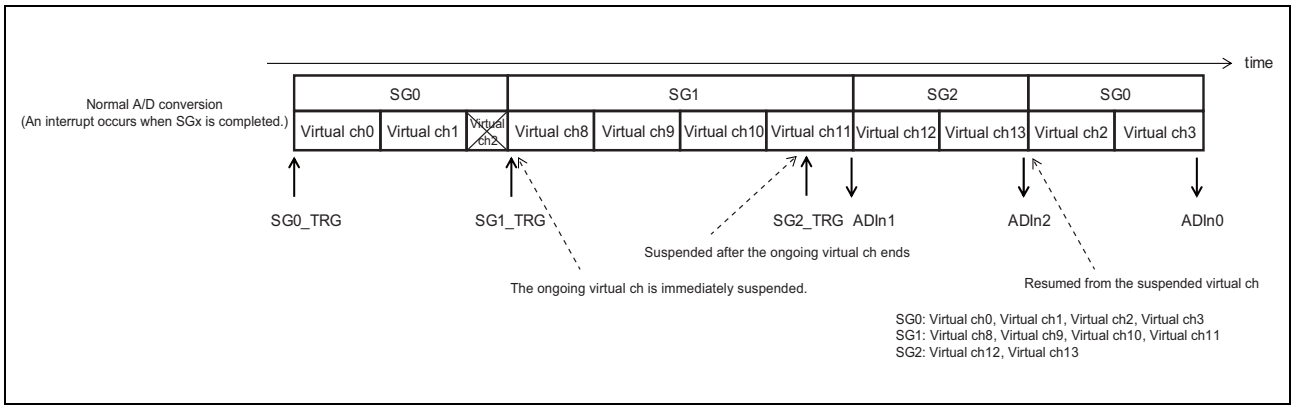


Figure 26.18 Operation Example of Synchronous and Asynchronous Combination Suspend

26.4.5 Interrupt Request Function

This function includes scan group x end interrupts and A/D error interrupts. An interrupt request signal is a pulse. You can start DMA/DTS by a scan group x end interrupt.

Interrupt output can be masked. Even if interrupt output is masked, the status register is set to 1 (an interrupt occurs).

26.4.5.1 Scan Group X End Interrupt

An end interrupt can occur at the following timing.

1. An interrupt occurs for each virtual channel.
   After A/D conversion for a virtual channel is completed, an interrupt occurs.
2. An interrupt occurs for each scan group.
   After A/D conversion for all the virtual channel assigned to the scan group is completed, an interrupt occurs.

In multiscan mode, an interrupt occurs in each A/D conversion for scan groups for an one-time scan. Therefore, if you perform multiscan two times in the above setting 2, an end interrupt occurs two times.

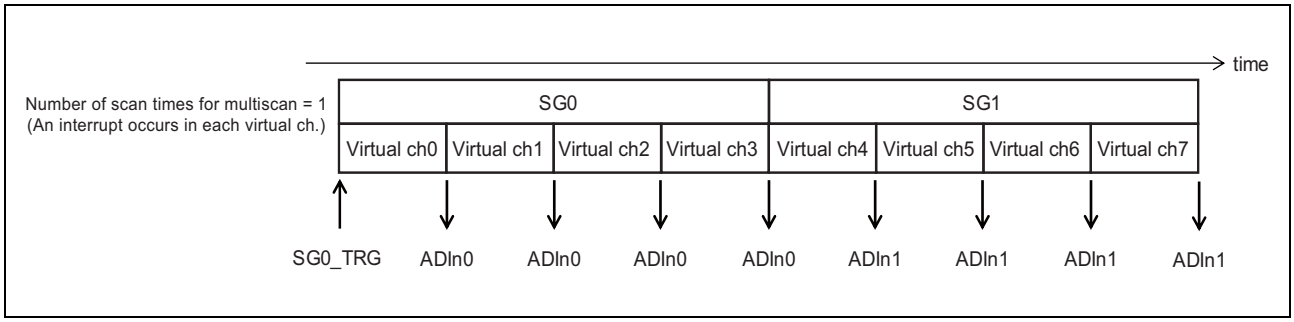


Figure 26.19 Scan Group X End Interrupt (An Interrupt Occurs for each Virtual Channel)

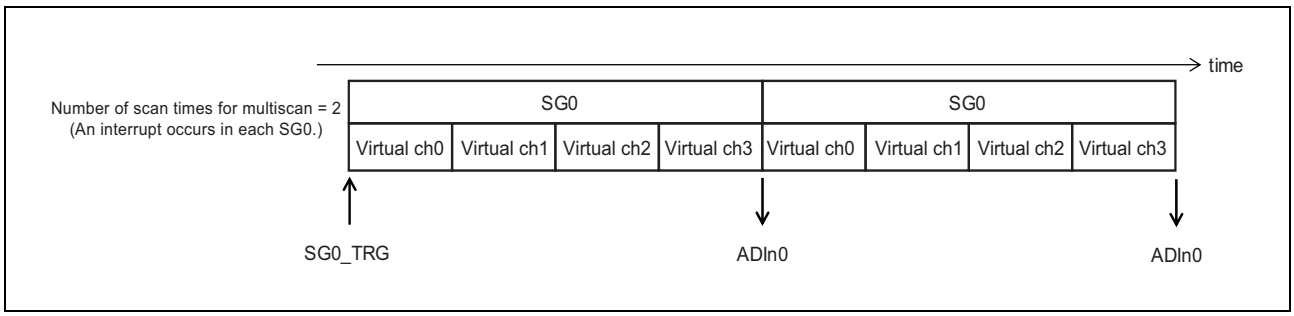


Figure 26.20 Scan Group X End Interrupt (An Interrupt Occurs for each SG)

26.4.5.2 A/D Error Interrupt Request

An A/D error interrupt can occur when the following A/D errors occur. The timing of the occurrence of A/D error interrupts is the same as that of end interrupts.

An A/D error interrupt request occurs when one of an upper-limit/lower-limit error, overwrite error, and an ID error occurs (OR condition). After an interrupt occurs, perform error status clear in the interrupt handler.

1. Upper-limit/lower-limit error
2. Overwrite error
3. ID error

(1) Upper-Limit/Lower-Limit Error

When an A/D converted value exceeds the specified upper-limit or lower-limit value, an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnULER.ULECAP[5:0] bits). You can select one from three tables for the upper-limit or lower-limit value.

An upper-limit/lower-limit error is judged by the addition result that is stored in the data register.

(2) Overwrite Error

When an A/D converted value has not been read (ADCCnDIRj.WFLG bit = 1) and the A/D converted value is updated (overwritten), an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnOWER.OWECAP[5:0] bits).

(3) ID Error

When a physical channel assigned to a virtual channel does not match with the physical channel for which conversion is actually performed, an error occurs. The number of the virtual channel in which an error occurs is retained in a register (ADCCnIDER.IDECAP[5:0] bits).

CAUTION

When an error has not been cleared, if the same error occurs again, the subsequent error information is discarded.

26.4.5.3 A/D Parity Error Trigger

A parity error interrupt can occur when a parity error occurs. A parity error trigger occurs when a data register (ADCCnDRj register) is read.

An A/D parity error trigger occurs by a parity error. An A/D parity error trigger is reported to the ECM.

When a data register is read, the contents are checked by the parity (ADCCnDIRj.PRTY bit) in the data supplementary information register. If a parity error is detected, the conversion is considered as an error. The number of the virtual channel in which an error occurs is retained in a register (ADCCnPER.PECAP[5:0] bits).

CAUTION

When an error has not been cleared, if the same error occurs again, the subsequent error information is discarded.

26.4.6 Function for Transferring Results of A/D Conversion to EMU

Each ADCC can output A/D conversion completion signals for virtual channels 0, 1, and 2, A/D converted data, and SG4 scan end signals to the EMU. An A/D conversion completion signal is a signal that indicates that the A/D conversion result value is stored in a data register and the A/D conversion is completed.

Data stored in data registers ADCCnDR0, 1, and 2 is output as A/D converted data. Note that the upper 4 bits of data to be output are fixed to 0 and the lower 12 bits are A/D converted data regardless of the data format setting (ADCCnADCR2.DFMT bit). Also, when the read and clear enable is enabled (ADCCnSFTCR.RDCLRE bit = 1), if the ADCCnDRj register or the ADCCnDIRj register is read, A/D converted data is cleared to 0000_H similarly to the ADCCnDR0, 1, and 2 registers.

CAUTION

When you use this function, virtual channels 0, 1, and 2 should be assigned to SG4. Specify the settings so that a scan group end interrupt occurs for each scan group (The settings in which an interrupt occurs for each virtual channel are prohibited).

Do not use this function in the addition A/D conversion settings (ADCCnVCRj.CNVCLS[2:0] bits = 4_H).

26.4.7 Self-Diagnostic Functions

Each ADCC has the following self-diagnostic functions. The self-diagnostic function compares the conversion result with the expected value and confirms whether the result is as expected.

- Pin-level self-diagnostic function
- A/D conversion circuit self-diagnostic function
- Wiring-break detection self-diagnostic circuit function

26.4.7.1 Pin-Level Self-Diagnostic Function

This is a function to check an abnormal path from pins. This function checks an abnormal path from pins by specifying an even physical subchannel and odd physical subchannel as a set, setting a different voltage by the pin-level self-diagnostic control register (ADCCnTDCR register), and then executing A/D conversion. As shown in **Figure 26.21**, disconnect ADCC from analog input pins, apply the specified voltage, and then perform A/D conversion.

You can use a combination of AVSS, AVCC, and 1/2 × AVCC for different voltage. Also, you can use any physical channel for diagnosis.

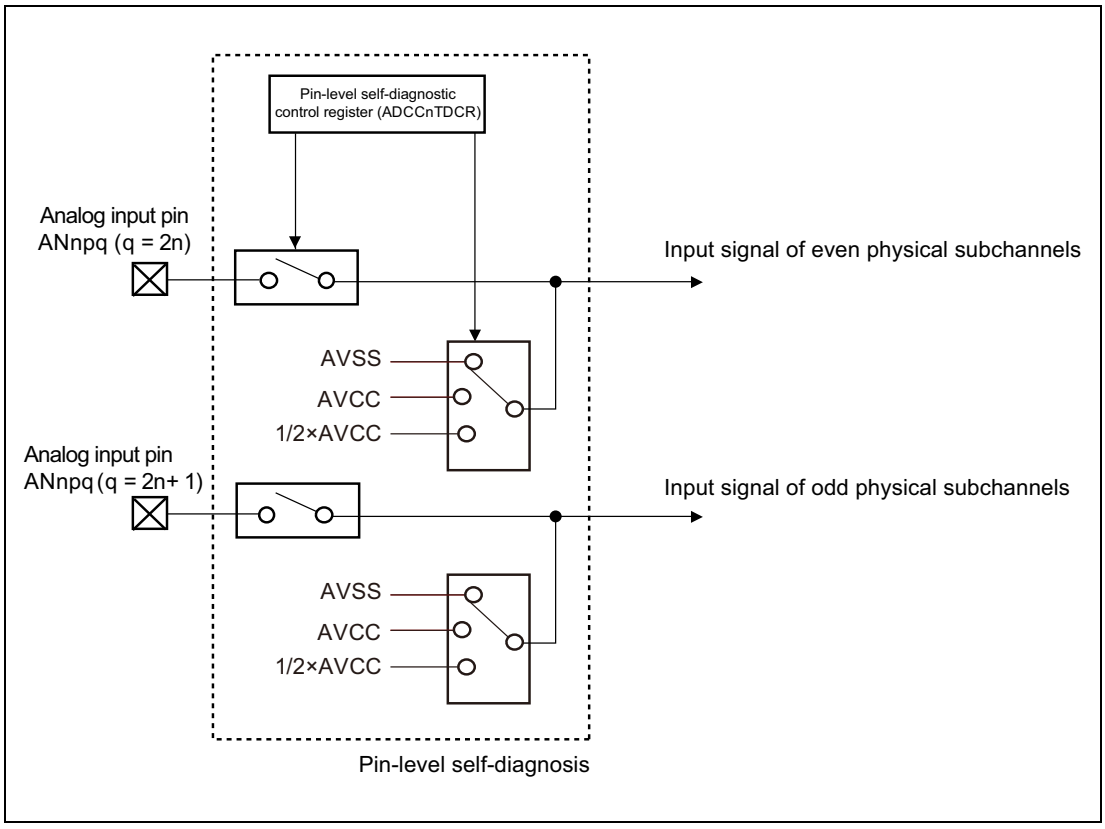


Figure 26.21 Functional Diagram of Pin-Level Self-Diagnosis

26.4.7.2 A/D conversion circuit self-diagnosis function

This is a function to input a self-diagnostic voltage level and check the A/D conversion circuit from the result of A/D conversion. You can select a self-diagnostic voltage level from AVREFH × 1, AVREFH × 3/4, AVREFH × 1/2, AVREFH × 1/4, and AVREFH × 0.

A/D conversion on the pin in use for self-diagnosis of the A/D conversion circuit proceeding beforehand may lead to the injection of current to the pin and thus affect the accuracy of A/D conversion. Therefore, select pins to which previous A/D conversion will not have led to the injection of current as the targets of A/D conversion for self- diagnosis.

The following shows the example of settings.

CAUTION

When the voltage applied to a pin exceeds the supplied power voltage or fall below the ground voltage, the current is being injected.

(1) When A/D conversion and A/D conversion circuit self-diagnosis are executed in the same SG, select the pin that the injection current is not applied to, before the A/D conversion circuit self-diagnosis.

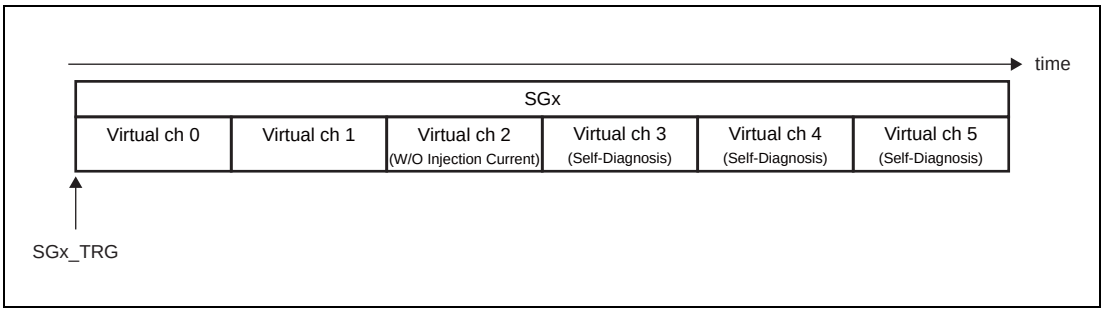


Figure 26.22 Examples of Setting for A/D Conversion and A/D Conversion Circuit Self-diagnosis in Same SG

(2) When a SG has the higher priority than the SG running A/D conversion circuit self-diagnosis, to run the A/D conversion of SG with high priority, after the A/D conversion circuit self-diagnosis is suspended, it may resume. In that case, convert the pin*1, that injection current is not applied to, at the end of SG with high priority.

Note 1. When unable to identify the pin which injection current is not applied to, apply the A/D conversion to a unused pin as a substitute.

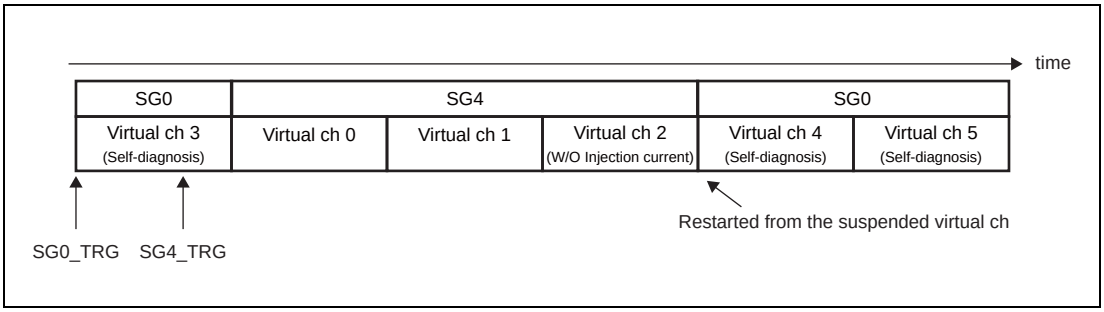


Figure 26.23 Examples of Setting for SG with Higher Priority than A/D Conversion Circuit Self-diagnosis

26.4.7.3 Wiring-break detection self-diagnosis circuit function

This is a function to detect a wiring-break in a pin due to solder separation.

Discharge the target analog pin for the specified time in the wiring-break detection control register (ADCCnODCR register) and then perform A/D conversion. If the conversion result attenuates to approximately 0 V, you can determine that a wiring-break is present.

26.5 Procedure

26.5.1 Procedure for Setting A/D Conversion

Figure 26.24 shows the flow of A/D conversion settings, and **Figure 26.25** shows the flow of initial settings. Perform the initial settings in a state in which trigger factors for all scan groups are disabled and all scan groups and T&H are stopped. If they are active, perform the A/D conversion stop settings.

Set the value after reset for the setting values of unused functions.

In the following flow, before starting A/D conversion, pin-level self-diagnosis is performed in a state in which A/D conversion is disabled, and then A/D conversion start settings are performed.

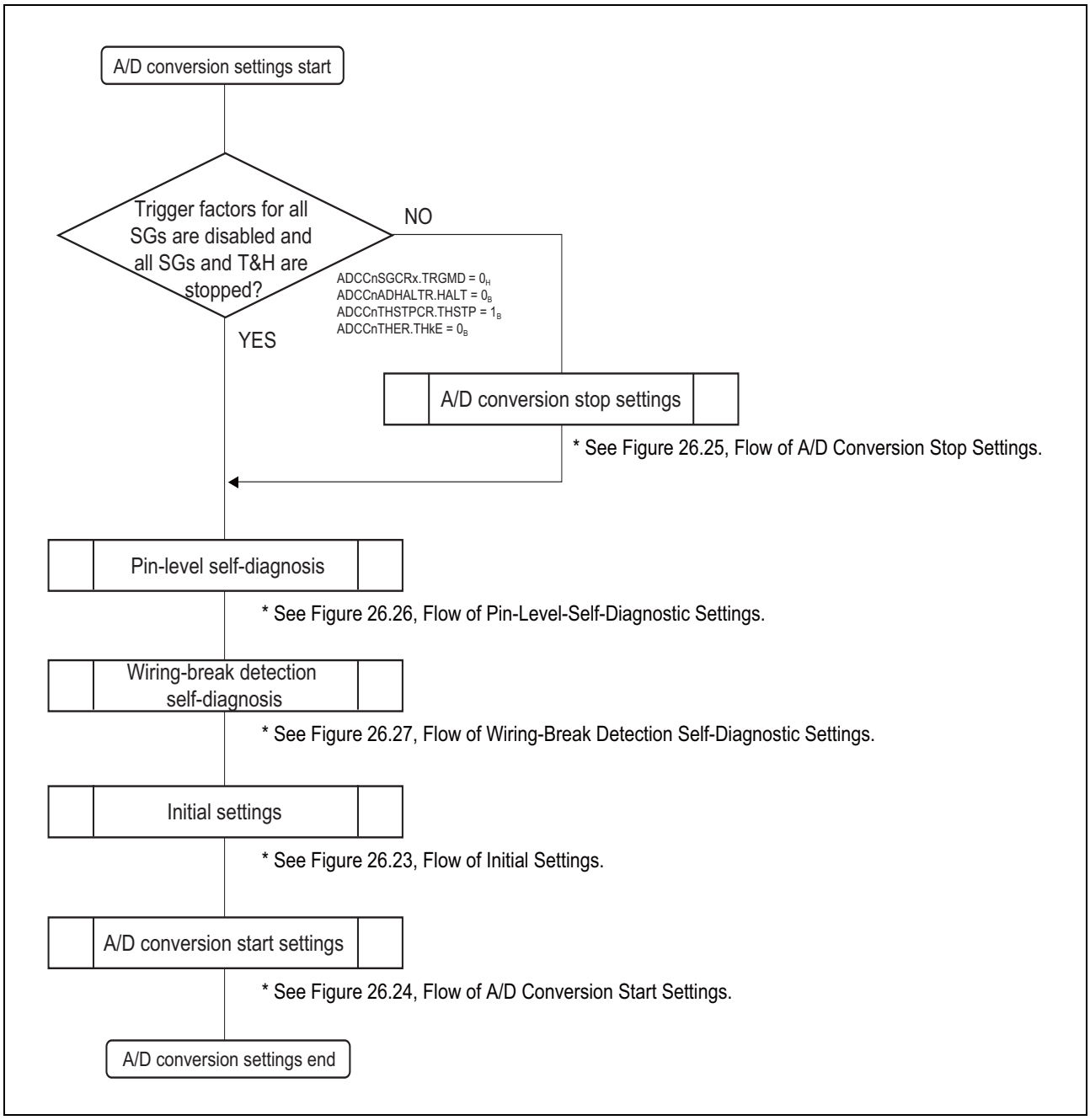


Figure 26.24 Flow of A/D Conversion Settings

The following initial settings flow is a flow for performing the basic settings of ADCC such as the A/D conversion mode settings.

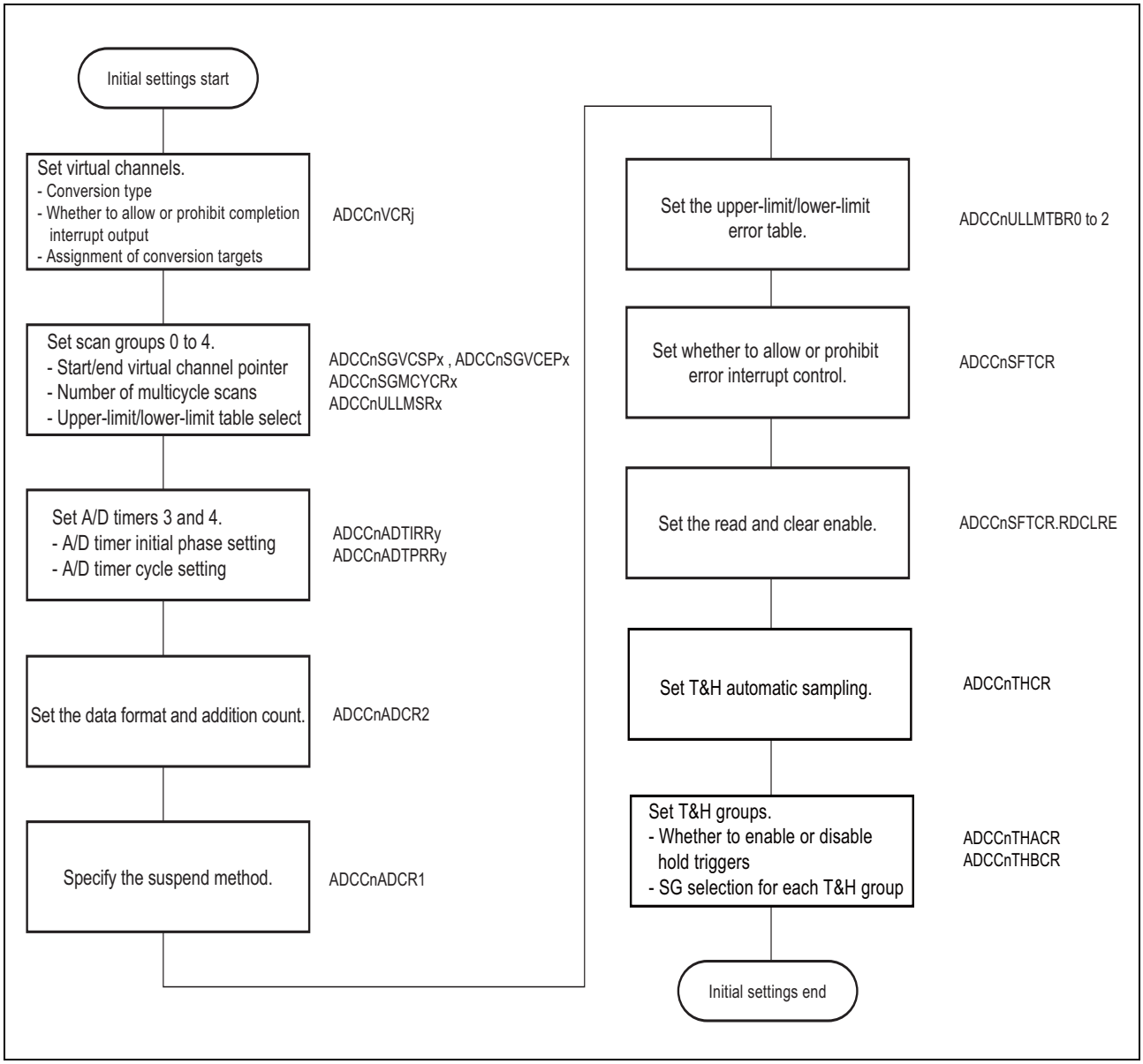


Figure 26.25 Flow of Initial Settings

26.5.2 Procedure for Starting A/D Conversion

Figure 26.26 shows the flow of A/D conversion start settings.

The following flow is a flow for starting A/D conversion operation by using a HW trigger. When you use the simultaneous track and hold function, you should perform hold after starting T&H sampling and then waiting 30 clkad or more samplings.

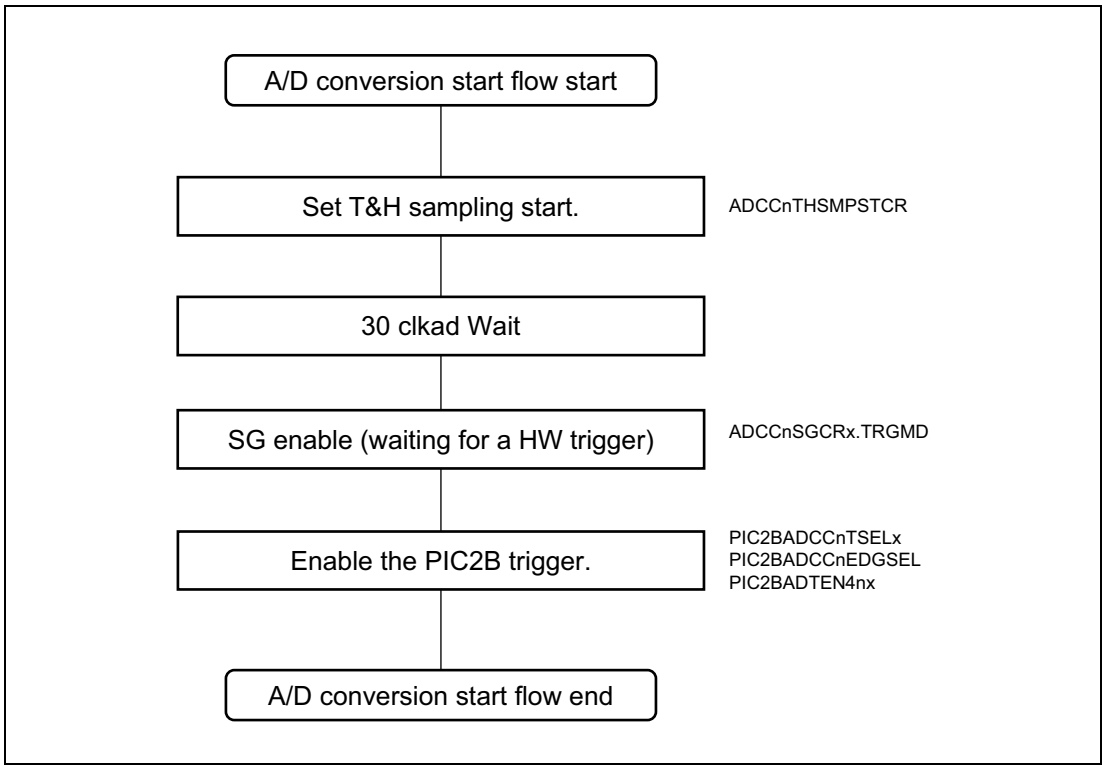


Figure 26.26 Flow of A/D Conversion Start Settings

26.5.3 Procedure for Stopping A/D Conversion

Figure 26.27 shows the flow of A/D conversion stop settings.

The following flow is a flow for stopping A/D conversion operation by disabling triggers for all scan groups and stopping operation of all scan groups and T&H.

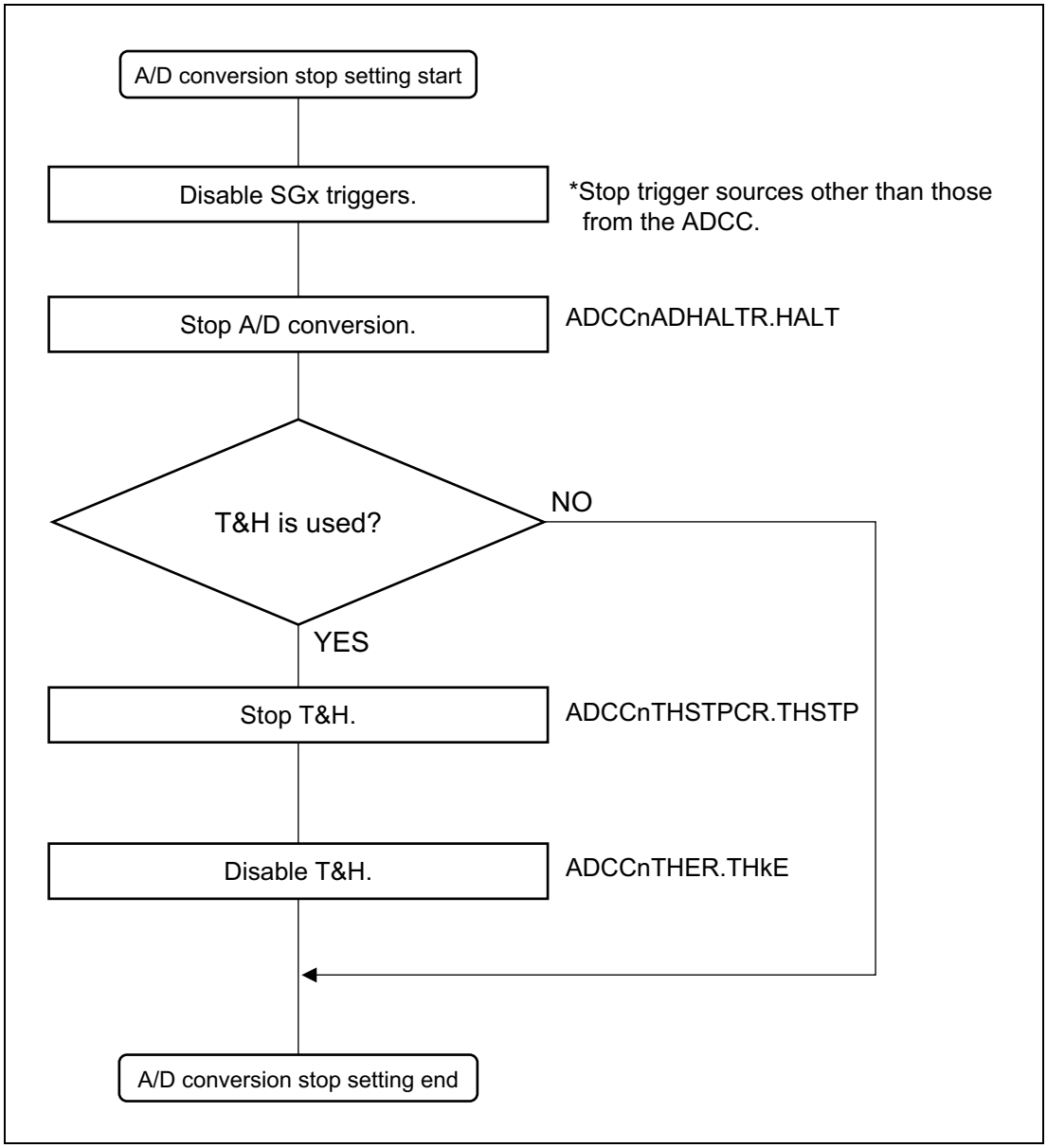


Figure 26.27 Flow of A/D Conversion Stop Settings

26.5.4 Procedure for Setting Pin-Level Self-Diagnosis

Figure 26.28 shows the flow of pin-level self-diagnostic settings.

The following flow uses an example in which all the pins of ADCC0 are assigned to virtual channels, and pin-level self-diagnosis is performed by switching applied voltage to even physical subchannels and odd physical subchannels. Also, it is assumed that the flow is performed before starting A/D conversion. Do not run self-diagnosis of analog input pins while current is being injected.

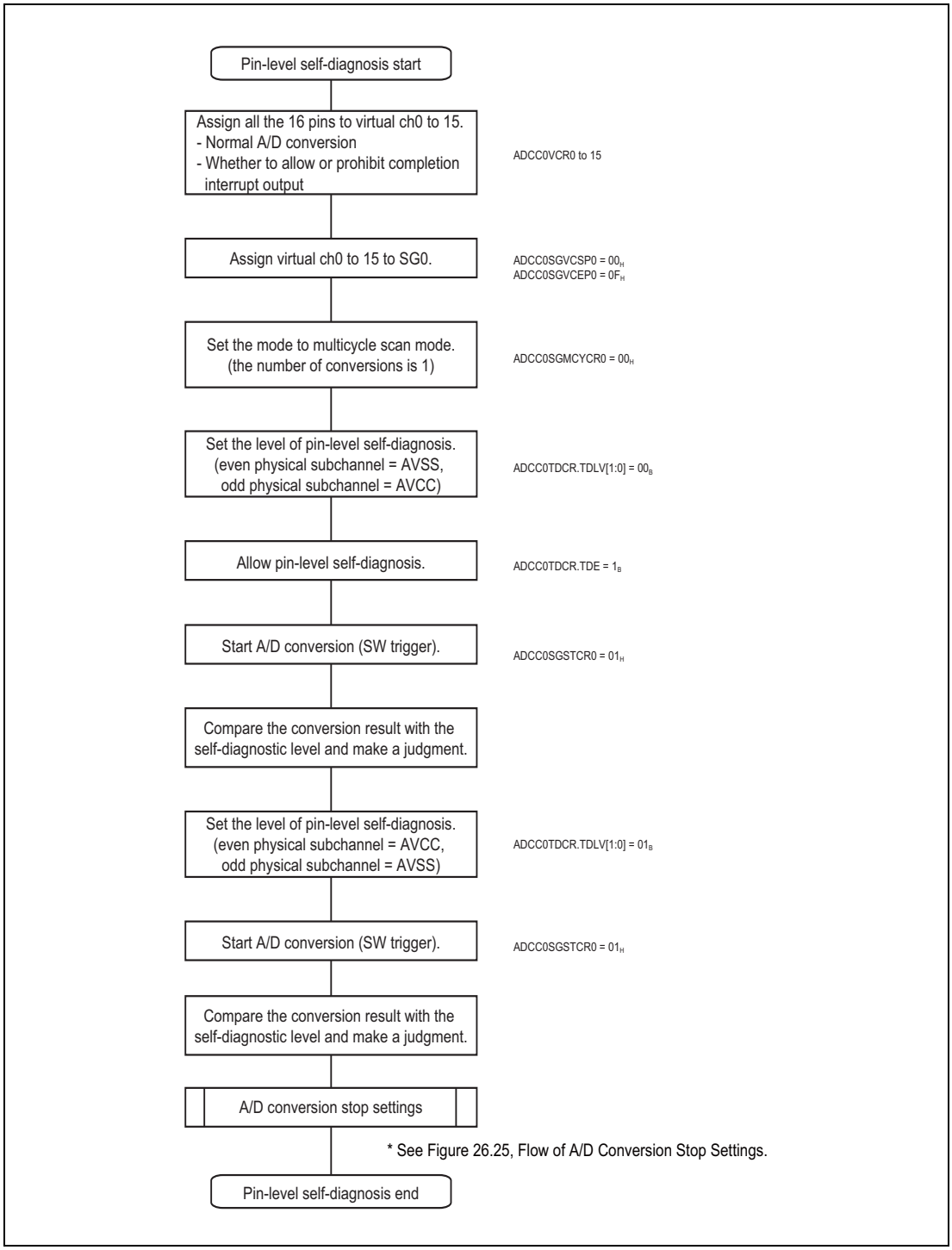
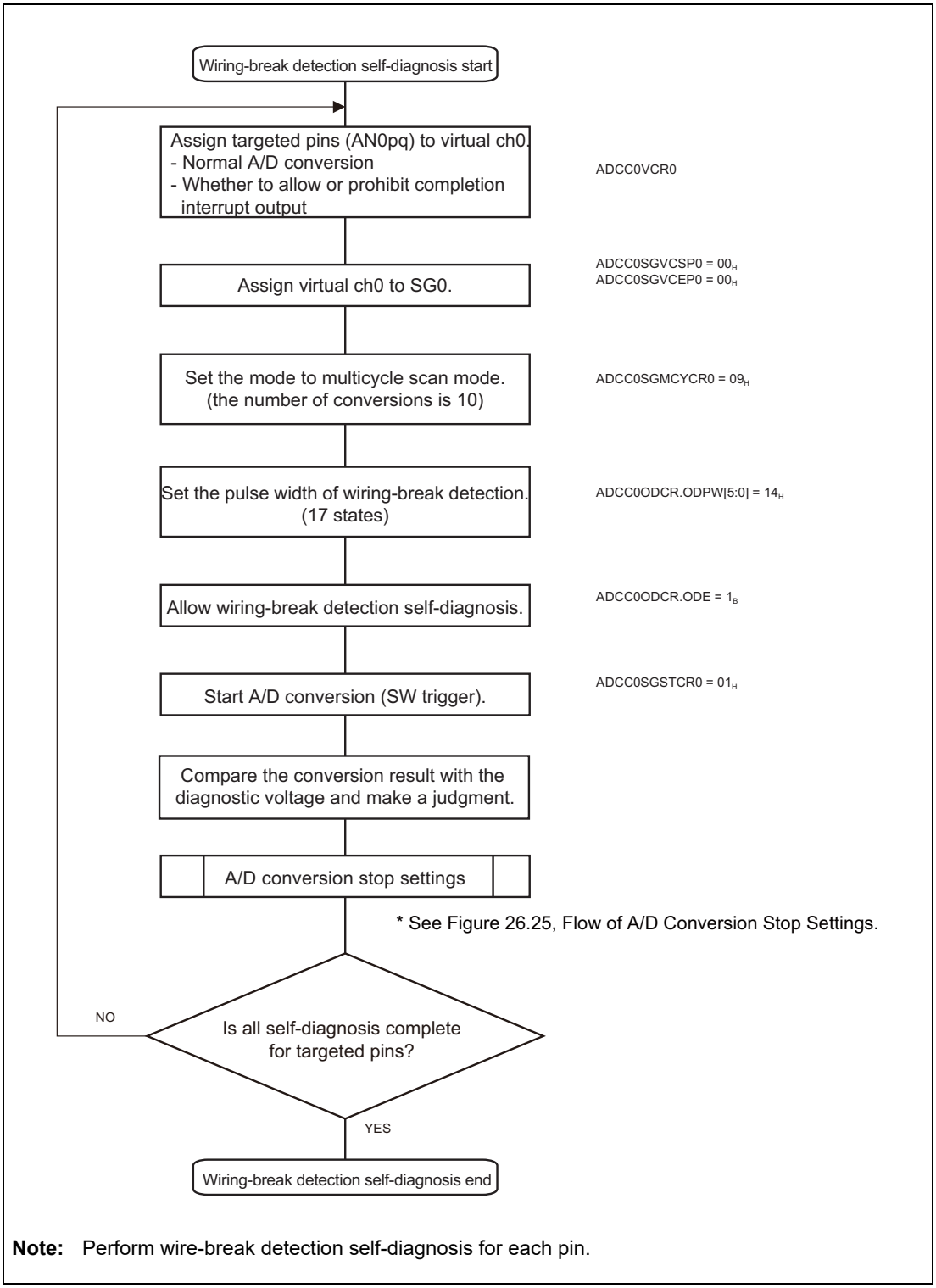


Figure 26.28 Flow of Pin-Level-Self-Diagnostic Settings

26.5.5 Procedure for Setting Wiring-Break Detection Self-Diagnosis

Figure 26.29 shows the flow of wiring-break detection self-diagnostic settings.

The following flow uses an example in which each pin of ADCC0 is assigned to virtual channels, and wiring-break detection self-diagnosis is performed by applying diagnostic voltage (other than the neighborhood of 0 V) to targeted pins. Also, it is assumed that the flow is performed before starting A/D conversion.



Note: Perform wire-break detection self-diagnosis for each pin.

Figure 26.29 Flow of Wiring-Break Detection Self-Diagnostic Settings

26.6 Definition of A/D Conversion Accuracy

A/D conversion accuracy is defined below.

- Resolution
  Number of digital output codes of the A/D converter
- Quantization error
  An error essentially contained in A/D converters, which is given as 1/2LSB (**Figure 26.30**).
- Offset error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from the minimum voltage value 000_H to 001_H. However, the quantization error is not included (**Figure 26.30**).
- Full-scale error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from FFE_H to FFF_H. However, the quantization error is not included (**Figure 26.30**).
- DNL (Differential nonlinear error)
  Deviation between the ideal digital output code width (Vq) and the actual digital output code width (Va), which is given as (Va - Vq)/Vq. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 26.30**).
- INL (Integral nonlinear error)
  Deviation of the actual value from the ideal A/D conversion characteristics, from the zero voltage to the full-scale voltage, which is given as an integral of DNL from 000_H to a digital output code. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 26.30**).
- Absolute accuracy
  Deviation between the digital value and the analog input value. The offset error, the full-scale error, the quantization error, DNL, and INL are included (**Figure 26.30**).

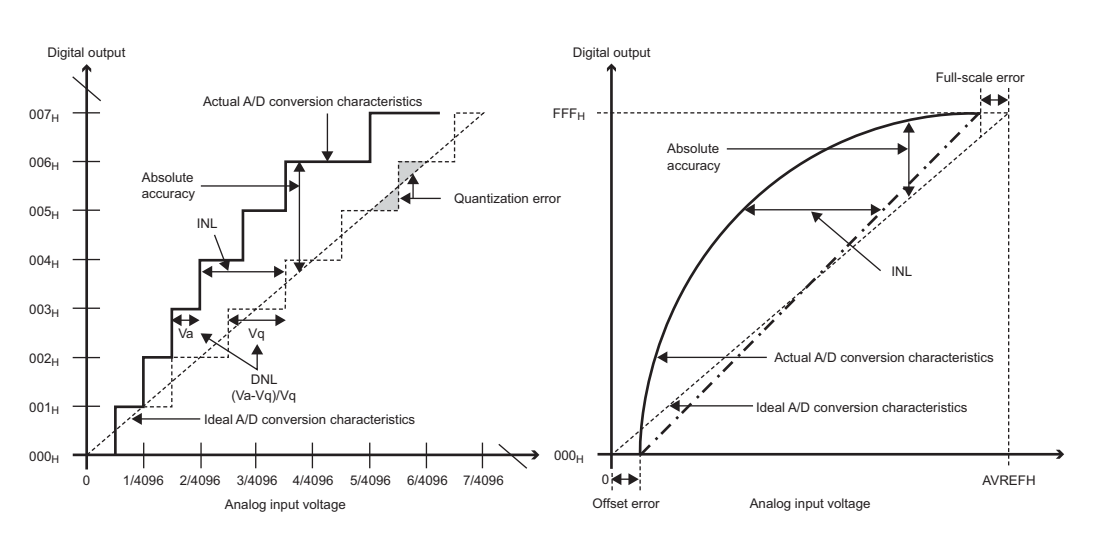


Figure 26.30 Definition of A/D Conversion Accuracy

26.7 Notes

26.7.1 Notes for the Register Setting

1. When you use the simultaneous track and hold function, be sure to select asynchronous suspend.
2. When you use the simultaneous track and hold function, be sure to select the multicycle scan mode and set the number of scans to 1.
3. When you use the simultaneous track and hold function, be sure to hold first to perform A/D conversion.
4. When you use the simultaneous track and hold function, be sure to disable an AD timer trigger input to SGx to be selected in T&H group A or B.
5. Do not perform wiring-break detection in A/D conversion using the simultaneous track and hold function. Set the ADCCnTHER.THkE bit to 0 or the ADCCnTHSTPCR.THSTP bit to 1 before using the wiring-break detection function.
6. Do not run self-diagnosis of analog input pins while current is being injected.

The following table shows notes on setting registers.

Before setting values for the following registers, set and check the contents of Check Steps

Table 26.49 Notes on Setting Registers (1/2)

Registers	Check Steps
ADCCnVCRj ADCCnADCR1 ADCCnADCR2 ADCCnSFTCR ADCCnTDCR ADCCnODCR ADCCnULLMTBR0 to 2	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of all scan groups is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of all scan groups is 0. 3. The scan group status (ADCCnSGSRx.SGACT bit) of all scan groups is 0 (before starting scan groups).
ADCCnTHCR ADCCnTHGSR	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group status (ADCCnSGSRx.SGACT bit) of SGx specified in the scan group select (ADCCnTHACR.SGS[1:0] bits and ADCCnTHBCR.SGS[1:0] bits) of T&H groups A and B is 0 (before starting scan groups). 3. The ADCCnTHER.THkE bit of all T&H is 0 (all T&H are stopped).
ADCCnTHACR.HLDCTE bit, SGS[1:0] bits ADCCnTHBCR.HLDCTE bit, SGS[1:0] bits	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of all scan groups is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of all scan groups is 0. 3. The scan group status (ADCCnSGSRx.SGACT bit) of all scan groups is 0 (before starting scan groups). 4. The ADCCnTHER.THkE bit of all T&H is 0 (all T&H are stopped).
ADCCnTHER	1. The hold trigger enable (ADCCnTHACR.HLDTE bit and ADCCnTHBCR.HLDTE bit) of T&H groups A and B is 0. 2. The scan group status (ADCCnSGSRx.SGACT bit) of SGx specified in the scan group select (ADCCnTHACR.SGS[1:0] bits and ADCCnTHBCR.SGS[1:0] bits) of T&H groups A and B is 0 (before starting scan groups).

Table 26.49 Notes on Setting Registers (2/2)

Registers	Check Steps
ADCCnSGCRx.SCANMD bit, ADIE bit ADCCnSGMCYCRx ADCCnULLMSRx ADCCnSGVCSPx ADCCnSGVCEPx	1. When the scan group select (ADCCnTHACR.SGS[1:0] bits) of T&H group A is specified to SGx, the hold trigger enable (ADCCnTHACR.HLDTE bit) of T&H group A is 0. 2. When the scan group select (ADCCnTHBCR.SGS[1:0] bits) of T&H group B is specified to SGx, the hold trigger enable (ADCCnTHBCR.HLDTE bit) of T&H group B is 0. 3. The scan group synchronization start enable (ADCCnSGCRx.ADSTARTE bit) of SGx is 0 and the trigger mode (ADCCnSGCRx.TRGMD bit) of SGx is 0. 4. The scan group status (ADCCnSGSRx.SGACT bit) of SGx is 0 (before starting scan groups).
ADCCnTHCR ADCCnTHACR ADCCnTHBCR ADCCnTHER ADCCnTHGSR ADCCnSGCRx ADCCnSGVCSPx ADCCnSGVCEPx	When setting the registers shown in the left column, write these registers after they have been read. If this procedure is not followed, the written register value may not be properly reflected, resuting in malfunction.

26.7.2 Notes during Injection Current Application

While injection current is being applied. When A/D conversion using the simultaneous track and hold function is executed, the A/D conversion accuracy can be affected.

In order to avoid affecting the A/D conversion accuracy by injection current, follow the contents below for the setting.

CAUTION

When the voltage applied to a pin exceeds the supplied voltage or fall below the ground voltage, the current is being injected.

(1) When using the simultaneous track and hold function, you must not apply the injection current to “Pins in which injection current affects the conversion accuracy” shown in **Table 26.50**.

Table 26.50 Pin Combination Affecting Conversion Accuracy and T&H Circuit

Unit Name	T&H Circuit for Conversion (Unit Signal Name)	Pins in which Injection Current Affects the Conversion Accuracy	Applicable Product*1	
			C1H	C1M
ADCC0	T&H circuit 0 (AN030)	AN000	√	√
	T&H circuit 1 (AN031)	AN001	√	—
	T&H circuit 2 (AN032)	AN002	√	—
	T&H circuit 3 (AN033)	AN003	√	√
	T&H circuit 4 (AN022)	AN010	√	—
	T&H circuit 5 (AN023)	AN011	√	—
ADCC1	T&H circuit 0 (AN130)	AN100	√	√
	T&H circuit 1 (AN131)	AN101	√	√
	T&H circuit 2 (AN132)	AN102	√	√
	T&H circuit 3 (AN140)	— (No applicable pin)	—	—
	T&H circuit 4 (AN141)	AN110	√	√
	T&H circuit 5 (AN142)	AN111	√	√

Note 1. Applicable: √, Non applicable (pin not included): —

(2) When using the simultaneous track and hold function, setting the bits 4 and 3 of ADCCnVCRj.GCTRL bit to the value other than 00_B can change the "Pins in which injection current affects the conversion accuracy".
The following shows the correspondence of "Pins in which injection current affects the conversion accuracy" and the setting value (except 00_B) of bit 4 and 3 of ADCCnVCRj.GCTRL bit. Follow **Table 26.51** and set up the bits 4 and 3 of ADCCnVCRj.GCTRL bit, to obtain the pins without injection current or to avoid the applicable pins for "Pins in which injection current affects the conversion accuracy".

Table 26.51 Pin Combination Affecting Conversion Accuracy and T&H Circuit ADCCnVCRj.GCTRL Bit Used

Unit Name	T&H Circuit for Conversion (Unit Signal Name)	Set Value of GCTRL[4:3] (Except 00_B)	Pins in which Injection Current Affects the Conversion Accuracy
ADCC0	T&H circuit 0 (AN030)	11_B	— (No applicable pin)
	T&H circuit 1 (AN031)	11_B	— (No applicable pin)
	T&H circuit 2 (AN032)	11_B	— (No applicable pin)
	T&H circuit 3 (AN033)	11_B	— (No applicable pin)
	T&H circuit 4 (AN022)	11_B	— (No applicable pin)
	T&H circuit 5 (AN023)	11_B	— (No applicable pin)
ADCC1	T&H circuit 0 (AN130)	01_B	AN120
		10_B	AN140
		11_B	AN160
	T&H circuit 1 (AN131)	01_B	AN121*1
		10_B	AN141
		11_B	AN161
	T&H circuit 2 (AN132)	01_B	AN122*1
		10_B	AN142
		11_B	AN162
	T&H circuit 3 (AN140)	—	— (No applicable pin)
	T&H circuit 4 (AN141)	11_B	— (No applicable pin)
	T&H circuit 5 (AN142)	11_B	— (No applicable pin)

Note 1. Pin not implemented in C1M. Select this pin as a workaround in C1M.

26.7.3 Notes when Using Simultaneous Track-and-Hold

If A/D conversion is performed by using simultaneous track-and-hold under either of the following conditions, this may affect the accuracy of A/D conversion and the monitor output of a resolver signal.

1. A/D conversion is performed with a combination of a T&H circuit and the applicable pin given in **Table 26.53**.
2. The resolver signal monitor output is used with a combination of a T&H circuit and the applicable pin given in **Table 26.53**.

An error due to the effects on the accuracy of A/D conversion is calculated from the following formula and added to the errors (absolute error, etc.) specified in the A/D converter characteristics.

$$sampling \quad error\ (LSB) = \left[\left(\frac{1}{T2} \times C1 \times V3 \times \mathrm{Re}\right)\right] \times \frac{4096}{Vavrefh}$$

Figure 26.31 A/D Conversion Influential Formula

Table 26.52 A/D Conversion Influential Formula

Item	Symbol	Reference	Unit
Signal source impedance	Re	Depends on user board	kΩ
Conversion cycle of T&H circuit	T2		ms
AnVREFH voltage (n = 0, 1)	Vavrefh		V
Parasitic capacitance of the last stage of channel multiplexer	C1	10	pF
| AnVCC voltage /2 – measured pin voltage | (n = 0, 1)	V3	Depends on user board	V

Therefore, settings should be made in consideration of the following.

(1) Use of simultaneous track-and-hold affects the applicable pins listed in **Table 26.53**.

Table 26.53 Pin Combination Affecting Conversion Accuracy and T&H Circuit

Unit Name	T&H Circuits which Affect Conversion Accuracy (Unit Signal Name)	Pins to be Affected	Applicable Product*1	
			C1H	C1M
ADCC0	T&H circuit 0 (AN030)	AN000/RDC20SINMNT	√	√
	T&H circuit 1 (AN031)	AN001	√	—
	T&H circuit 2 (AN032)	AN002	√	—
	T&H circuit 3 (AN033)	AN003/RDC20COSMNT	√	√
	T&H circuit 4 (AN022)	AN010	√	—
	T&H circuit 5 (AN023)	AN011	√	—
ADCC1	T&H circuit 0 (AN130)	AN100	√	√
	T&H circuit 1 (AN131)	AN101	√	√
	T&H circuit 2 (AN132)	AN102	√	√
	T&H circuit 3 (AN140)	— (No applicable pin)	—	—
	T&H circuit 4 (AN141)	AN110	√	√
	T&H circuit 5 (AN142)	AN111	√	√

Note 1. Applicable: √, Non applicable (pin not included): —

(2) When simultaneous track-and-hold is in use, the “pins to be affected” can be changed by setting bits 4 and 3 of the ADCCnVCRj.GCTRL bits to a value other than 00_B.
The table below shows the correspondence between the settings of bits 4 and 3 of the ADCCnVCRj.GCTRL bits and the “pins to be affected”. Set bits 4 and 3 of the ADCCnVCRj.GCTRL bits according to **Table 26.54** so that there are no pins applicable as the “pins to be affected” or the pins do not create a problem for the system.

Table 26.54 Pin Combination Affecting Conversion Accuracy and T&H Circuit ADCCnVCRj.GCTRL Bit Used

Unit Name	T&H Circuits which Affect Conversion Accuracy (Unit Signal Name)	Set Value of GCTRL[4:3]	Pins to be Affected
ADCC0	T&H circuit 0 (AN030)	11_B	— (No applicable pin)
	T&H circuit 1 (AN031)	11_B	— (No applicable pin)
	T&H circuit 2 (AN032)	11_B	— (No applicable pin)
	T&H circuit 3 (AN033)	11_B	— (No applicable pin)
	T&H circuit 4 (AN022)	11_B	— (No applicable pin)
	T&H circuit 5 (AN023)	11_B	— (No applicable pin)
ADCC1	T&H circuit 0 (AN130)	01_B	AN120
		10_B	AN140
		11_B	AN160
	T&H circuit 1 (AN131)	01_B	AN121*1
		10_B	AN141
		11_B	AN161
	T&H circuit 2 (AN132)	01_B	AN122*1
		10_B	AN142
		11_B	AN162
	T&H circuit 3 (AN140)	—	— (No applicable pin)
	T&H circuit 4 (AN141)	11_B	— (No applicable pin)
	T&H circuit 5 (AN142)	11_B	— (No applicable pin)

Note 1. Pin not implemented in C1M. Select this pin as a workaround in C1M.

Section 27 Functional Safety

27.1 Overview

This section describes the failure detection functions provided to detect the LSI failures in the early stage. Here, the failures include both the recoverable transient failures such as software errors of a memory and the unrecoverable permanent failures.

The following lists the failure detection functions provided by this LSI.

ECC and EDC

Detect failures of memories and data transfer paths; correct some types of failures.

Lockstep

Detects failures of the CPU1 in the early stage.

CAUTION

The lockstep function is disabled during debugging.

Failures are not detected even when the failure detection function is used.

Memory Protection

Detects erroneous access to memories and peripheral circuits to protect the data in these elements against erroneously access.

MISG

Monitors write access to a specific address by the CPU, generates the signatures based on the written data, and automatically compares the generated signatures with each other.

Clock Monitor

Monitors the clock operation to detect abnormal operation.

BIST

Detects failures of the failure detection function itself.

Error Control Module (ECM)

Monitors various failure detection states in the LSI and defines the operation to be carried out upon failure detection.

27.2 ECC and EDC

27.2.1 Overview

27.2.1.1 ECC

This product incorporates ECC for the following memories. The ECC enables detection and correction of errors of the data retained in the memories. The ECC also enables detection and correction of errors produced between the ECC encoder and memories; and memories and ECC decoder.

Table 27.1 ECC Overview

Applicable Memory	Applicable Data Width [bits]	Operation upon Error Detection				Failure Insertion
		Detection/ Correction	Notice to ECM	Error Status	Address Capture	
Code flash	128	SEC-DED	Possible	Possible	Possible	Possible
Data flash Local RAM (CPU1) Local RAM (CPU2: only C1H) Global RAM	32	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (data)	64	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (tag)	32	SED-DED	Possible	Possible	Possible	Possible
RAM for DTS	32	SEC-DED	Possible	Possible	Possible	Possible
Peripheral RAM (32 bits)	32	SEC-DED	Possible	Possible	Possible	Possible

Applicable Data Width

ECC encoding is applied to the data of the shown width. If narrower data is to be written, the following processes are required. Here, ECC is also checked when data is read out in (1).

(1) Read data to which ECC encoding is applied, including data to be rewritten.

(2) Replace data to be rewritten.

(3) Write back the data generated in (2).

Detection/Correction

SEC-DED: 1-bit errors can be detected and corrected, and 2-bit errors can only be detected.

SED-DED: 1-bit errors and 2-bit errors can only be detected.

Notice to ECM

An error detected can be notified to the ECM (error control module).

Error Status

The status of an error detected is retained.

Address Capture

The address of an error detected is retained.

Failure Insertion

An ECC error can be intentionally caused to enable self-diagnosis of the ECC decoder operation.

27.2.1.2 Address Parity

This product incorporates address parity for the following memories. The address parity enables detection of errors during address decoding. The address parity also enables detection of errors produced at addresses between the parity encoder and memories.

Table 27.2 Address Parity Overview

Applicable Memory	Parity Bit	Notice to ECM	Error Status	Address Capture	Failure Insertion
Code flash	1 bit	Possible	Possible	Possible	Possible
Local RAM (CPU1, CPU2: only C1H)	2 bits*1	Possible	Possible	Possible	Possible
Global RAM	1 bit	Possible	Possible	Possible	Possible

Note 1. The parity bits corresponding to the written address are written to two locations in the memory. When they are compared to the parity bits corresponding to the read address and errors are detected in both of the two parity bits stored in the memory, it is handled as an address parity error. When an error is detected only at one of the parity bits, it is handled as a parity bit error.

27.2.1.3 Data Parity

This product incorporates data parity for the particular data transfer. The data parity enables detection of errors of the transferred data. For details, refer to **Section 27.2.9, Data Parity for Data Transfer Paths**.

27.2.2 Code Flash ECC and Address Parity

27.2.2.1 Overview

The code flash ECC is summarized in the table below.

Table 27.3 Overview of the Code Flash ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. The register values after a reset enable the full functionality, that is, the detection of 2-bit errors and the detection and correction of 1-bit errors are notified.
Address parity	Address parity check can be either enabled or disabled. Address parity is checked during data read. The register values after a reset enable this function.
Error notification	Indicators of the occurrence of ECC and address parity errors are conveyed to the ECM. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable error notification upon detection of a 2-bit ECC error, and disable error notification upon detection of a 1-bit ECC error. **Parity Error:** • Error notification can be either enabled or disabled upon detection of an address parity error. The register values after a reset enable error notification upon detection of an address parity error. The error notification signal is issued to the ECM, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Data in the ROM and the ECC and address parity bits can all be read directly. Desired values can also be written to the ROM and to the ECC and address parity bits.
Others	The ECM can initiate a transition to the safe state in response to the detection of a 2-bit ECC error during the fetching of an instruction.

The ECC decoder and address parity generator are provided each for the read ports (CPU1, CPU2, and interconnect) connected to the code flash interface. The address parity checker is provided in the access controller for the code flash. For details, see **Figure 27.1**.

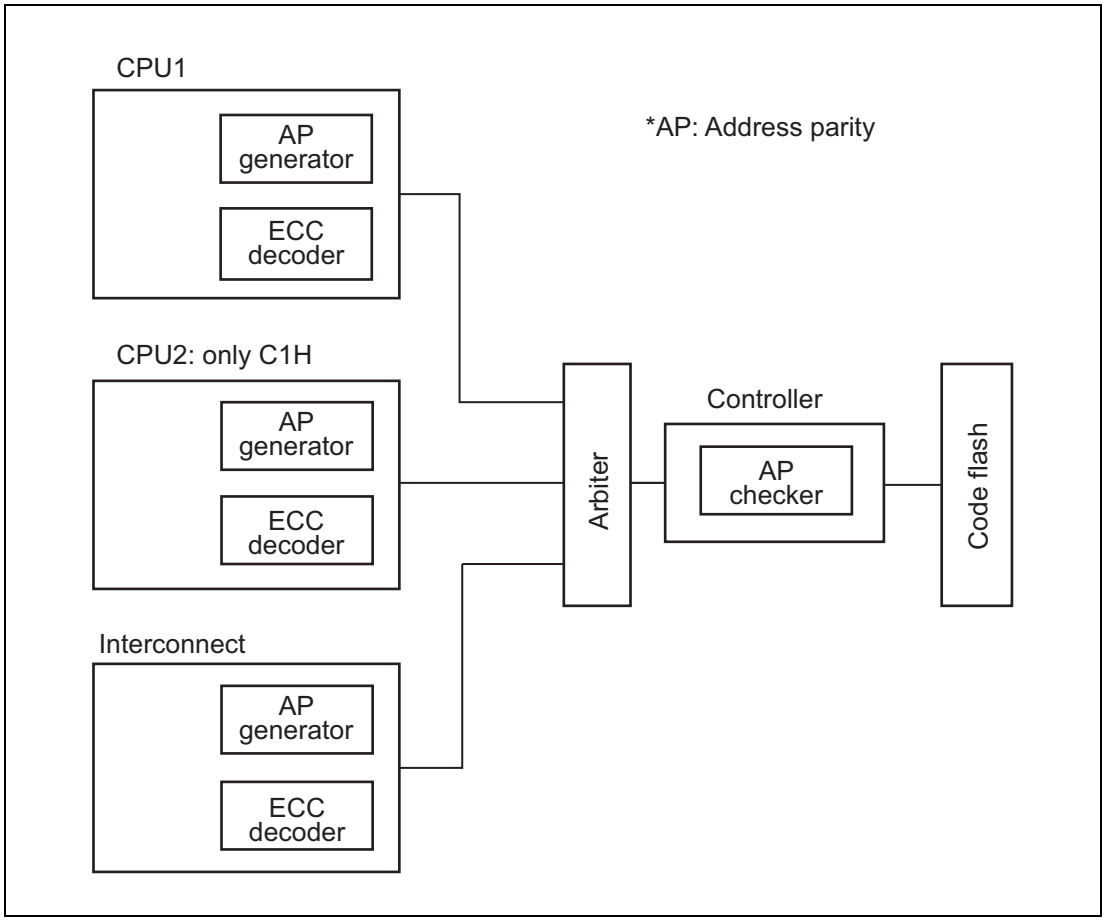


Figure 27.1 ECC and Address Parity of Code Flash

27.2.2.2 List of Registers

Table 27.4 List of Registers

Module Name	Address	Symbol*1	Register Name	R/W	Value after Reset	Access Size
ECCFLI	FFC6 2000$_H$	CFAPCTL	Code flash address parity control register	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2200$_H$	CFECCCTL_VCI	Code flash ECC control register (VCI)	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2204$_H$	CFERRINT_VCI	Code flash error information control register (VCI)	R/W	0000 0006$_H$	8/16/32
ECCFLI	FFC6 2208$_H$	CFSTCLR_VCI	Code flash status clear register (VCI)	W	0000 0000$_H$	8/16/32
ECCFLI	FFC6 220C$_H$	CFOVFSTR_VCI	Code flash error count overflow status register (VCI)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2210$_H$	CF1STERSTR_VCI	Code flash 1st error status register (VCI)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2250$_H$	CF1STEADR0_VCI	Code flash 1st error address register (VCI)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2350$_H$	CFSTSTCTL_VCI	Code flash sub-test control register (VCI)	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2400$_H$	CFECCCTL_PE1	Code flash ECC control register (PE1)	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2404$_H$	CFERRINT_PE1	Code flash error information control register (PE1)	R/W	0000 0006$_H$	8/16/32
ECCFLI	FFC6 2408$_H$	CFSTCLR_PE1	Code flash status clear register (PE1)	W	0000 0000$_H$	8/16/32
ECCFLI	FFC6 240C$_H$	CFOVFSTR_PE1	Code flash error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2410$_H$	CF1STERSTR_PE1	Code flash 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2450$_H$	CF1STEADR0_PE1	Code flash 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2550$_H$	CFSTSTCTL_PE1	Code flash sub-test control register (PE1)	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2600$_H$	CFECCCTL_PE2	Code flash ECC control register (PE2)	R/W	0000 0000$_H$	16/32
ECCFLI	FFC6 2604$_H$	CFERRINT_PE2	Code flash error information control register (PE2)	R/W	0000 0006$_H$	8/16/32
ECCFLI	FFC6 2608$_H$	CFSTCLR_PE2	Code flash status clear register (PE2)	W	0000 0000$_H$	8/16/32
ECCFLI	FFC6 260C$_H$	CFOVFSTR_PE2	Code flash error count overflow status register (PE2)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2610$_H$	CF1STERSTR_PE2	Code flash 1st error status register (PE2)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2650$_H$	CF1STEADR0_PE2	Code flash 1st error address register (PE2)	R	0000 0000$_H$	8/16/32
ECCFLI	FFC6 2750$_H$	CFSTSTCTL_PE2	Code flash sub-test control register (PE2)	R/W	0000 0000$_H$	16/32

Note 1. The registers with symbols “_VCI”, “_PE1”, and “_PE2” as suffixes are provided to the particular ECC controllers: the registers with “_VCI” are provided to the ECC controller for access from the system interconnect 1 to the code flash, the registers with “_PE1” are provided to the ECC controller for access from the CPU1, and the registers with "_PE2" are provided to the ECC controller for access from the CPU2.

27.2.2.3 Details of Registers

(1) CFAPCTL — Code Flash Address Parity Control Register

CFAPCTL enables or disables address parity check. Set the PROT[1:0] bits to 01_B when writing to CFAPCTL.

CFAPCTL is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	APTESTB	APTESTA	APARIDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 27.5 CFAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 3	—	Reserved. These bits are always read as 0. The write value should also be 0.
2	APTESTB	Address Parity Checker (Bank B) Test Sets the address parity checker to test mode. When APTESTB = 1, the parity generated by the address parity generator is inverted.
1	APTESTA	Address Parity Checker (Bank A) Test Sets the address parity checker to test mode. When APTESTA = 1, the parity generated by the address parity generator is inverted.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by the address parity circuit. 0: Enables address parity check. 1: Disables address parity check.

(2) CFECCCTL_VCI/PE1/PE2 — Code Flash ECC Control Register

CFECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01_B when writing to CFECCCTL.

CFAPCTL is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 27.6 CFECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(3) CFERRINT_VCI/PE1/PE2 — Code Flash Error Information Control Register

CFERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

CFERRINT is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 27.7 CFERRINT Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved. These bits are always read as 0. The write value should also be 0.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(4) CFSTCLR_VCI/PE1/PE2 — Code Flash Status Clear Register

CFSTCLR clears the error flags in the error status register (CF1STERSTR), the overflow flag in the error count overflow status register (CFOVFSTR), and the error address register (CF1STEADR). CFSTCLR is a write-only register and is always read as 0.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 27.8 CFSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	STCLR0	Error Overflow Flag Clear Writing 1 to this bit clears the APEF0, DEDF0, and SEDF0 flags in CF1STERSTR; ERROVF0 flag in CFOVFSTR; and CF1STEADR0.

(5) CFOVFSTR_VCI/PEI/PE2 — Code Flash Error Count Overflow Status Register

CFOVFSTR indicates overflows of the error counter. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.9 CFOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	ERROVF0	Error Overflow Flag This flag is set if the second error occurs while any of the error flags (APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(6) CF1STERSTR_VCI/PE1/PE2 — Code Flash 1st Error Status Register

CF1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The error flag is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error monitor flag is set.

If multiple errors occur simultaneously, all the corresponding error flags are set. CF1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.10 CF1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved. These bits are always read as 0. The write value should also be 0.
2	APEF0	Address Parity Error Monitor Flag 0: Cleared to 0 by setting the STCLR0 bit to 1 in CFSTCLR. 1: Indicates that an address parity error has occurred while the error flags DEDF0 and APEF0 are 0.
1	DEDF0	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLR0 bit to 1 in CFSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flags DEDF0 and APEF0 are 0.
0	SEDF0	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLR0 bit to 1 in CFSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags DEDF0, SEDF0, and APEF0 are 0.

(7) CF1STEADR0_VCI/PE1/PE2 — Code Flash 1st Error Address Register

CF1STEADR0 holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[24:4] of this register correspond to bits [24:4] of the real address. The real address can be calculated by appending the higher-order address bits [31:25] as a base address. CF1STEADR0 is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

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

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

Table 27.11 CF1STEADR0 Register Contents

Bit Position	Bit Name	Function
31 to 25	—	Reserved. These bits are always read as 0. The write value should also be 0.
24 to 4	EADR[24:4]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.
3 to 0	—	Reserved. These bits are always read as 0. The write value should also be 0.

(8) CFSTSTCTL_VCI/PE1/PE2 — Code Flash Sub-Test Control Register

CFSTSTCTL is used for the ECC test (self-diagnosis). This register is dedicated for the code flash. After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly. Set the PROT[1:0] bits to 01_B when writing to CFSTSTCTL.

CFSTSTCTL is initialized by an internal reset or an external reset.

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

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

Table 27.12 CFSTSTCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.

Correctly reading instructions from the code flash access port is not possible while ECC test mode is selected (ECCTST = 1). While the access port for the CPU is set to test mode (including during changes to the value of the ECCTST bit), the CPU must run a program from the local RAM or global RAM and must not fetch instructions from the code flash memory.

The CPU has a small data buffer. If an old value remains in this buffer, the correct value cannot be read even when the ECCTST bit is switched. When switching the ECCTST bit, be sure to clear the data buffer. For how to clear the data buffer, see **Section 3, CPU System**.

From the code flash access port with ECC test mode selected, access must be made by reading 4 bytes aligned to 16n address. The results of code flash reading are as follows:

Table 27.13 Results of Code Flash Reading

Bit Position	Contents
31 to10	Always 0
9	Address parity bit
8 to 0	ECC bits

27.2.2.4 Test Function

Through appropriate register setting, the code flash data, ECC bits, and address parity bit can be read out.

(1) Reading code flash data

(a) Set the ECCDIS bit in the code flash ECC control register to 1 to disable ECC error detection and correction.

(b) When ECCDIS = 1, neither error detection nor correction proceeds when the code flash is read; the data output from the code flash is read unchanged.

How to exit this test mode:

(c) Set the ECCDIS bit in the code flash ECC control register to 0 to enable ECC error detection and correction.

(2) Reading the ECC and address parity bits

(a) Set the ECCDIS bit in the code flash ECC control register to 1 to disable ECC error detection and correction.

(b) Set the ECCTST bit in the code flash sub-test control register to 1 to set test mode.

(c) When the code flash is read, the ECC and address parity bits are read instead of the code flash data.

How to exit this test mode:

(d) Set the ECCDIS bit in the code flash ECC control register to 0 to enable ECC error detection and correction.

(e) Set the ECCTST bit in the code flash sub-test control register to 0 to set normal mode.

(3) Self-diagnosis

Self-diagnosis of the ECC decoder and address parity decoder for the access ports is possible by writing incorrect data to the code flash memory beforehand (fault injection) and then reading this data. A 1- or 2-bit ECC error or an address parity error fault can be injected by generating correct ECC bits and address parity bit once and inverting only the appropriate bits.

For details on programming of the code flash, refer to “RH850/C1x Flash Memory User’s Manual: Hardware Interface”.

27.2.3 Data Flash ECC

27.2.3.1 Overview

The data flash ECC is summarized in the table below.

Table 27.14 Overview of the Data Flash ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. The register values after a reset enable the full functionality, that is, the detection of 2-bit errors and the detection and correction of 1-bit errors.
Error notification	An indicator of the occurrence of an ECC error is conveyed to ECM. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable error notification upon detection of a 2-bit ECC error, and disable error notification upon detection of a 1-bit ECC error. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Data in the ROM and the ECC bits can be read directly. Desired values can be written as ROM data and to the ECC bits.

27.2.3.2 List of Registers

(1) List of ECC Modules

Table 27.15 List of Registers

Name of ECC Modules And Register Base Address			
Master Side		Checker Side	
Module Name	Base Address <Base_addr>	Module Name	Base Address <Base_addr>
ECCEEP	FFC6 2C00$_H$	ECCEEPC	FFC6 2E00$_H$

(2) List of Registers

Table 27.16 List of Registers

Module Name	Symbol	Register Name	R/W	Value after Reset	Address	Access Size		
						8	16	32
ECCEEP	DFECCCTL	Data flash ECC control register	R/W	0000$_H$	<Base_addr>		√	
ECCEEP	DFERSTR	Data flash error status register	R	00$_H$	<Base_addr>+04$_H$	√		
ECCEEP	DFERSTC	Data flash error status clear register	W	00$_H$	<Base_addr>+08$_H$	√		
ECCEEP	DFOVFSTR	Data flash error overflow status register	R	00$_H$	<Base_addr>+0C$_H$	√		
ECCEEP	DFOVFSTC	Data flash error overflow status clear register	W	00$_H$	<Base_addr>+10$_H$	√		
ECCEEP	DFERRINT	Data flash error notification control register	R/W	02$_H$	<Base_addr>+14$_H$	√		
ECCEEP	DFEADR	Data flash 1st error address register	R	0000 0000$_H$	<Base_addr>+18$_H$			√
ECCEEP	DFTSTCTL	Data flash test control register	R/W	0000$_H$	<Base_addr>+1C$_H$		√	

(3) DFECCCTL — Data Flash ECC Control Register

DFECCCTL enables or disables ECC error detection and correction and 1-bit error correction.

DFECCCTL is initialized by an internal reset or an external reset.

Set the PROT[1:0] bits to 01_B when writing to DFECCCTL.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 27.17 DFECCCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	—	Reserved
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(4) DFERSTR — Data Flash Error Status Register

DFERSTR monitors occurrence of errors.

DFERSTR is initialized by an internal reset, an external reset, or setting the clear bit in the data flash error status clear register.

The SEDF bit is set if a 1-bit ECC error is detected while ECC error detection and correction are enabled, and the DEDF bit is set if a 2-bit ECC error is detected.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDF	SEDF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.18 DFERSTR Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved
1	DEDF	2-Bit ECC Error Monitor This bit is set if a 2-bit ECC error is generated while SEDF and DEDF are 0. Clearing conditions: A reset is generated. The ERRCLR bit is set in data flash error status clear register. Setting condition: A 2-bit ECC error is generated with both SEDF and DEDF being 0.
0	SEDF	1-Bit ECC Error Monitor This bit is set if a 1-bit ECC error is generated while SEDF and DEDF are 0. Clearing conditions: A reset is generated. The ERRCLR bit is set in data flash error status clear register. Setting condition: A 1-bit ECC error is generated with both SEDF and DEDF being 0.

(5) DFERSTC — Data Flash Error Status Clear Register

DFERSTC clears the error flags in the data flash error status register. DFERSTC is a write-only register and is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERRCLR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.19 DFERSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERRCLR	SEDF/DEDF Flag Clear Writing 1 to this bit clears the SEDF/DEDF flag.

(6) DFOVFSTR — Data Flash Error Overflow Status Register

DFOVFSTR monitors occurrence of data flash error overflow. The ERROVF flag is cleared by an internal reset, an external reset, or setting the ERROVFCLR bit to 1 in DFOVFSTC.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.20 DFOVFSTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERROVF	Error Overflow Flag This flag is set if an ECC error occurs while the error address register is full.

(7) DFOVFSTC — Data Flash Error Overflow Status Clear Register

DFOVFSTC clears the data flash error overflow flag. Setting the ERROVFCLR bit to 1 clears this flag.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVFCLR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.21 DFOVFSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERROVFCLR	Error Overflow Flag Clear Writing 1 to this bit clears the ERROVF flag. This bit is always read as 0.

(8) DFERRINT — Data Flash Error Notification Control Register

DFERRINT enables or disables generation of the error notification signal upon detection of a 2-bit ECC error or a 1-bit ECC error.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 27.22 DFERRINT Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved
1	DEDIE	2-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(9) DFEADR — Data Flash 1st Error Address Register

DFEADR holds the address at which an ECC error has occurred while both of the SEDF and DEDF bits in the data flash error status register are 0.

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

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

Table 27.23 DFEADR Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved
20 to 2	DFEADR[20:2]	ECC Error Address These bits are read-only and used to monitor the address at which an ECC error has occurred.
1, 0	—	Reserved

(10) DFTSTCTL - Data Flash Test Control Register

DFTSTCTL is used for the ECC test. After ECC test mode is enabled by setting ECCTST = 1, the ECC bits can be read.

Set the PROT[1:0] bits to 01B when writing to DFTSTCTL.

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

Table 27.24 DFTSTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	ECCTST	ECC Test Sets ECC test mode.

27.2.3.3 Test Function

Data in the ROM and the ECC bits can be read through the setting of the data flash test control register (DFTSTCTL).

(1) Reading the ROM data

(a) Set the ECCDIS bit in the data flash ECC control register to 1 to disable ECC error detection and correction.

(b) When ECCDIS = 1, neither error detection nor correction proceeds when the data flash is read; the data output from the data flash is read unchanged.

How to exit this test mode:

(c) Set the ECCDIS bit in the data flash ECC control register to 0 to enable ECC error detection and correction.

(2) Reading the ECC data

(a) Set the ECCDIS bit in the data flash ECC control register to 1 to disable ECC error detection and correction.

(b) Set the ECCTST bit in the data flash control register to 1 to set test mode.

(c) When the data flash is read, the 7 lower-order bits of read data are read as ECC data.

How to exit this test mode:

(d) Set the ECCDIS bit in the data flash ECC control register to 0 to enable ECC error detection and correction.

(e) Set the ECCTST bit in the data flash control register to 0 to set normal mode.

(3) Self-diagnosis

Self-diagnosis of the ECC decoder is possible by writing incorrect data to the data flash memory beforehand (fault injection) and then reading this data. A 1- or 2-bit ECC error fault can be injected by generating correct ECC bits once and inverting only the appropriate bits.

For details on programming of the data flash, refer to “RH850/C1x Flash Memory User’s Manual: Hardware Interface”.

27.2.4 Local RAM (CPU1/CPU2) ECC and Address Parity

27.2.4.1 Overview

The local RAM ECC of CPU1 and CPU2 are summarized in the table below.

Table 27.25 Overview of the Local RAM ECC of CPU1 and CPU2

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. The register values after a reset enable the full functionality, that is, the detection of 2-bit errors and the detection and correction of 1-bit errors.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the same parity bit is written to two locations in the RAM. During data read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the RAM. The error decoding specifications are shown in **Table 27.26, Address Parity Definitions**. The register values after a reset enable this function.
Error notification	Indicators of the occurrence of ECC and address parity errors are conveyed to ECM. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable notification of the 2-bit error and disable notification of the 1-bit error. **Parity Error:** • Error notification can be either enabled or disabled upon detection of an address parity error. • Error notification can be either enabled or disabled upon detection of a parity bit error. The register values after a reset enable notification of the address parity error and disable notification of the parity bit error. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error and a parity bit error are handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. The RAM data and the ECC and address parity bits can be read directly.

The definitions of the address parity error during read accesses are shown in the table below.

Table 27.26 Address Parity Definitions

RAM Macro Parity Bit 1	RAM Macro Parity Bit 2	Read Address Parity	Error Determination and Error Name
0	0	0	No error
0	0	1	Address parity error
0	1	0	Parity bit error
0	1	1	Parity bit error
1	0	0	Parity bit error
1	0	1	Parity bit error
1	1	0	Address parity error
1	1	1	No error

The local RAM of CPU1 and CPU2 has structures that can read and write a maximum of 128 bits of data simultaneously. In the meanwhile, ECC or address parity is provided for each 32-bit data, and the 32-bit data is divided into four banks (bank 0 to bank 3). The lower bits (including the LSB) are assigned to bank 0, and the upper bits (including the MSB) are assigned to bank 3.

The relationship between addresses and banks is described below.

Table 27.27 Relationship between Addresses and Banks

4 Lower-Order Bits of Address (Hexadecimal Notation)	F_H to C_H	B_H to 8_H	7_H to 4_H	3_H to 0_H
Bank number	Bank 3	Bank 2	Bank 1	Bank 0

27.2.4.2 List of Registers

Table 27.28 List of Registers

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
ECCCPU1	FFC6 5000$_H$	LRAPCTL_PE1	Local RAM address parity control register (PE1)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5020$_H$	LRAPCTL_PE2	Local RAM address parity control register (PE2)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5004$_H$	LRTSTCTL_PE1	Local RAM test control register (PE1)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5008$_H$	LRTDATBF0_PE1	Local RAM test data read buffer 0 (PE1)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 500C$_H$	LRTDATBF1_PE1	Local RAM test data read buffer 1 (PE1)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 5024$_H$	LRTSTCTL_PE2	Local RAM test control register (PE2)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5028$_H$	LRTDATBF0_PE2	Local RAM test data read buffer 0 (PE2)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 502C$_H$	LRTDATBF1_PE2	Local RAM test data read buffer 1 (PE2)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 5400$_H$	LRECCCTL_PE1	Local RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5404$_H$	LRERRINT_PE1	Local RAM error information control register (PE1)	R/W	0000 0006$_H$	8/16/32
ECCCPU1	FFC6 5408$_H$	LRSTCLR_PE1	Local RAM status clear register (PE1)	W	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 540C$_H$	LROVFSTR_PE1	Local RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 5410$_H$	LR1STERSTR_PE1	Local RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 5600$_H$	LRECCCTL_PE2	Local RAM ECC control register (PE2)	R/W	0000 0000$_H$	16/32
ECCCPU1	FFC6 5604$_H$	LRERRINT_PE2	Local RAM error information control register (PE2)	R/W	0000 0006$_H$	8/16/32
ECCCPU1	FFC6 5608$_H$	LRSTCLR_PE2	Local RAM status clear register (PE2)	W	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 560C$_H$	LROVFSTR_PE2	Local RAM error count overflow status register (PE2)	R	0000 0000$_H$	8/16/32
ECCCPU1	FFC6 5610$_H$	LR1STERSTR_PE2	Local RAM 1st error status register (PE2)	R	0000 0000$_H$	8/16/32

27.2.4.3 Details of Registers

(1) LRAPCTL_PE1/PE2 — Local RAM Address Parity Control Register

LRAPCTL enables or disables address parity check. Set the PROT[1:0] bits to 01_B when writing to LRAPCTL.

LRAPCTL is initialized by an internal reset or an external reset.

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

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

Table 27.29 LRAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by all the address parity circuits (bank 0 to bank 3). 0: Enables address parity check. 1: Disables address parity check.

(2) LRTSTCTL_PE1/PE2 — Local RAM Test Control Register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC and address parity bits. The DATSEL bit is used to select RAM data or the ECC and address parity bits.

By setting address parity test mode (APTESTi = 1; i = 0, 1, 2, 3), the parity to be input to the address parity checker is inverted. Set the PROT[1:0] bits to 01B when writing to LRTSTCTL.

LRTSTCTL is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	APTEST3	APTEST2	APTEST1	APTEST0	ECCTST	DATSEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.30 LRTSTCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 6	—	Reserved. These bits are always read as 0. The write value should also be 0.
5	APTEST3	Address Parity Checker (Bank3) Test Sets the address parity checker to test mode. When APTEST3 = 1, the parity generated through the address parity generator is inverted.
4	APTEST2	Address Parity Checker (Bank2) Test Sets the address parity checker to test mode. When APTEST2 = 1, the parity generated through the address parity generator is inverted.
3	APTEST1	Address Parity Checker (Bank1) Test Sets the address parity checker to test mode. When APTEST1 = 1, the parity generated through the address parity generator is inverted.
2	APTEST0	Address Parity Checker (Bank0) Test Sets the address parity checker to test mode. When APTEST0 = 1, the parity generated through the address parity generator is inverted.
1	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.
0	DATSEL	Data Select This bit is valid when ECCTST = 1. This bit selects the RAM bit which can be accessed when writing. 0: RAM data is selected. 1: The ECC bits and address parity bit are selected.

Note 1. When ECC test mode for the local RAM is enabled (ECCTST = 1), access to the local RAM should be in 4-byte units.

(3) LRTDATBFn_PE1/PE2 — Local RAM Test Data Read Buffer n (n = 0, 1)

In ECC test mode (self-diagnosis), the ECC and address parity bits can be read. If the local RAM is read while ECCTST = 1 in the local RAM test control register LRTSTCTL, reading from the local RAM reads out the ECC and address parity bits, and these bits are stored in this buffer.

This register is initialized by an internal reset or an external reset.

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

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

Table 27.31 LRDATBFn Register Contents

Bit Position	Bit Name	Function
31 to 25	—	Reserved. These bits are always read as 0. The write value should also be 0.
24 to 16	LRDATABF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding bank in the local RAM, the ECC bits for the local RAM (bank (2n + 1)) and the address parity bit are respectively stored in LRTDATABF[22:16] and LRTDATABF[24:23].
15 to 9	—	Reserved. These bits are always read as 0. The write value should also be 0.
8 to 0	LRDATABF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding bank in the local RAM, the ECC bits for the local RAM (bank (2n)) and the address parity bit are respectively stored in LRTDATABF[6:0] and LRTDATABF[8:7].

(4) LRECCCTL_PE1/PE2 — Local RAM ECC Control Register

LRECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01_B when writing to LRECCCTL.

LRECCCTL is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 27.32 LRECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(5) LRERRINT_PE1/PE2 — Local RAM Error Information Control

LRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, an address parity error, or a parity bit error.

LRERRINT is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PBEIE	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 27.33 LRERRINT Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved. These bits are always read as 0. The write value should also be 0.
3	PBEIE	Parity Bit Error Notification Enable Enables or disables generation of the error notification signal upon detection of a parity bit error when address parity check is enabled. 0: Disables notification of the parity bit error. 1: Enables notification of the parity bit error.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(6) LRSTCLR_PE1/PE2 — Local RAM Status Clear Register

LRSTCLR clears the error flags in the error status register (LR1STERSTR) and the overflow flag in the error count overflow status register (LROVFSTR). LRSTCLR is a write-only register and is always read as 0.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	STCLR3	STCLR2	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Table 27.34 LRSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved. These bits are always read as 0. The write value should also be 0.
3	STCLR3	Error Overflow Flag Clear (for bank 3) Writing 1 to this bit clears the PBEF3, APEF3, DEDF3, and SEDF3 flags in LR1STERSTR; ERROVF3 flag in LROVFSTR.
2	STCLR2	Error Overflow Flag Clear (for bank 2) Writing 1 to this bit clears the PBEF2, APEF2, DEDF2, and SEDF2 flags in LR1STERSTR; ERROVF2 flag in LROVFSTR.
1	STCLR1	Error Overflow Flag Clear (for bank 1) Writing 1 to this bit clears the PBEF1, APEF1, DEDF1, and SEDF1 flags in LR1STERSTR; ERROVF1 flag in LROVFSTR.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the PBEF0, APEF0, DEDF0, and SEDF0 flags in LR1STERSTR; ERROVF0 flag in LROVFSTR.

(7) LROVFSTR_PE1/PE2 — Local RAM Error Count Overflow Status Register

LROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in LRSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF3	ERROVF2	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.35 LROVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved. These bits are always read as 0. The write value should also be 0.
3	ERROVF3	Error Overflow Flag (for bank 3) This flag is set if the second error occurs while any of the error flags (PBEF3, APEF3, DEDF3, and SEDF3) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
2	ERROVF2	Error Overflow Flag (for bank 2) This flag is set if the second error occurs while any of the error flags (PBEF2, APEF2, DEDF2, and SEDF2) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
1	ERROVF1	Error Overflow Flag (for bank 1) This flag is set if the second error occurs while any of the error flags (PBEF1, APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (PBEF0, APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(8) LR1STERSTR_PE1/PE2 — Local RAM 1st Error Status Register

LR1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The error flag is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag or parity bit error flag is set.

If more than one error occurs simultaneously, all the corresponding error flags are set. LR1STERSTR is cleared by an internal reset, an external reset, or setting 1 to the STCLR bit in LRSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PBEF3	APEF3	DEDF3	SEDF3	—	—	—	—	PBEF2	APEF2	DEDF2	SEDF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PBEF1	APEF1	DEDF1	SEDF1	—	—	—	—	PBEF0	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.36 LR1STERSTR Register Contents

Bit Position	Bit Name	Function
7+8n:4+8n	—	Reserved. These bits are always read as 0. The write value should also be 0.
3+8n	PBEFn	Parity Bit Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a parity bit error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.
2+8n	APEFn	Address Parity Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that an address parity error has occurred while the error flags DEDFn and APEFn are 0. **CAUTION** This flag is set equally for read and write; the error generation source is not considered.
1+8n	DEDFn	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flags DEDFn and APEFn are 0.
0+8n	SEDFn	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting 1 to the STCLRn bit in LRSTCLR. 1: Indicates that a 1-bit ECC error has occurred while all the error flags PBEFn, APEFn, DEDFn and SEDFn are 0.

Note: n = 0 to 3, where “n” denotes a bank number.

27.2.4.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits. Also, data in the RAM and the ECC and address parity bits can all be read.

(1) Writing RAM data

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL bit in the local RAM test control register to 0 to select RAM data for access when writing.

(c) When data is written to the local RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

(d) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(2) Reading RAM data

(a) Set the ECCDIS bit in the local RAM ECC control register to 1 to disable ECC error detection and correction.

(b) Read the local RAM. Since neither error detection nor correction proceeds when the local RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(c) Set the ECCDIS bit in the local RAM ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and address parity bits

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL bit in the local RAM test control register to 1 to select the ECC bits and the address parity bit for access when writing.

(c) When data is written to the local RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bits [6:0] and bits [8:7] are respectively written to the ECC bits and to the address parity bit.

CAUTION

Writing desired values to the RAM data and the address parity bit should be in order of 1) writing the RAM data and then 2) writing the address parity bit.

When writing desired values to the RAM data and ECC bits, you can start either by writing to the RAM data or ECC bits.

How to exit this test mode:

(d) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(4) Reading the ECC and address parity bits

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) When the local RAM is read, the ECC and address parity bits are stored in the bank corresponding to local RAM test data read buffer 0 or local RAM test data read buffer 1.

How to exit this test mode:

(c) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(5) Self-diagnosis of the ECC check function

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the local RAM in normal mode and checking the result of error correction or detection.

(6) Self-diagnosis of the address parity check function

Self-diagnosis is enabled by following either of the two procedures below.

(a) Setting APTESTi (i = 0, 1, 2, 3) in the local RAM test control register to 1 inverts the result of address parity generation for the corresponding bank. That is, a fault can be injected to the address parity generator. Writing to the corresponding bank in the local RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.

(b) Desired data can be written to the address parity bit through the procedure described in (4) above. This enables injection of a 1-bit or 2-bit fault to the address parity bit by inverting the address parity bit. After that, self-diagnosis of the address parity bit checking function for reading is possible by reading the local RAM in normal mode and checking the result of parity error detection.

27.2.5 Global RAM ECC and Address Parity

27.2.5.1 Overview

The global RAM ECC is summarized in the table below.

Table 27.37 Overview of the Global RAM ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. The register values after a reset enable the full functionality, that is, the detection of 2-bit errors and the detection and correction of 1-bit errors.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the parity bit is written to only one location in the RAM. During data read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the memory. The register values after a reset enable this function.
Error notification	Indicators of the occurrence of ECC and address parity errors are conveyed to the ECM. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. The register values after a reset enable error notification upon detection of a 2-bit ECC error, and disable error notification upon detection of a 1-bit ECC error. **Parity Error:** • Error notification can be either enabled or disabled upon detection of an address parity error. The register values after a reset enable error notification upon detection of an address parity error. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. Data in the RAM and the ECC and address parity bits can be read directly.

The ECC encoder and decoder and the address parity generator are provided for the access ports (CPU1, CPU2, and interconnect) that are connected to the global RAM. Separate address checkers are provided for bank A and bank B of the global RAM. Also, the ECC decoder and ECC encoder for RMW processing* are provided for each of bank A and bank B. See **Figure 27.2**.

- RMW processing

  Bit manipulation instructions, 2-byte writing and 1-byte writing are executed in three steps: 1) reading of 32-bit data; 2) generation of write data by replacement of (modifying) the predetermined data; and 3) writing of 32-bit data. In this section, such operations are referred to as read-modify-write (RMW) processing. RMW processing for the global RAM proceeds in the controller for each bank.

  In RWM processing, the ECC is decoded for the read operation in step 1), while the ECC is encoded for the write operation in step 3).

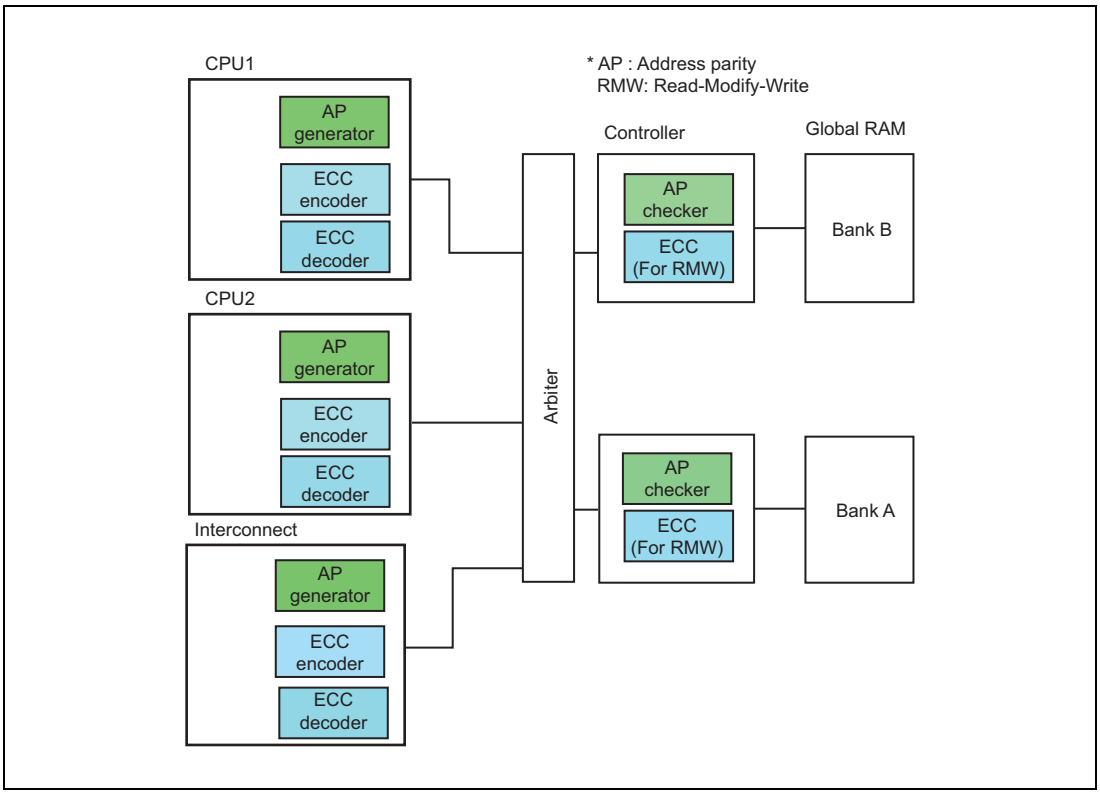


Figure 27.2 ECC of Global RAM and Address Parity

Up to 64 bits of data can be simultaneously read from or written to the global RAM. Meanwhile, the ECC and address parity are provided for each 32-bit data. That is, two each of the ECC decoder, ECC encoder, ECC circuit (for RMW), and address parity checker in **Figure 27.2** are provided, one for the 32 higher-order bits and the other for the 32 lower-order bits.

Table 27.38 Address and Corresponding ECC Circuit

3 Lower-Order Bits of the Address	7_H to 4_H	3_H to 0_H
Corresponding ECC circuit	32 higher-order bits	32 lower-order bits

CAUTION

In the local RAM (CPU1, CPU2), the locations for storage of data for units of the ECC are referred to as banks 0 to 3. Since the global RAM is divided into banks A and B with the address FEF0 0000_H as the boundary, banks A and B are respectively referred to as the 32 higher-order bits and the 32 lower-order bits to reduce confusion.

27.2.5.2 List of Registers

Table 27.39 List of Registers

Module Name	Address	Symbol*1	Register Name	R/W	Value after Reset	Access Size
ECCGRAM	FFC6 4000$_H$	GRECCCTL_GRAMC	Global RAM ECC control register (GRAMC)	R/W	0000 0000$_H$	16/32
ECCGRAM	FFC6 4004$_H$	GRTSTCTL	Global RAM test control register	R/W	0000 0000$_H$	16/32
ECCGRAM	FFC6 4008$_H$	GRTDATBF0	Global RAM test data read buffer 0	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 400C$_H$	GRTDATBF1	Global RAM test data read buffer 1	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4010$_H$	GRTDATBF2	Global RAM test data read buffer 2	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4014$_H$	GRTDATBF3	Global RAM test data read buffer 3	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4018$_H$	GRDECINBF0	Global RAM ECC decoder input data buffer 0	R/W	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 401C$_H$	GRDECINBF1	Global RAM ECC decoder input data buffer 1	R/W	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4200$_H$	GRECCCTL_VCI	Global RAM ECC control register (VCI)	R/W	0000 0000$_H$	16/32
ECCGRAM	FFC6 4204$_H$	GRERRINT_VCI	Global RAM error information control register (VCI)	R/W	0000 0006$_H$	8/16/32
ECCGRAM	FFC6 4208$_H$	GRSTCLR_VCI	Global RAM status clear register (VCI)	W	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 420C$_H$	GROVFSTR_VCI	Global RAM error count overflow status register (VCI)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4210$_H$	GR1STERSTR_VCI	Global RAM 1st error status register (VCI)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4250$_H$	GR1STEADR0_VCI	Global RAM 1st error (lower 32-bit data) address register (VCI)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4254$_H$	GR1STEADR1_VCI	Global RAM 1st error (upper 32-bit data) address register (VCI)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4400$_H$	GRECCCTL_PE1	Global RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
ECCGRAM	FFC6 4404$_H$	GRERRINT_PE1	Global RAM error information control register (PE1)	R/W	0000 0006$_H$	8/16/32
ECCGRAM	FFC6 4408$_H$	GRSTCLR_PE1	Global RAM status clear register (PE1)	W	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 440C$_H$	GROVFSTR_PE1	Global RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4410$_H$	GR1STERSTR_PE1	Global RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4450$_H$	GR1STEADR0_PE1	Global RAM 1st error (lower 32-bit data) address register (PE1)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4454$_H$	GR1STEADR1_PE1	Global RAM 1st error (upper 32-bit data) address register (PE1)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4600$_H$	GRECCCTL_PE2	Global RAM ECC control register (PE2)	R/W	0000 0000$_H$	16/32
ECCGRAM	FFC6 4604$_H$	GRERRINT_PE2	Global RAM error information control register (PE2)	R/W	0000 0006$_H$	8/16/32
ECCGRAM	FFC6 4608$_H$	GRSTCLR_PE2	Global RAM status clear register (PE2)	W	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 460C$_H$	GROVFSTR_PE2	Global RAM error count overflow status register (PE2)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4610$_H$	GR1STERSTR_PE2	Global RAM 1st error status register (PE2)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4650$_H$	GR1STEADR0_PE2	Global RAM 1st error (lower 32-bit data) address register (PE2)	R	0000 0000$_H$	8/16/32
ECCGRAM	FFC6 4654$_H$	GR1STEADR1_PE2	Global RAM 1st error (upper 32-bit data) address register (PE2)	R	0000 0000$_H$	8/16/32

Note 1. The registers with symbols "_VCI", "_PE1", and "_PE2" as suffixes are provided to the ECC controller for each access port: the registers with "_VCI" are provided to the ECC controller for access from the system interconnect 1 to the global RAM, the registers with "_PE1" are provided to the ECC controller for access from the CPU1 to the global RAM, and the registers with "_PE2" are provided to the ECC controller for access from the CPU2 to the global RAM. The registers with "_GRAMC" are control registers that are common to all access ports.

27.2.5.3 Details of Registers

(1) GRECCCTL_GRAMC — Global RAM ECC Control Register (GRAMC)

GRECCCTL_GRAMC is the ECC and address parity control register shared by global RAMs. GRECCCTL_GRAMC enables or disables address parity check and specifies ECC processing for Read-Modify-Write (RMW) processing. Set the PROT[1:0] bits to 01_B when writing to GRECCCTL_GRAMC.

GRECCCTL_GRAMC is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	APARIDIS	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 27.40 GRECCCTL_GRAMC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 3	—	Reserved. These bits are always read as 0. The write value should also be 0.
2	APARIDIS	Address Parity Check Disable Enables or disables address parity check. In the initial state, the parity check is enabled. 0: Enables address parity check. 1: Disables address parity check.
1	SECDIS	1-Bit Error Correction Disable for RMW Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable for RMW Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction. **CAUTION** The encoding function is enabled even though the error detection and correction are disabled.

(2) GRTSTCTL — Global RAM Test Control Register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC and address parity bits. The DATSEL0 or DATSEL1 bit is used to select the RAM data or the ECC and address parity bits. Also, input and output by the ECC decoder in the global RAM controller can be controlled for testing (self-diagnosis).

By setting address parity test mode (APTESTi = 1; i = 0, 1, 2, 3), the parity to be input to the address parity checker is inverted. Set the PROT[1:0] bits to 01B when writing to GRTSTCTL.

This register is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	APTEST3	APTEST2	APTEST1	APTEST0	ECCTST	DECINEN	DATSEL	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.41 GRTSTCTL Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 8	—	Reserved. These bits are always read as 0. The write value should also be 0.
7	APTEST3	Address Parity Checker (BankB, 32 higher-order bits) Test Sets the address parity checker to test mode. When APTEST3 = 1, the parity generated through the address parity generator is inverted.
6	APTEST2	Address Parity Checker (BankB, 32 lower-order bits) Test Sets the address parity checker to test mode. When APTEST2 = 1, the parity generated through the address parity generator is inverted.
5	APTEST1	Address Parity Checker (BankA, 32 higher-order bits) Test Sets the address parity checker to test mode. When APTEST1 = 1, the parity generated through the address parity generator is inverted.
4	APTEST0	Address Parity Checker (BankA, 32 lower-order bits) Test Sets the address parity checker to test mode. When APTEST0 = 1, the parity generated through the address parity generator is inverted.
3	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.
2	DECINEN	ECC Decoder Error Injection Enable for RMW This bit is valid when ECCTST = 1. This bit enables input of the value in the ECC decoder input buffer register to the ECC decoder for use in updating of data at the time of RMW access. 0: The value in the ECC decoder input buffer register is not selected. 1: The value in the ECC decoder input buffer register is selected.

Table 27.41 GRTSTCTL Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	DATSEL	Read Buffer Storage Data Select 0 and 1 These bits are valid when ECCTST = 1. These bits select the value to be stored in the read buffer GRTDATBFn and the value to be written to each field. 00: – GRTDATBFn: When reading involves an RMW cycle, the ECC and address parity bits are stored. – Global RAM: When writing involves an RMW cycle, the data area that is updated depends on the unit of access and the location accessed. The ECC bits are not updated. 01: – GRTDATBFn: When reading involves an RMW cycle, the ECC and address parity bits are stored. – Global RAM: When writing involves an RMW cycle, only the ECC and address parity bits are updated. The data area is not updated. 10: – GRTDATBFn: When access is RMW, this register holds the result of ECC decoding for the data read out in the read portion of the RMW cycle. Its value is not updated in the case of access that is not RMW. – Global RAM: Operates in the same way as in normal operating mode (i.e. when ECCTST = 0). 11: – GRTDATBFn : When access is RMW, this register holds the result of ECC decoding for use in the updating of data at the time of RMW access. Its value is not updated in the case of access that is not RMW. – Global RAM: Operates in the same way as in normal operating mode (i.e. when ECCTST = 0). In any case, the result of reading by the CPU, DMAC, etc. is the same value as would normally be read out.

(3) GRTDATBFn — Global RAM Test Data Read Buffer n (n = 0 to 3)

In test mode (ECCTST = 1), data in the RAM, the ECC bits, address parity bit, and ECC decoder output can all be read. When the RAM is read, the value selected by the DATSEL1 or DATSEL0 bit of the global RAM test control register is stored in this buffer.

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

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

Table 27.42 GRTDATBFn Register Contents

Bit Position	Bit Name	Function
31 to 0	GRTDATBF	These bits are valid when while ECCTST = 1 (selecting test mode) in the global RAM test control register. When (DATSEL1, DATSEL0) = (0, 0) or (0, 1) When reading from the RAM, the ECC bits and the address parity bit are respectively stored in GRTDATBF[6:0] and GRTDATBF[7]. 0 is stored in GRTDATBF[31:8]. When (DATSEL1, DATSEL0) = (1, 0) When access is RMW, the output data from the ECC decoder for reading (after updating) are stored in GRTDATBF[31:0]. When (DATSEL1, DATSEL0) = (1, 1) When access is RMW, the output data from the ECC decoder for use in updating of data (after updating) are stored in GRTDATBF[31:0].

Note: n = 0: BankA, 32 lower-order bits
n = 1: BankA, 32 higher-order bits
n = 2: BankB, 32 lower-order bits
n = 3: BankB, 32 higher-order bits

(4) GRECCCTL_VCI/PE1/PE2 — Global RAM ECC Control Register (VCI/PE1/PE2)

GRECCCTL_VCI/PE1/PE2 enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01_B when writing to GRECCCTL_VCI/PE1/PE2.

The setting of this register is used for accesses through the respective access port.

GRECCCTL_VCI/PE1/PE2 is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 27.43 GRECCCTL_VCI/PE1/PE2 Register Contents

Bit Position	Bit Name	Function
13 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(5) GRDECINBF0 — Global RAM ECC Decoder Input Data Buffer 0

This register (GRDECINBF0) holds input data for the ECC decoder to use in updating the data at the time of RMW access. The value of this register is treated by the ECC decoder as if it were 32 bits of data from RAM.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRDECINBF0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRDECINBF0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.44 GRDECINBF0 Register Contents

Bit Position	Bit Name	Function
31 to 0	GRDECINBF0	These bits are valid when ECCTST = 1 (selecting test mode) in the global RAM test control register. When DECINEN = 1, the value of this register is input to the ECC decoder as data for use in updating in response to the execution of an RMW instruction. The value is treated as if it were 32 bits of data from RAM. This register is common to banks A and B and provides the values for both the 32 higher-order bits and the 32 lower-order bits.

(6) GRDECINBF1 — Global RAM ECC Decoder Input Data Buffer 1

This register (GRDECINBF1) holds input data for the ECC decoder to use in updating the data at the time of RMW access. The value of this register is treated by the ECC decoder as if it were 7 bits of data from ECC.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	GRDECINBF1						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.45 GRDECINBF1 Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved. These bits are always read as 0. The write value should also be 0.
6 to 0	GRDECINBF1	These bits are valid when ECCTST = 1 (selecting test mode) in the global RAM test control register. When DECINEN = 1, the value of this register is input to the ECC decoder as data for use in updating in response to the execution of an RMW instruction. The value is treated as if it were 7 bits of data from ECC. This register is common to banks A and B and provides the values for both the 32 higher-order bits and the 32 lower-order bits.

(7) GRERRINT_VCI/PE1/PE2 — Global RAM Error Information Control Register

GRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

The setting of this register is used for accesses through the respective access port.

GRERRINT is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 27.46 GRERRINT Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved. These bits are always read as 0. The write value should also be 0.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(8) GRSTCLR_VCI/PE1/PE2 — Global RAM Status Clear Register

GRSTCLR clears the error flags in the error status register (GR1STERSTR), the overflow flag in the error count overflow status register (GROVFSTR), and the error address register (GR1STEADR). GRSTCLR is a write-only register and is always read as 0.

The setting of this register is used for accesses through the respective access port.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 27.47 GRSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	STCLR1	Error Overflow Flag Clear (for the 32 higher-order bits) Writing 1 to this bit clears the EXDEDF1, EXSEDF1, APEF1, DEDF1, and SEDF1 flags in GR1STERSTR; ERROVF1 flag in GROVFSTR; and GR1STEADR1.
0	STCLR0	Error Overflow Flag Clear (for the 32 lower-order bits) Writing 1 to this bit clears the EXDEDF0, EXSEDF0, APEF0, DEDF0, and SEDF0 flags in GR1STERSTR; ERROVF0 flag in GROVFSTR; and GR1STEADR0.

(9) GROVFSTR_VCI/PE1/PE2 — Global RAM Error Count Overflow Status Register

GROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.48 GROVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	ERROVF1	Error Overflow Flag (for the 32 higher-order bits) This flag is set if the second error occurs while any of the error flags (EXDEDF1, EXSEDF1, APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for the 32 lower-order bits) This flag is set if the second error occurs while any of the error flags (EXDEDF0, EXSEDF0, APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(10) GR1STERSTR_VCI/PE1/PE2 — Global RAM 1st Error Status Register

GR1STERSTR monitors occurrence of the first error. The error status is set if an error occurs while the error flag is 0. The error flag is overwritten if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error monitor flag is set.

If more than one error occurs simultaneously, all the corresponding error flags are set. GR1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EXDEDF1	EXSEDF1	—	APEF1	DEDF1	SEDF1	—	—	EXDEDF0	EXSEDF0	—	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.49 GR1STERSTR Register Contents

Bit Position	Bit Name	Function
7+8×n 6+8×n	—	Reserved. These bits are always read as 0. The write value should also be 0.
5+8×n	EXDEDFn	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLRn bit to 1 in GRSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0 (during RMW processing for the global RAM).
4+8×n	EXSEDFn	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLRn bit to 1 in GRSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags EXDEDFn, EXSEDFn, APEFn, DEDFn, and SEDFn are 0 (during RMW processing for the global RAM).
3+8×n	—	Reserved. These bits are always read as 0. The write value should also be 0.
2+8×n	APEFn	Address Parity Error Monitor Flag 0: Cleared to 0 by setting the STCLRn bit to 1 in GRSTCLR. 1: Indicates that an address parity error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0. **CAUTION** This flag is set equally for read and write; the error generation source is not considered.
1+8×n	DEDFn	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLRn bit to 1 in GRSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0.
0+8×n	SEDFn	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLRn bit to 1 in GRSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags EXDEDFn, EXSEDFn, APEFn, DEDFn, and SEDFn are 0.

Note: n = 0, 1, n = 0 represents the 32 lower-order bits and n = 1 represents the 32 higher-order bits.

(11) GR1STEADRn_VCI/PE1/PE2 — Global RAM 1st Error Address Register n (n = 0, 1)

GR1STEADRn holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[20:0] of this register correspond to bits [20:0] of the real address. The real address can be calculated by appending the higher-order address bits [31:21] as a base address. GR1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR. During accesses to the lower 32-bit data, the address is stored in GR1STEADR0. During accesses to the upper 32-bit data, the address is stored in GR1STEADR1.

The setting of this register is used for accesses through the respective access port.

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

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

Table 27.50 GR1STEADRn Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved. These bits are always read as 0. The write value should also be 0.
20 to 0	EADR[20:0]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

27.2.5.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits. Also, data in the RAM and the ECC and address parity bits and the ECC decoder output data for RMW can all be read.

It is possible to input the desired data in the ECC decoder for RMW.

(1) Writing RAM data

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL1 to 0 and DATSEL0 to 0 in the global RAM test control register to select RAM data for access when writing.

(c) When data is written to the global RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

(d) Set the ECCTST bit in the global RAM test control register to 0 to disable test mode (normal mode).

(2) Reading RAM data

(a) Set the ECCDIS bit in the global RAM ECC control register to 1 to disable ECC error detection and correction.

(b) Read the global RAM. Since neither error detection nor correction proceeds when the global RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(c) Set the ECCDIS bit in the global RAM ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and address parity bits

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL1 bit to 0 and DATSEL0 bit to 1 in the global RAM test control register to select the ECC bits and the address parity bit for access when writing.

(c) When data is written to the global RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bits [6:0] and bit [7] are respectively written to the ECC bits and to the address parity bit.

CAUTION

Writing desired values to the RAM data and the address parity bit should be in order of 1) writing the RAM data and then 2) writing the address parity bit.

When writing desired values to the RAM data and ECC bits, you can start either by writing to the RAM data or ECC bits.

How to exit this test mode:

(d) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(4) Reading the ECC and address parity bits

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Setting the DATSEL1 and DATSEL0 bits in the global RAM test control register to 0 and 1, respectively, to select the ECC and address parity bits for access when reading.

(c) When data in the global RAM is read, the ECC and address parity bits are stored in the corresponding register from among global RAM test data read buffers 0 to 3.

How to exit this test mode:

(d) Set the ECCTST bit in the global RAM test control register to 0 to disable test mode (normal mode).

(5) Self-diagnosis of the ECC check function for the access ports

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the global RAM in normal mode and checking the result of error correction or detection.

(6) Self-diagnosis of the address parity check function

(a) Self-diagnosis is enabled by following either of the two procedures below.
Setting APTESTi (i = 0, 1, 2, 3) in the global RAM test control register to 1 inverts the result of address parity generation for the corresponding data area (the 32 higher-order bits or the 32 lower-order bits) of the corresponding bank (bank A or B). That is, a fault can be injected to the address parity generator. Writing to the corresponding data area of the corresponding bank in the global RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.

(b) Desired data can be written to the address parity bits through the procedure described in (4) above. This enables injection of a fault to the address parity bit by inverting the address parity bit. After that, self-diagnosis of the address parity bit checking function for reading is possible by reading the global RAM in normal mode and checking the result of parity error detection.

(7) Self-diagnosis of the ECC decoder for the data read out in an RMW operation

(a) Suitable erroneous values are injected as RAM data or to the ECC bits by following procedure (1) or (3) above.

(b) Setting the DATSEL1 and DATSEL0 bits in the test control registers for the global RAM to 1 and 0, respectively, makes the output data from the ECC decoder for the data read out in an RMW operation the target for reading.

(c) After RMW processing for the global RAM proceeds, the data read out in an RMW operation for global RAM are stored in the corresponding register from among global RAM test data read buffers 0 to 3. Checking the result allows self-diagnosis of the ECC decoder for the data read out in an RMW operation.

(8) Self-diagnosis of the ECC decoder for the data being updated at the time of RMW access

(a) Setting the DATSEL1 and DATSEL0 bits in the global RAM test control register to 1 makes the output data from the ECC decoder for the data being updated at the time of RMW access the target for reading.

(b) Through the above setting, the input data from the ECC decoder for the data being updated at the time of RMW access is switched to ECC decoder input buffer 0 or 1 (GRDECINBF0 or 1) from write data sent from the access ports. As a result, suitable erroneous values can be injected by setting an appropriate value in ECC decoder input buffer 0 or 1.

(c) After RMW processing for the global RAM proceeds, the data being updated at the time of RMW access for global RAM are stored in the corresponding register from among global RAM test data read buffers 0 to 3. Checking the result allows self-diagnosis of the ECC decoder when data are updated.

27.2.6 Instruction Cache ECC and EDC

27.2.6.1 Overview

The instruction cache ECC is summarized in the table below.

Table 27.51 Overview of the Instruction Cache ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. The register values after a reset enable the full functionality, that is, the detection of 2-bit errors and the detection and correction of 1-bit errors.
Address parity	None
Error notification	An indicator of the occurrence of an ECC error is conveyed to the ECM. **ECC Error:** • Enabling or disabling of error notification upon detection of a 2-bit ECC error can be selected. • Enabling or disabling of error notification upon detection of a 1-bit ECC error can be selected. The register values after a reset disable notification of the 2-bit ECC error and disable notification of the 1-bit ECC error. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the states of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status is set, the corresponding status is set. The error status can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status serves as the enable bit of the capture address.
Self-diagnosis	A cache instruction is used to write the desired values as RAM data and to the ECC bits. Similarly, data in the RAM and the ECC bits can be read directly. Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

27.2.6.2 List of Registers

Table 27.52 List of Registers (1/2)

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
ECCIC1	FFC6 0400$_H$	IDECCCTL_PE1	Instruction cache data RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
ECCIC1	FFC6 0404$_H$	IDERRINT_PE1	Instruction cache data RAM error information control register (PE1)	R/W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0408$_H$	IDSTCLR_PE1	Instruction cache data RAM error status clear register (PE1)	W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 040C$_H$	IDOVFSTR_PE1	Instruction cache data RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0410$_H$	ID1STERSTR_PE1	Instruction cache data RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0450$_H$	ID1STEADR0_PE1	Instruction cache data RAM (bank 0) 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0454$_H$	ID1STEADR1_PE1	Instruction cache data RAM (bank 1) 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0600$_H$	IDECCCTL_PE2	Instruction cache data RAM ECC control register (PE2)	R/W	0000 0000$_H$	16/32
ECCIC1	FFC6 0604$_H$	IDERRINT_PE2	Instruction cache data RAM error information control register (PE2)	R/W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0608$_H$	IDSTCLR_PE2	Instruction cache data RAM error status clear register (PE2)	W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 060C$_H$	IDOVFSTR_PE2	Instruction cache data RAM error count overflow status register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0610$_H$	ID1STERSTR_PE2	Instruction cache data RAM 1st error status register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0650$_H$	ID1STEADR0_PE2	Instruction cache data RAM (bank 0) 1st error address register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 0654$_H$	ID1STEADR1_PE2	Instruction cache data RAM (bank 1) 1st error address register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1400$_H$	ITECCCTL_PE1	Instruction cache tag RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
ECCIC1	FFC6 1404$_H$	ITERRINT_PE1	Instruction cache tag RAM error information control register (PE1)	R/W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1408$_H$	ITSTCLR_PE1	Instruction cache tag RAM error status clear register (PE1)	W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 140C$_H$	ITOVFSTR_PE1	Instruction cache tag RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1410$_H$	IT1STERSTR_PE1	Instruction cache tag RAM 1st error status clear register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1450$_H$	IT1STEADR0_PE1	Instruction cache tag RAM 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1600$_H$	ITECCCTL_PE2	Instruction cache tag RAM ECC control register (PE2)	R/W	0000 0000$_H$	16/32
ECCIC1	FFC6 1604$_H$	ITERRINT_PE2	Instruction cache tag RAM error information control register (PE2)	R/W	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1608$_H$	ITSTCLR_PE2	Instruction cache tag RAM error status clear register (PE2)	W	0000 0000$_H$	8/16/32

Table 27.52 List of Registers (2/2)

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
ECCIC1	FFC6 160C$_H$	ITOVFSTR_PE2	Instruction cache tag RAM error count overflow status register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1610$_H$	IT1STERSTR_PE2	Instruction cache tag RAM 1st error status register (PE2)	R	0000 0000$_H$	8/16/32
ECCIC1	FFC6 1650$_H$	IT1STEADR0_PE2	Instruction cache tag RAM 1st error address register (PE2)	R	0000 0000$_H$	8/16/32

27.2.6.3 Details of Registers

(1) IDECCCTL_PE1/PE2 — Instruction Cache Data RAM ECC Control Register

IDECCCTL enables or disables ECC error detection and correction and 1-bit error correction for cache data RAM. Set the PROT[1:0] bits to 01_B when writing to IDECCCTL.

IDECCCTL is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R\W	R\W

Table 27.53 IDECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(2) IDERRINT_PE1/PE2 — Instruction Cache Data RAM Error Information Control Register

IDERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache data RAM.

IDERRINT is initialized by an internal reset or an external reset.

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

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

Table 27.54 IDERRINT Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(3) IDSTCLR_PE1/PE2 — Instruction Cache Data RAM Error Status Clear Register

IDSTCLR clears the error flags in the error status register (ID1STERSTR), the overflow flag in the error count overflow status register (IDOVFSTR), and the error address register (ID1STEADR). IDSTCLR is a write-only register and is always read as 0.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 27.55 IDSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	STCLR1	Error Overflow Flag Clear (for bank 1) Writing 1 to this bit clears the DEDF1and SEDF1 flags in ID1STERSTR; ERROVF1 flag in IDOVFSTR; and ID1STEADR1.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the DEDF0and SEDF0 flags in ID1STERSTR; ERROVF0 flag in IDOVFSTR; and ID1STEADR0.

(4) IDOVFSTR_PE1/PE2 — Instruction Cache Data RAM Error Count Overflow Status Register

IDOVFSTR monitors occurrence of error overflow in cache data RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.56 IDOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	ERROVF1	Error Overflow Flag (for bank 1) This flag is set if the second error occurs while any of the error flags (DEDF1 and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(5) ID1STERSTR_PE1/PE2 — Instruction Cache Data RAM 1st Error Status Register

ID1STERSTR monitors occurrence of the first error in cache data RAM. The error status is set if an error occurs while all the error flags for the same banks are 0. The applicable error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error monitor flag is set.

This register can not identify the WAY in which an error occurs.

ID1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDF1	SEDF1	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.57 ID1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 10	—	Reserved. These bits are always read as 0. The write value should also be 0.
9	DEDF1	2-Bit ECC Error Monitor Flag (for bank 1) 0: Cleared to 0 by setting the STCLR1 bit to 1 in IDSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flag DEDF1 is 0.
8	SEDF1	1-Bit ECC Error Monitor Flag (for bank 1) 0: Cleared to 0 by setting the STCLR1 bit to 1 in IDSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags DEDF1 and SEDF1 are 0.
7 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag (for bank 0) 0: Cleared to 0 by setting the STCLR0 bit to 1 in IDSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag (for bank 0) 0: Cleared to 0 by setting the STCLR0 bit to 1 in IDSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(6) ID1STEADRn_PE1/PE2 — Instruction Cache Data RAM (Bank n) 1st Error Address Register (n = 0, 1)

ID1STEADR holds the address at which an error has occurred in cache data RAM.

The error address is set if an error occurs while all the error flags for the relevant banks are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set for a first error. Once a 2-bit ECC error occurs, the address is not updated.

ID1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

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

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

Table 27.58 ID1STEADRn Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved. These bits are always read as 0. The write value should also be 0.
8 to 0	EADRn[8:0]	1st Error Address (for bank n) These bits monitor the address of the first error. EADR[6:0] indicate the cache address [10:4]. EADR[7] is fixed to 0. EADR[8] indicates Way group. The error address is set if an error occurs while all the error flags for bank n are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address is not updated. For configuration of the instruction cache, see **Section 3, CPU System**.

(7) ITECCCTL_PE1/PE2 — Instruction Cache Tag RAM ECC Control Register

ITECCCTL enables or disables ECC error detection in cache tag RAM. Set the PROT[1:0] bits to 01_B when writing to ITECCCTL.

ITECCCTL is initialized by an internal reset or an external reset.

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

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

Table 27.59 ITECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writing to this register. These bits are always read as 0.
13 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	ECCDIS	ECC Disable Enables or disables ECC error detection 0: Enables ECC error detection. 1: Disables ECC error detection.

(8) ITERRINT_PE1/PE2 — Instruction Cache Tag RAM Error Information Control Register

ITERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache tag RAM.

ITERRINT is initialized by an internal reset or an external reset.

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

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

Table 27.60 ITERRINT Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection is enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection is enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(9) ITSTCLR_PE1/PE2 — Instruction Cache Tag RAM Error Status Clear Register

ITSTCLR clears the error flags in the error status register (IT1STERSTR), the overflow flag in the error count overflow status register (ITOVFSTR), and the error address register (IT1STEADR). ITSTCLR is a write-only register and is always read as 0.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 27.61 ITSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the DEDF0 and SEDF0 flags in IT1STERSTR; ERROVF0 flag in ITOVFSTR; and IT1STEADR0.

(10) ITOVFSTR_PE1/PE2 — Instruction Cache Tag RAM Error Count Overflow Status Register

ITOVFSTR monitors occurrence of error overflow in cache tag RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.62 ITOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(11) IT1STERSTR_PE1/PE2 — Instruction Cache Tag RAM 1st Error Status Register

IT1STERSTR monitors occurrence of the first error in cache tag RAM.

The error status is set if an error occurs while all the error flags are 0. Also the applicable error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error monitor flag is set.

This register can not identify the WAY in which an error occurs.

IT1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.63 IT1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLR0 bit to 1 in ITSTCLR. 1: Indicates that a 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag 0: Cleared to 0 by setting the STCLR0 bit to 1 in ITSTCLR. 1: Indicates that a 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(12) IT1STEADR0_PE1/PE2 — Instruction Cache Tag RAM 1st Error Address Register

IT1STEADR0 holds the address at which an error has occurred in cache tag RAM.

The error address is set if an error occurs while all the error flags are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address is not updated.

IT1STEADR0 is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

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

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

Table 27.64 IT1STEADR0 Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved. These bits are always read as 0. The write value should also be 0.
8 to 0	EADR[8:0]	1st Error Address These bits monitor the address of the first error. EADR[6:0] indicate the cache address [10:4]. EADR[7] is fixed to 0. EADR[8] indicates Way group. The error address is set if an error occurs while all the error flags are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated. For configuration of the instruction cache, see **Section 3, CPU System**.

27.2.6.4 Test Function

A cache instruction is used to write the desired values as RAM data and to the ECC bits, and read data in the RAM and the ECC bits directly.

Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

For details, see *RH850G3M Family User's Manual: Software*.

27.2.7 DTS RAM ECC

See **Section 7, DMA Controller**.

27.2.8 ECC for Peripheral RAM (32 Bits)

27.2.8.1 Overview

This is an ECC module for the RAM of the following peripheral modules.

RS-CAN and CSIH

Error Detection and Correction

Seven-bit ECC data is appended to the 32-bit RAM data.

This ECC module provides 2-bit ECC error detection and 1-bit ECC error detection and correction.

CAUTION

This module is not capable of reliably detecting errors in three or more bits.

If errors occur in three or more bits, the module may detect the errors as 1- or 2-bit ECC errors or not detect any errors. Depending on the settings, this may lead to the correction of a bit that was not actually inverted.

Enabling or Disabling ECC Error Detection and Correction

- ECC error detection can be either enabled or disabled.
- One-bit ECC error correction can be either enabled or disabled.
- If all the bits of RAM output data are stuck at 0 or 1, it is detected as a 2-bit ECC error.

Error Notification

- The ECM is notified when 2-bit ECC errors are detected (this can be enabled or disabled).
- The ECM is notified when 1-bit ECC errors are detected (this can be enabled or disabled).

Once the ECM has been notified of an error, even if another ECC error is detected, the ECM is not notified until the error status bit corresponding to the initial error is cleared.

Error Status

- Detection of 2- and 1-bit ECC errors can be monitored.
- Special registers are provided to clear error status.

Address Capture

- Only one address at which an ECC error has occurred can be captured.
- When a 2-bit or 1-bit ECC error is detected, the error-causing address is captured (when the first (1-bit or 2-bit) error is detected after the flag is cleared).

Testing Function (Error Insertion)

- By setting the test mode, register values can be used as the data to be output to the RAM. When a peripheral module writes to the RAM, the ECEDB[31:0] register value can be written to the RAM data section, and the ECERDB[6:0] register value can be written to the ECC redundant bit section.
- By setting the test mode, the ECC redundant bit section can be latched when RAM data is read, and the value can be confirmed.
- By setting the test mode, the ECC redundant bit (encoding circuit) and syndrome code (decoding circuit), which are generated from the input data, can be confirmed.

27.2.8.2 List of Registers

(1) List of ECC Modules

The RAMs of the multiple peripheral functions are provided with the ECC modules. The following table shows the peripheral functions provided with the ECC modules, the corresponding ECC module names, and base addresses of the ECC modules.

Table 27.65 Modules Provided with the ECC and Base Addresses

Supported Peripheral Functions	ECC Module Names and Register Base Addresses			
	Master Side*1		Checker Side*1	
	Module Name	Base Address <base_addr>	Module Names	Base Address <base_addr>
RS-CAN	E7RC0M	FFC7 1000_H	E7RC0C	FFC7 1200_H
CSIH0	E7CS0M	FFC7 0000_H	E7CS0C	FFC7 0200_H
CSIH1	E7CS1M	FFC7 0400_H	E7CS1C	FFC7 0600_H

Note 1. Two ECC modules are provided to support BIST, one for the master and the other for the checker. For details, refer to **Section 27.7, BIST**.

(2) List of Registers

Each ECC module has the registers shown in the following table.

Table 27.66 List of Registers

Module Name	Register Name	Symbol	R/W	Value after Reset	Address	Access Size
E7RC0M/ E7CS0M/ E7CS1M	ECC control register*1	E710CTL	R/W	$001X_H$	<base_addr>+00_H	16/8
E7RC0M/ E7CS0M/ E7CS1M	ECC test mode control register	E710TMC	R/W	0000_H	<base_addr>+04_H	16/8
E7RC0M/ E7CS0M/ E7CS1M	ECC redundant bit data control test register	E710TRC	R/W	0000 0000_H	<base_addr>+08_H	32
E7RC0M/ E7CS0M/ E7CS1M	ECC encoder and decoder data test register	E710TED	R/W	0000 0000_H	<base_addr>+$0C_H$	32
E7RC0M/ E7CS0M/ E7CS1M	ECC error address register	E710EAD	R/W	0000 0000_H	<base_addr>+10_H	32

Note 1. The reset value of the LSB in the ECC control register is undefined.

(3) Register Map

Table 27.67 Register Map

Abbreviation	31 24	23 16	15 8	7 0	Address
E710CTL	— (00_H)	— (00_H)	ECCTL[15:8]	ECCTL[7:0]	nn00_H
E710TMC	— (00_H)	— (00_H)	ECTMC[15:8]	ECTMC[7:0]	nn04_H
E710TRC	ECSYND[7:0]	ECHORD[7:0]	ECECRD[7:0]	ECERDB[7:0]	nn08_H
E710TED	ECEDB[31:24]	ECEDB[23:16]	ECEDB[15:8]	ECEDB[7:0]	nn$0C_H$
E710EAD	ECEAD[31:24]	ECEAD[23:16]	ECEAD[15:8]	ECEAD[7:0]	nn10_H

27.2.8.3 Details of Registers

(1) E710CTL — ECC Control Register

E710CTL controls the status and modes of the ECC module.

E710CTL can be read and written to using the 16-bit or 8-bit manipulation instruction.

However, only the 16-bit manipulation instruction is valid for writing to bit 7.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	—	ECER2C	ECER1C	—	ECTHM	—	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	ECEMF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	Un-defined
R/W	W*1	W*1	R	R	R	W*1	W*1	R/W	R/W	R	R/W	R/W	R/W	R	R	R

Note 1. This bit is always read as 0.

Table 27.68 E710CTL Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	EMCA[1:0]	Access Control 1 and 0 to ECC Mode Select Bit These bits reserve the write trigger to bit 7. The read value is always 0.
10	ECER2C	2-Bit ECC Error Detection Flag Clear Clears the bit 2 (ECER2F) status flag. The read value is always 0, and writing 0 to ECER2C does not change the value. If writing 1 to ECER2C conflicts with the condition for setting bit 2, the former takes priority. Writing 1 to ECER2C while ECER2F is set clears ECER2F.
9	ECER1C	1-Bit ECC Error Detection Flag Clear Clears the bit 1 (ECER1F) status flag. The read value is always 0, and writing 0 to ECER1C does not change the internal state. If writing 1 to ECER1C conflicts with the condition for setting bit 1, the former takes priority. Writing 1 to ECER1C while ECER1F is set clears ECER1F.
7	ECTHM	ECC Function Disable Select Selects the ECC decoding operation. Write access to ECTHM is enabled when the value of bits 15 and 14 is 01_B. Therefore, only the 16-bit manipulation instruction is valid. Setting ECTHM to 1 disables error detection and bit correction. Here, if the data to be output to the peripheral module contains an error, the data is output without bit correction. Setting ECTHM to 1 has no effect on the encoder side. 0: ECC detection and correction enabled 1: ECC detection and correction disabled
5	EC1ECP	1-Bit ECC Error Correction Enable 0: Enables 1-bit error correction upon error detection. 1: Disables 1-bit error correction upon error detection.
4	EC2EDIC	2-Bit ECC Error Detection Notification Enable 0: When a 2-bit error is detected, the ECM is not notified of the error. 1: When a 2-bit error is detected, the ECM is notified of the error.
3	EC1EDIC	1-Bit ECC Error Detection Notification Enable 0: When a 1-bit error is detected, the ECM is not notified of the error. 1: When a 1-bit error is detected, the ECM is notified of the error.

Table 27.68 E710CTL Register Contents (2/2)

Bit Position	Bit Name	Function
2	ECER2F	2-Bit ECC Error Detection Flag Indicates that errors have been detected in two bits among bits 0 to 38 of data read from the RAM while error detection is enabled. This bit is read-only. 0: A 2-bit error has not occurred. 1: A 2-bit error has occurred. [Clearing conditions] (1) Reset is applied. (2) 1 is written to ECER2C. (3) ECC detection and correction being disabled (ECTHM = 1).
1	ECER1F	1-Bit ECC Error Detection And Correction Flag Indicates that an error has been detected in one bit among bits 0 to 38 of data read from the RAM while error detection is enabled. This bit is read-only. 0: A 1-bit error has not occurred. 1: A 1-bit error has occurred. [Clearing conditions] (1) Reset is applied. (2) 1 is written to ECER1C. (3) ECC detection and correction being disabled (ECTHM = 1).
0	ECEMF	ECC Error Message Flag Indicates that the current read data bus contains an error. ECEMF is updated every time RAM data is output. Since the RAM value after a reset is undefined, it is determined to be an error, and thus ECEMF may be set. Therefore, the ECEMF bit value after a reset is undefined. 0: The current read data bus contains no bit errors. 1: The current read data bus contains bit errors. ECEMF remains set as long as RAM data containing a bit error is output with error detection being enabled. [Clearing conditions] (1) ECC detection and correction being disabled (ECTHM = 1). (2) Decoding circuit input data contains no 1-bit errors.

(2) E710TMC — ECC Test Mode Control Register

E710TMC is a 16-bit register to switch the mode to test mode and controls the mode.

E710TMC can be read and written to using the 16-bit or 8-bit manipulation instruction.

However, only the 16-bit manipulation instruction is valid for writing to bit 7.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETMA[1:0]		—	—	—	—	—	—	ECTMCE	—	—	ECTRRS	ECREOS	ECENS	ECDCS	ECREIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Note 1. This bit is always read as 0.

Table 27.69 E710TMC Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	ETMA[1:0]	Access Control 1 and 0 to ECC Test Mode Control Enable Bit These bits reserve the write trigger to bit 7. The read value is always 0.
7	ECTMCE	ECC Test Mode Control Enable ECTMCE enables or disables access to the test registers and test control bits. Write access to ETCMCE is enabled when the value of bits 15 and 14 is 10_B. 0: Disables access to the test registers and test control bits. 1: Enables access to the test registers and test control bits.
4	ECTRRS	ECC RAM Read Test Mode Control Enable ECTRRS enables the read status of RAM to be generated by reading the E710TED register, and also allows the RAM output data to be read out when the E710TRC:ECERDB[7:0] bits and the E710TED register are read. Write access to ECTRRS is enabled only when ECTMCE is 1 (can be set simultaneously). ECTRRS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Disables generation of RAM read status for testing when E710TED is read. 1: Enables generation of RAM read status for testing when E710TED is read. When E710TRC:ECERDB[7:0] and E710TED are read, the values of the RAM output data pin are read out.
3	ECREOS	ECC Redundant Bit Output Data Select ECREOS selects either the ECC encoder output data or the ECERDB register value to be output as the ECC redundant bit output. Write access to ECREOS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREOS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows encoding result to be output as the ECC redundant bit output. 1: Allows the E710TRC:ECERDB[6:0] value to be output as the ECC redundant bit output.
2	ECENS	ECC Encoder Input Select ECENS selects either the data value from the peripheral module or the internal test register value (E710TED:ECEDB[31:0]) as the input signal to be encoded. Write access to ECENS is enabled only when ECTMCE is 1 (can be set simultaneously). ECENS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the RAM write data from the peripheral module to be input as the ECC encoder input data. 1: Allows the E710TED:ECEDB[31:0] value to be input as the ECC encoder input data.

Table 27.69 E710TMC Register Contents (2/2)

Bit Position	Bit Name	Function
1	ECDCS	ECC Decoder Input Select ECDCS selects either the lower 32-bit data value from the RAM or the internal test register value (E710TED:ECEDB[31:0]) as the lower 32-bit data of the input signal to be decoded. Write access to ECDCS is enabled only when ECTMCE is 1 (can be set simultaneously). ECDCS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the lower 32-bit RAM output data to be input to the data area (32 lower-order bits) to the decoder. 1: Allows the E710TED:ECEDB[31:0] value to be input to the data area to the decoder.
0	ECREIS	ECC Redundant Bit Input Data Select ECREIS selects either the upper 7-bit data value from the RAM (redundant bit area) or the test register value (E710TRC:ECERDB[6:0]) as the upper 7-bit data of the input signal to be decoded. Write access to ECREIS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREIS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the upper 7-bit RAM output data to be input to the ECC redundant bit area to the decoder. 1: Allows the E710TRC:ECERDB[6:0] value to be input to the ECC redundant bit area to the decoder.

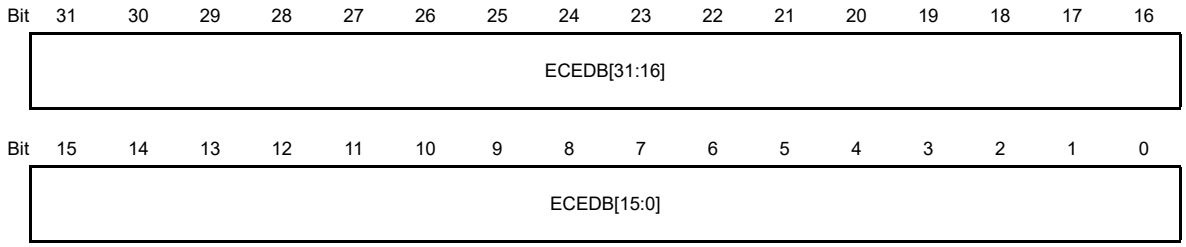
(3) E710TED — ECC Encoder and Decoder Data Test Register

E710TED is a 32-bit data test register for ECC encoding and decoding.

When ECTMCE = 1, E710TED can be read and written to using the 32-bit manipulation instruction.

When ECTMCE = 0, E710TED is always read as 0.

In test mode, the E710TED value can be used as the input data to the encoding circuit and decoding circuit.



NOTE

Changing ECTMCE from 1 to 0 resets E710TED synchronously.

When E710TMC:ECENS = 1, the ECEDB value is input to the encoding circuit and supplied to the RAM.

When E710TMC:ECDCS = 1, the ECEDB value is input as the bits 31-0 of the input data to the decoding circuit.

When E710TMC:ECTRRS = 1, reading ECEDB returns the RAM output data instead of the data written to ECEDB.

(4) E710TRC — ECC Redundant Bit Data Control Test Register

E710TRC is a 32-bit test register consisting of four fields (ECSYND, ECHORD, ECECRD, and ECRODM) corresponding to the ECC redundant bit area. Each field can be accessed as the 8-bit register with the same name. For details of each field, refer to the descriptions of these four registers.

When ECTMCE = 0, E710TRC is always read as 0.

When ECTMCE = 1, E710TRC can be read using the 32-bit manipulation instruction.

NOTE

Changing ECTMCE from 1 to 0 resets E710TRC synchronously.

(5) ECSYND — ECC Decoder Syndrome Data Register

ECSYND is a read-only register to confirm the syndrome code generated by the decoding circuit in test mode (ECTMCE = 1).

Write access to ECSYND is ignored.

Bit	31	30	29	28	27	26	25	24
	—	SYND[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

When read, the ECSYND bits return the value of the syndrome code (synd[6:0]) generated based on the input data to the decoding circuit.

The ECSYND bits are not holding circuits; the register value changes as the input signal changes.

ECSYND is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(6) ECHORD — ECC 7-Bit Redundant Data Holding Test Register

ECHORD holds the 7-bit ECC redundant area (upper 7-bit RAM data), which cannot be confirmed by the peripheral module, when the peripheral module accesses the RAM for reading in test mode (ECTMCE = 1).

Bit	23	22	21	20	19	18	17	16
	—	HORD[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

The ECHORD bits are loaded with the upper 7-bit RAM output data at the next rising edge of the operating clock signal after the peripheral module accesses the RAM for reading data in test mode (ECTMCE = 1).

The ECHORD bits are also loaded with the data on the input pins EC7TERI38 to EC7TERI32 at the next operating clock pulse when the ECEDB[15:0] register is read while E710TMC:ECTRRS = 1.

ECHORD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(7) ECECRD — ECC Encoder Test Register

ECECRD is a read-only register to read the 7-bit redundant section generated by the encoding circuit in test mode (ECTMCE = 1).

Bit	15	14	13	12	11	10	9	8
	—	ECRD[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

ECECRD is used to confirm the redundant bits generated by the input data from the peripheral module. Here, the read-out data is the result of encoding (ecc[6:0]), not the output value.

ECECRD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(8) ECERDB — ECC Redundant Bit Input and Output Substitution Buffer Register

ECERDB is a buffer register for the data that substitutes for the input and output data for the 7-bit ECC redundant data area in test mode (ECTMCE = 1).

ECERDB can be read and written to in ECC test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	ERDB[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When ECREOS = 1, the ECERDB value, instead of the seven redundant bits generated by the encoding circuit, is output to the pin to be supplied to the RAM.

When ECREIS = 1, the ECERDB value, instead of the upper seven data bits to be input to the decoding circuit, is handled by the decoding circuit.

When ECTRRS = 1, reading ECERDB returns the signal value supplied to RAM instead of the data written to ECERDB.

(9) E710EAD — ECC Error Address Register

E710EAD is a read-only register to hold the address at which an ECC error has occurred.

If an ECC error is detected while ECC error detection is enabled, the RAM address is latched using the detection signal as a trigger, and the address is stored in E710EAD as the address at which the ECC error has occurred.

The address is stored upon detection of the first ECC error while no error status is set. However, if a 1-bit error is followed by a 2-bit error, the address of the latter is stored.

Only one address can be held in E710EAD.

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

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

27.2.8.4 Notification to ECM

This module can detect two error signals through programming. The ECM is notified of the detected error.

- 1-bit error notification

  If an error is detected in one bit among bits 0 to 38 of data read from the RAM while 1-bit error notification is enabled (EC1EDIC = 1_B), the 1-bit error is indicated. However, it will not be indicated if ECER1F or ECER2F is already set.

- 2-bit error notification

  If errors are detected in two bits among bits 0 to 38 of data read from the RAM while 2-bit error notification control is enabled (EC2EDIC = 1_B), the 2-bit error is indicated. However, it will not be indicated if ECER2F is already set.

27.2.8.5 Test Function

(1) Writing to RAM Data

Write data to the peripheral RAM. However, the ECC corresponding to the written data will be written to the ECC bits simultaneously. In order to write a specified value to the ECC bits, use the ECC test mode described in (3) below.

(2) Reading RAM Data

(a) Set the ECTHM bit in the ECC control register to 1 to disable ECC error detection and correction.

(b) Read the peripheral RAM. Since neither error detection nor correction proceeds when the peripheral RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(c) Set the ECTHM bit in the ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) Write the value for writing to the ECC bits to the E710TRC.ECERDB[6:0] bits.

(c) Set the ECREOS bit in the ECC test mode control register to 1 to select writing of the value of the E710TRC.ECERDB[6:0] bits to the ECC bits.

(d) When data are written to the peripheral RAM, the value in the E710TRC.ECERDB[6:0] bits will be written to the ECC bits.

How to exit this test mode:

(e) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

(4) Reading the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) When data in the peripheral RAM is read, the ECC bits are stored in the E710TRC.ECHORD[6:0] bits.

How to exit this test mode:

(c) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

27.2.9 Data Parity for Data Transfer Paths

The following shows the transfer paths to which data parity is applied. Data parity errors can be detected during data transfer from the access sources to destinations shown below. If a parity error is detected in any of the paths, the error is notified to the ECM.

Table 27.70 Transfer Paths to which Data Parity is Applied

Access Source (Master)	Access Destination (Slave)
CPU1, CPU2, DMAC, DTS	INTC2, DMAC, DTS, CSIH, ADCC, port group
DMAC, DTS	CPU1/CPU2 local RAM, global RAM

When a data parity error occurs during read operation, 0 is returned to the CPU of the bus master and error data is returned to other masters. When a data parity error occurs during write operation, the write operation is cancelled if the slave is INTC2, CSIH, or ADCC.

Some of the parity-applied modules have the control registers in the parity controllers (encoder and decoder), which can hold status upon detection of an error. If a parity error is detected in these modules, it is possible to identify the access that caused the error.

On the other hand, parity controllers which do not have control registers always perform parity detection. Though the status upon detection of an error is not retained in these controllers, whether an error has been detected or not is retained in the ECM.

27.2.9.1 List of Registers

Table 27.71 List of Registers

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
APDP	<Base_addr>+0_H	APDPERRST_xx	P-Bus data parity status register xx	R	0000 0000_H	8/16/32
APDP	<Base_addr>+4_H	APDPERRSTC_xx	P-Bus data parity status clear register xx	W	0000 0000_H	8/16/32
APDP	<Base_addr>+8_H	APDPTMC_xx	P-Bus data parity test mode control register xx	R/W	0000 0000_H	16/32
APDP	<Base_addr>+C_H	APDPERRADR_xx	P-Bus data parity error address register xx	R	0000 0000_H	32

The symbol xx in the above table indicates a parity-applied module for which the control register is provided. **Table 27.73** shows the parity-applied modules and base addresses <base_addr>.

Table 27.72 List of Data Parity Control Modules

Module to Which Parity is Applied	xx	<base_addr>
INTC2	INTC2	FFC6 8800_H
DMA_DTS	PDMA	FFC6 8900_H
CSIH0 (group A)*1	CS0A	FFF9 6000_H
CSIH0 (group B)*1	CS0B	FFF9 6020_H
CSIH1 (group A)*1	CS1A	FFF9 6040_H
CSIH1 (group B)*1	CS1B	FFF9 6060_H
ADCC0	ADC0	FFC8 8000_H
ADCC1	ADC1	FFC8 8020_H
Port group	PT	FFC8 5000_H

Note 1. The CSIHx registers are divided into the following two groups and controlled separately.
Group A: CSIHnCTL0 to CSIHnCTL2, CSIHnSTR0, and CSIHnSTCR0
Group B: Other than above.

27.2.9.2 Details of Registers

(1) APDPERRST_xx — P-Bus Data Parity Status Register

For the symbol xx, see **Table 27.72, List of Data Parity Control Modules**

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

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

Table 27.73 APDPERRST Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	APDPERR	Data Parity Error Monitor Flag Indicates occurrence of the parity error. APDPERR is cleared by setting the data parity error monitor flag clear bit (APDPERRC) to 1. 0: Indicates that a parity error has not occurred. 1: Indicates that a parity error has occurred.

(2) APDPERRSTC_xx — P-Bus Data Parity Status Clear Register

For the symbol xx, see **Table 27.72, List of Data Parity Control Modules**

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

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

Table 27.74 APDPERRSTC Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved. These bits are always read as 0. The write value should also be 0.
0	APDPERRC	Clears the data parity error monitor flag (APDPERR). Writing 1 to this bit with the APDPERR bit set clears APDPERR. This bit is always read as 0.

(3) APDPTMC_xx — P-Bus Data Parity Test Mode Control Register

For the symbol xx, see **Table 27.72, List of Data Parity Control Modules**

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APDPTMC[1:0]		—	—	—	—	—	—	—	—	—	—	APDPEIC3	APDPEIC2	APDPEIC1	APDPEIC0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 27.75 APDPTMC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15, 14	APDPTMC[1:0]	Data Parity Test Mode Control Setting APDPTMC[1:0] to 01_B enables write-accesses to the error insertion control bits 3 to 0 (APDPEIC3 to APDPEIC0). When APDPTMC[1:0] are not 01_B, APDPEIC3 to APDPEIC0 are not written to even though a write-access is attempted. These bits are always read as 0.
13 to 4	—	Reserved. These bits are always read as 0. The write value should also be 0.
3	APDPEIC3	Byte-Lane 3 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 3 (bits 31 to 24) to odd parity. APDPEIC3 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
2	APDPEIC2	Byte-Lane 2 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 2 (bits 23 to 16) to odd parity. APDPEIC2 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
1	APDPEIC1	Byte-Lane 1 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 1 (bits 15 to 8) to odd parity. APDPEIC1 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
0	APDPEIC0	Byte-Lane 0 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 0 (bits 7 to 0) to odd parity. APDPEIC0 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)

(4) APDPERRADR_xx — P-Bus Data Parity Error Address Register

For the symbol xx, see **Table 27.72, List of Data Parity Control Modules**

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

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

Table 27.76 APDPERRADR Register Contents

Bit Position	Bit Name	Function
31 to 0	APDPERRADR [31:0]	Data Parity Error Address Holds the address at which the first parity error has occurred while the data parity error monitor flag (APDPAERR) is not set. The retained information is not updated even though the next parity error occurs while DPAERR is set.

CAUTIONS

The representation of addresses in the APDPERRADR_xx register depends on the target module for parity checking.

- **In the case of the INTC2 and DMA_DTS modules, the indicated address is the actual address (32 bits).**
- **In the case of other modules, the indicated address is only the 12 lower-order bits of the address, so add the base address of the given module to obtain the actual address.**
  The indicated address also indicates the data alignment, so the 2 lower-order bits are fixed to 0.

27.3 Lockstep

This product incorporates the CPU1 with the lockstep function to quickly detect CPU failures without special software. The CPU1 executes the program using two different cores, that is, master core and checker core, and constantly compares the execution results of the two cores. When the results do not agree, the CPU1 determines that an error has occurred in one of the cores, and notifies the lock step compare error to the ECM.

Bus outputs to be compared are outputs to the local RAM for CPU1, global RAM, a CPU peripheral, Interconnect, the P-bus, code flash memory, and the tag RAM and data RAM of the instruction cache.

The lockstep function of the CPU1 features failure insertion, with which errors can be intentionally caused and thus self-diagnosis of the lockstep operation is possible.

27.3.1 List of Registers

Table 27.77 List of Registers

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
TESTCOMP	FFFE ED00$_H$	TESTCOMPREG0	Comparator test register 0	R/W	0000 0000$_H$	8/16/32
TESTCOMP	FFFE ED04$_H$	TESTCOMPREG1	Comparator test register 1	R/W	0000 0000$_H$	8/16/32

These registers are placed in the CPU Peripheral of the CPU1. These registers can only be accessed by the CPU1.

27.3.2 Details of Registers

27.3.2.1 TESTCOMPREG0 — Comparator Test Register 0

TESTCOMPREG0 is test register 0 used for the lockstep function of the CPU1.

Combining TESTCOMPREG0 with TESTCOMPREG1 enables self-diagnosis of the lockstep function. The following gives an example of self-diagnosis procedure.

(1) Write arbitrary value to TESTCOMPREG0.

(2) Write a different value to TESTCOMPREG1.

(3) Read TESTCOMPREG0. The different values are read and sent to the master core and checker core.

(4) Using the values read, run the comparator to be diagnosed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG0[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.78 TESTCOMPREG0 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG0[31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG0[31:0] value is read. PE1C: TESTCOMPREG1[31:0] value is read.

27.3.2.2 TESTCOMPREG1 — Comparator Test Register 1

TESTCOMPREG1 is test register 0 used for the lockstep function of the CPU1.

Combining TESTCOMPREG1 with TESTCOMPREG0 enables self-diagnosis of the lockstep function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG1[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.79 TESTCOMPREG1 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG1[31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG1[31:0] value is read. PE1C: TESTCOMPREG0[31:0] value is read.

27.4 Memory Protection

27.4.1 Overview

This product incorporates the memory protection function to prevent erroneous accesses to data in memories and control registers of the peripheral circuits.

- MPU
  The CPU1 and CPU2 protect memories against illegal accesses from the CPU1 and CPU2 themselves. Accesses to addresses that are prohibited by the MPU are never issued by the CPU1 or CPU2. For details, refer to the *RH850G3M Family User's Manual: Software*.
- Slave Guard
  A specific memory is protected against illegal accesses from any bus master. Slave guard includes the following guard types. The details of each type are given in the following sections.

  – PEG
    The local RAM is protected against illegal accesses. However, accesses from the CPU incorporating the local RAM itself are excluded. For example, accesses from the CPU1 to local RAM in the CPU1, or accesses from the CPU2 to local RAM in the CPU2 are not rejected by the PEG.
    For details, refer to **Section 3, CPU System**.
  – IPG
    The CPU Peripheral is protected against illegal accesses. For details, refer to **Section 3, CPU System**.
  – GRG
    The global RAM is protected against illegal accesses.
  – PBG
    The control registers in the peripheral circuits and memories are protected against illegal accesses. For details, refer to **Section 27.4.3**.

27.4.1.1 Identifiers for Slave Guard

For the slave guard function, the type of illegal accesses to be rejected can be designated using the following identifiers.

Table 27.80 Identifiers for Slave Guard

Identifier	Function
UM	When the CPU makes an access, the operating mode of the CPU is indicated. 0: Supervisor mode 1: User mode When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register. When the other master makes an access, the value of this identifier is always 0.
SPID	When the CPU makes an access, the system protection identifier SPID that is assigned to the CPU is indicated. When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register. When the other master makes an access, the value of this identifier is always 00_B.
PEID	The access source bus master is indicated. 000_B: Reserved 001_B: CPU1 010_B: CPU2 011_B: Reserved 100_B: Other bus master 101_B: Reserved 110_B: Reserved 111_B: Reserved When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register.

27.4.2 GRG (Global RAM Guard)

This product is provided with 4-channel GRG, which is described in detail in the following sections.

27.4.2.1 List of Registers

Table 27.81 List of Registers

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
MGDGR	FFC4 9000$_H$	MGDGRPROT0	GRG protection setting register 0	R/W	07FF FFF0$_H$	8/16/32
MGDGR	FFC4 9004$_H$	MGDGRBAD0	GRG compare base address register 0	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9008$_H$	MGDGRADV0	GRG valid compare address register 0	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9010$_H$	MGDGRPROT1	GRG protection setting register 1	R/W	07FF FFF0$_H$	8/16/32
MGDGR	FFC4 9014$_H$	MGDGRBAD1	GRG compare base address register 1	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9018$_H$	MGDGRADV1	GRG valid compare address register 1	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9020$_H$	MGDGRPROT2	GRG protection setting register 2	R/W	07FF FFF0$_H$	8/16/32
MGDGR	FFC4 9024$_H$	MGDGRBAD2	GRG compare base address register 2	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9028$_H$	MGDGRADV2	GRG valid compare address register 2	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9030$_H$	MGDGRPROT3	GRG protection setting register 3	R/W	07FF FFF0$_H$	8/16/32
MGDGR	FFC4 9034$_H$	MGDGRBAD3	GRG compare base address register 3	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9038$_H$	MGDGRADV3	GRG valid compare address register 3	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9100$_H$	MGDGRSCTL_VCI	GRG control register (VCI)	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9104$_H$	MGDGRSSTAT_VCI	GRG error status register (VCI)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 9108$_H$	MGDGRSAD_VCI	GRG error address register (VCI)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 910C$_H$	MGDGRSTYPE_VCI	GRG error access type register (VCI)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 9200$_H$	MGDGRSCTL_PE1	GRG control register (PE1)	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9204$_H$	MGDGRSSTAT_PE1	GRG error status register (PE1)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 9208$_H$	MGDGRSAD_PE1	GRG error address register (PE1)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 920C$_H$	MGDGRSTYPE_PE1	GRG error access type register (PE1)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 9300$_H$	MGDGRSCTL_PE2	GRG control register (PE2)	R/W	0000 0000$_H$	8/16/32
MGDGR	FFC4 9304$_H$	MGDGRSSTAT_PE2	GRG error status register (PE2)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 9308$_H$	MGDGRSAD_PE2	GRG error address register (PE2)	R	0000 0000$_H$	8/16/32
MGDGR	FFC4 930C$_H$	MGDGRSTYPE_PE2	GRG error access type register (PE2)	R	0000 0000$_H$	8/16/32

- MGDGRPROTn, MGDGRBADn, and MGDGRADVn set the protection specifications for each channel (n: 0 to 3).
- MGDGRSCTL_*, MGDGRSSTAT_*, MGDGRSAD_*, and MGDGRSTYPE_* indicate error information on each access port. "_VCI" represents access from the system interconnect 1 to the global RAM, "_PE1" represents access from the CPU1 to the global RAM, and "_PE2" represents access from the CPU2 to the global RAM.

27.4.2.2 Details of Registers

(1) MGDGRPROTn — GRG Protection Setting Register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	EN	—	—	—	—	UM	PEID[7:0]								—
Value after reset	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

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

Table 27.82 MGDGRPROTn Register Contents

Bit Position	Bit Name	Function
31	—	Reserved. This bit is always read as 0. The write value should also be 0.
30	EN	Protection Enable 0: Disables protection. 1: Enables protection. The access permitted by this register is only allowed.
29 to 27	—	Reserved. These bits are always read as 0. The write value should also be 0.
26	—	Reserved. This bit is always read as 1. The write value should also be 1.
25	UM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24 to 17	PEID[7:0]	PEID Access The PEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables multiple ID values simultaneously. For example, setting the PEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
16 to 9	—	Reserved. These bits are always read as 1. The write value should also be 1.
8 to 5	SPID[3:0]	SPID Access The SPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the SPID field to 0101_B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.
4	DEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Be sure to set this bit to 1. If the setting is 0, the debugger or RAM monitor tool may not operate correctly.
3 to 0	—	Reserved. These bits are always read as 0. The write value should also be 0.

(2) MGDGRBADn — GRG Compare Base Address Register n (n = 0 to 3)

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

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

Table 27.83 MGDGRBADn Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved. These bits are always read as 0. The write value should also be 0.
20 to 9	AD[20:9]	Compare base address
8 to 0	—	Reserved. These bits are always read as 0. The write value should also be 0.

(3) MGDGRADVn — GRG Valid Compare Address Register n (n = 0 to 3)

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

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

Table 27.84 MGDGRADVn Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved. These bits are always read as 0. The write value should also be 0.
20 to 9	ADV[20:9]	Valid Compare Address Bits of addresses corresponding to bits from among MGDGRADVn[20:9] which have the value 1 are compared. When MGDGRADVn[20:9] are all 1, the 512 bytes from the address specified by MGDGRBAD are protected. When MGDGRADVn[20:9] are all 0, the entire global RAM is subject to protection.
8 to 0	—	Reserved. These bits are always read as 0. The write value should also be 0.

Setting example: When MGDGRBADn[20:9] = 800_H and MGDGRADVn[20:9] = $FF7_H$, the global RAM guard protection areas n are $FEF00000_H$ to $FEF001FF_H$ and $FEF01000_H$ to $FEF011FF_H$.

Concept: When MGDGRBADn[20:9] = 800_H, the base address is $FEF00000_H$ and the base address of the range to which the setting applies is indicated within [ ] below.

1111 1110 111 [1 0000 0000 000] 0 0000 0000

F E F 0 0 0 0 0

When MGDGRADVn[20:9] = $FF7_H$, the setting of the Lower bits 9-0 is 0 so the corresponding bit of addresses is not compared, and the 512 bytes represented by

1111 1110 111[1 0000 000X 000] X XXXX XXXX,

i.e.

F E F 0 0 0 0 0 to

F E F 0 0 1 F F,

and

F E F 0 1 0 0 0 to

F E F 0 1 1 F F

(1 Kbyte in total) are protected.

(4) MGDGRSCTL_VCI/PE1/PE2 — GRG Control Register

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

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

Table 27.85 MGDGRSCTL Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	ERRCLO	Clears the error entry overflow flag. 0: No operation 1: Clears the overflow flag.
0	ERRCLE	Clears the error detection flag. 0: No operation 1: Clears the error detection flag. However, set the bit simultaneously with ERRCLO as shown in the table below.

Table 27.86 Settings of the ERRCLO and ERRCLE Bits

ERRCLO	ERRCLE	Function
0	0	Clears neither of the bits.
0	1	Clears neither of the bits (setting ignored).
1	0	Clears the OVF bit.
1	1	Clears both of the bits.

(5) MGDGRSSTAT_VCI/PE1/PE2 — GRG Error Status Register

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

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

Table 27.87 MGDGRSSTAT Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow 1: Overflow occurred Since the error entry depth of GRG is 1, when another guard violation occurs after the error detection flag was set due to the earlier guard violation, the error entry overflow occurs and this flag is set. Occurrence of overflow is not reported to ECM. Guard violation error information is not captured in case of an overflow.
0	ERR	Error Detection Flag 0: No error 1: An error occurred

(6) MGDGRSAD_VCI/PE1/PE2 — GRG Error Address Register

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

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

Table 27.88 MGDGRSAD Register Contents

Bit Position	Bit Name	Function
31 to 26	—	Reserved. These bits are always read as 0. The write value should also be 0.
25	GRIFREQAP	Address Parity Bit upon Error Occurrence
24 to 21	—	Reserved. These bits are always read as 0. The write value should also be 0.
20 to 0	GRIFA[20:0]	Address upon Error Occurrence

(7) MGDGRSTYPE_VCI/PE1/PE2 — GRG Error Access Type Register

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—	—	—	SPID[1:0]		—	UM	—	TYPE[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.89 MGDGRSTYPE Register Contents

Bit Position	Bit Name	Function
31 to 18	—	Reserved. These bits are always read as 0. The write value should also be 0.
17, 16	PACKETID[1:0]	Packet ID upon Error Occurrence (Internal bus and interconnect signals. Reference information.)
15 to 13	PEID[2:0]	PEID upon Error Occurrence
12 to 10	—	Reserved. These bits are always read as 0. The write value should also be 0.
9, 8	SPID[1:0]	SPID upon Error Occurrence
7	—	Reserved. This bit is always read as 0. The write value should also be 0.
6	UM	UM upon Error Occurrence
5	—	Reserved. This bit is always read as 0. The write value should also be 0.
4 to 0	TYPE[4:0]	Transfer Type upon Error Occurrence (Internal bus and interconnect signals. Reference information.)

27.4.3 PBG

The PBG module is divided into several PBG groups, each of which is provided with a maximum of 16 protection channels. A single PBG channel can designate the access against which a single peripheral circuit should be protected. Each PBG group can hold the information of the access that has been rejected.

The following table lists the peripheral circuits to be protected, the corresponding PBG group names, and the PBG channel numbers.

Table 27.90 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers

PBG Group	PBG Channel Number	Module to be Protected
PBG0	0	INTC2
	1	DMA_DTS
PBG1	4	GRG (control register)
	5	GRG status (VCI2GRAM)
	6	GRG status (PE1)
	7	GRG status (PE2)
PBG2	0	RDC20
	1	RDC21 (only C1H)
	2	PBG2 itself
PBG3	0	ADCC0
	1	ADCC1
	2	ECM (master)
	3	ECM (checker)
	4	ECM (common part)
	5	port group
	6	DCRA0
	7	DCRA1
	8	PBG3 itself
PBG4	0	TAUD0
	1	TAUD1
	2	TAPA0
	3	TAPA1
	4	TAPA2
	5	TAPA3
	6	TSG30
	7	TSG31
	8	PBG4 itself
PBG5	0	CSIH0 (group A)*1
	1	CSIH0 (group B)*1
	2	CSIH1 (group A)*1
	3	CSIH1 (group B)*1
	4	RS-CAN
	5	EMU20
	6	EMU21
	7	PBG5 itself

Note 1. The CSIHx registers are divided into the following two groups and controlled separately.
Group A: CSIHnCTL0 to CSIHnCTL2, CSIHnSTR0, and CSIHnSTCR0
Group B: Other than above

27.4.3.1 List of Registers

The following table lists the register provided for each PBG channel.

Table 27.91 List of Registers

Module Name	Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size
PBG	FSGDxxDPROTn	PBGxx protection register n	R/W	PBG0-1: 07FF FFFF$_H$ PBG2-5: 0006 0255$_H$	<base_addr0> + 4*n	8/16/32

The following table lists the registers provided for each PBG group.

Table 27.92 List of Registers (for PBG Group)

Module Name	Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size
PBG	ERRSLVxxCTL	PBGxx error control register	W	0000 0000$_H$	<base_addr1> + 0$_H$	8/16/32
PBG	ERRSLVxxSTAT	PBGxx error status register	R	0000 0000$_H$	<base_addr1> + 4$_H$	8/16/32
PBG	ERRSLVxxADDR	PBGxx error address register	R	0000 0000$_H$	<base_addr1> + 8$_H$	32
PBG	ERRSLVxxTYPE	PBGxx error type register	R	0000 0000$_H$	<base_addr1> + C$_H$	16/32

In the above tables, “xx” and “n” in the register names and symbols represents the PBG group numbers and PBG channel numbers, respectively. The table below shows the base address values <base_addr0> and <base_addr1>, which correspond to each of the PBG group numbers and PBG channel numbers.

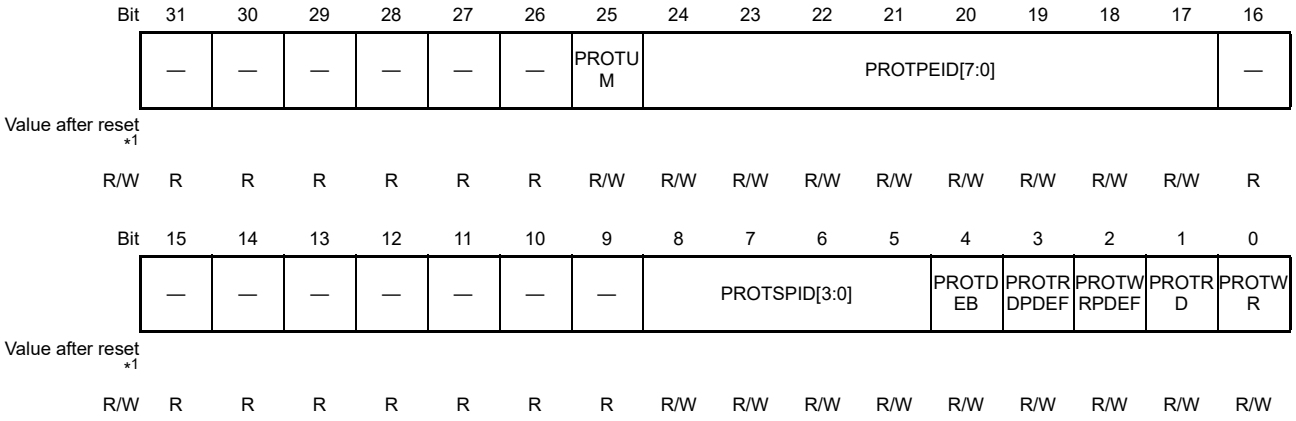
Table 27.93 Base Address Values for the PBG Channel Numbers

PBG Group	PBG Channel Number	<base_addr0>	<base_addr1>
PBG0	0, 1	FFC4 C000$_H$	FFC4 C800$_H$
PBG1	4 to 7	FFC4 C100$_H$	FFC4 C900$_H$
PBG2	0 to 2	FFDC 0000$_H$	FFDC 0200$_H$
PBG3	0 to 8	FFC4 0000$_H$	FFC4 0200$_H$
PBG4	0 to 8	FFDD D000$_H$	FFDD D200$_H$
PBG5	0 to 7	FFF9 4000$_H$	FFF9 4200$_H$

27.4.3.2 Details of Registers

(1) FSGDxxDPROTn — PBGxx Protection Register n

FSGDxxDPROTn designates the access to be rejected against which the peripheral circuit control registers and RAM should be protected. Any access that is disabled using any of the identifiers is rejected as an illegal access.



Note 1. The value after reset is as follows:
PBG0-1: 0000 0111 1111 1111 1111 1111 1111 1111_B
PBG2-5: 0000 0000 0000 0110 0000 0010 0101 0101_B

Table 27.94 FSGDxxDPROTn Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved. These bits are always read as 0. The write value should also be 0.
26	—	Reserved. Reading this bit returns the value after reset. The write value should also be the value after reset.
25	PROTUM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24 to 17	PROTPEID[7:0]	PEID Access The PROTnPEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables multiple ID values simultaneously. For example, setting the PROTnPEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
16 to 9	—	Reserved. Reading these bits returns the value after reset. The write value should also be the value after reset.
8 to 5	PROTSPID[3:0]	SPID Access The PROTnSPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the PROTnSPID field to 0101_B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.
4	PROTDEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Be sure to set this bit to 1. If the setting is 0, the debugger or RAM monitor tool may not operate correctly.
3	PROTRDPDEF	Default Read Protection 0: Enables read access from any master. 1: Only enables reading by the access-permitted master.

Table 27.94 FSGDxxDPROTn Register Contents (2/2)

Bit Position	Bit Name	Function
2	PROTWRPDEF	Default Write Protection 0: Enables write access from any master. 1: Only enables write access from the access-permitted master.
1	PROTRD	Read Permission 0: Reading by any master is prohibited. 1: Reading by the access-permitted master is only permitted.
0	PROTWR	Write Permission 0: Writing by any master is prohibited. 1: Writing by the access-permitted master is only permitted.

(2) ERRSLVxxCTL — PBGxx Error Control Register

ERRSLVxxCTL clears the status in the error status register.

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

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

Table 27.95 ERRSLVxxCTL Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	CLRO	Clears the error entry overflow flag. 0: No operation. 1: Clears the overflow flag.
0	CLRE	Clears the error detection flag. 0: No operation. 1: Clears the error detection flag.

(3) ERRSLVxxSTAT — PBGxx Error Status Register

ERRSLVxxSTAT holds the status of the illegal access rejected with the PBGxx.

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

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

Table 27.96 ERRSLVxxSTAT Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow 1: Overflow occurred Since the error entry depth of PBG is 1, when another guard violation occurs after the error detection flag was set due to the earlier guard violation, the error entry overflow occurs and this flag is set. Occurrence of overflow is not reported to ECM. Guard violation error information is not captured in case of an overflow.
0	ERR	Error Detection Flag 0: No error 1: An error occurred

(4) ERRSLVxxADDR — PBGxx Error Address Register

ERRSLVxxADDR holds the address of the illegal access rejected with the PBGxx.

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

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

Table 27.97 ERRSLVxxADDR Register Contents

Bit Position	Bit Name	Function
31 to 24	—	Reserved. These bits are always read as 0. The write value should also be 0.
23 to 0	ADDR[23:0]	When an illegal access occurs, the access address is calculated by addition of FF00 0000_H to read value of these bits.

(5) ERRSLVxxTYPE — PBGxx Error Type Register

ERRSLVxxTYPE holds the type of the illegal access rejected with the PBGxx.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—	—	—	SPID[1:0]		—	UM	—	STRB[3:0]				WRITE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.98 ERRSLVxxTYPE Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. These bits are always read as 0. The write value should also be 0.
15 to 13	PEID[2:0]	PEID upon Error Occurrence
12 to 10	—	Reserved. These bits are always read as 0. The write value should also be 0.
9 and 8	SPID[1:0]	SPID upon Error Occurrence
7	—	Reserved. This bit is always read as 0. The write value should also be 0.
6	UM	UM upon Error Occurrence
5	—	Reserved. This bit is always read as 0. The write value should also be 0.
4 to 1	STRB[3:0]	Strobe signals upon Error Occurrence (it's signals of internal bus and internal connection. Reference information)
0	WRITE	Write signal upon Error Occurrence (it's signal of Internal bus and internal connection. Reference information)

27.5 Multi-Input Signature Generator (MISG)

27.5.1 Overview

This LSI incorporates multi-input signature generators (MISG) for self-diagnosis by the CPUs.

The table below shows the overview of the MISG specifications.

Table 27.99 Specification Overview

Item	Description
Generating polynomials	Two polynomials are available for use in signature generation. • MISR1: $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$ • MISR2: $G(x) = x^{32} + x^{22} + x^2 + x + 1$ *To support signature generation for 64-bit data, each CPU is connected with an MISG that has two MISRs (MISR1 and MISR2).
Signature generation	Signature generation can be enabled or disabled. • Signature generation in MISR1 is enabled or disabled. • Signature generation in MISR2 is enabled or disabled. The following two conditions can be selected as conditions for signature generation • Writing to the register A signature is generated by writing to the MISR1 calculation register (MISRCDRL) • Monitoring write access A signature is generated if writing to the address area specified for monitoring occurs when writing by the CPU is being monitored. The address area is specified by the monitoring area base address register and the monitoring area address mask register.
Automatic signature comparison	Two signature generation units are selected for comparison of signatures. Each signature generation unit has a data counter and comparison proceeds when the values of the data counters in the MISGs selected as the target for comparison match. The data counter counts the number of write accesses to the MISRCDRL register or the address area being monitored.
Error notification	When signatures are compared and do not match, the ECM is notified of an error. Enabling or disabling of error notification to the ECM can be selected. An interrupt request for the INTC is not made directly.

27.5.2 Block Diagram

27.5.2.1 MISG

The figure below is a block diagram of the MISG. The MISG consists of two signature generation units and a signature comparison unit.

Write monitoring mode for signature generation conditions (**(2) Write monitoring mode**) can only be executed between the corresponding signature generation unit and the CPU. Therefore, each signature generation unit is given a name corresponding to the CPU number (MISG_PE1 or MISG_PE2). MISG_PE1 can monitor CPU1's write accesses, and MISG_PE2 can monitor CPU2's write accesses.

When register write mode is the signature generation condition, there is no correspondence between a CPU and the signature generation unit. Any CPU can generate a signature in any signature generation unit.

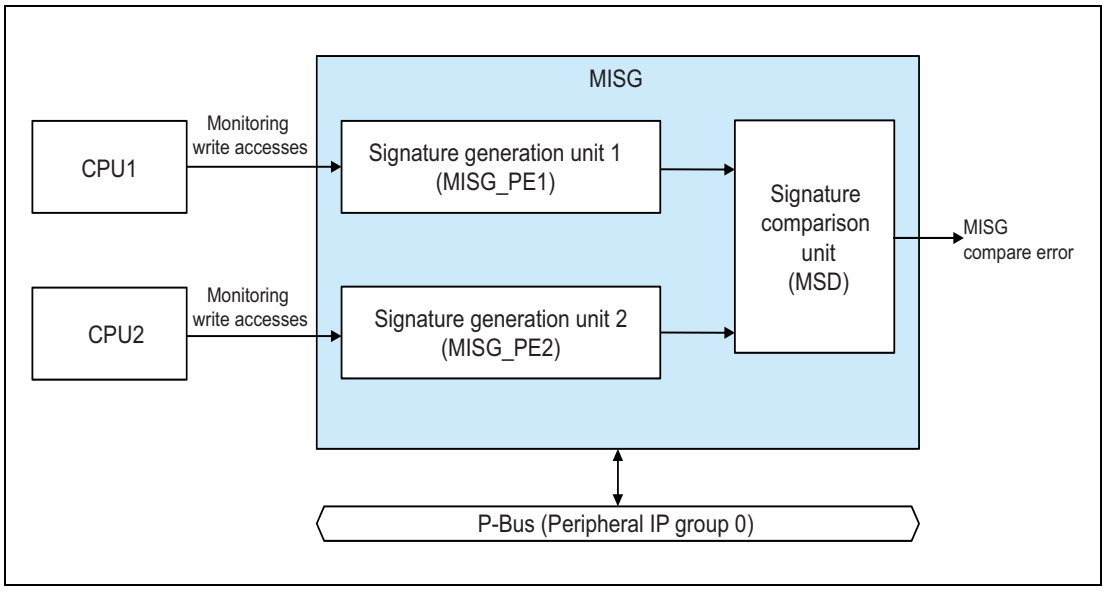


Figure 27.3 MISG Block Diagram (RH850/C1H)

27.5.2.2 Signature Generation

The figure below shows the flow of data in signature generation. MISR1 and MISR2 consist of two 32-bit signature generation units (MISR1H and MISR1L, MISR2H and MISR2L).

MISR1H and MISR2H can generate a signature from the 32 higher-order bits of write data of the CPU to be monitored. MISR1L and MISR2L can generate signatures from the 32 lower-order bits of data written by the CPU to be monitored or data written to MISRCDRL.

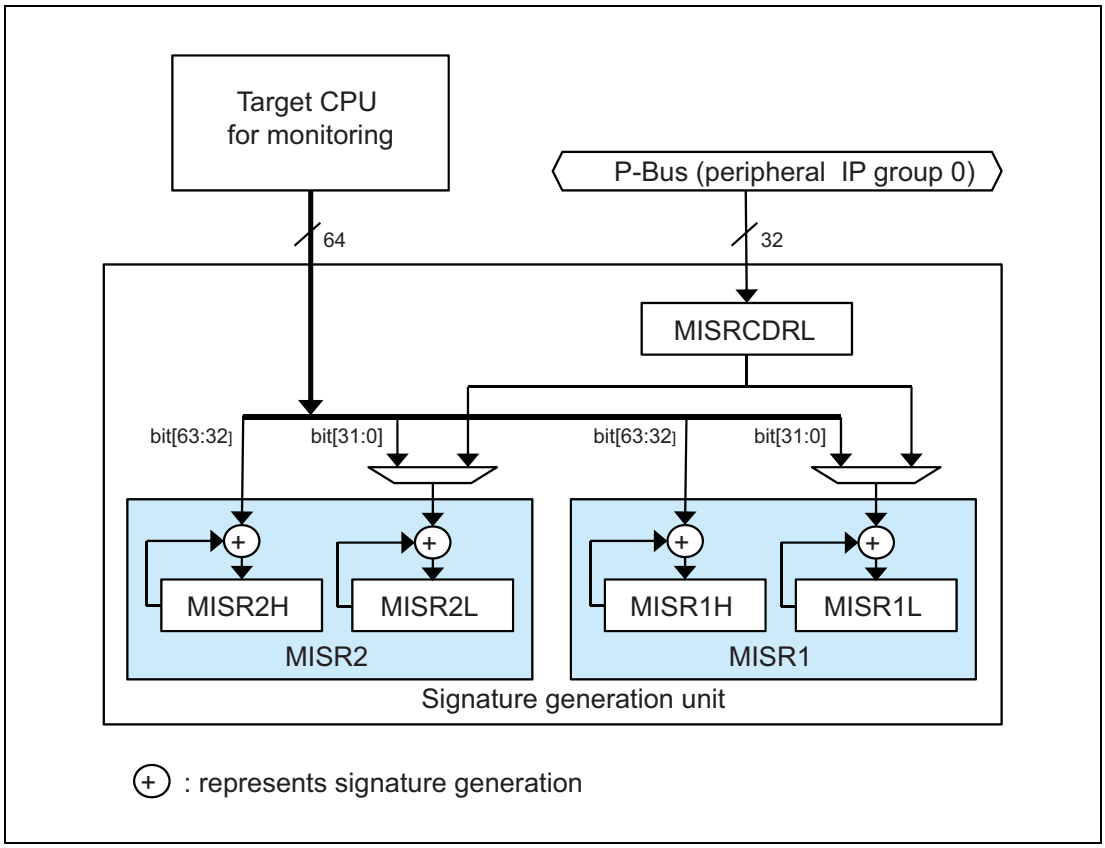


Figure 27.4 Signature Generation Units

Block diagrams of signature generation in MISR1H and MISR1L and the polynomials used for this are shown below.

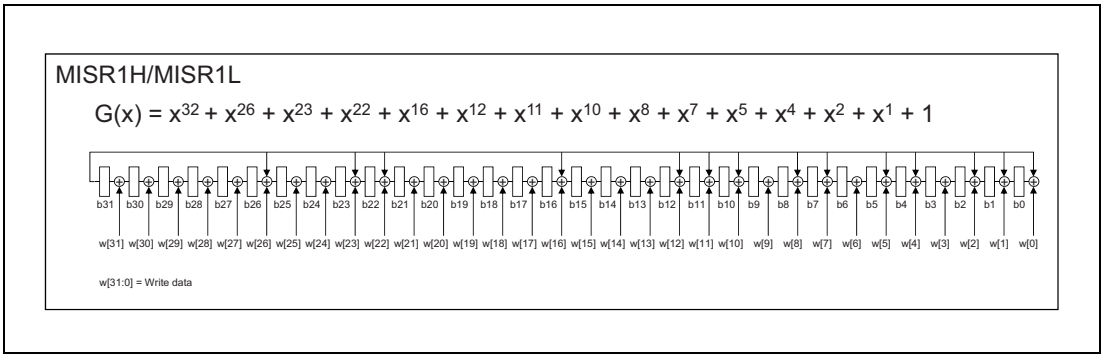


Figure 27.5 Block Diagram of Signature Generation in MISR1H and MISR1L and the Polynomials

Block diagrams of signature generation in MISR2H and MISR2L and the polynomials used for this are shown below.

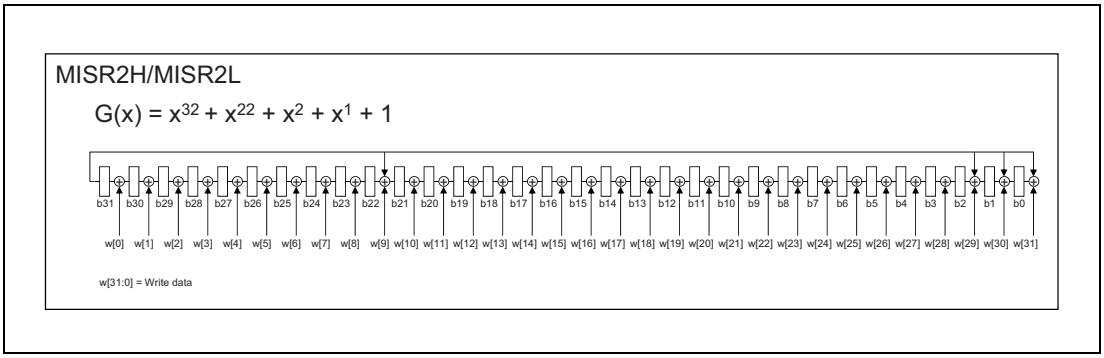


Figure 27.6 Block Diagram of Signature Generation in MISR2H and MISR2L and the Polynomials

27.5.3 Functional Specification

27.5.3.1 Conditions for Signature Generation

The conditions for signature generation in MISR1 and MISR2 can be selected by the setting of the MISR control register (MISRCR).

Signature generation conditions in MISRi (i = 1, 2)

Table 27.100 Signature Generation Conditions in MISRi

MISRCR. MISRiEN	MISRCR. MISEiCND	Signature Generation Conditions
0	—	MISRi does not generate a signature.
1	0	Register write mode MISRi generates a signature after write access to MISRCDRL register.
1	1	Write monitoring mode MISRi generates a signature after the corresponding CPU writes to an address specified for monitoring.

(1) Register write mode

Writing to the MISR calculation register (MISRCDRL) while MISR1 is in register write mode leads to the generation of a 32-bit signature in MISR1 from the value held in the multi-input signature register 1L (MISR1L) and the data written to MISRCDRL, and the result is retained in MISR1L. Similarly, a 32-bit signature is generated in MISR2 from the value held in the multi-input signature register 2L (MISR2L) and the data written to MISRCDRL, and the result is retained in MISR2L. In register write mode, the values of MISR1H and MISR2H are not updated.

Writing to MISRCDRL can proceed in 8-, 16-, or 32-bit units, and bits to which a value is not written are treated as 0. For example, writing to the 16 lower-order bits of MISRCDRL leads to the generation of a signature with the 16 higher-order bits of write data treated as 0. Similarly, writing to the 16 higher-order bits of MISRCDRL leads to the generation of a signature with the 16 lower-order bits of write data treated as 0. However, these cases will not arise if access is by using the IO header file.

MISR1 and MISR2 do not identify the bus master which writes to the MISRCDRL. Writing by any bus master, whether a CPU, DMAC, or debugging master, produces a signature.

(2) Write monitoring mode

Writing to the address area specified for monitoring by the CPU while MISR1 is in write monitoring mode leads the generation of two 32-bit signatures in MISR1 from the value held in MISR1H and MISR1L and the data written to the CPU, and the result retained in MISR1H and MISR1L. At this time, MISR1H is for the 32 higher-order bits of write data, while MISR1L is for the 32 lower -order bits of write data. Similarly, writing to the address area specified for monitoring by the CPU while MISR2 is in write monitoring mode leads the generation of two 32-bit signatures in MISR2 from the value held in MISR2H and MISR2L and the data written to the CPU, and the result retained in MISR2H and MISR2L.

The address area specified for monitoring for signature generation is set by the MISR monitoring area base address register (MISRBASEADR) and the MISR monitoring area address mask register (MISRADRMSK). If the CPU write address is within the address area specified for monitoring, a signature is generated.

Write monitoring mode monitors write access by the corresponding CPU in 8-, 16-, 32-, or 64-bit units. When writing proceeds in 8-, 16-, or 32-bit units, remaining bits to which a value is not written are treated as 0, and 64-bit data is always input to MISR1 or MISR2. Write data are allocated to the lower-order side regardless of the destination address. For example, when 32 bits are written to an address of the form 8N+4, the 32 bits are allocated to the lower-order side and a signature is generated with the 32 higher-order bits treated as 0.

The CPU write access destinations that can be monitored by MISR1 and MISR2 are as follows:

Local RAM, global RAM, CPU peripheral (local APB),

Peripheral circuit connected to the interconnect (GVCI) or to the P-Bus (global APB)

The following store operations are not subject to monitoring.

(1) Instructions that identify write data in slave responses: BitOp, CAXI, and STC.W

(2) Instructions for saving more than 64 bits on the stack: PREPARE and PUSHSP

Write monitoring mode can only be executed between the specified signature generation unit and the corresponding CPU.

In this product, the following monitoring operations are possible.

- Monitoring of CPU1 write access by signature generation unit 1
- Monitoring of CPU2 write access by signature generation unit 2 (this function is only available in the dual-core product).

A signature is not generated if write monitoring mode is set for a signature generation unit that does not support write monitoring mode. At this time, reference to the values of MISRBASEADR and MISRADRMSK is not possible from anywhere.

27.5.3.2 Automatic Signature Comparison

Of the signature generation units, two signature generation units are selected for the comparison of signatures by the MISR1CMPEN0 or MISR2CMPEN0 bit. Each signature generation unit has a data counter, and the signatures are compared if the values of the data counters generated in two signature generation units selected as targets for comparison match.

If two signatures are to be compared, use MISR1CMPEN0 or MISR2CMPEN0 in the MISRCMPCTL register to enable signature comparison by two signature generation units.

27.5.3.3 Data Counter

The MISR data counter register (MISRDCNT) counts how many times writing to MISRCDRL or the address range specified by BASEADR and ADRMSK proceeds. When the CNTSTA bit = 1 and the CNTTRG bit = 0 in the data counter control register (MISRDCNTCTL), MISRDCNT counts the number of write accesses to MISRCDRL. When the CNTSTA bit = 1 and the CNTTRG bit = 1 in MISRDCNTCTL, MISRDCNT counts the number of write accesses by the corresponding CPU to the address range specified by MISRBASEADR and MISRADRMSK.

MISRDCNT can count the number of times either MISR1 or MISR2 or both MISR1 and MISR2 generate signatures by setting the trigger for counting up by MISRDCNT as the signature generation condition in MISR1 or MISR2. Note, however, that if the signature generation conditions for MISR1 and MISR2 and the trigger for counting up by the data counter are not consistent, the value of MISRDCNT and the signature generation count will not match.

CAUTION

1. **When the MISR1EN and MISR2EN bits in the MISRCR register are both 0, counting by MISRDCNT is not incremented even if writing to MISRCDRL and the address range specified by MISRBASEADR and MISRADRMSK proceeded.**
2. **In the signature generation unit that does not support write monitoring mode, the counting by MISRDCNT is not incremented when the CPU write access to the monitoring address area is set as the trigger for counting up by the data counter.**

27.5.3.4 Error Notification

When the CMPERREN bit in the error notification control register (MISRERRCTL) is set to 1, the ECM is notified of an error when signatures are compared and do not match. At the same time, the error flag in the compare error status register is set.

An interrupt request for the INTC is not generated.

27.5.4 Register Specifications

27.5.4.1 Register Map

The table below lists the registers of the signature generation units.

The registers with symbols ending in "_PE1" are those of signature generation unit 1 (MISG_PE1).

The registers with symbols ending in "_PE2" are those of signature generation unit 2 (MISG_PE2).

Note that the endings of the register symbols ("_PE1" and "_PE2") are omitted if signature generation units 1 and 2 do not require identification.

MISG_PE1_base = FFC5 1000_H

MISG_PE2_base = FFC5 2000_H

Table 27.101 Registers of the Signature Generation Units (1/2)

Module Name	Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size		
						8	16	32
MISG	MISR1L_PE1	Multi input signature register 1L (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+000_H			√
MISG	MISR1H_PE1	Multi input signature register1H (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+004_H			√
MISG	MISR2L_PE1	Multi input signature register2L (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+008_H			√
MISG	MISR2H_PE1	Multi input signature register2H (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+$00C_H$			√
MISG	MISRCDRL_PE1	MISR calculation data register L (PE1)	W	—	<MISG_PE1_base>+010_H	√	√	√
MISG	MISRCR_PE1	MISR control register (PE1)	R/W	00_H	<MISG_PE1_base>+018_H	√		
MISG	MISRBASEADR_PE1	MISR monitoring area base address register (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+$01C_H$	√	√	√
MISG	MISRADRMSK_PE1	MISR monitoring area base address register (PE1)	R/W	0000 0000_H	<MISG_PE1_base>+020_H	√	√	√
MISG	MISRDCNTCTL_PE1	MISR data count control register (PE1)	R/W	00_H	<MISG_PE1_base>+024_H	√		
MISG	MISRDCNT_PE1	MISR data count register (PE1)	R/W	0000_H	<MISG_PE1_base>+028_H		√	
MISG	MISR1L_PE2	Multi input signature register 1L (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+000_H			√
MISG	MISR1H_PE2	Multi input signature register 1H (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+004_H			√
MISG	MISR2L_PE2	Multi input signature register 2L (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+008_H			√
MISG	MISR2H_PE2	Multi input signature register 2H (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+$00C_H$			√
MISG	MISRCDRL_PE2	MISR calculation data register L (PE2)	W	—	<MISG_PE2_base>+010_H	√	√	√
MISG	MISRCR_PE2	MISR control register (PE2)	R/W	00_H	<MISG_PE2_base>+018_H	√		
MISG	MISRBASEADR_PE2	MISR monitoring area base address register (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+$01C_H$	√	√	√

Table 27.101 Registers of the Signature Generation Units (2/2)

Module Name	Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size		
						8	16	32
MISG	MISRADRMSK_PE2	MISR monitoring area base address register (PE2)	R/W	0000 0000_H	<MISG_PE2_base>+020_H	√	√	√
MISG	MISRDCNTCTL_PE2	MISR data count control register(PE2)	R/W	00_H	<MISG_PE2_base>+024_H	√		
MISG	MISRDCNT_PE2	MISR data count register (PE2)	R/W	0000_H	<MISG_PE2_base>+028_H		√	

The table below lists the registers of the signature comparison unit (MSD sub-block).

MSD_base = FFC50000

Table 27.102 Registers of the Signature Comparison Unit (MSD Sub-Block)

Module Name	Register symbol	Register Name	R/W	Value after Reset	Address	Access Size		
						8	16	32
MISG	MISRCMPCTL	MISR comparator control register	R/W	0000_H	<MSD_base>+00_H	√	√	
MISG	MISRCMPERSTR	MISR compare error status register	R	00_H	<MSD_base>+04_H	√		
MISG	MISRCMPERRSTC	MISR compare error status clear register	W	00_H	<MSD_base>+08_H	√		
MISG	MISRERRCTL	MISR error notification control register	R/W	00_H	<MSD_base>+$0C_H$	√		

Note 1. In case of access to a register in a unit smaller than 32 bits, the bits that are not specified are ignored when writing and read as 0.

The following describes the control registers of the signature comparison unit.

27.5.4.2 MISRCDRL_PE1/PE2 — MISR Calculation Data Register

The MISR calculation data register is a 32-bit write-only register.

When signature generation in register write mode is selected, a signature is generated in MISR1 or MISR2 by writing to this register. Data written to this register is the data input to the multi-input signature register 1L (MISR1L) or the multi-input signature register 2L (MISR2L). For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

This register can be written in 8-, 16-, or 32-bit units. When writing proceeds in 8- or 16-bit units, remaining bits to which a value is not written are treated as 0, and 32-bit data is always input to MISR1 or MISR2

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

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

Table 27.103 MISRCDRL Register Contents

Bit position	Bit name	Functions
31 to 0	MISRCDRL31 to MISRCDRL0	Calculation Data Input data to MISR1 or MISR2. When the MISR1EN bit = 1 and the MISR1CND bit = 0 in the MISRCR register, or the MISR2EN bit = 1 and the MISR2CND bit = 0, a new signature is generated each time the MISRCDR register is written, and the result is retained in MISR1 or MISR2.

Note 1. If a signature is generated by writing to this register, the operation is only based on the 32 lower-order bits in MISR1 or MISR2 (MISR1L or MISR2L) and not on the 32 higher-order bits (MISR1H or MISR2H), the values of which remain the same.

27.5.4.3 MISR1L_PE1/PE2 — Multi-Input Signature Register 1L

The multi-input signature register 1L is a 32-bit writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR1L[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR1L[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.104 MISR1L Register Contents

Bit position	Bit Name	Functions
31 to 0	MISR1L31 to MISR1L0	Multi-Input Signature Register 1L When read, a new signature is always read.

27.5.4.4 MISR1H_PE1/PE2 — Multi-Input Signature Register 1H

The multi-input signature register 1H is a 32-bit writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR1H[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR1H[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.105 MISR1H Register Contents

Bit position	Bit Name	Function
31 to 0	MISR1H31 to MISR1H0	Multi-Input Signature Register 1H When read, a new signature is always read.

Note 1. In register write mode, this register is not updated even when the signature generation condition is met. This register is also provided in the signature generation unit that does not support write monitoring mode, but does not generate signatures.

27.5.4.5 MISR2L_PE1/PE2 — Multi-Input Signature Register 2L

The multi-input signature register 2L is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{22} + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR2L[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR2L[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.106 MISR2L Register Contents

Bit position	Bit Name	Function
31 to 0	MISR2L31 to MISR2L0	Multi-Input Signature Register 2L When read, a new signature is always read.

27.5.4.6 MISR2H_PE1/PE2 — Multi-Input Signature Register 2H

The multi-input signature register 2H is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{22} + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR2H[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR2H[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.107 MISR2H Register Contents

Bit position	Bit Name	Functions
31 to 0	MISR2H31 to MISR2H0	Multi-Input Signature Register 2H When read, a new signature is always read.

Note 1. In register write mode, this register is not updated even when the signature generation condition is met. This register is also provided in the signature generation unit that does not support write monitoring mode, but does not generate signatures.

27.5.4.7 MISRCR_PE1/PE2 — MISR Control Register

The MISR control register is an 8-bit readable and writable register.

The MISR1EN and MISR2EN bits are used to enable/disable signature generation in MISR1 and MISR2. When the MISR1EN or MISR2EN bit is 1, MISR1 or MISR2 generates a signature and retains the generated value. When the MISR1EN or MISR2EN bit is 0, MISR1 or MISR2 does not generate signatures and the values of these registers are not updated.

When the MISR1EN or MISR2EN bit is 1, the MISR1CND or MISR2CND bit selects the signal generation condition in MISR1 or MISR2. Setting the MISR1CND or MISR2CND bit to 0 selects signature generation in register write mode by MISR1 or MISR2, so writing to MISRCDRL leads to signature generation. Setting the MISR1CND or MISR2CND bit to 1 selects signature generation in write monitoring mode by MISR1 or MISR2, writing by the CPU to the address range specified by the MISRBASEADR and MISRADRMSK registers leads to signature generation. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MISR2CND	MISR1CND	MISR2EN	MISR1EN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 27.108 MISRCR Register Contents

Bit position	Bit Name	Functions
7 to 4	—	Reserved. These bits are always read as 0. The write value should also be 0.
3	MISR2CND	MISR2 Signature Write Control This bit sets the signature generation condition in MISR2 when the MISR2EN bit = 1. 0: Register write mode 1: Write monitoring mode
2	MISR1CND	MISR1 Signature Write Control This bit sets the signature generation condition in MISR1 when the MISR1EN bit = 1. 0: Register write mode 1: Write monitoring mode
1	MISR2EN	MISR2 Enable 0: MISR2 does not generate a signature. 1: MISR2 generates a signature and the MISR2H and MISR2L values are updated.
0	MISR1EN	MISR1 Enable 0: MISR1 does not generate a signature. 1: MISR1 generates a signature and the MISR1H and MISR1L values are updated.

27.5.4.8 MISRBASEADR_PE1/PE2 — MISR Monitoring Area Base Address Register

MISRBASEADR specifies the CPU write access area to be monitored by MISG when signature generation in write monitoring mode is selected. In combination with the setting of the monitoring area mask address register, this register specifies the address range of the monitoring area. For the conditions for signature generation, see **Section 27.5.3.1, Conditions for Signature Generation**.

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

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

Table 27.109 MISRBASEADR Register Contents

Bit Position	Bit Name	Function
31 to 0	BASEADR31 to BASEADR0	Monitoring Area Base Address Register

For the mechanism to judge access to the monitoring area, see **Section 27.5.4.9, MISRADRMSK_PE1/PE2 — MISR Monitoring Area Address Mask Register**.

27.5.4.9 MISRADRMSK_PE1/PE2 — MISR Monitoring Area Address Mask Register

This register sets the area for which MISG monitors the CPU's write accesses, based on the conditions for generating signatures in write monitoring mode. Specify the settings of the monitoring area base address register and the address range of the monitoring area. For details about the conditions for generating signatures, see **Section 27.5.3.1, Conditions for Signature Generation**.

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

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

Table 27.110 MISRADRMSK Register Contents

Bit Position	Bit Name	Function
31 to 0	ADRMSK31 to ADRMSK0	Monitoring area mask address register

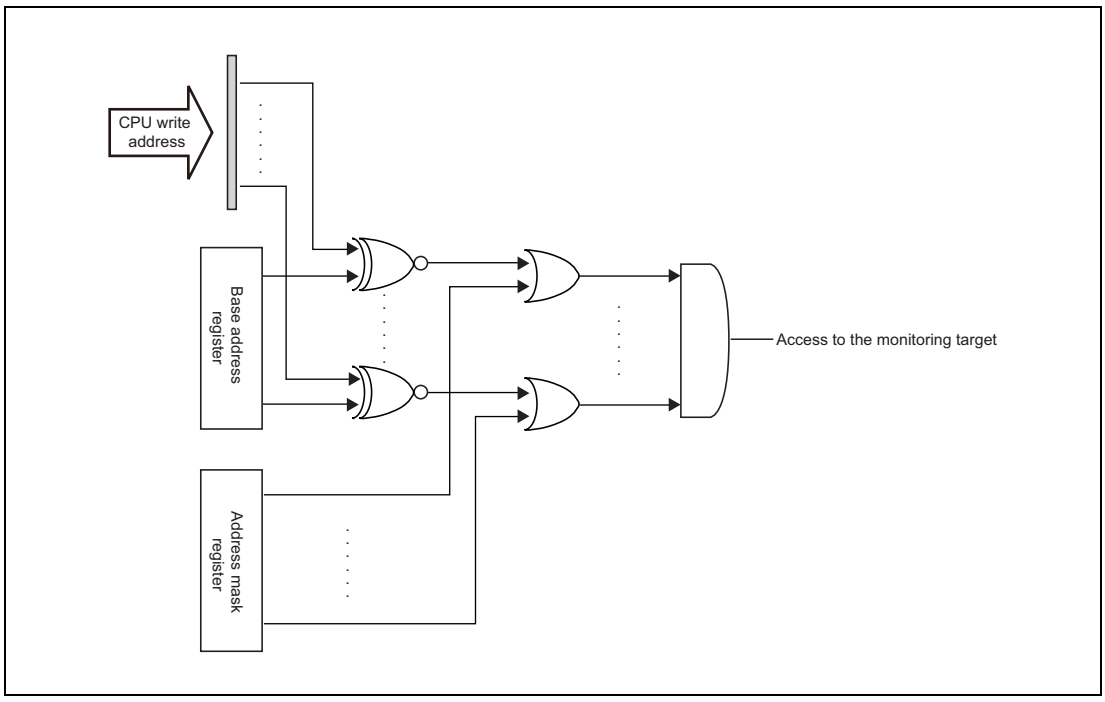


Figure 27.7 Mechanism of Judging the Access to the Monitoring Area

27.5.4.10 MISRDCNTCTL_PE1/PE2 — MISR Data Counter Control Register

This register controls operation of the MSIR data counter register. If the event selected by the CNTTRG bit occurs while the CNTSTA bit is 1, the data counter is incremented. For operation of the data counter, see **Section 27.5.3.3, Data Counter**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CNTTRG	CNTSTA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 27.111 MISRDCNTCTL Register Contents

Bit Position	Bit Name	Functions
7 to 2	—	Reserved. These bits are always read as 0. The write value should also be 0.
1	CNTTRG	Count-Up Trigger Select This bit selects the trigger for counting up by the data counter. 0: Write access to the MISRCDRL register 1: Write access to the address area specified by the MISRBASEADR and MISRADRMSK registers
0	CNTSTA	Data Counter Start This bit is an enable bit for the data counter. If the event selected by the CNTTRG bit occurs while CNTSTA = 1, the data counter is incremented. 0: The data counter is stopped. 1: The data counter is operating.

27.5.4.11 MISRDCNT_PE1/PE2 — MISR Data Counter Register

The data counter is a 16-bit readable and writable register.

If the data counter values of two signature generation units selected for comparison match, automatic comparison of the signatures proceeds.

When the CNTTRG bit in the data counter control register is 0, the data counter is incremented by write access to the MISR calculation data register. When the CNTTRG bit in the data control register is 1, the data counter is incremented by write access by the corresponding CPU to the address area specified by the MISRBASEADR and MISRADRMSK registers. For operation of the data counter, see **Section 27.5.3.3, Data Counter**.

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

Table 27.112 MISRDCNT Register Contents

Bit Position	Bit Name	Function
15 to 0	DCNT15 to DCNT0	Data Counter Register

The following describes the control registers of the MSD unit (signature comparison unit).

27.5.4.12 MISRCMPCTL — MISR Comparator Control Register

MISRCMPCTL is a 16-bit readable and writable register.

This register controls the comparator that compares signatures generated in the signature generation units. For automatic comparison of signatures, see **Section 27.5.3.2, Automatic Signature Comparison**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MISR2 CMP0	—	—	MISR1 CMP0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W

Table 27.113 MISRCMPCTL Register Contents

Bit Position	Bit Name	Functions
15 to 4	—	Reserved. These bits are always read as 0. The write value should always be 0.
3	MISR2CMPEN0	MISR2 Signature Compare Enable 0 This bit controls comparison of signatures retained in MISR2 of MISG_PE2 and MISG_PE1. 0: Disables comparison 1: Enables comparison
2, 1	—	Reserved. These bits are always read as 0. The write value should always be 0.
0	MISR1CMPEN0	MISR1 Signature Compare Enable 0 This bit controls comparison of signatures retained in MISR1 of MISG_PE1 and MISG_PE2. 0: Disables comparison 1: Enables comparison

27.5.4.13 MISRCMPERSTR — MISR Compare Error Status Register

The compare error status register is an 8-bit readable register.

If a mismatch occurs in signature comparison enabled by the comparator control register, the corresponding error flag is set.

The error flag is cleared by writing 1 to the corresponding clear bit in the compare error status clear register. The flag is also cleared by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MISR2ERR0	—	—	MISR1ERR0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.114 MISRCMPERSTR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved. These bits are always read as 0. The write value should always be 0.
3	MISR2ERR0	MISR2 Signature Compare Error Flag 0 This flag is set when signatures retained in MISR2 of MISG_PE1 and MISG_PE2 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
2, 1	—	Reserved. These bits are always read as 0. The write value should always be 0.
0	MISR1ERR0	MISR1 Signature Compare Error Flag 0 This flag is set when signatures retained in MISR1 of MISG_PE1 and MISG_PE2 are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.

27.5.4.14 MISRCMPERRSTC — MISR Compare Error Status Clear Register

The compare error status clear register is an 8-bit write-only register.

When an error flag in the compare error status register is 1, writing 1 to corresponding clear bit clears the error flag. Read the MISR compare error status register and write 1 to the clear bit for the flag being 1.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MISR2ERR0	—	—	MISR1ERR0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	R	R	W

Table 27.115 MISRCMPERRSTC Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved
3	MISR2CLR0	MISR2 Signature Compare Error Clear 0 Writing 1 to this bit clears the MISR2ERR0 bit in the CMPERRST register.
2, 1	—	Reserved
0	MISR1CLR0	MISR1 Signature Compare Error Clear 0 Writing 1 to this bit clears the MISR1ERR0 bit in the CMPERRST register.

27.5.4.15 MISRERRCTL — MISR Error Notification Control Register

The error notification control register is an 8-bit readable and writable register.

This register enables or disables error notification when signatures are compared by the automatic signature comparison and do not mach. For the automatic signature comparison and error notification, see **Section 27.5.3.2, Automatic Signature Comparison**, and **Section 27.5.3.4, Error Notification**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CMPERREN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 27.116 MISRERRCTL Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	CMPERREN	Compare Error Notification Enable This bit enables or disables error notification when signatures are compared and do not mach. 0: Error notification does not proceed even if a mismatch has occurred in signature comparison. 1: Error notification proceeds if a mismatch has occurred in signature comparison.

27.5.5 Usage

27.5.5.1 Usage Example 1

The self-diagnostic program eases diagnosis. By using the MISG to compress intermediate transitions, the self-diagnostic program makes the saving and comparison of all intermediate results unnecessary. Results of self-diagnosis including intermediate transitions can be judged by comparing the results of compression by the MISG with the expected results when the program ends.

This has the effect of reducing requirements for memory capacity and times for processing comparisons (substitution by CRCs is also possible).

Example of Settings (when PE1 is running the self-diagnostic program)

The descriptions of registers of an MISG below applies to the registers of signature generation unit 1 (MISG_PE1), which is for PE1.

(1) Initialize the multi-input signature register 1 (MISR1H/MISR1L), multi-input signature register 2 (MISR2H/MISR2L) and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 1 to select signature generation by MISR1 or MISR2 in write monitoring mode.

(3) Use the MISR monitoring area base address register (MISRBASEADR) and MISR monitoring area address mask register (MISRADRMSK) to set the address area for monitoring.

(4) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(5) Run the self-diagnostic program on PE1.

(6) Upon completion of execution of the self-diagnostic program, MISR1H and MISR1L or MISR2H and MISR2L data is compared with the expected value in flash memory.

27.5.5.2 Usage Example 2

Use multiple processors to run the same processing (including the self-diagnostic program) to confirm the correctness of results. Comparing results from different hardware raises reliability.

Example of Settings (when PE1 and PE2 are running the same task)

The descriptions of registers of an MISG below applies to the MISG registers of PE1 and PE2.

(1) Initialize the multi-input signature register 1 (MISR1H or MISR1L), multi-input signature register 2 (MISR2H or MISR2L), and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 0 to select signature generation by MISR1 or MISR2 in register write mode.

(3) Set MISR1CMPEN1 or MISR2CMPEN1 in the MISR comparator control register (MISRCMPCTL) to 1 to enable signature comparison by the comparator.

(4) Set the CMPERREN bit in the MISR error notification control register to 1 to enable error notification to the ECM.

(5) Set the CNTTRG bit in the MISR data counter control register (MISRDCNTCTL) to 0 to set writing to MISRCDRL as the trigger for counting up by the data counter. Set the CNTSTA bit to 1 to enable operation of the data counter.

(6) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(7) Run the self-diagnostic program on all CPUs.

(8) The self-diagnostic program stores intermediate results while the program is running in MISRCDRL of the signature generation units. The signatures in the MISR1 and MISR2 registers of MISG_PE1 and MISG_PE2 are compared whenever the values of the data counter registers (MISRDCNT) of MISG_PE1 and MISG_PE2 match.

(9) Check the comparison status register to see if there were errors in comparison.

27.6 Clock Monitors

27.6.1 Overview

This product incorporates the clock monitors to monitor the clock operation by detecting the abnormal frequency of the clock to be monitored. The clock monitors provide the following functions.

- Monitors to see if the frequency of the clock to be monitored is within the specified range based on the sampling clock.
- Issues an error notice to the ECM upon detection of the abnormal state of the clock.

Table 27.117 shows the clocks monitored by the clock monitors and the sampling clocks used.

Table 27.117 List of Clocks Monitored by Each Clock Monitor and Sampling Clocks Used

Clock Monitor Channel	Monitored Clock	Sampling Clock
CLMA0	40 MHz SSCG	40 MHz clean
CLMA1	40 MHz clean	10 MHz (1/2 main OSC)
CLMA2	WDTA count clock (1/80 main OSC)	CLK_LIOSC

27.6.2 List of Registers

27.6.2.1 Clock Monitor Channel Registers

Table 27.118 List of Registers

Module Name	Address	Symbol	Register Name	R/W	Value after Reset	Access Size
CLMAn	<Base_adr + 00_H>	CLMAnCTL0	CLMAn control register 0	R/W	00_H	8
CLMAn	<Base_adr + 08_H>	CLMAnCMPL	CLMAn compare register L	R/W	0001_H	16
CLMAn	<Base_adr + $0C_H$>	CLMAnCMPH	CLMAn compare register H	R/W	$03FF_H$	16
CLMAn	<Base_adr + 10_H>	CLMAnPCMD	CLMAn protection command register	W	00_H	8
CLMAn	<Base_adr + 14_H>	CLMAnPS	CLMAn protection command status register	R	00_H	8

The base addresses of the registers are shown below.

Table 27.119 Register Base Addresses

Clock Monitor Channel	<Base_addr>
CLMA0	$FFF8\ 8400_H$
CLMA1	$FFF8\ 8420_H$
CLMA2	$FFF8\ 8440_H$

27.6.2.2 Common Registers

Table 27.120 Common Registers

Module Name	Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size	Protection
CLMAC	CLMATEST	CLMA self-test register	R/W	0000 0000$_H$	FFF8 8204$_H$	32	PROT1PHCMD*1
CLMAC	CLMATESTS	CLMA self-test status register	R/W	0000 0000$_H$	FFF8 8208$_H$	32	

Note 1. For this register, refer to **Section 10, Clock Controller**.

27.6.3 Details of Registers

27.6.3.1 CLMAnCTL0 — CLMAn Control Register 0

CLMAnCTL0 controls the operation of the clock monitors. CLMAnCTL0 is protected by the CLMAnPCMD register.

CLMAnCTL0 can be initialized by either an internal reset or an external reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAnCLME
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W*1

Note 1. CLMAnCTL0 can be cleared by a reset. Writing 0 is ignored.

Table 27.121 CLMAnCTL0 Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved. These bits are always read as 0. The write value should always be 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	CLMAnCLME	Clock Monitor Operation 0: Disables operation. 1: Enables operation.

27.6.3.2 CLMAnCMPL — CLMAn Compare Register L

CLMAnCMPL sets the lower limit of the normal frequency range used for comparison.

CLMAnCMPL can be initialized by either an internal reset or an external reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAn CMPL11	CLMAn CMPL10	CLMAn CMPL9	CLMAn CMPL8	CLMAn CMPL7	CLMAn CMPL6	CLMAn CMPL5	CLMAn CMPL4	CLMAn CMPL3	CLMAn CMPL2	CLMAn CMPL1	CLMAn CMPL0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.122 CLMAnCMPL Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Reserved. These bits are always read as 0. The write value should always be 0.
11 to 0	CLMAnCMPL [11:0]	Lower Limit of Normal Frequency Range CLMAnCMPL can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to CLMAnCMPL is invalid.

27.6.3.3 CLMAnCMPH — CLMAn Compare Register H

CLMAnCMPH sets the upper limit of the normal frequency range used for comparison.

CLMAnCMPH can be initialized by either an internal reset or an external reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAn CMPH11	CLMAn CMPH10	CLMAn CMPH9	CLMAn CMPH8	CLMAn CMPH7	CLMAn CMPH6	CLMAn CMPH5	CLMAn CMPH4	CLMAn CMPH3	CLMAn CMPH2	CLMAn CMPH1	CLMAn CMPH0
Value after reset	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.123 CLMAnCMPH Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Reserved. These bits are always read as 0. The write value should always be 0.
11 to 0	CLMAnCMPH [11:0]	Upper Limit of Normal Frequency Range CLMAnCMPH can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to CLMAnCMPH is invalid.

27.6.3.4 CLMAnPCMD — CLMAn Protection Command Register

CLMAnPCMD is a special sequential register for CLMAnCTL0.

CLMAnPCMD can be initialized by either an internal reset or an external reset.

Bit	7	6	5	4	3	2	1	0
	CLMAnPCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

See **Section 27.6.6.1**, regarding this protection method on details.

27.6.3.5 CLMAnPS — CLMAn Protection Command Status Register

CLMAnPS is a special sequential register for CLMAnCTL0.

CLMAnPS can be initialized by either an internal reset or an external reset

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAn PRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.124 CLMAnPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved. These bits are always read as 0.
0	CLMAnPRERR	Protection Error Detection 0: A protection error has not been generated. 1: A protection error has been generated.

- Operating conditions of the CLMAnPRERR bit
  Setting condition: Access to CLMAnCTL0, which is the target for protection specified by CLMAnPCMD, without following the protection unlock sequence.
  Clearing condition: Writing of $A5_H$ to the CLMAnPCMD register (step 1 in the protection unlock sequence).

27.6.3.6 CLMATEST — CLMA Self-Test Register

CLMATEST is used for self-testing of CLMA2 to CLMA0.

CLMATEST can be protected by the PROT1PHCMD register.

CLMATEST is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLMA2TESTEN	CLMA1TESTEN	CLMA0TESTEN	ERRMSK	MONCLKMSK	RESCLM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.125 CLMATEST Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved. These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 3	CLMA2TESTEN CLMA1TESTEN CLMA0TESTEN	These bits enable or disable self-testing of CLMA2 to CLMA0. 0: Disables self-testing of the corresponding CLMAn. 1: Enables self-testing of the corresponding CLMAn.
2	ERRMSK	Masks an error notification to the ECM when CLMAn detects an error. When the ERRMSK is set for a certain CLMAn, the associated CLMAn does not issue an error notification to the ECM even if it detects an error. ERRMSK setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not mask an error notification to the ECM. 1: Masks an error notification to the ECM.
1	MONCLKMSK	Fixes the level of the clock input to the CLMAn that should be monitored, to the low level. MONCLKMSK setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not fix the clock input to the CLMAn that should be monitored to the low level. 1: Fixes the clock input to the CLMAn that should be monitored to the low level.
0	RESCLM	Initializes CLMAn forcibly. RESCLM setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not initialize CLMAn. 1: Initializes CLMAn.

27.6.3.7 CLMATESTS — CLMA Self-Test Status Register

CLMATESTS indicates the self-testing result of CLMA2 to CLMA0.

CLMATESTS is initialized by an internal reset or an external reset.

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLMA2 ERRS	CLMA1 ERRS	CLMA0 ERRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.126 CLMATESTS Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved. These bits are always read as 0.
2 to 0	CLMA2ERRS CLMA1ERRS CLAM0ERRS	These bits indicate whether or not CLMA2 to CLMA0 have detected an error. These bits are not affected by CLMATEST.ERRMSK. 0: The corresponding CLMAn has not yet detected an error. 1: The corresponding CLMAn has detected an error.

27.6.4 Detection of Abnormal Clock Frequency

Method of detection

1. CLMAn counts the number of rising edges of the monitored clock signal within 16 sampling clock cycles, and compares the count with the specified thresholds.
   - The lower threshold is specified using the CLMAnCMPL[11:0] bits in CLMAnCMPL register.
   - The upper threshold is specified using the CLMAnCMPH[11:0] bits in CLMAnCMPH register.
2. When the monitored clock frequency is so low*1 that the count falls below the value set in the CLMAnCMPL[11:0] bits in the CLMAnCMPL register, CLMAn notifies the ECM of the abnormal clock. CLMAn also notifies the ECM of the abnormal clock when the clock frequency is so high that the count exceeds the value set in CLMAnCMPH[11:0] in the CLMAnCMPH register.

   Note that even if the frequency of the monitored clock fluctuates during the sampling period, an error is not notified as long as the number of detected edges falls within the specified range.

Note 1. The abnormal state of the clock may not be detected when the monitored clock comes to a complete stop.

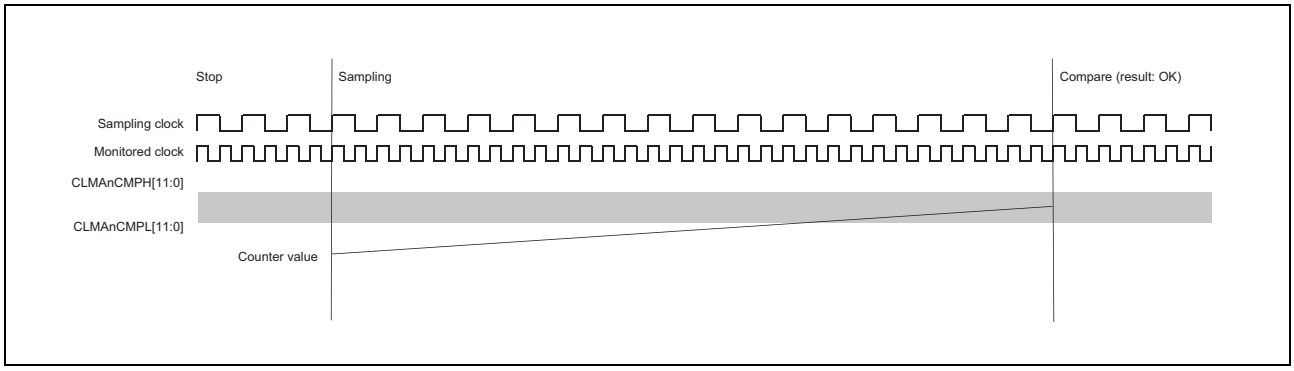


Figure 27.8 Operation when Clock Frequency is within the Specified Range

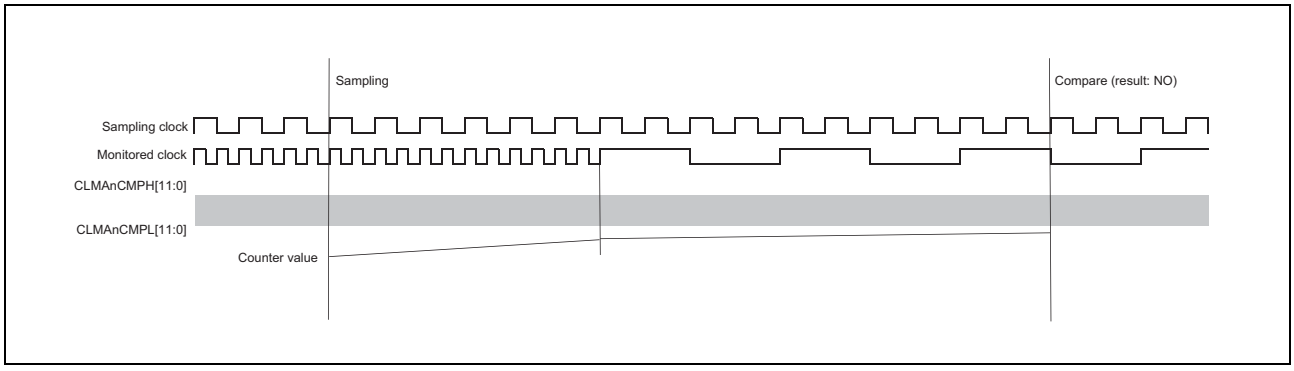


Figure 27.9 Operation when Clock Frequency is Lower than the Specified Range

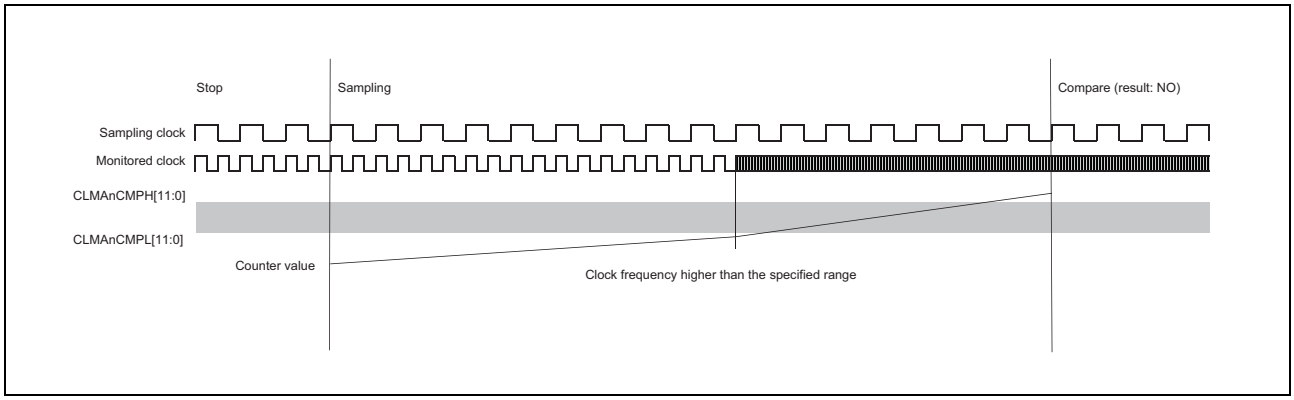


Figure 27.10 Operation when Clock Frequency is Higher than the Specified Range

(1) Calculation method of the thresholds CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]

The compare registers CLMAnCMPL and CLMAnCMPH are configured with the minimum and maximum values of clock cycles of the monitored clock CLMATMON that are assumed to be valid within 16 cycles of the sampling clock CLMATSMP.

The expected number of clock cycles is denoted by N.

$$\frac{16}{f_{CLMATSMP}} = \frac{N}{f_{CLMATMON}}$$

$$N = \frac{f_{CLMATMON}}{f_{CLMATSMP}} \times 16$$

Considering the allowed frequency deviations of CLMATMON and CLMATSMP, the threshold values can be calculated by the following formulas:

$$\text{Lower threshold} = N_{min} = \frac{f_{CLMATMON(min)}}{f_{CLMATSMP(max)}} \times 16 - 1$$

$$\text{Upper threshold} = N_{max} = \frac{f_{CLMATMON(max)}}{f_{CLMATSMP(min)}} \times 16 + 1$$

NOTE

PLL jitter is covered by “+1” and “–1” as in the formulas.

Example

When $f_{CLMATSMP}$ = 240 kHz (±8%) and $f_{CLMATMON}$ = 16 MHz (±5%), the recommended thresholds are:

$$N_{min} = 15{,}200 / 259.2 \times 16 - 1$$
$$= 937.27$$
$$CLMAnCMPL = 937 = 03A9_H$$

$$N_{max} = 16{,}800 / 220.8 \times 16 + 1$$
$$= 1218.39$$
$$CLMAnCMPH = 1219 = 04C3_H$$

Minimum thresholds

The following restrictions must be taken into account:

- $CLMAnCMPL \geq 0001_H$
- $CLMAnCMPH \geq CLMAnCMPL + 0003_H$

(2) Definition of the input of initial values to the threshold registers

The values of the threshold registers after a reset are as follows.

- $CLMAnCMPL[11:0] = 001_H$
- $CLMAnCMPH[11:0] = 3FF_H$

27.6.5 Self-Diagnosis

Self-diagnosis of the clock monitor is available as described below. In the description, the clock monitor to be self-diagnosed is operating.

(1) Set the threshold for clock monitor to be self-diagnosed (by setting CLMAnCMPL/ CLMAnCMPH). In this case, set the threshold so as an error always occurs.

(2) Specify the clock monitor to be self-diagnosed.
Setting the CLMATEST.CLMAnTESEN bit to 1 enables specifying the corresponding clock monitor to be self-diagnosed.

(3) To prevent error notice to the ECM based on self-diagnosis, set CLMATEST.ERRMSKL simultaneously with step (2).

(4) Enable the operation of clock monitor by setting the CLMAnCTL0.CLMAnCLME bit to 1.

(5) Wait for the time long enough to allow error occurrence, read the CLMATESTS register to see if an error has been generated in the clock monitor to be self-diagnosed. The time from the start of self-diagnosis to the error occurrence depends on the CLMAnCMPL setting and the start point of self-diagnosis in the sampling period. A maximum of two sampling cycles are required.

(6) Clear the error generated by self-diagnosis.
Setting CLMATEST.RESCLM to 1 enables initializing the clock monitor to be self-diagnosed.

(7) Terminate self-diagnosis.
Setting all the bits in CLMATEST to 0 enables terminating self-diagnosis.

Before restarting the clock monitor that has been self-diagnosed, set the registers again as required.

27.6.6 Notes on Register Setting

27.6.6.1 Writing to Protected Registers

Writing to the CLMAnCTL0 register (n = 0 to 2) of each clock monitor is only possible with the protection unlock sequence described below.

Step 1. Write the fixed value ($A5_H$) to the CLMAnPCMD register.

Step 2. Write the new setting to the CLMAnCTL0 register. Write the value after a reset to the reserved bits.

Step 3. Write the bitwise inverse of the setting value to the CLMAnCTL0 register. Write the inverse of the value after a reset to the reserved bits.

Step 4. Write the new setting to the CLMAnCTL0 register. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the CLMAnPS.CLMAnPRERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the CLMAnPS.CLMAnPRERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module, writing to the protected register fails and the CLMAnPS.CLMAnPRERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

For writing to CLMATEST, see the description of the PROT1PHCMD register.

27.6.6.2 Setting CLMAnCMPL/CLMAnCMPH Registers

The CLMAnCMPL/CLMAnCMPH registers should be set so that the following conditions are satisfied. If the clock monitor is otherwise used, operation cannot be guaranteed.

- $1 \leq$ CLMAnCMPL
- CLMAnCMPL + $3 \leq$ CLMAnCMPH (n = 0, 1, or 2)

27.7 BIST

This product incorporates the function to detect failures of the failure detection function itself, which is referred to as BIST. The functions shown in the table below have the dual hardware configuration, and a BIST error notice is issued to the ECM if a failure occurs in any function.

Table 27.127 Failure Detection

BIST-Applied Function	Remark
Lockstep comparator	Lockstep comparator for CPU1
ECC decoder	
Address parity decoder	

For the ECC decoder for the peripheral RAM and data flash, the hardware including the control registers has dual configuration. When using the BIST function, set the same values to the control registers of the master and checker.

For the other ECC decoders, address parity checker, and control registers do not have dual configuration.

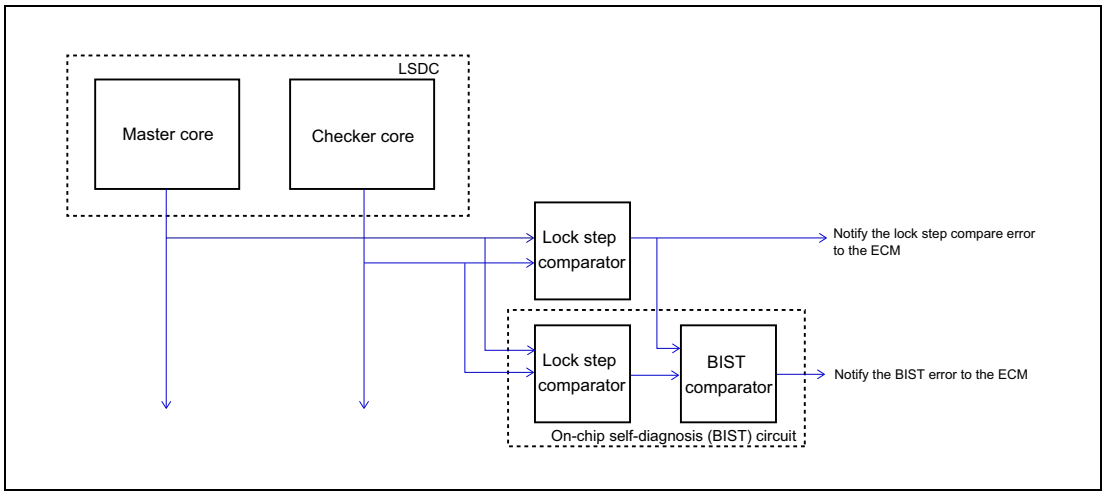


Figure 27.11 Lock Step Configuration

27.8 ECM

The ECM monitors various failure detection states in the LSI chip, and defines the operation to be carried out upon failure detection. For the details of the ECM, refer to **Section 28, Error Control Module (ECM)**.

Section 28 Error Control Module (ECM)

This section describes an error control module (ECM).

The first part of this section describes all RH850/C1x specific properties, such as the number of units, register base addresses, etc. The remainder of the section describes the functions and registers of the ECM.

28.1 Features of RH850/C1x ECM

28.1.1 Number of Units

This microcontroller has the following number of units of the ECM.

Table 28.1 Number of Units

Product	RH850/C1x
Number of Units	1
Name	ECM

28.1.2 Register Base Address

ECM base addresses are listed in the following table.

ECM register addresses are given as offsets from the base addresses in general.

Table 28.2 Register Base Address

Base Address Name	Base Address
ECM master <ECMM_base>	FFCB 0000_H
ECM checker <ECMC_base>	FFCB 1000_H
ECM common part <ECM_base>	FFCB 2000_H

28.1.3 Clock Supply

ECM clock is listed in the following table.

Table 28.3 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
ECM	PCLK	CLK_LSB (Low-speed peripheral clock (peripheral clock))

28.1.4 Interrupt and DMA/DTS

ECM interrupt requests are listed in the following table.

Table 28.4 Interrupt Request

Interrupt Name (Overview)	Interrupt Number	DMA Trigger Number	DTS Trigger Number
Error control module NMI interrupt	(FEINT)	—	—
Error control module interrupt	8	—	—

28.1.5 Reset Sources

ECM reset sources are listed in the following table. ECM is initialized by these reset sources.

However, the ECM master/checker error source status register is initialized only by external reset.

For details, see **Section 28.3, Registers**.

Table 28.5 Reset Sources

Unit Name	Reset Source
ECM	Reset by all reset sources.

28.1.6 External Input/Output Signals

External input/output signals of ECM are listed in the following table.

Table 28.6 External Input/Output Signals

Unit Signal Name	Description	Alternative Port Pin Signal Name
$\overline{\text{ERROROUT\_M}}$	ERROROUT pin (master)	$\overline{\text{ERROROUT\_M}}$
$\overline{\text{ERROROUT\_C}}$	ERROROUT pin (checker)	$\overline{\text{ERROROUT\_C}}$

28.2 Overview

28.2.1 Function Overview

The error control module (ECM) collects error signals coming from different error sources and monitoring circuits. It also outputs error signals from the ERROROUT pins ($\overline{\text{ERROROUT\_M}}$ and $\overline{\text{ERROROUT\_C}}$) and generates interrupts and internal reset signals. **Table 28.7** shows the specification overview of ECM.

Table 28.7 Function Overview

Item	Description
Safety processing	ECM can handle the following processing in response to error signal inputs from individual modules. • Error flag set • Maskable interrupt generation Maskable interrupt generation can be controlled (enabled/disabled) for individual errors. • FE level interrupt generation FE level interrupt generation can be controlled (enabled/disabled) for individual errors. • Internal reset generation Internal reset generation can be controlled (enabled/disabled) for individual errors. • ERROROUT output Pin output mask can be controlled (enabled/disabled) for individual errors. Output can be toggled in response to a timer input or made at a fixed level.
Error status	ECM incorporates the ECM master/checker error source status register, which can be used to confirm the error status from the error flag. The error flags are only cleared by an external reset or by writing the corresponding bit of the ECM error source status clear trigger register to 1. In case of an internal reset, the error flags are kept and the reset generation source can be confirmed by reading the ECM master/checker error source status register after reset.
Debug, self-diagnosis	• Pseudo errors can be generated for debug and self-diagnosis. The operation during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for the mask to the ERROROUT output, interrupt, or internal reset apply in the same way. • ECM incorporates a loop-back function of the ERROROUT output that is used to diagnose the path to the ERROROUT pin. The status of the ERROROUT pin is reflected to the ECM master/checker error source status register and can be confirmed by reading the register.
Timeout function	ECM incorporates a function that generates an ERROROUT output or internal reset when the value counted by the delay timer matches with the ECM delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The value counted by the delay timer is in cycles of the low-speed peripheral clock.
Register protection	A write-protection with a special sequence is incorporated to protect registers from inadvertent write access.
Others	ECM is duplexed. The ERROROUT pin is duplexed to the two pins of the master pin and the checker pin. The ERROROUT outputs from the ECM master and ECM checker are constantly compared. If they do not match, an ECM compare error (error source 26) occurs.

28.2.2 Block Diagram

Figure 28.1 is a block diagram of the ECM.

The ERROROUT output, internal reset, and ERROROUTZ signals are active low, while the maskable and FE level interrupt signals are active high.

Note that the ERROROUTZ signal for PIC1A is not toggled even if the ECM is in dynamic mode.

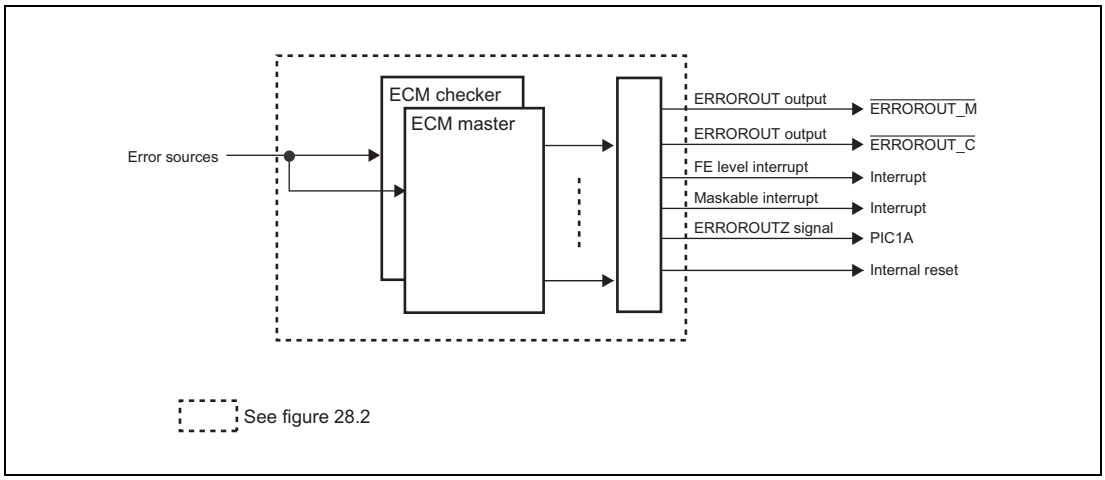


Figure 28.1 Outline of ECM

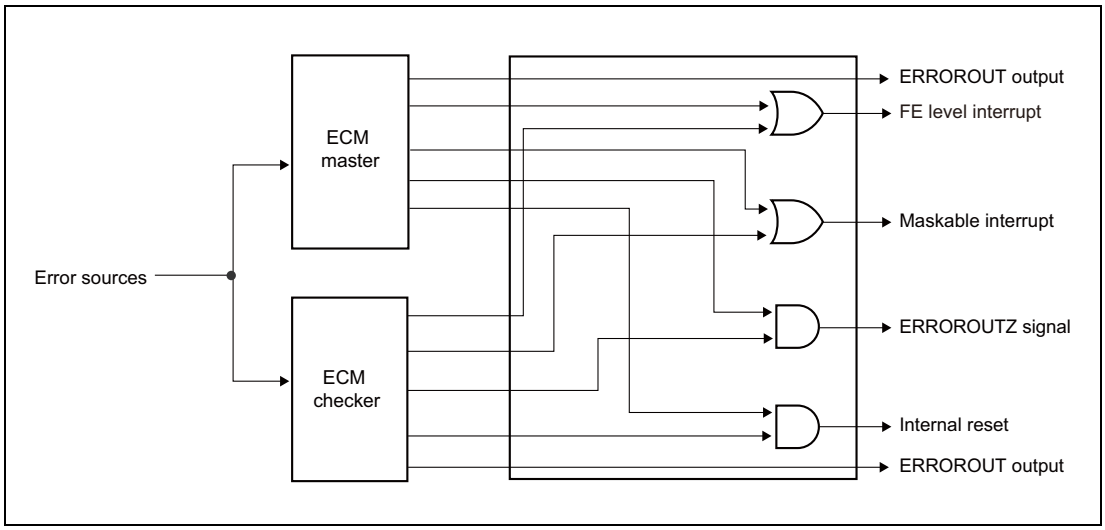


Figure 28.2 Connection of ECM

28.2.3 Error Sources and Safety Processing

Table 28.8 lists the error sources and safety processing of the ECM of RH850/C1x.

Table 28.8 List of Error Sources and Safety Processing (1/2)

Error Source No.	Module	Error Source	Error Flag Set	Maskable Interrupt	Non-maskable Interrupt	FE Level Internal Reset	ERRROUT Output	Delay Timer Start
0	WDTA	WDTA error*2	√	√	√	√*1	√	√
1	Reserved		—	—	—	—	—	—
2	Reserved		—	—	—	—	—	—
3	Reserved		—	—	—	—	—	—
4	Lock step	Lock step compare error*3	√	√	√	√	√	√
5	MISG	MISG compare error*3	√	√	√	√	√	√
6	RAM	Local RAM (CPU1, CPU2) 2-bit ECC error and local RAM (CPU1) address parity error*3	√	√	√	√	√	√
7		Local RAM (CPU1, CPU2) 1-bit ECC error and local RAM (CPU1) parity bit error*3	√	√	√	√	√	√
8		Global RAM 2-bit ECC error and address parity error*3	√	√	√	√	√	√
9		Global RAM 1-bit ECC error*3	√	√	√	√	√	√
10		Peripheral RAM (RS-CAN, CSIH, DTS) 2-bit ECC error*3	√	√	√	√	√	√
11		Peripheral RAM (RS-CAN, CSIH, DTS) 1-bit ECC error*3	√	√	√	√	√	√
12	Code flash	Code flash 2-bit ECC error and address parity error*3	√	√	√	√	√	√
13		Code flash 1-bit ECC error*3	√	√	√	√	√	√
14	Instruction cache	Instruction cache data (CPU1, CPU2) 2-bit ECC error*3	√	√	√	√	√	√
15		Instruction cache data (CPU1, CPU2) 1-bit ECC error*3	√	√	√	√	√	√
16		Instruction cache tag (CPU1, CPU2) 2-bit ECC error*3	√	√	√	√	√	√
17		Instruction cache tag (CPU1, CPU2) 1-bit ECC error*3	√	√	√	√	√	√
18	Data flash	Data flash 2-bit ECC error*3	√	√	√	√	√	√
19		Data flash 1-bit ECC error*3	√	√	√	√	√	√
20	PE guard (PEG)	PEG error*4	√	√	√	√	√	√
21	Global RAM guard (GRG)	GRG error*3	√	√	√	√	√	√
22	PBG	PBG error*3	√	√	√	√	√	√
23	Reserved		—	—	—	—	—	—
24	Reserved		—	—	—	—	—	—
25	Data parity	Data parity error*3	√	√	√	√	√	√
26	ECM	ECM compare error*5	√	√	√	√	√	√

Table 28.8 List of Error Sources and Safety Processing (2/2)

Error Source No.	Module	Error Source	Error Flag Set	Maskable Interrupt	Non-maskable Interrupt	FE Level Internal Reset	ERROROUT Output	Delay Timer Start
27	Clock monitor	Clock monitor error (main oscillation)*3	√	√	√	√	√	√
28		Clock monitor error (PLL0 (SSCG))*3	√	√	√	√	√	√
29		Clock monitor error (PLL1 (clean))*3	√	√	√	√	√	√
30 to 36	Reserved		—	—	—	—	—	—
37	ADCC	AD parity error*6	√	√	√	√	√	√
38	Flash	Flash access error*7	√	√	√	√	√	√
39		FACI reset transfer error*7,*13	√	—	—	—	√	—
40	Reserved		—	—	—	—	—	—
41	DMAC	DTS RAM Data ECC SEC-DED*8	√	√	√	√	√	√
42	BIST	Error detection using on-chip self-diagnostic (BIST) circuit*3	√	√	√	√	√	√
43	DMAC	DMA violation access notice*8	√	√	√	√	√	√
44	OSTM	OSTM1 interrupt *9	√	√	√	√	√	√
45	TSG3	TSG3 error signal (INTTSG3nIER)*10	√	√	√	√	√	√
46	EMU2	EMU2 error signal*11	√	√	√	√	√	√
47	RDC2	RDC2 error detection signal*12	√	√	√	√	√	√
48 to 60	Reserved		—	—	—	—	—	—
61	ECM	Time out function of the ECM delay timer*5	√	—	—	√	√	—
62		Error set by ECMmESET*5	√	—	—	—	—	—
63		Error output loop back state*5	√	—	—	—	—	—

Note 1. In the initial state, generation of an internal reset is allowed.
Note 2. For the details of this error, see **Section 15, Window Watchdog Timer A (WDTA)**.
Note 3. For the details of this error, see **Section 27, Functional Safety**.
Note 4. For the details of this error, see **Section 3, CPU System**.
Note 5. For the details of this error, see **Table 28.7, Function Overview**.
Note 6. For the details of this error, see **Section 26, A/D Converter (ADCC)**.
Note 7. For the details of the errors No. 38 and No. 39, refer to the *RH850/ C1x Flash Memory User's Manual: Hardware Interface*.
Note 8. For the details of this error, see **Section 7, DMA Controller**.
Note 9. For the details of this error, see **Section 16, OS Timer (OSTM)**.
Note 10. For the details of this error, see **Section 19, Motor Control Timer (TSG3)**.
Note 11. For the details of this error, see **Section 24, Enhanced Motor Control Unit (EMU2)**.
Note 12. For the details of this error, see **Section 25, R/D Converter (RDC2)**.
Note 13. Operation of the device is not guaranteed if an FACI reset transfer error has occurred. When clearing the error signal output by the ERROROUT pin after release from the reset state, confirm that an FACI reset transfer error has not occurred by checking that an error state is not being indicated again.

Error sources are merged as described in **Table 28.9**.

Table 28.9 Merging of Error Sources

Error Source No.	Module	Error Source	Note
0	WDTA	WDTA error (WDTA0, WDTA1)	Errors of WDTA0 and WDTA1 are merged. In the initial state, generation of an internal reset is allowed.
6	RAM	Local RAM (CPU1, CPU2): 2-bit ECC error Local RAM (CPU1): Address parity error	2-bit ECC errors (local RAM for CPU1, CPU2) and address parity errors are merged.
7		Local RAM (CPU1, CPU2)): 1-bit ECC error Local RAM (CPU1): Parity bit error	1-bit ECC errors (local RAM for CPU1, CPU2) and parity bit errors are merged.
8		Global RAM: 2-bit ECC error and address parity error	2-bit ECC errors and address parity errors when global RAM is accessed from each master are merged.
9		Global RAM: 1-bit ECC error	1-bit ECC errors when global RAM is accessed from each master are merged.
10		Peripheral RAM (CAN, CSIH, DTS): 2-bit ECC error	2-bit ECC errors of RAM for peripheral circuits are merged.
11		Peripheral RAM (CAN, CSIH, DTS): 1-bit ECC error	1-bit ECC errors of RAM for peripheral circuits are merged.
12	Code flash	Code flash 2-bit ECC error Code flash address parity error	2-bit ECC errors when code flash is accessed from each master are merged.
13		Code flash 1-bit ECC error	1-bit ECC errors when code flash is accessed from each master are merged.
14	Instruction cache	Instruction cache data (CPU1, CPU2): 2-bit ECC error	2-bit ECC errors of PE1, PE2 cache data array are merged.
15		Instruction cache data (CPU1, CPU2): 1-bit ECC error	1-bit ECC errors of PE1, PE2 cache data array are merged.
16		Instruction cache tag (CPU1, CPU2): 2-bit ECC error	2-bit ECC errors of PE1, PE2 cache tag array are merged.
17		Instruction cache tag (CPU1, CPU2): 1-bit ECC error	1-bit ECC errors of PE1, PE2 cache tag array are merged.
20	PE guard (PEG)	PEG error	PEG errors of CPU1 and CPU2 are merged.
21	Global RAM guard (GRG)	GRG error	GRG errors of CPU1, CPU2, and DMA are merged.
22	PBG	PBG error	PBG errors for each peripheral circuit are merged.
25	Data parity	Data parity error	Data parity errors of each access path are merged.
37	ADCC	AD parity error	AD parity errors of ADCC0, 1 are merged.
41	DMA	DTS RAM Data ECC SEC-DED	2-bit and 1-bit ECC errors of DTS RAM are merged.
42	BIST	Error detection using on-chip self-diagnostic (BIST) circuit	Errors of each self-diagnostic circuit are merged.
45	TSG3	TSG3 error (INTTSG3nIER)	TSG30 and TSG31 errors are merged.
46	EMU2	EMU2 error signal	EMU20 interrupt 4 and EMU21 interrupt 4 are merged.
47	RDC2	RDC2 error detection	RDC20 and RDC21 errors are merged

Note: Error sources not listed in **Table 28.9** are not merged.

28.3 Registers

28.3.1 List of Registers

ECM registers are assigned to three address areas: the address area for ECM common registers, the address area for ECM master registers, and the address area for ECM checker registers. The address area for ECM common registers is used for both master and checker registers. Writing to this common area is performed by master and checker registers simultaneously. Reading from the common area reads the master register's value. ECM master and checker registers can be written separately.

The ECM master registers are listed in the following table.

See **Section 28.1.2, Register Base Address** for <ECMM_base>, <ECMC_base>, and <ECM_base>.

Table 28.10 List of Registers (ECM Master)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECM	ECM master error set trigger register	ECMMESET	Protected	<ECMM_base> + 00_H
ECM	ECM master error clear trigger register	ECMMECLR	Protected	<ECMM_base> + 04_H
ECM	ECM master error source status register 0	ECMMESSTR0	Not protected	<ECMM_base> + 08_H
ECM	ECM master error source status register 1	ECMMESSTR1	Not protected	<ECMM_base> + $0C_H$
ECM	ECM master protection command register	ECMMPCMD0	Not protected	<ECMM_base> + 10_H

The ECM checker registers are listed in the following table.

Table 28.11 List of Registers (ECM Checker)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECM	ECM checker error set trigger register	ECMCESET	Protected	<ECMC_base> + 00_H
ECM	ECM checker error clear trigger register	ECMCECLR	Protected	<ECMC_base> + 04_H
ECM	ECM checker error source status register 0	ECMCESSTR0	Not protected	<ECMC_base> + 08_H
ECM	ECM checker error source status register 1	ECMCESSTR1	Not protected	<ECMC_base> + $0C_H$
ECM	ECM checker protection command register	ECMCPCMD0	Not protected	<ECMC_base> + 10_H

The ECM common part registers are listed in the following table.

Table 28.12 List of Registers (ECM Common Part)

Module Name	Register Name	Symbol	Protection by Sequence	Address
ECM	ECM error pulse configuration register	ECMEPCFG	Protected	<ECM_base> + 00_H
ECM	ECM maskable interrupt configuration register 0	ECMMICFG0	Protected	<ECM_base> + 04_H
ECM	ECM maskable interrupt configuration register 1	ECMMICFG1	Protected	<ECM_base> + 08_H
ECM	ECM FE level interrupt configuration register 0	ECMNMICFG0	Protected	<ECM_base> + $0C_H$
ECM	ECM FE level interrupt configuration register 1	ECMNMICFG1	Protected	<ECM_base> + 10_H
ECM	ECM internal reset configuration register 0	ECMIRCFG0	Protected	<ECM_base> + 14_H
ECM	ECM internal reset configuration register 1	ECMIRCFG1	Protected	<ECM_base> + 18_H
ECM	ECM error mask register 0	ECMEMK0	Protected	<ECM_base> + $1C_H$
ECM	ECM error mask register 1	ECMEMK1	Protected	<ECM_base> + 20_H
ECM	ECM error source status clear trigger register 0	ECMESSTC0	Protected	<ECM_base> + 24_H
ECM	ECM error source status clear trigger register 1	ECMESSTC1	Protected	<ECM_base> + 28_H
ECM	ECM protection command register	ECMPCMD1	Not protected	<ECM_base> + $2C_H$
ECM	ECM protection status register	ECMPS	Not protected	<ECM_base> + 30_H
ECM	ECM pseudo error trigger register 0	ECMPE0	Protected	<ECM_base> + 34_H
ECM	ECM pseudo error trigger register 1	ECMPE1	Protected	<ECM_base> + 38_H
ECM	ECM delay timer control register	ECMDTMCTL	Protected	<ECM_base> + $3C_H$
ECM	ECM delay timer register	ECMDTMR	Not protected	<ECM_base> + 40_H
ECM	ECM delay timer compare register	ECMDTMCMP	Protected	<ECM_base> + 44_H
ECM	ECM delay timer configuration register 0	ECMDTMCFG0	Protected	<ECM_base> + 48_H
ECM	ECM delay timer configuration register 1	ECMDTMCFG1	Protected	<ECM_base> + $4C_H$
ECM	ECM delay timer configuration register 2	ECMDTMCFG2	Protected	<ECM_base> + 50_H
ECM	ECM delay timer configuration register 3	ECMDTMCFG3	Protected	<ECM_base> + 54_H

28.3.2 ECMmESET (m = M/C) — ECM Master/Checker Error Set Trigger Register

The ECM master/checker error set trigger register is for selecting output of the error signal from the ERROROUT pin. When the ECMmEST bit is set to 1, the ERROROUT pin immediately outputs the error signal. The output cannot be masked. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base>
<ECMC_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmEST
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 28.13 ECMmESET Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmEST	Error Set Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to error output.

CAUTIONS

Setting the ERROROUT output from the ERROROUT pin via the ECMmESET register will set the ECMmSSE026 bit of the ECMmESSTR0 register (ECM compare error). Therefore, the ECMmESET register has to be set following the sequence below.

1. Set the ECMEMK026 bit of the ECMEMK0 register to “masked”.
2. Prevent the generation of interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “prohibited” and the ECMNMIE026 bit of the ECMNMICFG0 register to “prohibited”.
3. Prevent generation of an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “prohibited”.
4. Set the ERROROUT output with the ECMmESET register.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accord with the condition of usage for the ECM compare error.
   - To allow output of the ERROROUT signal from the ERROROUT pin, set the ECMEMK026 bit of the ECMEMK0 register to “not masked”.
   - To enable generation of interrupts, set the ECMMIE026 bit of the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit of the ECMNMICFG0 register to “enabled”.
   - To enable generation of an internal reset, set the ECMIRE026 bit of the ECMIRCFG0 register to “enabled”.

28.3.3 ECMmECLR (m = M/C) — ECM Master/Checker Error Clear Trigger Register

The ECM master/checker error clear trigger register is for setting the error signal from the ERROROUT pin to normal output. When the ECMmECT bit is set to 1, the ERROROUT pin immediately outputs a signal to indicate normal operation as long as no sources lead to setting of the ERROROUT pin to indicate an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base> + 04_H
<ECMC_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmECT
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 28.14 ECMmECLR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmECT	Error Clear Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to normal output.

CAUTIONS

Clearing of the ERROROUT output is only possible if all errors, not masked by ECMEMK0/1, are cleared beforehand.

Clearing the ERROROUT output via the ECMmECLR register will set the ECMmSSE026 bit of the ECMmESSTR0 register (ECM compare error). Therefore, the ECMmECLR register has to be set following the sequence below.

1. Set the ECMEMK026 (ECM compare error) bit of the ECMEMK0 register to “masked”.
2. Disable interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “disabled” and the ECMNMIE026 bit of the ECMNMICFG0 register to “disabled”.
3. Disable an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “disabled”.
4. Use the ECMmECLR register to clear the ERROROUT output.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accord with the condition of usage for the ECM compare error.
   - To allow output of the ERROROUT signal from the ERROROUT pin, set the ECMEMK026 bit of the ECMEMK0 register to “not masked”.
   - To enable generation of interrupts, set the ECMMIE026 bit of the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit of the ECMNMICFG0 register to “enabled”.
   - To enable generation of an internal reset, set the ECMIRE026 bit of the ECMIRCFG0 register to “enabled”.

28.3.4 ECMmESSTR0 (m = M/C) — ECM Master/Checker Error Source Status Register 0

The ECM master/checker error source status register 0 indicates the state of individual internal error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by an external reset or by writing 1 to the corresponding bit of the ECM error source status clear trigger register 0. An internal reset will not affect this register.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + 08_H
<ECMC_base> + 08_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMm SSE029	ECMm SSE028	ECMm SSE027	ECMm SSE026	ECMm SSE025	—	—	ECMm SSE022	ECMm SSE021	ECMm SSE020	ECMm SSE019	ECMm SSE018	ECMm SSE017	ECMm SSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMm SSE015	ECMm SSE014	ECMm SSE013	ECMm SSE012	ECMm SSE011	ECMm SSE010	ECMm SSE009	ECMm SSE008	ECMm SSE007	ECMm SSE006	ECMm SSE005	ECMm SSE004	—	—	—	ECMm SSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.15 ECMmESSTR0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value returned is undefined.
29 to 25	ECMmSSE029 to ECMmSSE025	Error Source Status ECMmSSE029 to ECMmSSE025 correspond to error sources 29 to 25. 0: Error not occurred 1: Error occurred
24, 23	—	Reserved. When read, the value returned is undefined.
22 to 4	ECMmSSE022 to ECMmSSE004	Error Source Status ECMmSSE022 to ECMmSSE004 correspond to error sources 22 to 4. 0: Error not occurred 1: Error occurred
3 to 1	—	Reserved. When read, the value returned is undefined.
0	ECMmSSE000	Error Source Status ECMmSSE000 corresponds to error source 0. 0: Error not occurred 1: Error occurred

28.3.5 ECMmESSTR1 (m = M/C) — ECM Master/Checker Error Source Status Register 1

The ECM master/checker error source status register 1 indicates the state of individual internal error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by an external reset or by writing 1 to the corresponding bit of the ECM error source status clear trigger register 1.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + $0C_H$
<ECMC_base> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMmSSE131	ECMmSSE130	ECMmSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMmSSE115	ECMmSSE114	ECMmSSE113	ECMmSSE112	ECMmSSE111	ECMmSSE110	ECMmSSE109	—	ECMmSSE107	ECMmSSE106	ECMmSSE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.16 ECMmESSTR1 Register Contents

Bit Position	Bit Name	Function
31	ECMmSSE131	Indicates the ERROROUT output loopback status. 0: ERROROUT output is error output. 1: ERROROUT output is normal output.
30	ECMmSSE130	Indicates the ECMmESET write status. 0: No error 1: Error is set by the ECMmEST bit of the ECMmESET register
29	ECMmSSE129	Indicates whether delay timer overflow has occurred. 0: Delay timer overflow not occurred 1: Delay timer overflow occurred
28 to 16	—	Reserved. When read, the value returned is undefined.
15 to 9	ECMmSSE115 to ECMmSSE109	Error Source Status ECMmSSE115 to ECMmSSE109 correspond to error sources 47 to 41. 0: Error not occurred 1: Error occurred
8	—	Reserved. When read, the value returned is undefined.
7 to 5	ECMmSSE107 to ECMmSSE105	Error Source Status ECMmSSE107 to ECMmSSE105 correspond to error sources 39 to 37. 0: Error not occurred 1: Error occurred
4 to 0	—	Reserved. When read, the value returned is undefined.

28.3.6 ECMmPCMD0 (m = M/C) — ECM Master/Checker Protection Command Register

The ECM master/checker protection command register protects the write-protected registers against illegal writing due to incorrect operation of the program, etc.

Refer to **Section 28.3.1, List of Registers**, for the registers to be protected by the ECM master/checker protection command register.

Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECMM_base> + 10_H
<ECMC_base> + 10_H

Value after reset: Undefined

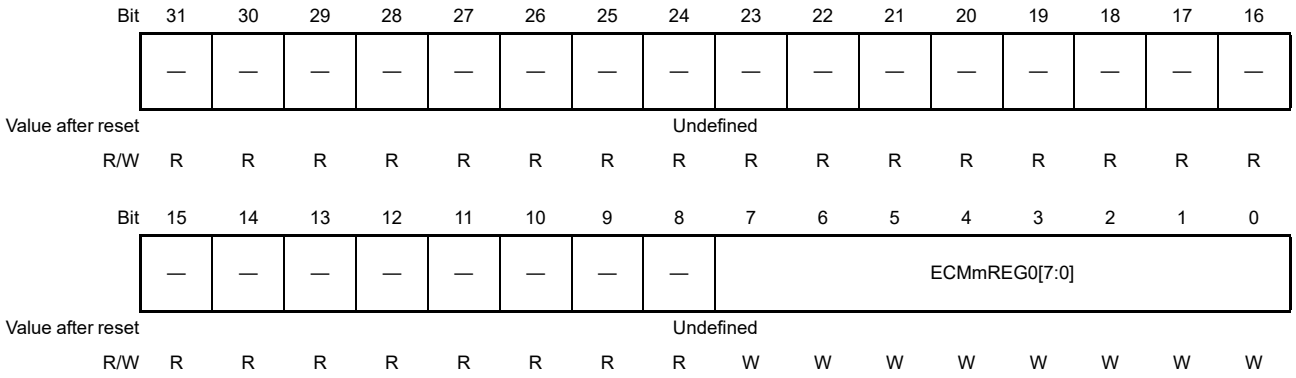


Table 28.17 ECMmPCMD0 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved. When writing to these bits, write 0.
7 to 0	ECMmREG0[7:0]	Protection command that enables writing to write protected ECMm registers.

28.3.7 ECMEPCFG — ECM Error Pulse Configuration Register

The ECM error pulse configuration register sets the type of ERROROUT signal that is output on the ERROROUT pin. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 8-bit units.

Address: <ECM_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMSL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 28.18 ECMEPCFG Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMSL0	ERROROUT Pin Operation Configuration Setting for the type of ERROROUT signal that is output on the ERROROUT pin 0: Non-dynamic mode 1: Dynamic mode

28.3.8 ECMMICFG0 — ECM Maskable Interrupt Configuration Register 0

The ECM maskable interrupt configuration register 0 is used to set the generation of ECM maskable interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 04_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMMIE029	ECMMIE028	ECMMIE027	ECMMIE026	ECMMIE025	—	—	ECMMIE022	ECMMIE021	ECMMIE020	ECMMIE019	ECMMIE018	ECMMIE017	ECMMIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMMIE015	ECMMIE014	ECMMIE013	ECMMIE012	ECMMIE011	ECMMIE010	ECMMIE009	ECMMIE008	ECMMIE007	ECMMIE006	ECMMIE005	ECMMIE004	—	—	—	ECMMIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.19 ECMMICFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMMIE029 to ECMMIE025	ECM Maskable Interrupt Generation Control ECMMIE029 to ECMMIE025 correspond to error sources 29 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMMIE022 to ECMMIE004	ECM Maskable Interrupt Generation Control ECMMIE022 to ECMMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMMIE000	ECM Maskable Interrupt Generation Control ECMMIE000 corresponds to error source 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

28.3.9 ECMMICFG1 — ECM Maskable Interrupt Configuration Register 1

The ECM maskable interrupt configuration register 1 is used to set the generation of ECM maskable interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 08_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMMIE115	ECMMIE114	ECMMIE113	ECMMIE112	ECMMIE111	ECMMIE110	ECMMIE109	—	—	ECMMIE106	ECMMIE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 28.20 ECMMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMMIE115 to ECMMIE109	ECM Maskable Interrupt Generation Control ECMMIE115 to ECMMIE109 correspond to error sources 47 to 41. 0: Interrupt generation disabled 1: Interrupt generation enabled
8, 7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMMIE106, ECMMIE105	ECM Maskable Interrupt Generation Control ECMMIE106 and ECMMIE105 correspond to error sources 38 and 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.10 ECMNMICFG0 — ECM FE Level Interrupt Configuration Register 0

The ECM FE level interrupt configuration register 0 is used to set the generation of FE level interrupt. The generation of FE level interrupt in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMN MIE029	ECMN MIE028	ECMN MIE027	ECMN MIE026	ECMN MIE025	—	—	ECMN MIE022	ECMN MIE021	ECMN MIE020	ECMN MIE019	ECMN MIE018	ECMN MIE017	ECMN MIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMN MIE015	ECMN MIE014	ECMN MIE013	ECMN MIE012	ECMN MIE011	ECMN MIE010	ECMN MIE009	ECMN MIE008	ECMN MIE007	ECMN MIE006	ECMN MIE005	ECMN MIE004	—	—	—	ECMN MIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.21 ECMNMICFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMNMIE029 to ECMNMIE025	ECM FE Level Interrupt Generation Control ECMNMIE029 to ECMNMIE025 correspond to error sources 29 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMNMIE022 to ECMNMIE004	ECM FE Level Interrupt Generation Control ECMNMIE022 to ECMNMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMNMIE000	ECM FE Level Interrupt Generation Control ECMNMIE000 corresponds to error source 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

28.3.11 ECMNMICFG1 — ECM FE Level Interrupt Configuration Register 1

The ECM FE level interrupt configuration register 1 is used to set the generation of FE level interrupt. The generation of FE level interrupt in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 10_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMNMIE115	ECMNMIE114	ECMNMIE113	ECMNMIE112	ECMNMIE111	ECMNMIE110	ECMNMIE109	—	—	ECMNMIE106	ECMNMIE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.22 ECMNMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMNMIE115 to ECMNMIE109	ECM FE Level Interrupt Generation Control ECMNMIE115 to ECMNMIE109 correspond to error sources 47 to 41. 0: Interrupt generation disabled 1: Interrupt generation enabled
8, 7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMNMIE106, ECMNMIE105	ECM FE Level Interrupt Generation Control ECMNMIE106 and ECMNMIE105 correspond to error sources 38 and 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.12 ECMIRCFG0 — ECM Internal Reset Configuration Register 0

The ECM internal reset configuration register 0 is used to set the generation of an internal reset in response to an internal error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 14_H

Value after reset: 0000 $000F_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMIRE029	ECMIRE028	ECMIRE027	ECMIRE026	ECMIRE025	—	—	ECMIRE022	ECMIRE021	ECMIRE020	ECMIRE019	ECMIRE018	ECMIRE017	ECMIRE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMIRE015	ECMIRE014	ECMIRE013	ECMIRE012	ECMIRE011	ECMIRE010	ECMIRE009	ECMIRE008	ECMIRE007	ECMIRE006	ECMIRE005	ECMIRE004	—	—	—	ECMIRE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.23 ECMIRCFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMIRE029 to ECMIRE025	ECM Internal Reset Generation Control ECMIRE029 to ECMIRE025 correspond to error sources 29 to 25. 0: Internal reset generation disabled 1: Internal reset generation enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMIRE022 to ECMIRE004	ECM Internal Reset Generation Control ECMIRE022 to ECMIRE004 correspond to error sources 22 to 4. 0: Internal reset generation disabled 1: Internal reset generation enabled
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMIRE000	ECM Internal Reset Generation Control ECMIRE000 corresponds to error source 0. 0: Internal reset generation disabled 1: Internal reset generation enabled

28.3.13 ECMIRCFG1 — ECM Internal Reset Configuration Register 1

The ECM internal reset configuration register 1 is used to set the generation of an internal reset in response to an internal error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 18_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMIRE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMIRE115	ECMIRE114	ECMIRE113	ECMIRE112	ECMIRE111	ECMIRE110	ECMIRE109	—	—	ECMIRE106	ECMIRE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 28.24 ECMIRCFG1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMIRE129	ECM Internal Reset Generation Control Corresponds to delay timer overflow. 0: Internal reset generation disabled 1: Internal reset generation enabled
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMIRE115 to ECMIRE109	ECM Internal Reset Generation Control ECMIRE115 to ECMIRE109 correspond to error sources 47 to 41. 0: Internal reset generation disabled 1: Internal reset generation enabled
8, 7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMIRE106, ECMIRE105	ECM Internal Reset Generation Control ECMIRE106 and ECMIRE105 correspond to error sources 38 and 37. 0: Internal reset generation disabled 1: Internal reset generation enabled
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.14 ECMEMK0 — ECM Error Mask Register 0

The ECM error mask register 0 is used to mask the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $1C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMEMK029	ECMEMK028	ECMEMK027	ECMEMK026	ECMEMK025	—	—	ECMEMK022	ECMEMK021	ECMEMK020	ECMEMK019	ECMEMK018	ECMEMK017	ECMEMK016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMEMK015	ECMEMK014	ECMEMK013	ECMEMK012	ECMEMK011	ECMEMK010	ECMEMK009	ECMEMK008	ECMEMK007	ECMEMK006	ECMEMK005	ECMEMK004	—	—	—	ECMEMK000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.25 ECMEMK0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
29 to 25	ECMEMK029 to ECMEMK025	ECM ERROROUT Output Mask Control ECMEMK029 to ECMEMK025 correspond to error sources 29 to 25. 0: ERROROUT output not masked 1: ERROROUT output masked
24, 23	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
22 to 4	ECMEMK022 to ECMEMK004	ECM ERROROUT Output Signal Mask Control ECMEMK022 to ECMEMK004 correspond to error sources 22 to 4. 0: ERROROUT output not masked 1: ERROROUT output masked
3 to 1	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
0	ECMEMK000	ECM ERROROUT Output Signal Mask Control ECMEMK000 corresponds to error source 0. 0: ERROROUT output not masked 1: ERROROUT output masked

28.3.15 ECMEMK1 — ECM Error Mask Register 1

The ECM error mask register 1 is used to mask the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 20_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMEMK129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMEMK115	ECMEMK114	ECMEMK113	ECMEMK112	ECMEMK111	ECMEMK110	ECMEMK109	—	ECMEMK107	ECMEMK106	ECMEMK105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R

Table 28.26 ECMEMK1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
29	ECMEMK129	Corresponds to delay timer overflow. 0: ERROROUT output not masked 1: ERROROUT output masked
28 to 16	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.
15 to 9	ECMEMK115 to ECMEMK109	ECM ERROROUT Output Mask Control ECMEMK115 to ECMEMK109 correspond to error sources 47 to 41. 0: ERROROUT output not masked 1: ERROROUT output masked
8	—	Reserved. When read, the value returned is undefined. When writing to this bit, follow the procedure of the protection unlock sequence to write 1.
7 to 5	ECMEMK107 to ECMEMK105	ECM ERROROUT Output Mask Control ECMEMK107 to ECMEMK105 correspond to error sources 39 to 37. 0: ERROROUT output not masked 1: ERROROUT output masked
4 to 0	—	Reserved. When read, the value returned is undefined. When writing to these bits, follow the procedure of the protection unlock sequence to write 1.

28.3.16 ECMESSTC0 — ECM Error Source Status Clear Trigger Register 0

The ECM error source status clear trigger register 0 is used to clear the states of the individual error sources of the ECM master/checker error source status register 0. Both the error states of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 24_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMCLSSE029	ECMCLSSE028	ECMCLSSE027	ECMCLSSE026	ECMCLSSE025	—	—	ECMCLSSE022	ECMCLSSE021	ECMCLSSE020	ECMCLSSE019	ECMCLSSE018	ECMCLSSE017	ECMCLSSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	W	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMCLSSE015	ECMCLSSE014	ECMCLSSE013	ECMCLSSE012	ECMCLSSE011	ECMCLSSE010	ECMCLSSE009	ECMCLSSE008	ECMCLSSE007	ECMCLSSE006	ECMCLSSE005	ECMCLSSE004	—	—	—	ECMCLSSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	W

Table 28.27 ECMESSTC0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMCLSSE029 to ECMCLSSE025	ECM Error Status Clear ECMCLSSE029 to ECMCLSSE025 correspond to error sources 29 to 25. 0: Corresponding error status unchanged 1: Corresponding error status cleared
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMCLSSE022 to ECMCLSSE004	ECM Error Status Clear ECMCLSSE022 to ECMCLSSE004 correspond to error sources 22 to 4. 0: Corresponding error status unchanged 1: Corresponding error status cleared
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMCLSSE000	ECM Error Status Clear ECMCLSSE000 corresponds to error source 0. 0: Corresponding error status unchanged 1: Corresponding error status cleared

28.3.17 ECMESSTC1 — ECM Error Source Status Clear Trigger Register 1

The ECM error source status clear trigger register 1 is used to clear the states of the individual error sources of the ECM master/checker error source status register 1. Both the error states of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 28_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	ECMCLSSE130	ECMCLSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMCLSSE115	ECMCLSSE114	ECMCLSSE113	ECMCLSSE112	ECMCLSSE111	ECMCLSSE110	ECMCLSSE109	—	ECMCLSSE107	ECMCLSSE106	ECMCLSSE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	R	W	W	W	R	R	R	R	R

Table 28.28 ECMESSTC1 Register Contents (1/2)

Bit Position	Bit Name	Function
31	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
30, 29	ECMCLSSE130, ECMCLSSE129	ECM Error Status Clear ECMCLSSE130 and ECMCLSSE129 correspond to the write status of the ECMmESET register and delay timer overflow. 0: Corresponding error status unchanged 1: Corresponding error status cleared
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMCLSSE115 to ECMCLSSE109	ECM Error Status Clear ECMCLSSE115 to ECMCLSSE109 correspond to error sources 47 to 41. 0: Corresponding error status unchanged 1: Corresponding error status cleared
8	—	Reserved. When read, the value after reset is returned. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 5	ECMCLSSE107 to ECMCLSSE105	ECM Error Status Clear ECMCLSSE107 to ECMCLSSE105 correspond to error sources 39 to 37. 0: Corresponding error status unchanged 1: Corresponding error status cleared

Table 28.28 ECMESSTC1 Register Contents (2/2)

Bit Position	Bit Name	Function
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.18 ECMPCMD1 — ECM Protection Command Register

Refer to **Section 28.3.1, List of Registers**, for the registers to be protected by the ECM protection command register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + $2C_H$

Value after reset: Undefined

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECMREG1[7:0]							
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 28.29 ECMPCMD1 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved. When writing to these bits, write 0.
7 to 0	ECMREG1[7:0]	Protection command that enables writing to write protected ECM registers.

28.3.19 ECMPS — ECM Protection Status Register

The ECM protection status register indicates whether the write protected register has been written successfully or not. For details, refer to **Section 28.4.5, Write Protected Registers**.

Access: This register can be read in 8-bit units.

Address: <ECM_base> + 30_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMPRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 28.30 ECMPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When read, the value after reset is returned.
0	ECMPRERR	ECM Protection Status Indicates whether the write protected register has been written successfully or not. 0: Writing was successfully completed. 1: Writing failed

28.3.20 ECMPE0 — ECM Pseudo Error Trigger Register 0

The ECM pseudo error trigger register 0 is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 34_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMPE029	ECMPE028	ECMPE027	ECMPE026	ECMPE025	—	—	ECMPE022	ECMPE021	ECMPE020	ECMPE019	ECMPE018	ECMPE017	ECMPE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	W	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMPE015	ECMPE014	ECMPE013	ECMPE012	ECMPE011	ECMPE010	ECMPE009	ECMPE008	ECMPE007	ECMPE006	ECMPE005	ECMPE004	—	—	—	ECMPE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	W

Table 28.31 ECMPE0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMPE029 to ECMPE025	ECM Pseudo Error Trigger ECMPE029 to ECMPE025 correspond to error sources 29 to 25. 0: Pseudo error not generated 1: Generates corresponding pseudo error
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMPE022 to ECMPE004	ECM Pseudo Error Trigger ECMPE022 to ECMPE004 correspond to error sources 22 to 4. 0: Pseudo error not generated 1: Generates corresponding pseudo error generated
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMPE000	ECM Pseudo Error Trigger ECMPE000 corresponds to error source 0. 0: Pseudo error not generated 1: Generates corresponding pseudo error generated

28.3.21 ECMPE1 — ECM Pseudo Error Trigger Register 1

The ECM pseudo error trigger register 1 is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 38_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMPE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMPE115	ECMPE114	ECMPE113	ECMPE112	ECMPE111	ECMPE110	ECMPE109	—	ECMPE107	ECMPE106	ECMPE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	R	W	W	W	R	R	R	R	R

Table 28.32 ECMPE1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMPE129	ECM Pseudo Error Trigger Corresponds to delay timer overflow. 0: Pseudo error not generated 1: Generates corresponding pseudo error
28 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMPE115 to ECMPE109	ECM Pseudo Error Trigger ECMPE115 to ECMPE109 correspond to error sources 47 to 41. 0: Pseudo error not generated 1: Generates corresponding pseudo error
8	—	Reserved. When read, the value after reset is returned. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 5	ECMPE107 to ECMPE105	ECM Pseudo Error Trigger ECMPE107 to ECMPE105 correspond to error sources 39 to 37. 0: Pseudo error not generated 1: Generates corresponding pseudo error
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.22 ECMDTMCTL — ECM Delay Timer Control Register

The ECM delay timer control register is used to control the delay timer. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be written in 8-bit units.

Address: <ECM_base> + $3C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ECMSTP	ECMSTA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 28.33 ECMDTMCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	ECMSTP	Delay Timer Stop Writing 1 to this bit initializes the delay timer counter, causing the delay timer to stop. The ECMSTA bit is set to 0 at the same time.
0	ECMSTA	Delay Timer Start Specifies the operation of the delay timer when an interrupt is occurred. Writing 1 to this bit initializes the delay timer counter, causing the timer to start operation. Writing 0 to this bit initializes the delay timer counter, causing the timer to stop. 0: Timer stops 1: Timer starts

28.3.23 ECMDTMR — ECM Delay Timer Register

The ECM delay timer register indicates the value of the delay timer counter. Writing 1 to the ECMSTP bit of the ECM delay timer control register (ECMDTMCTL) or writing 0 to the ECMSTA bit initializes the delay timer counter.

Access: This register can be read in 16-bit units.

Address: <ECM_base> + 40_H

Value after reset: 0000_H

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

28.3.24 ECMDTMCMP — ECM Delay Timer Compare Register

When the value of the ECM delay timer compare register matches with the value of the delay timer counter, a delay timer overflow signal is generated to set the ECMmSSE129n bit. Writing data to this register has to be conducted while the delay timer is stopped. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 16-bit units.

Address: <ECM_base> + 44_H

Value after reset: 0000_H

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

28.3.25 ECMDTMCFG0 — ECM Delay Timer Configuration Register 0

The ECM delay timer configuration register 0 is used to enable or disable the delay timer to be started by a maskable interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 48_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMTE029	ECMTE028	ECMTE027	ECMTE026	ECMTE025	—	—	ECMTE022	ECMTE021	ECMTE020	ECMTE019	ECMTE018	ECMTE017	ECMTE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE015	ECMTE014	ECMTE013	ECMTE012	ECMTE011	ECMTE010	ECMTE009	ECMTE008	ECMTE007	ECMTE006	ECMTE005	ECMTE004	—	—	—	ECMTE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.34 ECMDTMCFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMTE029 to ECMTE025	ECM Delay Timer Start Control ECMTE029 to ECMTE025 correspond to maskable interrupts generated by error sources 29 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE022 to ECMTE004	ECM Delay Timer Start Control ECMTE022 to ECMTE004 correspond to maskable interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMTE000	ECM Delay Timer Start Control ECMTE000 corresponds to the maskable interrupt generated by error source 0. 0: Delay timer start disabled 1: Delay timer start enabled

28.3.26 ECMDTMCFG1 — ECM Delay Timer Configuration Register 1

The ECM delay timer configuration register 1 is used to enable or disable the delay timer to be started by a maskable interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $4C_H$

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE115	ECMTE114	ECMTE113	ECMTE112	ECMTE111	ECMTE110	ECMTE109	—	—	ECMTE106	ECMTE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 28.35 ECMDTMCFG1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMTE115 to ECMTE109	ECM Delay Timer Start Control ECMTE115 to ECMTE109 correspond to maskable interrupts generated by error sources 47 to 41. 0: Delay timer start disabled 1: Delay timer start enabled
8, 7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE106, ECMTE105	ECM Delay Timer Start Control ECMTE106 and ECMTE105 correspond to maskable interrupts generated by error sources 38 and 37. 0: Delay timer start disabled 1: Delay timer start enabled
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.3.27 ECMDTMCFG2 — ECM Delay Timer Configuration Register 2

The ECM delay timer configuration register 2 is used to enable or disable the delay timer to be started by an FE level interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 50_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMTE229	ECMTE228	ECMTE227	ECMTE226	ECMTE225	—	—	ECMTE222	ECMTE221	ECMTE220	ECMTE219	ECMTE218	ECMTE217	ECMTE216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE215	ECMTE214	ECMTE213	ECMTE212	ECMTE211	ECMTE210	ECMTE209	ECMTE208	ECMTE207	ECMTE206	ECMTE205	ECMTE204	—	—	—	ECMTE200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.36 ECMDTMCFG2 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29 to 25	ECMTE229 to ECMTE225	ECM Delay Timer Start Control ECMTE229 to ECMTE225 correspond to FE level interrupts generated by error sources 29 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE222 to ECMTE204	ECM Delay Timer Start Control ECMTE222 to ECMTE204 correspond to FE level interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3 to 1	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMTE200	ECM Delay Timer Start Control ECMTE200 corresponds to the FE level interrupt generated by error source 0. 0: Delay timer start disabled 1: Delay timer start enabled

28.3.28 ECMDTMCFG3 — ECM Delay Timer Configuration Register 3

The ECM delay timer configuration register 3 is used to enable or disable the delay timer to be started by an FE level interrupt in response to an error. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 28.4.5, Write Protected Registers**, for details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 54_H

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE315	ECMTE314	ECMTE313	ECMTE312	ECMTE311	ECMTE310	ECMTE309	—	—	ECMTE306	ECMTE305	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 28.37 ECMDTMCFG3 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
15 to 9	ECMTE315 to ECMTE309	ECM Delay Timer Start Control ECMTE315 to ECMTE309 correspond to FE level interrupts generated by error sources 47 to 41. 0: Delay timer start disabled 1: Delay timer start enabled
8, 7	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE306, ECMTE305	ECM Delay Timer Start Control ECMTE306 and ECMTE305 correspond to FE level interrupts generated by error sources 38 and 37. 0: Delay timer start disabled 1: Delay timer start enabled
4 to 0	—	Reserved. When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

28.4 Functions

28.4.1 Operations for ERROROUT Output

During reset and after release from the reset state, the output on the $\overline{\text{ERROROUT\_M}}$ pin indicates an error. Follow the procedure described in **CAUTIONS** in **Section 28.3.3** to clear the error output before using the ECM. Since the $\overline{\text{ERROROUT\_C}}$ pin is a multi-functional pin that can be used also as a general-purpose I/O pin or for other functions, select the $\overline{\text{ERROROUT\_C}}$ function before using this function. Refer to **Section 2, Pins**, for how to configure the port.

The ERROROUT output signal can be configured for two different modes of operation, non-dynamic or dynamic. Furthermore, when the ERROROUT output is selected and the signal is in dynamic mode, output on the pin indicates errors in synchronization with errors as they occur regardless of the pulse cycle.

Table 28.38 ERROROUT Output Operation

Error State ECMmSSE031 to ECMmSSE000 ECMmSSE115 to ECMmSSE100	Operating Mode ECMSL0 Bit	ERROROUT Output Operating Mode	ERROROUT Output Level	Error State
0	0	Non-dynamic	High level	Normal
	1	Dynamic	Toggles*1 (according to timer input*2)	Normal
1	0	Non-dynamic	Low level	Error
	1	Dynamic	Low level	Error

Note 1. The ERROROUTZ signal for PIC1A is not toggled.
Note 2. For details, see **Section 16, OS Timer (OSTM)**.

28.4.1.1 Enabling Dynamic Mode

1. Initialize OSTM0.
2. Set the ERROROUT output to normal output by setting the ECMmECT bit (m = M/C) in the ECM master/checker error clear trigger register to 1.
3. Set the ECMSL0 bit in the ECM error pulse configuration register to 1 to specify dynamic mode.
4. Start up OSTM0.

28.4.1.2 Disabling Dynamic Mode

1. Set the ERROROUT output to error output by setting the ECMmEST bit (m = M/C) in the ECM master/checker error set trigger register to 1.
2. Stop OSTM0.
3. Clear the ECMSL0 bit in the ECM error pulse configuration register to 0 to specify non-dynamic mode.

28.4.2 Loop-Back Function

ECM incorporates a loop-back function that is used to check the path to the ERROROUT pin. The output level of the ERROROUT pin can be checked with the ECMmSSE131 bit (m = M/C) in the ECM master/checker error source status register 1.

28.4.3 Pseudo Error Generation

ECM incorporates a function that can generate pseudo errors for test or debug purposes. The operation of the ECM during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for error masks, interrupt, internal reset, or delay timer apply in the same way.

28.4.4 Error State

The error state is indicated by the ECM master/checker error source status register 0 and the ECM master/checker error source status register 1. The error state is only cleared by an external reset or by writing 1 to the corresponding bit of the ECM error source status clear trigger register. In case of an internal reset, the error state is kept and the error of the reset source can be confirmed by reading the ECM master/checker error source status register 0 and the ECM master/checker error source status register 1 after release from the reset state.

28.4.5 Write Protected Registers

Write protected registers are protected from inadvertent write access due to erroneous program execution, etc.

28.4.5.1 Sequence of Writing to the Write-Protected Registers

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value (0000 00A5$_H$) to the ECM protection command register ECMPCMD1 and ECM master/checker protection command register ECMmPCMD0.
2. Write the new setting to the write-protected registers in the ECM and ECMm. Write the value after a reset*1 to the reserved bits.
3. Write the bitwise inverse of the setting value to the same registers as in step 2. Write the inverse of the value after a reset*1 to the reserved bits.
4. Write the new setting to the same registers as in step 2. Write the value after a reset*1 to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Note 1. For the reserved bits in the ECMEMK0/1 register, write 1 in steps 2 and 4, and 0 in step 3.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the ECMPRERR bit of the ECM protection status register ECMPS is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the ECMPRERR bit of the ECM protection status register ECMPS is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*2, writing to the protected register fails and the protection status bit in the protection status register is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 2. As to ECM, all registers listed in **Section 28.3.1, List of Registers** are treated as “another register in the same module”.

28.4.6 Timeout Function for Interrupt Processing

The ECM incorporates a function that generates an ERROROUT output or internal reset when the count value of the delay timer incorporated in the ECM matches with the value of the ECM delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The timer counting is not stopped when a break occurs.

The counting of the delay timer always starts from 0. Configure the duration until an internal reset or the ERROROUT output is generated with the settings of the ECM delay timer compare register.

If another error by which the delay timer is to be started occurs while the delay timer is running, the current value counted by the delay timer is not reset but the timer continues to run.

28.5 Usage Notes

A clock hazard (in the form of up to 20 ns of skew) output may be produced on the $\overline{\text{ERROROUT\_M}}$ pin within 2 μs after de-assertion of the signal on the external reset pin ($\overline{\text{RESET}}$).

When using the signal on the ERROROUT pin, take the following measures against this.

(1) When both the $\overline{\text{ERROROUT\_M}}$ (master) and $\overline{\text{ERROROUT\_C}}$ (checker) pins are in use

After both the master and checker pins detect no error condition (both pins are in the output direction), start the external monitoring.

NOTE

Since the power-supply monitoring IC (RAA270000KFT) manufactured by Renesas Electronics Corporation is not affected by the hazard described above, this measure is not necessary.

(2) When only the $\overline{\text{ERROROUT\_M}}$ (master) pin is used

To eliminate the hazard, insert a filter with a time constant of at least 50 to 100 ns in the output from the $\overline{\text{ERROROUT\_M}}$ pin.

Section 29 Data CRC (DCRA)

This section contains a generic description of the data CRC function A (DCRA).

The first section describes all properties specific to the RH850/C1x, such as units, register base addresses, input/output signal names, etc.
The remainder of the section describes the functions and registers of the DCRA.

29.1 Features of RH850/C1x DCRA

29.1.1 Number of Units

This microcontroller has the following number of units of the DCRA.

Table 29.1 Number of Units

Product	RH850/C1x
Units	2
Name	DCRAn (n = 0 to 1)

Table 29.2 Index

Index	Meaning
n	Throughout this section, the individual channels of a DCRA is identified by the index “n” (n = 0 to 1), for example, DCRAnCTL for the DCRAn control register.

29.1.2 Register Base Addresses

DCRA base addresses are listed in the following table.

DCRA register addresses are given as offsets from the individual base address.

Table 29.3 Register Base Addresses

Base Address Name	Base Address
<DCRA0_base>	FFF7 0000_H
<DCRA1_base>	FFF7 1000_H

29.1.3 Clock Supply

The DCRA clock supply is listed in the following table.

Table 29.4 DCRAn Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
DCRAn	PCLK	CLKC_LSB (low-speed peripheral clock)

29.1.4 Reset Source

The DCRA reset sources are shown below. DCRA is initialized by the reset sources below.

Table 29.5 Reset Source

Unit Name	Reset Source
DCRAn	Reset by any reset source

29.2 Overview

29.2.1 Functional Overview

The data CRC function A can be used to verify or generate CRC protected data streams of arbitrary length and different bit widths.

- 32-bit Ethernet CRC
  $(X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X^1 + 1)$
- 16-bit CCITT CRC
  $(X^{16} + X^{12} + X^5 + 1)$
- CRC of an arbitrary data block length can be generated.
- After initialization of the CRC data register, every write access to the CRC input register generates a new CRC according to the selected polynomial, and the result is stored in the CRC data register.

29.2.2 Block Diagram

The following picture shows the block diagram of the data CRC function A.

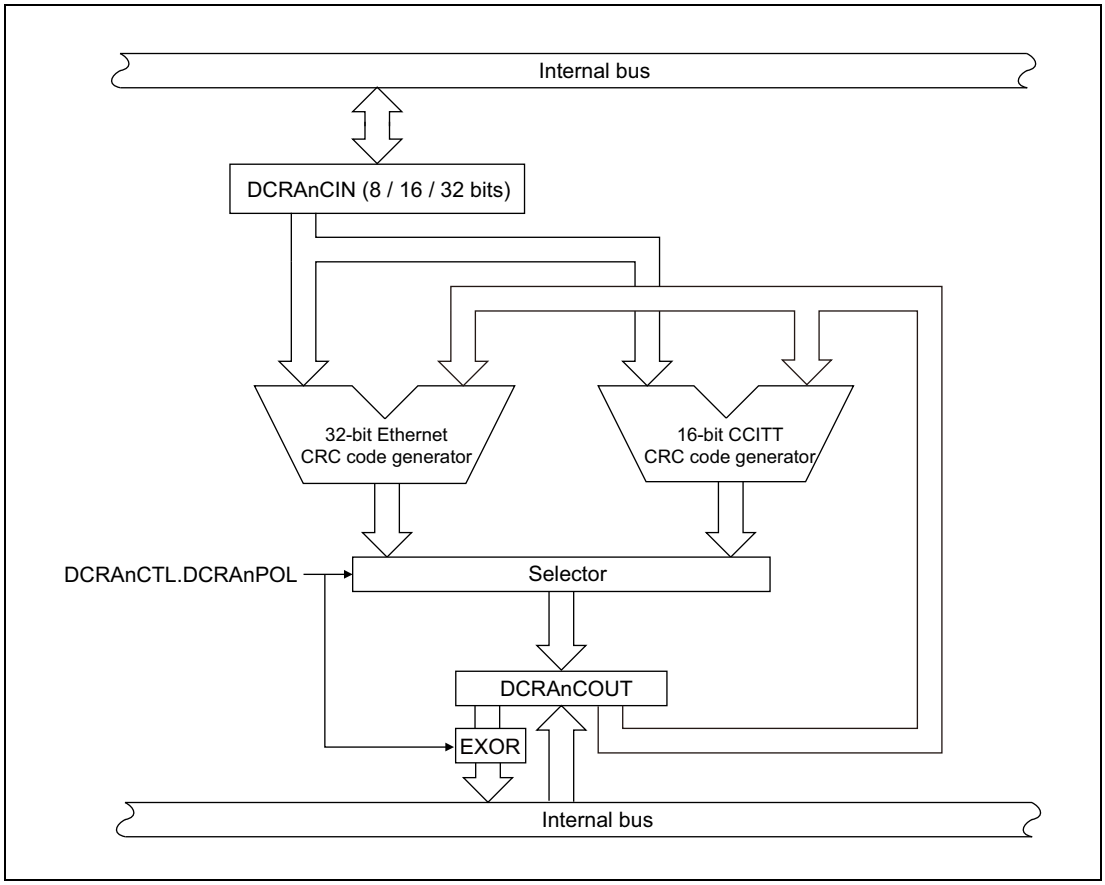
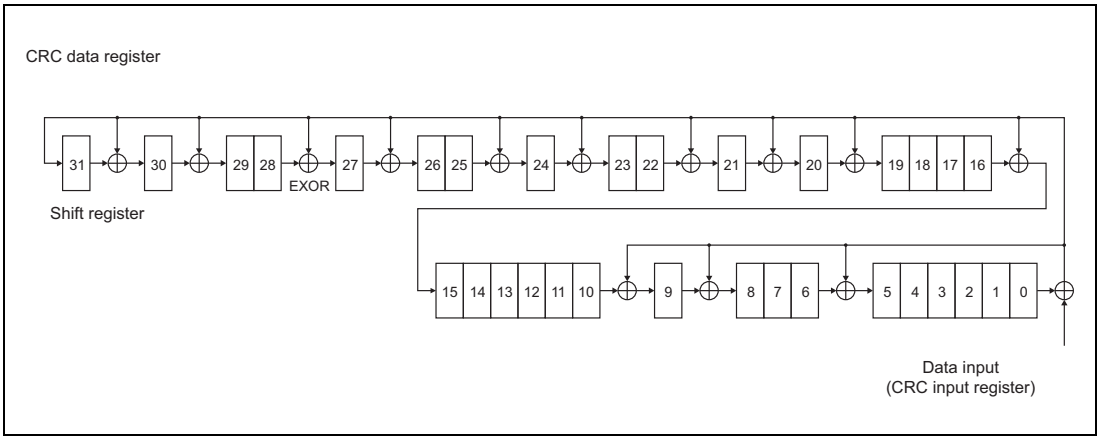


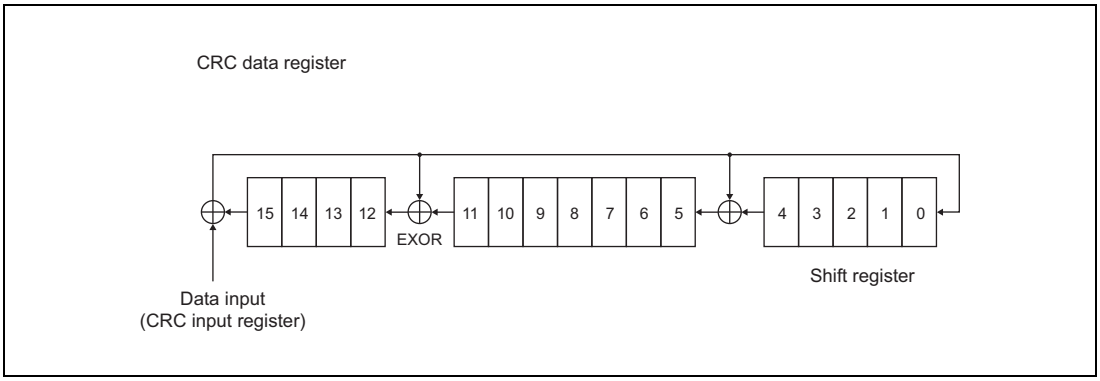
Figure 29.1 Block Diagram of Data CRC Function A

29.2.3 Operation Circuit

- 32-bit Ethernet



- 16-bit CCITT



29.3 Registers

29.3.1 List of Registers

The DCRA registers are listed in the following table.

See **Section 29.1.2, Register Base Addresses** for <DCRAn_base>.

Table 29.6 List of Registers

Module Name	Register Name	Symbol	Address
DCRAn	CRC input register	DCRAnCIN	<DCRAn_base> + 00_H
DCRAn	CRC data register	DCRAnCOUT	<DCRAn_base> + 04_H
DCRAn	CRC control register	DCRAnCTL	<DCRAn_base> + 20_H

29.3.2 DCRAnCIN — CRC Input Register

This register holds the input data for the CRC calculation. The effective bit width used for CRC calculation must be set by DCRAnCTL.DCRAnISZ[1:0].

When data is written to this register, the CRC code is generated.

The CRC calculation is immediately started after the DCRAnCIN register is written. The DCRAnCOUT register must be initialized, with the initial start value, before the first data of the data block is written to DCRAnCIN register.

Access: This register can be read/written in 32-bit units.

Address: <DCRAn_base>

Value after reset: $0000\ 0000_H$

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

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

Table 29.7 DCRAnCIN Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRAnCIN [31:0]	Input Data for CRC Calculation The valid bits are: • for 32 bits, the effective bit width: DCRAnCIN[31:0] • for 16 bits, the effective bit width: DCRAnCIN[15:0] • for 8 bits, the effective bit width: DCRAnCIN[7:0]

29.3.3 DCRAnCOUT — CRC Data Register

This register stores the result of the CRC code generated by the 32-bit Ethernet or 16-bit CCITT polynomial.

Access: This register can be read/written in 32-bit units.

Address: <DCRAn_base> + 4_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCRAnCOUT[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCRAnCOUT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W*1	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. The read value after reset is 0000 0000_H since the 32-bit Ethernet CRC polynomial is selected as the CRC generating function after reset.

Table 29.8 DCRAnCOUT Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRAnCOUT [31:0]	Result of the CRC Code Generation When the 16-bit CCITT polynomial is enabled, bits 15 to 0 show the CRC result. Bits 31 to 16 are undefined. Read value of this register results in Ex-ORed with the value below. • Using 32-bit Ethernet polynomial: FFFF $FFFF_H$ • Using 16-bit CCITT polynomial: 0000_H For example, AAAA $AAAA_H$ is read when DCRAnCOUT = 5555 5555_H by the 32-bit Ethernet polynomial.

CAUTION

This register must be initialized (the initial start value must be set) before the first data of the data block is written to the DCRAnCIN register.

29.3.4 DCRAnCTL — CRC Control Register

This register controls the CRC generation process.

Access: This register can be read/written in 8-bit units.

Address: <DCRAn_base> + 20_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DCRAnISZ[1:0]		DCRAnPOL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 29.9 DCRAnCTL Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	The write value should be the value after reset.
2, 1	DCRAnISZ[1:0]	Specify the CRC input bit width: 00: 32 bits (DCRAnCIN[31:0]) 01: 16 bits (DCRAnCIN[15:0]) 10: 8 bits (DCRAnCIN[7:0]) 11: Setting prohibited
0	DCRAnPOL	Specifies the CRC generating function: 0: 32-bit Ethernet CRC polynomial generation. The byte order of the DCRAnCIN register is LSB (least significant bit) first. This means that, if the input bit width is 8 bits (DCRAnISZ[1:0] = 10_B), bit positions 7 to 0 of the DCRAnCIN register contain the input data and bit position 0 (LSB) is the start bit of the input data. 1: 16-bit CCITT CRC polynomial generation. The byte order of the DCRAnCIN register is MSB (most significant bit) first. This means that, if the input bit width is 8 bits (DCRAnISZ[1:0] = 10_B), bit positions 7 to 0 of the DCRAnCIN register contain the input data and bit position 7 (MSB) is the start bit of the input data.

CAUTIONS

- After changing the CRC generating function (DCRAnCTL.DCRAnPOL), the DCRAnCOUT register must be initialized (the initial start value must be set).
- The CRC input bit width (DCRAnCTL.DCRAnISZ[1:0]) must be set according to the data block bit width. Switching the CRC bit input width is not allowed during processing of a data block (a data block consists of N bytes, half words or words). After the final CRC result is read from the DCRAnCOUT register, the CRC input bit width can be changed. In this case, the DCRAnCOUT register must be initialized (the initial start value must be set) afterwards.

29.4 Function

The data CRC function A generates a CRC (cyclic redundancy check) of an arbitrary data block length. The data is forwarded to the data CRC function in 8-, 16- or 32-bit units. The CRC polynomial can either be selected for 32-bit Ethernet or 16-bit CCITT,

The initial start value must be set at the DCRAnCOUT register before the first write access to the CRC input register (DCRAnCIN) is performed.

The flowchart below shows the CRC generating procedure.

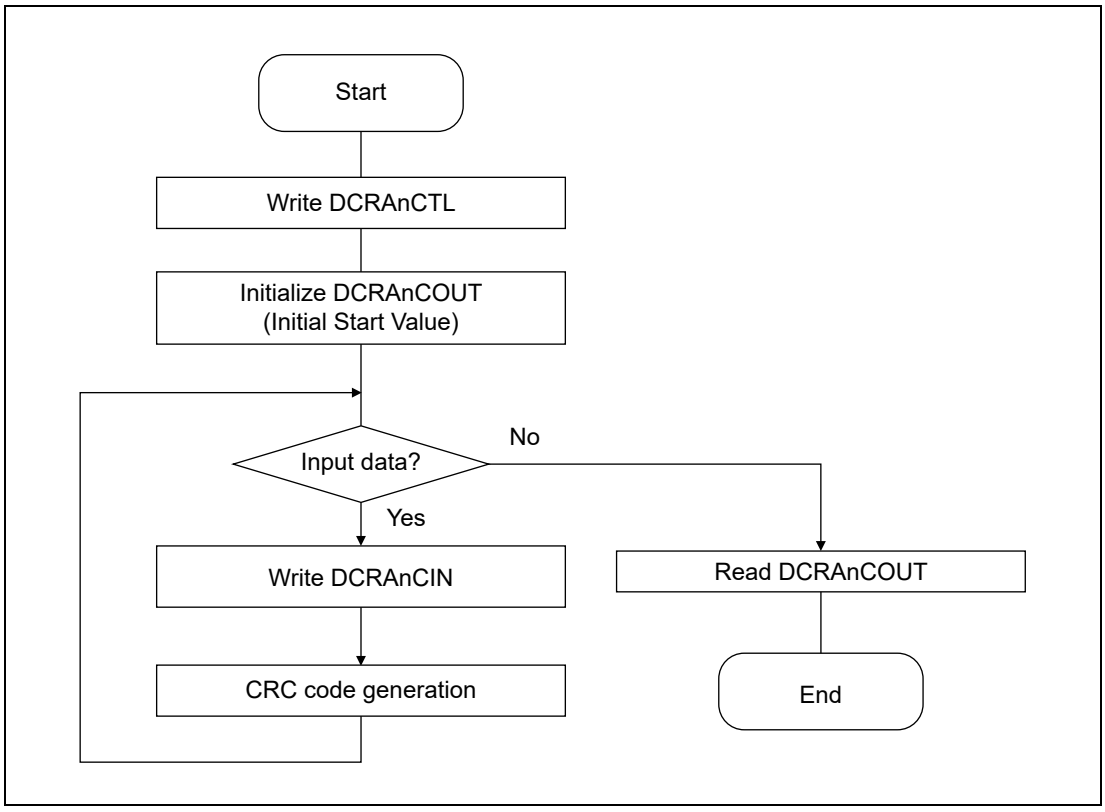


Figure 29.2 Flowchart of Data CRC Function A

NOTES

1. Before writing the first data to DCRAnCIN, the CRC output register DCRAnCOUT must be initialized (the initial start value must be set).
2. DCRAnCOUT must be re-initialized by setting the initial start value when the polynomial is changed by changing DCRAnCTL.DCRAnPOL.
3. Setting example of the initial start values of the respective polynomials
   The following is the example of setting values.

Table 29.10 Setting Example of Initial Start Values (when read at a reset)

	Initial Start Value	EXOR Value	DCRAnCOUT Read Value
32-bit Ethernet	FFFF FFFF$_H$	FFFF FFFF$_H$	0000 0000$_H$
16-bit CCITT	XXXX FFFF$_H$	XXXX 0000$_H$	XXXX FFFF$_H$

Section 30 On-Chip Debugging Unit (OCD)

30.1 Debug Function

This microcontroller has an on-chip debug function. By using the on-chip debug emulator, programs can be debugged with the microcontroller mounted in the target system.

CAUTION

The debug functions described in this section are supported by the microcontroller but whether they are usable depends on the debugger. For details on debugging, see the user's manual of the debugger.

(1) Debug Interface

This microcontroller supports the NEXUS JTAG Interface, and low-pin debug interface (4 pins) (hereinafter referred to as LPD (4 pins)) as debug interfaces. Also it incorporates the AUD-RAM monitor and supports the AUD-RAM monitor interfaces to monitor and tune the on-chip RAM, data peripheral registers, etc.

For the AUD-RAM monitor, see **Section 30.4, AUD-RAM Monitor (AUDR)**.

(2) Debug Monitor Function

In debug mode, the monitoring program is executed in the debugging-only area.

The basic debug functions below can be used by running a monitoring program.

- Downloading user programs
- Reading and writing user resources including the memory and registers while the user program is suspended
- Running the user program starting at any address

(3) On-chip Break Function

12 break points are included in each CPU. Four of them can be designated for any access (access address and access data).

(4) Software Break Function

A software break point can be designated for any address.

(5) Forced Break Function

Execution of the user program can be forcibly suspended.

(6) Forced Reset Function

The microcontroller (this product) can be forcibly reset.

(7) Real-time RAM Monitoring (RRM)

The memory can be read during program execution. Because this read access uses debug-dedicated DMA, it has minimal effect on program execution.

(8) Dynamic Memory Modification (DMM)

The memory can be written during program execution. Because this write access uses debug-dedicated DMA, it has minimal effect on program execution.

(9) Timer Function

Using a 32-bit counter, the time for running the user program can be measured based on the clock for debug.

(10) Mask Function

A reset factor (external reset, software reset, and ECM reset) can be masked.

(11) Event Detection Function

Events can be detected by the following: execution address, access address, access data, range (comparison in size), and sequential execution.

(12) Hot Plug-in Function

Debugging can be started in normal operating mode without an input of an external reset.

(13) Security Function

To prevent the contents of the flash memory from being read by an unauthorized person, a 128-bit ID code (OCD_ID) can be written to the microcontroller. If the code the user inputs when starting a debugger does not match the ID code written to the microcontroller, the flash memory cannot be accessed.

(14) Tracing Function

Execution history, data changes, etc. of the user program can be obtained. For details, see **Section 30.2, Trace Control Function**.

(15) Multi Core Debug Function

The following functions are supported as the multi-core debugger for the CPU1 and CPU2:

- Synchronization functions (including reset, execution, and break)
- Synchronous setting
- Simultaneous tracing for multiple cores.

30.2 Trace Control Function

This product provides several trace functions including branch PC trace of the CPU, data trace, and DMA data trace.

(1) Trace RAM

This product has 32 KB for the trace RAM.

The trace information in the trace RAM is accessible via the debug interface: NEXUS, LPD (4 pins).

(2) Software Trace

This function enables the obtaining of history of user program execution, data changes, etc.

The software trace information can be output via the debug interface, LPD (4 pins).

30.3 Peripheral Break Control

The peripheral break function stops the peripheral modules if the user program is stopped, for instance upon a breakpoint hit.

The on-chip modules can be classified into two by its operation at the time of peripheral break as follows:

1. Modules that are unconditionally stopped: WDTA0, WDTA1*1
2. Modules that can select abeyance or continuation*2: OSTM0, OSTM1, OSTM2*1, TAPA0, TAPA1, TAPA2, TAPA3, TSG30, TSG31, CSIH0, CSIH1, TAUJ0, TAUD0, TAUD1, ENCA0, ENCA1, TPBA0, TPBA1*1

Note 1. Not provided for RH850/C1M.

Note 2. TAPAn forcibly places the output pins of TAUD and TSG3n into Hi-Z when its function is stopped by peripheral break. The value of the TAPAnHOF[10:8] bits of the TAPAnFLG register becomes 111_B. The applicable output pins are as follow:
TAPAnUN, TAPAnUP, TAPAnVN, TAPAnVP, TAPAnWN, TAPAnWP, and TSG3nO1 to TSG3nO6

30.4 AUD-RAM Monitor (AUDR)

30.4.1 Overview

This product includes the AUD (Advanced User Debugger) -RAM monitor (AUDR) to support debugging of a user program under conditions as if it were actually mounted in the system. The AUDR is a function to read and write the resources mapped to memory spaces including an on-chip memory and a peripheral register during LSI operation.

The AUDR is provided for only RH850/C1H (252-pin BGA).

Table 30.1 lists the outlines of the AUDR and **Figure 30.1** shows the block diagram of the AUDR.

Table 30.1 Outlines of the AUDR

Item	Outline
Transfer method	Clock-synchronous parallel interface (4 bits)
Transfer clock generation	Transfer clock is generated at the external host (RAM monitor tool) side.
Transfer clock frequency	Maximum 20 MHz
Access area	Physical address area on the system bus
Access data size	8, 16, 32, and 64 bits
Access address input bit width	8, 16, 24, and 32 bits The same values as those of the previous access address are used for the upper bits of access address, which are not input.
Data transfer method	• Single transfer: Single data is transferred to the access address that has been input. • Continuous transfer: Up to 16 data are continuously transferred from the access address that has been input. The access address is automatically incremented depending on data size.
I/O pin	7 pins ($\overline{\text{AUDRST}}$, AUDCK, $\overline{\text{AUDSYNC}}$, and AUDATA3 to AUDATA0)
Function	• RAM monitor function: Read and write are performed from a system bus to an accessible physical address area. This function enables reference to and modification of an on-chip memory, a peripheral register, and the like. • Configuration information retention (startup communication) function: The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained. This function is used for communication with a RAM monitor tool. • Synchronization communication (message board) function: This flag register is used for communication of the firmware operated by the CPU with a RAM monitor tool.

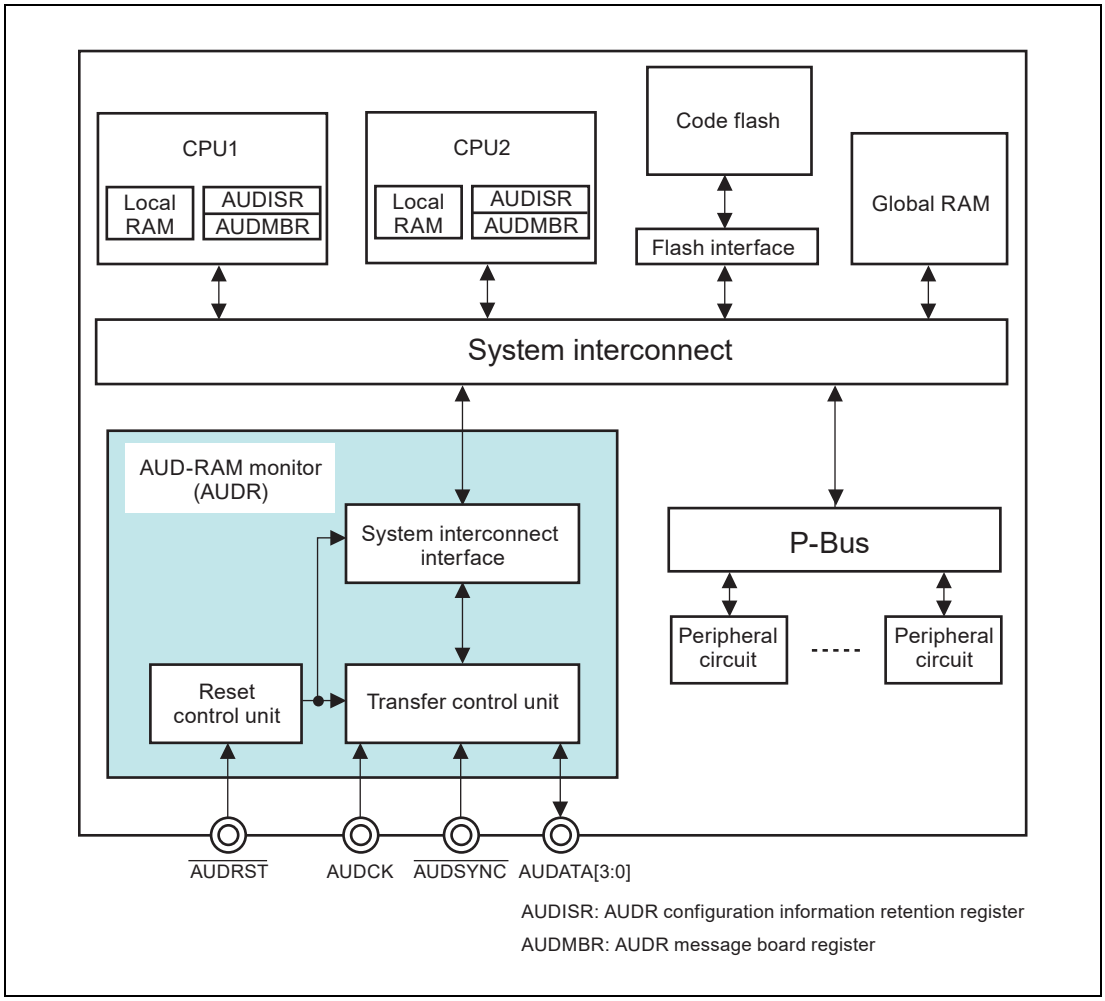


Figure 30.1 Block Diagram of the AUDR

30.4.2 I/O Pins

Table 30.2 lists the I/O pins of the AUDR.

Table 30.2 I/O Pins of AUDR

Pin Name	I/O	Description
$\overline{\text{AUDRST}}$	Input	AUDR reset input pin Inputting L to this pin resets the AUDR, but it does not initialize the AUDISR, AUDMBR and AUDMBRC (described below). When this pin is not connected, it is internally pulled-down.
AUDCK	Input	External clock input pin The frequency of clock that can be input is 20 MHz or under. When this pin is not connected, it is internally pulled-up.
$\overline{\text{AUDSYNC}}$	Input	Timing control signal input pin L: A command, an address, and write data are input and the status flag is output. H: The read-out data is output, and the pin enters into an idle state. When this pin is not connected, it is internally pulled-up. **CAUTION** This pin should not be negated (to a high level) until the ready state is entered by inputting a command or the like from the outside to the AUDATA. For details, see the protocol mentioned below.
AUDATA3 to AUDATA0	Input/Output	4-bit parallel data I/O pin. The following information is input and output by time division. • Command (input) • Address (input) • Write data (input)/Read data (output) • Status flag (output) When this pin is not connected, it is internally pulled-up.

30.4.3 Description of Registers

Table 30.3 lists the registers related to the AUDR.

Table 30.3 List of Registers

Register Name	Abbreviation	Value after reset	R/W	Address	Access Size	Reference section
AUDR configuration information retention register	AUDISR	$000x_H$*1	R	FA00 5000_H (CPU)*4 F900 5000_H (AUDR)*4 F800 5000_H (AUDR)*4	16*5	**30.4.3.1**
AUDR message board register	AUDMBR AUDMBRC	0000_H	R/W *2	FA00 5004_H (CPU)*4 F900 5004_H (AUDR)*4 F800 5004_H (AUDR)*4	16*5	**30.4.3.2**
			R/W *2,*3	FA00 5008_H (CPU)*4 F900 5008_H (AUDR)*4 F800 5008_H (AUDR)*4	16*5	

Note 1. The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained in bits 3 to 0.

Note 2. Only 1 can be written to the bits that have been set to 0. 0 cannot be written to the bits that have been set to 1.

Note 3. All bits are cleared to 0 after read.

Note 4. Different addresses are used for the AUDISR and AUDMBR/AUDMBRC bits when they are accessed from the CPU and when they are accessed from the AUDR.
Use FA00 $500x_H$ for an access from the CPU. Use F900 $500x_H$ for an access from the AUDR to the CPU1 register. Use F800 $500x_H$ for an access from the AUDR to the CPU2 register.
Although this addresses are “reserved area”, it is capable of accessing to this addresses for communication with AUDR tool.

Note 5. If an access in data sizes other than 16 bits (half-word) is performed, correct operation is not guaranteed.

30.4.3.1 AUDISR — AUDR Configuration Information Retention Register

AUDISR is a 16-bit readable register. It can be read out from the CPU and the AUDR.

Use FA00 5000_H for an access from the CPU to the AUDISR.

Use F900 5000_H for an access from the AUDR to the AUDISR of CPU1.

Use F800 5000_H for an access from the AUDR to the AUDISR of CPU2.

The AUDISR register is not initialized by AUDR reset. The AUDISR can be also read out from the CPU during reset.

This register is used in the way described below.

- Retention of configuration information

  On release from an internal reset, the register retains the values corresponding to the levels on pins AUDATA3 to AUDATA0. Setting the levels on pins AUDATA3 to AUDATA0 in the emulator configuration allows programs to judge the connection configuration of the emulator.

  If nothing is connected to the AUD RAM monitoring-related pins, pins AUDATA3 to AUDATA0 are pulled up and the register is read as $000F_H$. If an AUD RAM monitoring tool is connected, the levels on pins AUDATA3 to AUDATA0 will correspond to a value other than $000F_H$. Reading of the AUDISR by the CPU can then be used to judge that a tool is connected.

  In addition, the values from pins AUDATA3 to AUDATA0 can be used to identify the vendor that is the source of the connected tool.

For details, see **Table 30.3, List of Registers** about access size, address and value after reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DATA			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	Values of the AUDATA3 to AUDATA0 pins			
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 30.4 AUDISR Register Contents

Bit Position	Bit Name	Function
15 to 4	—	Reserved These bits are always read as 0.
3 to 0	DATA	The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained.

30.4.3.2 AUDMBR/AUDMBRC — AUDR Message Board Register

The AUDMBR/AUDMBRC registers are 16-bit readable/writable registers. The registers can be read and written from the CPU and the AUDR.

Use FA00 5004_H or FA00 5008_H for an access from the CPU to the AUDMBR.

Use F900 5004_H or F900 5008_H for an access from the AUDR to the AUDMBR of CPU1.

Use F800 5004_H, F800 5008_H for an access from the AUDR to the AUDMBR of CPU2.

When the CPU reads the AUDMBRC from FA00 5008_H, all bits of the AUDMBR/AUDMBRC are cleared to 0 after reading.

When the CPU reads the AUDMBR from FA00 5004_H, the bits of the AUDMBR/AUDMBRC are not cleared.

When the AUDR reads the AUDMBR from F900 5008_H, all bits of the AUDMBR/AUDMBRC are cleared to 0 after reading.

When the AUDR reads the AUDMBR from F900 5004_H, the bits of the AUDMBR/AUDMBRC are not cleared. It is the same about the AUDMBR of CPU2.

The CPU and the AUDR can write to the AUDMBR/AUDMBRC from the above addresses. However, writing 0 to the bits that have been set to 1 will be ignored (they can only be set to 1).

Table 30.6 summarizes the recommended accesses to the AUDMBR.

The AUDMBR register is not initialized by AUDR reset. The AUDMBR can be also read and written from the CPU during AUDR reset.

AUDMBR and AUDMBRC are used in the way described below.

- Synchronous communications (message board) function

  The firmware (program) run by the CPU uses AUDMBR as a flag register for use in communications with the emulator, which reads AUDMBR by using the RAM monitoring function. This allows the emulator to follow the state of firmware (program) operations.

For details, see **Table 30.3, List of Registers** about access size, address and value after reset.

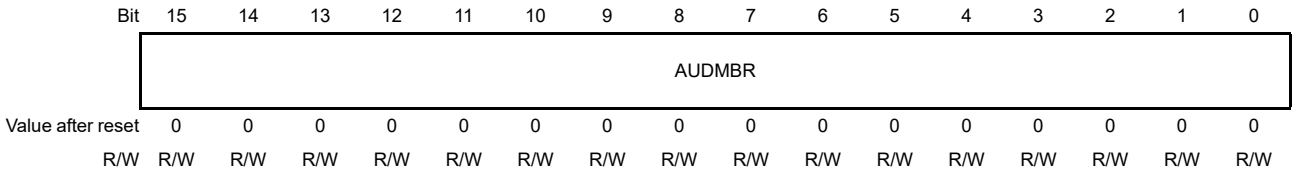


Table 30.5 AUDMBR/AUDMBRC Register Contents

Bit Position	Bit Name	Function
15 to 0	AUDMBR	Communication flag between the AUDR and the CPU

Table 30.6 Recommended Access to the AUDMBR/AUDMBRC

Accessing Master	Address	R/W	Access	Remarks
CPU	FA00 5004_H (AUDMBR)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	Not cleared after reading.
AUDR	AUDMBR(CPU1): F900 5008_H AUDMBR(CPU2): F800 5008_H (AUDMBRC)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	All bits are cleared to 0 after reading.

30.4.4 RAM Monitoring

30.4.4.1 Communication Protocol

Input a command, counter value, address, and data to the AUDATA pin in the format shown in **Figure 30.2**. For details, see **Section 30.4.4.2, Operation**.

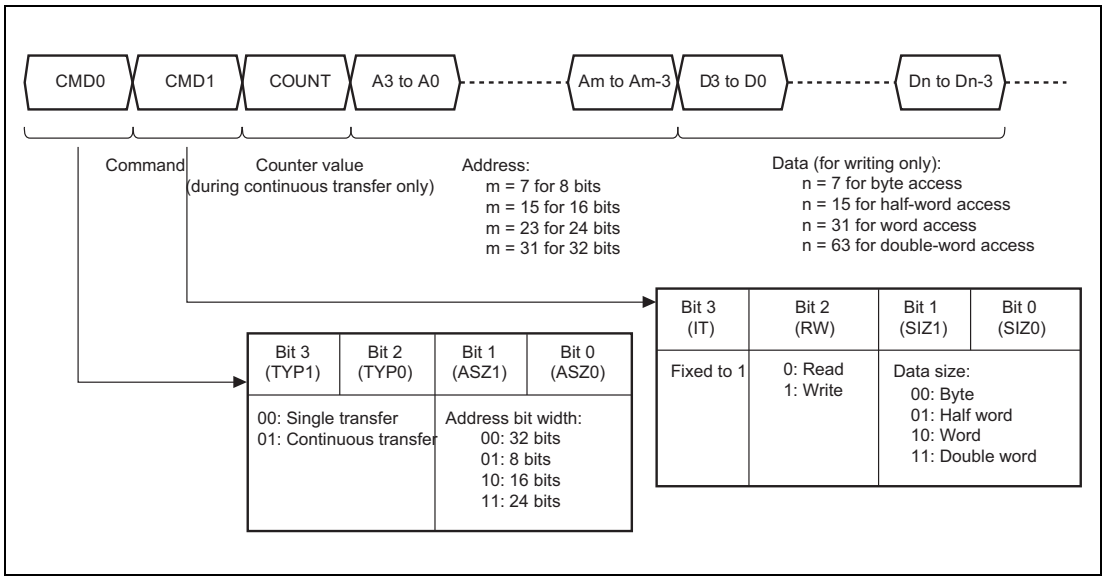


Figure 30.2 AUDATA Pin Input Format

30.4.4.2 Operation

(1) Single Transfer

Single transfer is a method of transmitting single data to an address that has been input for access. **Figure 30.3** shows an example of reading operation during single transfer and **Figure 30.4** shows an example of writing operation during single transfer.

When the $\overline{\text{AUDSYNC}}$ pin is asserted and a command, address and data (only for writing) are input to the AUDATA pin in the format shown in **Figure 30.2**, the AUDR starts reading or writing operation for the specified address. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. (See **Figure 30.3** and **Figure 30.4**).

For reading, after the Ready flag is output and the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin (see **Figure 30.3**).

After output of the read data and until input of the next command, at least one AUDCK cycle is required for switching of the input/output states of the pins.

In addition, after the $\overline{\text{AUDSYNC}}$ pin is negated following the completion of processing for write access, when the next command is input, at least one AUDCK cycle is required for switching of the input/output states of the pins. Therefore, the $\overline{\text{AUDSYNC}}$ pin is negated for two cycles of AUDCK.

If a command other than those shown in **Figure 30.2** is input to CMD0 and CMD1, the AUDR disables this processing as a command error and sets the CFLG bit in the Ready flag to 1. When a bus error occurs during the internal execution, the AUDR disables this processing and sets the BFLG bit in the Ready flag to 1 (see **Figure 30.5**).

When an error is detected, the read data is not output.

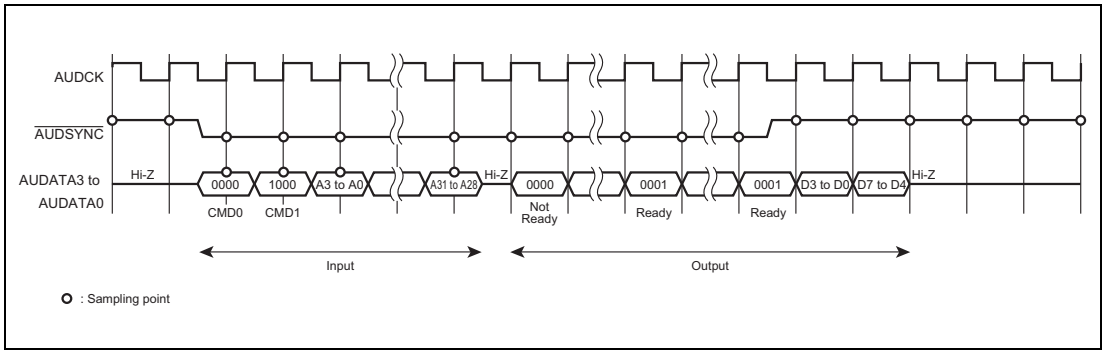


Figure 30.3 Example of Reading Operation in Single Transfer (address 32 bits; byte read)

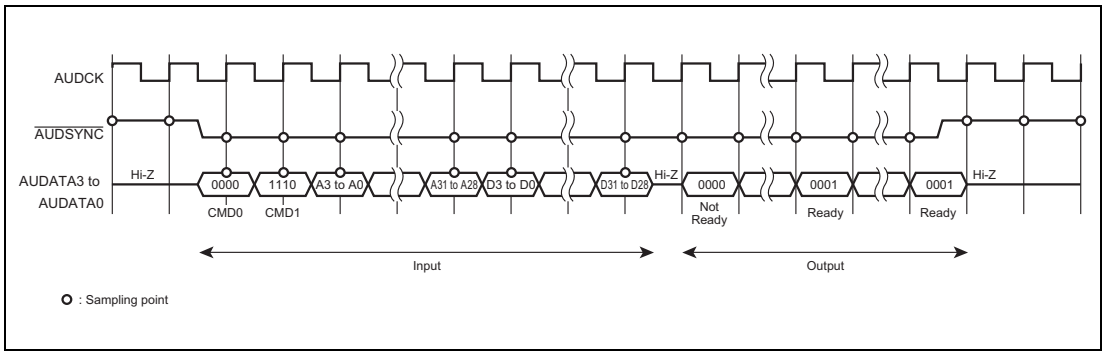


Figure 30.4 Example of Writing Operation in Single Transfer (address 32 bits; word write)

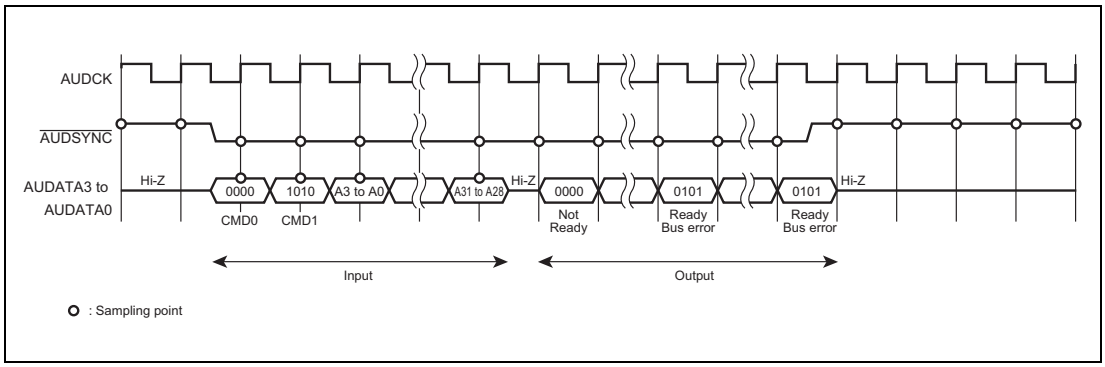


Figure 30.5 Example of Error Occurrence in Single Transfer (address 32 bits; word read)

(2) Continuous Transfer

Continuous transfer is a continuous transmitting method of up to 16 data to the address that has been input. The address to be accessed is automatically incremented depending on the data size after completion of each data transfer. **Figure 30.6** shows an example of reading operation during continuous transfer and **Figure 30.7** shows an example of writing operation during continuous transfer.

The first data transfer is equivalent to single transfer except that COUNT (counter value) shown in **Figure 30.2** should be input. The number of data to be transferred minus 1 is input to the COUNT.

In the second and subsequent data transfer, the input of CMD0, CMD1, COUNT is skipped, and addresses are skipped to input. The second and subsequent data transfer for reading and writing proceed as follows.

For reading, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. Then after the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin. This operation is repeated until the number of data specified in COUNT is read (see **Figure 30.6**).

For writing, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR inputs data to be written to the AUDATA pin. After that, the AUDR starts the internal execution. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this operation. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. This operation is repeated until the number of data specified in COUNT are written (see **Figure 30.7**).

When the $\overline{\text{AUDSYNC}}$ pin is re-asserted to input the next write data, at least two cycles of AUDCK are required for negating the $\overline{\text{AUDSYNC}}$ pin.

If a bus error occurs during the internal execution, the AUDR suspends data transfer operation and sets the BFLG bit in the Ready flag to 1 (see **Figure 30.8**).

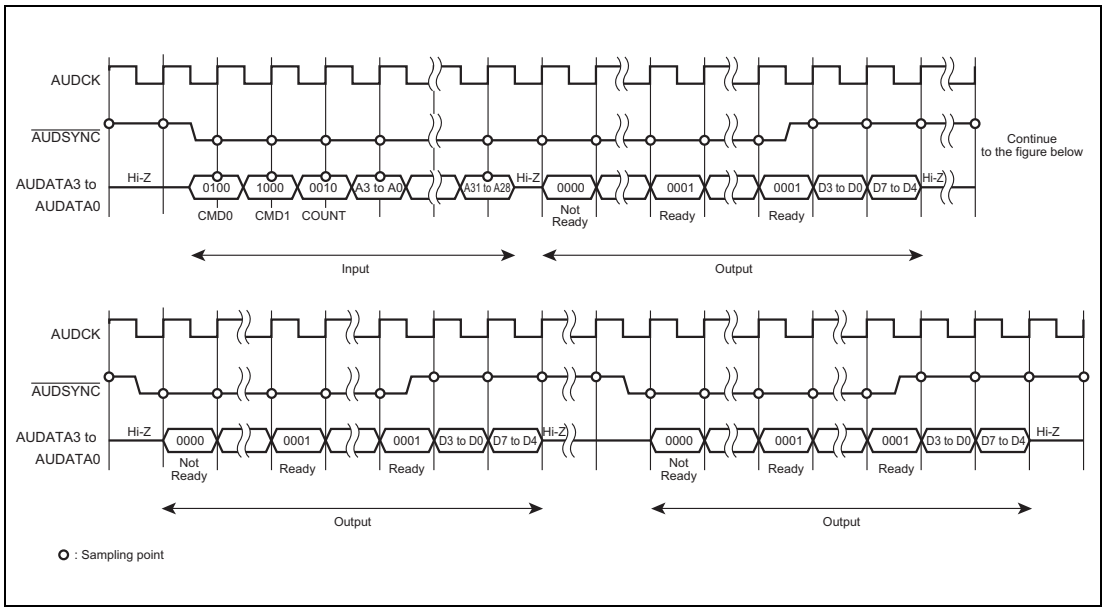


Figure 30.6 Example of Reading Operation in Continuous Transfer (address 32 bits; byte read × three units of data)

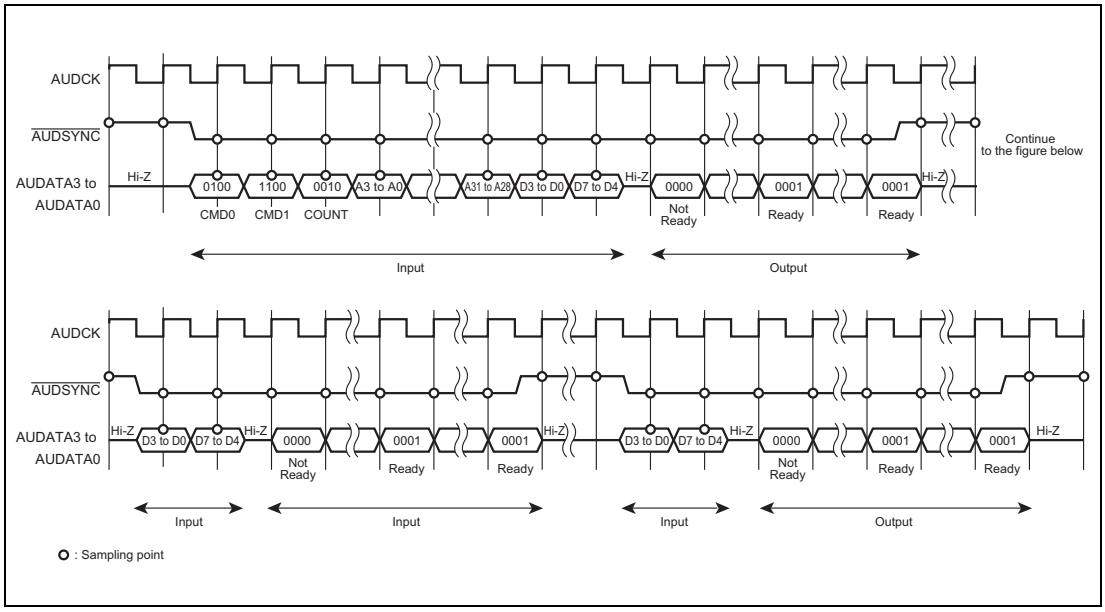


Figure 30.7 Example of Writing Operation in Continuous Transfer (address 32 bits; byte write × three units of data)

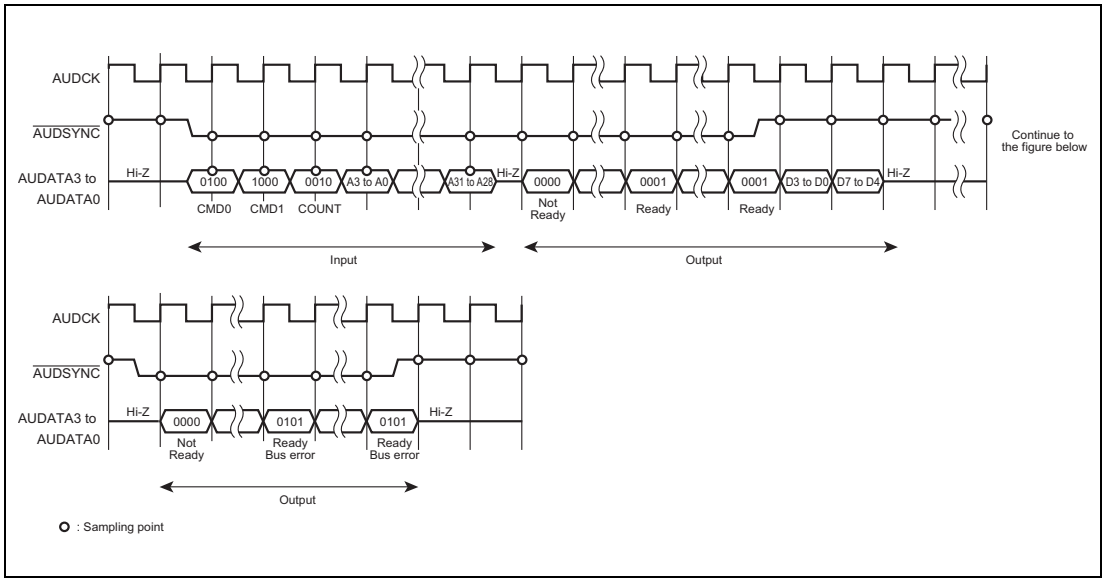


Figure 30.8 Example of Error Occurrence in Continuous Transfer (address 32 bits; byte read × three units of data)

(3) Command Error Conditions

Table 30.7 Command (CMD0) Error Conditions

Bit 3 (TYP1)	Bit 2 (TYP0)	Bit 1 (ASZ1)	Bit 0 (ASZ0)	Description
0	0	0	0	Single transfer Address bit width: 32 bits
0	0	0	1	Single transfer Address bit width: 8 bits
0	0	1	0	Single transfer Address bit width: 16 bits
0	0	1	1	Single transfer Address bit width: 24 bits
0	1	0	0	Continuous transfer Address bit width: 32 bits
0	1	0	1	Continuous transfer Address bit width: 8 bits
0	1	1	0	Continuous transfer Address bit width: 16 bits
0	1	1	1	Continuous transfer Address bit width: 24 bits
0	×	×	×	Command error

Table 30.8 Command (CMD1) Error Conditions

Bit 3 (TYP1)	Bit 2 (TYP0)	Bit 1 (SIZ1)	Bit 0 (SIZ0)	Description
0	×	×	×	Command error
1	0	0	0	Read: Byte
1	0	0	1	Read: Half word
1	0	1	0	Read: Word
1	0	1	1	Read: Double word
1	1	0	0	Write: Byte
1	1	0	1	Write: Half word
1	1	1	0	Write: Word
1	1	1	1	Write: Double word

(4) Bus Error Conditions

- Half-word access is made to the addresses of 4n + 1 and 4n + 3.
- Word access is made to the addresses of 4n + 1, 4n + 2, and 4n + 3.
- Double-word access is made to the addresses of 8n + 1, 8n + 2, 8n + 3, 8n + 4, 8n + 5, 8n + 6, and 8n + 7.
- An error response is received from the system bus.

(5) AUDATA Pin Input Format

Table 30.9 Input Format Bit Position

Input Order	Format Name	Bit Position				√: Required —: Not Required			
		AUDATA3	AUDATA2	AUDATA1	AUDATA0				
Earlier	CMD0	TYP1	TYP0	ASZ1	ASZ0	√			
	CMD1	IT	RW	SIZ1	SIZ0	√			
	COUNT	C3	C2	C1	C0	—: Single transfer; √: Continuous transfer			
	Address					8 bits	16 bits	24 bits	32 bits
		A3	A2	A1	A0	√	√	√	√
		A7	A6	A5	A4	√	√	√	√
		A11	A10	A9	A8	—	√	√	√
		A15	A14	A13	A12	—	√	√	√
		A19	A18	A17	A16	—	—	√	√
		A23	A22	A21	A20	—	—	√	√
		A27	A26	A25	A24	—	—	—	√
		A31	A30	A29	A28	—	—	—	√
	Data (for writing only)					Byte write	Half-word write	Word write	Double-word write
		D3	D2	D1	D0	√	√	√	√
		D7	D6	D5	D4	√	√	√	√
		D11	D10	D9	D8	—	√	√	√
		D15	D14	D13	D12	—	√	√	√
		D19	D18	D17	D16	—	—	√	√
		D23	D22	D21	D20	—	—	√	√
		D27	D26	D25	D24	—	—	√	√
		D31	D30	D29	D28	—	—	√	√
		D35	D34	D33	D32	—	—	—	√
		D39	D38	D37	D36	—	—	—	√
		D43	D42	D41	D40	—	—	—	√
		D47	D46	D45	D44	—	—	—	√
		D51	D50	D49	D48	—	—	—	√
		D55	D54	D53	D52	—	—	—	√
		D59	D58	D57	D56	—	—	—	√
Later		D63	D62	D61	D60	—	—	—	√

Table 30.10 CMD0 Format

Bit Name	Function	Description
TYP[1:0]	Type of transfer	00: Single transfer 01: Continuous transfer
ASZ[1:0]	Specify an address bit width	These bits specify the bit width of an address input from the AUDATA pin. When 8-, 16-, or 24-bits are specified, the same values as the previous access address are used for the upper bits that are not input from the AUDATA pin. Input a 32-bit address for the first access after reset release or occurrence of a command or bus error. 00: 32 bits 01: 8 bits 10: 16 bits 11: 24 bits

Note 1. Settings other than the above cause a command error.

Table 30.11 CMD1 Format

Bit Name	Function	Description
IT	Specifies access space	Set this bit to 1.
RW	Specifies read or write	0: Read 1: Write
SIZ[1:0]	Specify data size	These bits specify the size of data to be accessed. 00: Byte (8 bits) 01: Half word (16 bits) 10: Word (32 bits) 11: Double word (64 bits)

Note 1. Settings other than the above cause a command error.

Table 30.12 COUNT Format

Bit Name	Function	Description
C3 to C0	Specify the number of data to be transferred	These bits specify the number of data to be transferred in continuous transfer. 0000: 1 data 0001: 2 data 0010: 3 data 0011: 4 data 0100: 5 data 0101: 6 data 0110: 7 data 0111: 8 data 1000: 9 data 1001: 10 data 1010: 11 data 1011: 12 data 1100: 13 data 1101: 14 data 1110: 15 data 1111: 16 data

Table 30.13 Address Format

Bit Name	Function	Description
A31 to A0	Specify address	These bits specify an address to be accessed. The required number of bits depends on the setting of the ASZ[1:0] bits in CMD0 (for details, see **Table 30.9**).

Table 30.14 Write Data Format

Bit Name	Function	Description
D63 to D0	Specify write data	These bits specify write data. The required number of bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 30.9**).

(6) AUDATA Pin Output Format

Table 30.15 Ready Flag Format

Bit Position	Bit Name	Function	Description
AUDATA3	0	—	—
AUDATA2	BFLG	This bit indicates a bus error.	0: Normal 1: A bus error occurred.
AUDATA1	CFLG	This bit indicates a command error.	0: Normal 1: A command error occurred.
AUDATA0	RFLG	This bit indicates completion of AUDR operation.	0: Not Ready 1: Ready

Table 30.16 Read Data Bit Position

Output Order	Bit Position				√: Required; —: Not Required			
	AUDATA3	AUDATA2	AUDATA1	AUDATA0	Byte read	Half-word read	Word read	Double-word read
Earlier	D3	D2	D1	D0	√	√	√	√
	D7	D6	D5	D4	√	√	√	√
	D11	D10	D9	D8	—	√	√	√
	D15	D14	D13	D12	—	√	√	√
	D19	D18	D17	D16	—	—	√	√
	D23	D22	D21	D20	—	—	√	√
	D27	D26	D25	D24	—	—	√	√
	D31	D30	D29	D28	—	—	√	√
	D35	D34	D33	D32	—	—	—	√
	D39	D38	D37	D36	—	—	—	√
	D43	D42	D41	D40	—	—	—	√
	D47	D46	D45	D44	—	—	—	√
	D51	D50	D49	D48	—	—	—	√
	D55	D54	D53	D52	—	—	—	√
	D59	D58	D57	D56	—	—	—	√
Later	D63	D62	D61	D60	—	—	—	√

Table 30.17 Read Data Format

Bit Name	Function	Description
D63 to D0	Output read data	The required number of bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 30.16**).

30.4.4.3 Usage Notes on the AUDR Function

- Do not negate the $\overline{\text{AUDSYNC}}$ pin until one cycle of AUDCK has elapsed after a command is input to the AUDATA pin and the Ready flag has been returned.
- When uninitialized memory is accessed through the AUDR, a bus error may occur due to ECC error detection.

30.4.4.4 Enabling and Disabling RAM Monitoring

Enabling or disabling of the AUDR can be specified by option byte (AUDREN).

AUDREN: Enable Bit of the AUDR

For setting of option byte, see **Section 31, Flash Memory**.

NOTE

The AUDR is disabled in serial programming mode regardless of this setting.

30.5 Cautions on Using On-Chip Debugger

(1) Treatment of Devices Used for Debugging

Do not install a device that was used for debugging on a mass-produced product, because the flash memory was rewritten during system debugging and thus the number of write/erase count of flash memory cannot be guaranteed.

(2) Hot plug-out

This product does not support hot plug-out for power off (including removal of the connector) of the debug tool in debug mode. Do not turn off the power of the NEXUS tool (including removal of the connector) in debug mode.

(3) Processing when ending on-chip debugging

When ending on-chip debugging, set the $\overline{\text{DCUTRST}}$ pin and external reset pin to the low level.

(4) Notes on using the debugger

When a debugger is used, the program written to the microcomputer before preparation for communications between the OCD emulator and microcomputer is complete is executed from the reset vector. For this reason, take care to ensure that this does not lead to any operations which you do not expect or intend.

This communications preparation period depends on the host PC environment of the OCD emulator and the operating frequency of the microcomputer.

Section 31 Flash Memory

This microcontroller incorporates code flash memory and data flash memory.

31.1 Features

- Code flash memory capacity: 4 Mbytes of user area (two banks configuration) and 32 Kbytes of user boot area
- Data flash memory capacity: 32 Kbytes of data area
- Methods of programming
  - Programming through transfer by a dedicated flash-memory programmer via a serial interface (serial programming)
  - Programming of flash memory by a user program (self-programming)
- Support for security functions to protect against illicit tampering with or reading out of data in flash memory
- Support for protection functions to protect against erroneous overwriting of the flash memory
- Support for the detection and correction of errors in the flash memory
- Support for the BGO (Back Ground Operation) function
  - Code flash memory can be read while data flash memory is being programmed.
- In the initial settings of this product, an expansion area of the flash memory (option bytes) can be set.

31.2 Structure of Memory

Figure 31.1 illustrates the mapping of the code flash memory. The user area of the code flash memory in this product is divided into 8- and 32-Kbyte blocks, which serve as the units of erasure. The user area is available as areas for storing the user program.

Furthermore, a 32-Kbyte user boot area, which is protected against programming by self-programming, is incorporated as a single block. This area is thus available as an area for storing boot programs, etc., for which programming while the user program is running has to be prohibited.

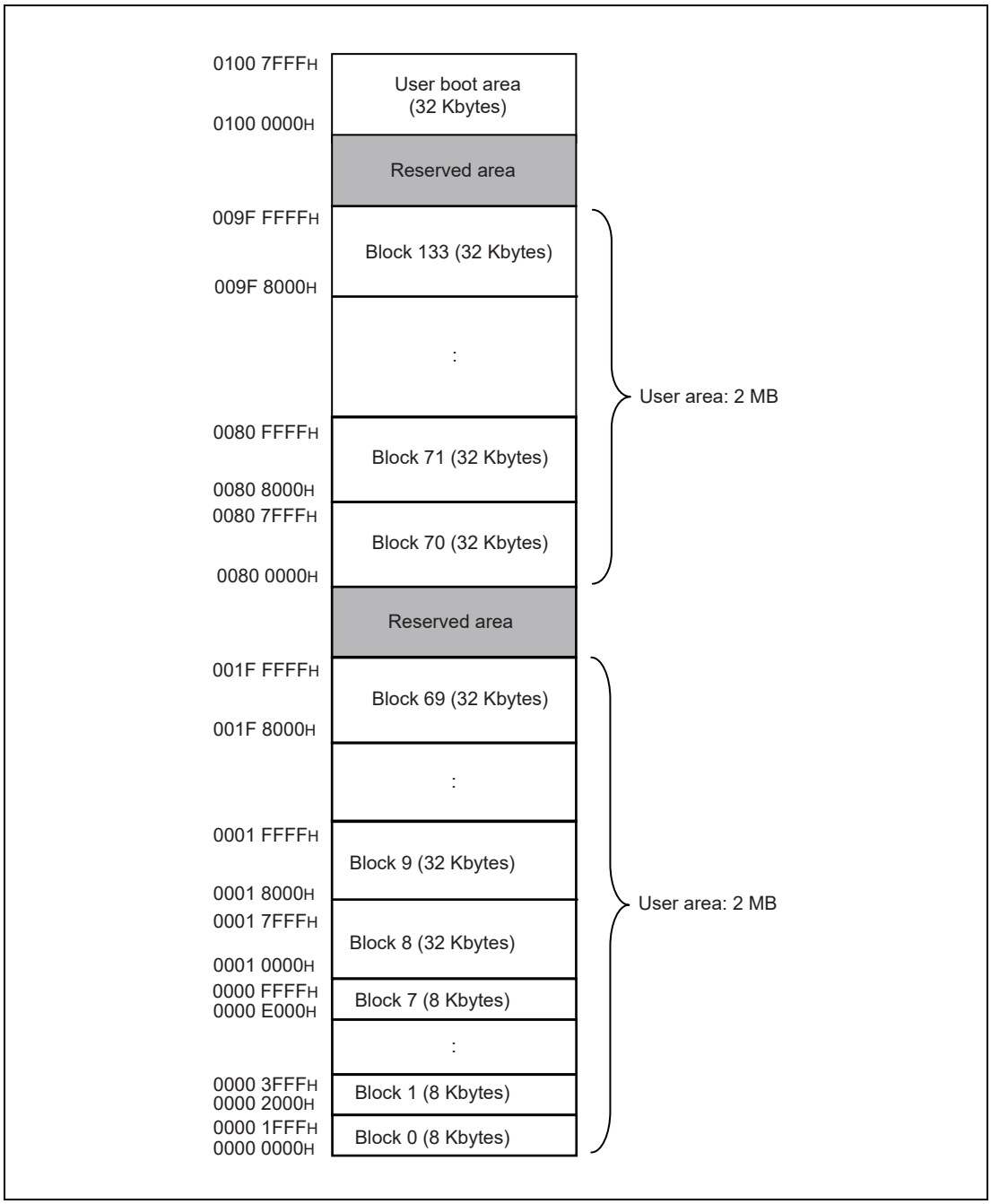


Figure 31.1 Code Flash Memory Map (8 Kbytes × 8 + 32 Kbytes × 126 configuration) _C1H

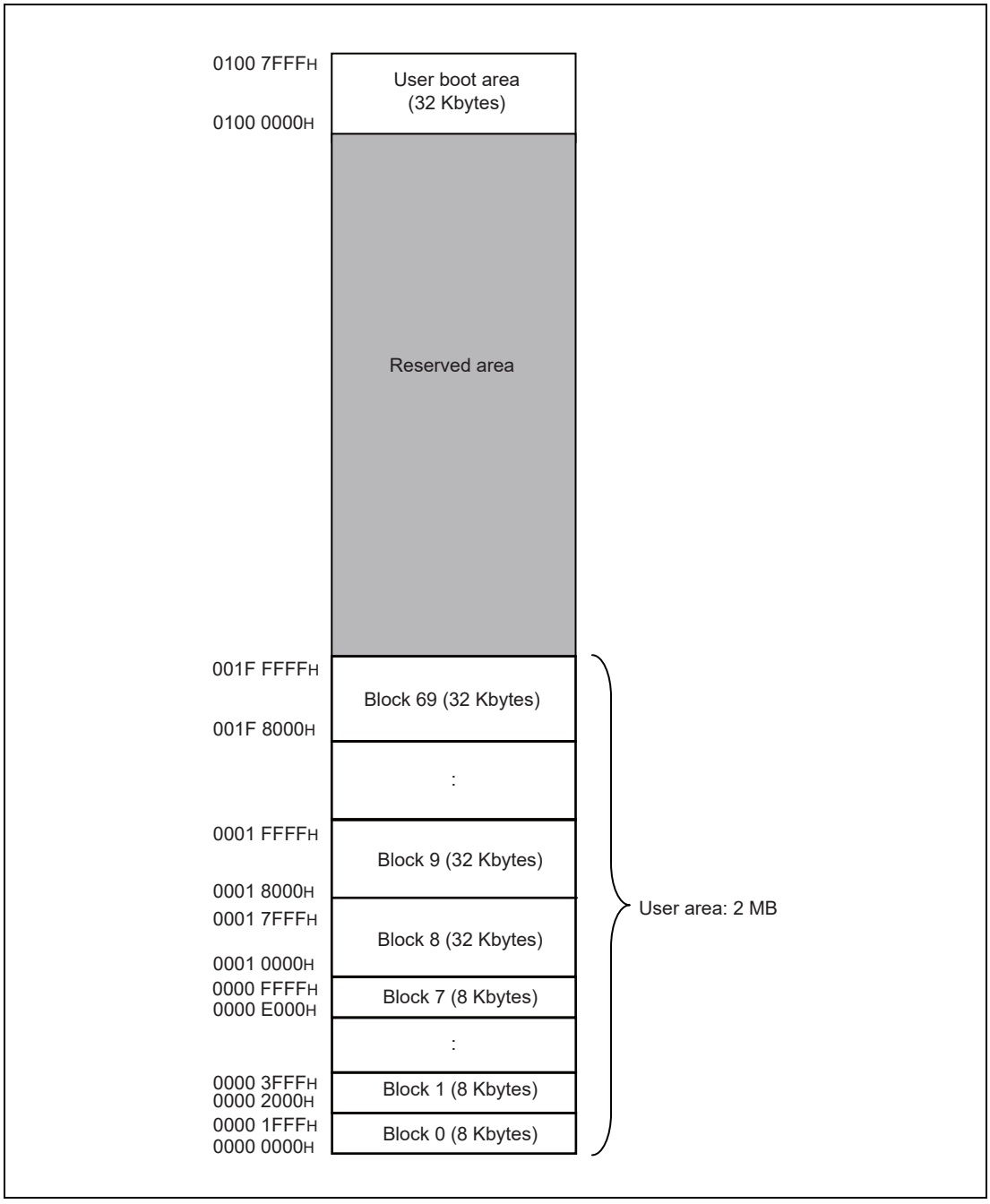


Figure 31.2 Code Flash Memory Map (8 Kbytes × 8 + 32 Kbytes × 62 configuration) _C1M

The data area of the data flash memory in this product is divided into 64-byte blocks, with each being a unit for erasure. **Figure 31.3** shows the mapping of the data flash memory.

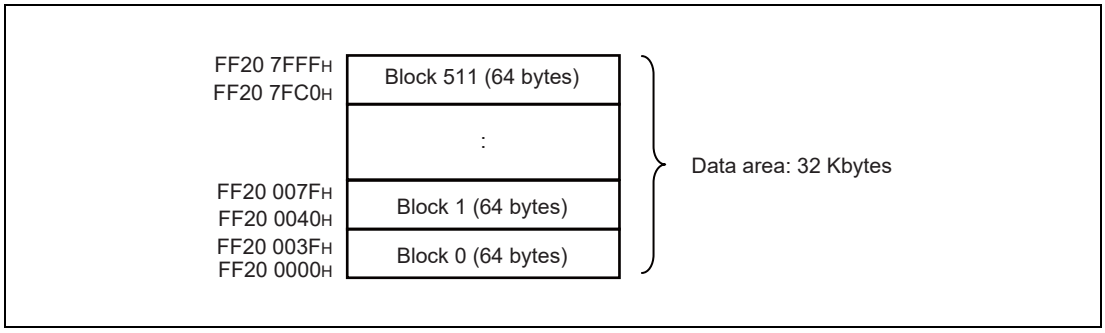


Figure 31.3 Data Flash Memory Map (64 bytes × 512 configuration)

31.3 Operating Modes Associated with Flash Memory

Figure 31.4 is a diagram of the mode transitions associated with the flash memory. For the procedures for setting the modes, refer to **Section 5, Operating Mode**.

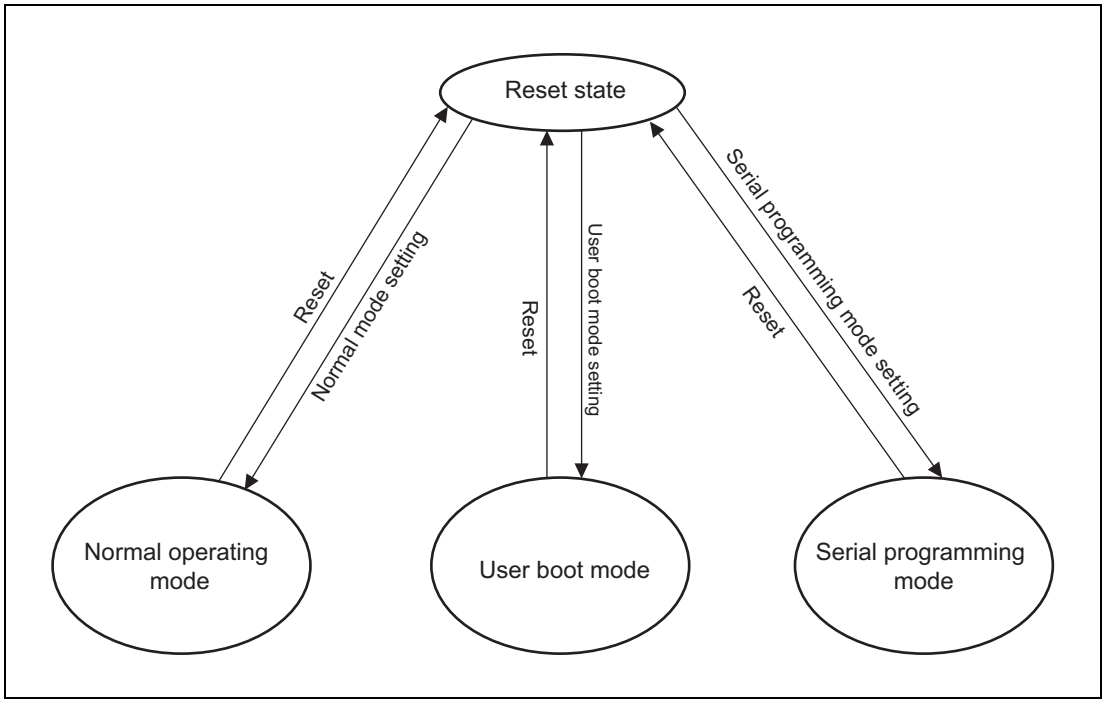


Figure 31.4 Mode Transition Associated with Flash Memory

The flash memory areas which are programmable and erasable and the boot program after a reset depend on the selected mode. The differences between modes are indicated in **Table 31.1**.

Table 31.1 Differences between Modes

Item	Normal Operating Mode*1	User Boot Mode	Serial Programming Mode
Programmable and erasable area	User area Data area	User area Data area	User area User boot area Data area
Boot program at a reset	Program in user area	Program in user boot area	Embedded program for serial programming

Note 1. “Normal operating mode” refers to the user boot mode in which the boot area is in the user area.

31.4 Functional Overview

By using a dedicated flash-memory programmer to program the flash memory of this device via a serial interface (serial programming), the device can be programmed regardless of whether this is before or after it is mounted on the target system.

Furthermore, security functions to prohibit over-writing of the user program written to the flash memory are incorporated, and this can prevent tampering by third parties and so on.

Programming by the user program (self-programming) is available to suit applications where the application on the target system may require updating after manufacturing or shipment. Protection features for the safe overwriting of the flash memory area are also incorporated. Furthermore, interrupt processing during self-programming is supported so programming can proceed at the same time as processing for external communications etc., and this is the case in various situations.

Table 31.2 lists the overview of the methods of programming and the corresponding operating modes.

Table 31.2 Programming Methods

Programming Method	Functional Overview	Operating Mode
Serial programming	A dedicated flash-memory programmer is capable of on-board programming the flash memory after the device is mounted on the target system.	Serial programming mode
	A dedicated flash-memory programmer and dedicated programming adapter board allow off-board programming of the flash memory, i.e. programming of the device before it is mounted on the target system.	
Self-programming	The user program that is written to code flash memory in advance of serial programming executing is also capable of programming the flash memory. The background operation capability makes it possible to fetch instructions or otherwise read data from code flash memory while the data flash memory is being programmed. For this reason, it is possible to program the data flash memory by executing a program written to the code flash memory. Instructions in the code flash memory cannot be fetched and data cannot be accessed while the code flash memory is being programmed by self-programming. In such cases, a program for programming from the local RAM or global RAM must be transferred in advance and executed.	Normal operating mode, user boot mode

When executing self-programming, see the *RH850/C1x Flash Memory User's Manual: Hardware Interface* which this product targets.

Table 31.3 lists the functions of the on-chip flash memory. Dedicated flash-memory programmer commands realize serial programming, while reading of the on-chip flash memory by interface operation of flash memory or the user program realizes self-programming.

Table 31.3 Basic Functions at a Glance

Function	Description in Overview	Level of Support (○: Supported, △: Conditionally Supported, ×: Not Supported)	
		Serial programming	Self-programming
Blank checking	This is used to check a specified block to ensure that writing to it has not already proceeded. Results of reading from code flash memory and data flash memory to which nothing has been written after erasure are not guaranteed, so use blank checking to confirm that writing to memory has not proceeded after erasure.	○	○
Block erasure	This is for erasing the contents of a specified block of memory.	○	○
Programming	This is for writing to a specified address.	○	○
Verification and checksum	Data that are read out from flash memory are compared with data transferred from the flash memory programmer.	○	○
Reading	Data that have been written to the flash memory are read out.	○	○
Setting for OTP (one-time programming)	A specified block of code flash memory is set for OTP (OTP can only be set, that is, it is not possible to release a block's OTP setting).	○	○
Setting an ID	An ID setting is made for use in controlling the connection of a dedicated flash-memory programmer for serial programming, and programming of the code flash memory by self-programming.	○	○
Security settings	Security settings are for use in serial programming.	○	△ (only when setting is prohibited after being permitted)
Protection settings	Lock bits for all blocks of code flash memory are provided and the value of the reset vector is variable.	○	○
Setting of option bytes	Option bytes are set to change them from the value after resets for this product.	○	○
Clearing the configuration	ID setting, security settings, protection settings, and option byte settings are initialized.	○	×

For details on serial programming, refer to the *PG-FP5 Flash Memory Programmer Users' Manual* and *Renesas Flash Programmer Flash Programming Software User's Manual.*

For details on self-programming, refer to the *RH850/C1x Flash Memory User's Manual: Hardware Interface* which this product targets.

The on-chip flash memory supports various protection functions.

The OTP setting and authentication of the ID code are security functions for use with serial programming and self-programming.

In serial programming, authentication of the ID code, prohibiting connection of a dedicated flash-memory programmer, and prohibition of commands (for block erasure, programming, and reading) are available for use as security functions.

Table 31.4, Summary of Security Functions lists security functions that are supported by the on-chip flash memory and **Table 31.5, Available Operations and Security Settings** lists security setting operations.

Table 31.4 Summary of Security Functions

Function	Description
OTP	OTP can be individually set for each block of the user area and the user boot area of code flash memory. When the OTP setting is made for an area, programming by serial programming and by self-programming is prohibited. Once set, the OTP setting cannot be released. Furthermore, since execution of the configuration clearing command is prohibited for any area for which OTP has been set, changing a security setting from "prohibited" to "permitted" is not possible.
ID authentication	The result of ID authentication can be used to control the connection of a dedicated flash memory programmer for serial programming. The result of ID authentication can also be used to control enabling of code flash memory programming by self-programming.
Prohibition of connection of a dedicated flash memory programmer	The connection of a dedicated flash memory programmer for serial programming is prohibited. Since execution of the configuration clearing command is also prohibited when the connection of a dedicated flash memory programmer is prohibited, changing a security setting from "prohibited" to "permitted" is not possible.
Prohibition of block erasure commands	Block erasure commands at the time of serial programming are prohibited. Since execution of the configuration clearing command is also prohibited when block erasure commands are prohibited, changing a security setting from "prohibited" to "permitted" is not possible.
Prohibition of programming commands	Block erasure commands and programming commands at the time of serial programming are prohibited. Block erasure commands can only be executed by erasing all user areas in the following order starting from block 0: erasure of all areas → erasure of user boot area → erasure of data areas starting from block 0. Only through execution of the configuration clearing command, the prohibition can be lifted.
Prohibition of read commands	Read commands at the time of serial programming are prohibited. Only through execution of the configuration clearing command, the prohibition can be lifted.

Table 31.5 Available Operations and Security Settings

Function	All Security Settings and Erasure, Programming, and Read Operations (○: Executable, ×: Not Executable, —: Not Supported)		Point for Caution Regarding the Security Setting	
	Serial programming	Self-programming	Serial programming	Self-programming
OTP	• Areas for which OTP is set – Block erasure commands: × – Programming commands: × – Read commands: ○ • Areas for which OTP is not set – Block erasure commands: ○ – Programming commands: ○ – Read commands: ○	• Areas for which OTP is set – Block erasure: × – Programming: × – Reading: ○ • Areas for which OTP is not set – Block erasure: ○ – Programming: ○ – Reading: ○	• The OTP setting cannot be released. • Execution of the configuration clearing command is not possible.	The OTP setting cannot be released.
ID authentication	• When the ID codes do not match – Block erasure commands: × – Programming commands: × – Read commands: × • When the ID codes match – Block erasure commands: ○ – Programming commands: ○ – Read commands: ○	• When the ID codes do not match – Code flash memory - Block erasure: × - Programming: × - Reading: ○ – Data flash memory - Block erasure: ○ - Programming: ○ - Reading: ○ • When the ID codes match – Block erasure: ○ – Programming: ○ – Reading: ○	• The configuration clearing command can initialize the setting for prohibition. • The setting for prohibition of block erasure commands is not available. • The setting for prohibition of programming commands is not available. • The setting for prohibition of read commands is not available.	ID authentication is always in effect.
Prohibition of the connection of a dedicated flash memory programmer	• Block erasure commands: × • Programming commands: × • Read commands: ×	• Block erasure: ○ • Programming: ○ • Reading: ○	Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of block erasure commands	• Block erasure commands: × • Programming commands: ○ • Read commands: ○	• Block erasure: ○ • Programming: ○ • Reading: ○	• Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible. • The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of programming commands	• Block erasure commands: ×*1 • Programming commands: × • Read commands: ○	• Block erasure: ○ • Programming: ○ • Reading: ○	• Executing the configuration clearing command only can initialize the setting for prohibition. • The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of read commands	• Block erasure commands: ○ • Programming commands: ○ • Read commands: ×	• Block erasure: ○ • Programming: ○ • Reading: ○		

Note 1. Block erasure commands can only be executed by erasing all user areas in the following order starting from block 0: erasure of all areas → erasure of user boot area → erasure of data areas starting from block 0.

The on-chip flash memory supports various protection functions. **Table 31.6** lists the protection functions supported by the on-chip flash memory.

Table 31.6 Summary of Protection Functions

Function	Description
Block protection	Lock bit settings can be individually made to enable or disable programming and erasure of each block of the user area of code flash memory. Programming and erasure by self-programming of an area for which the lock bit is set and the lock bit function is enabled are prohibited. Programming or erasure can proceed again when the lock bit function is disabled after having been enabled. When a block of code flash memory is erased, the lock bit for that block is also erased.
User boot protection	Programming or erasure of the user boot area by self-programming is prohibited. Programming or erasure of the user boot area by serial programming is available.

31.5 Serial Programming

A dedicated flash-memory programmer can be used to program the flash memory in serial programming mode.

Serial Programming

The microcontroller is mounted on the system board at the time of serial programming. Providing a connector to the board enables programming of the target microcontroller by the flash memory programmer to proceed.

31.5.1 Environments for Programming

The recommended environments for programming the flash memory of the microcontroller with data are described below.

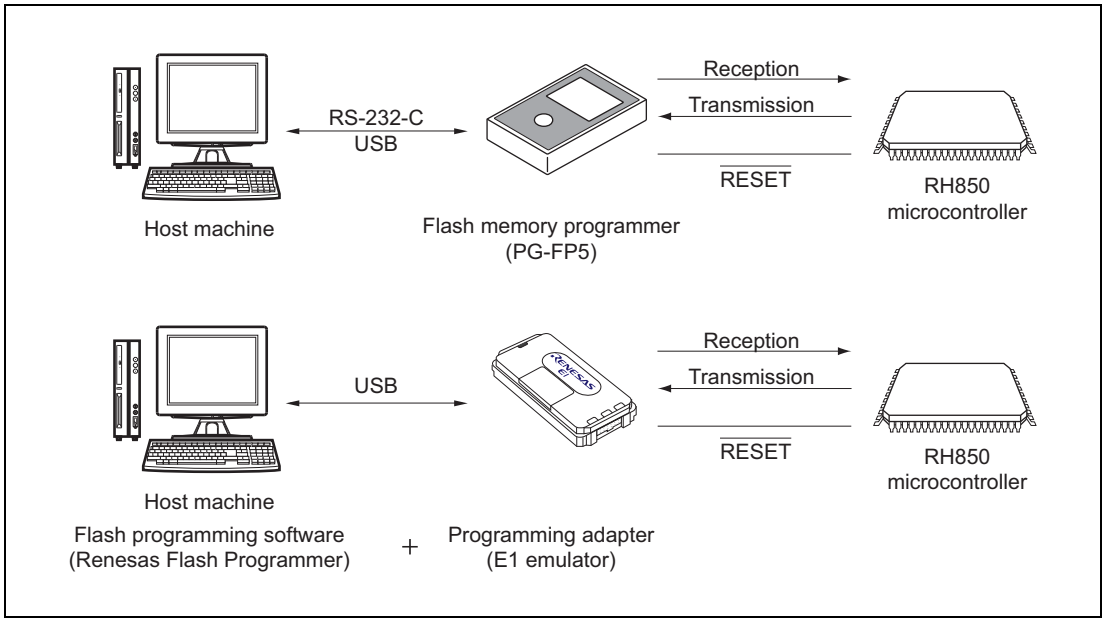


Figure 31.5 Environments for Programming Flash Memory

By using the PG-FP5 flash memory programmer or the combination of the Renesas Flash Programmer (software for writing to flash memory) running on the host machine and the E1 emulator as an programming adapter, the user is easily able to erase, program, and verify the contents of the on-chip memory of flash-memory-equipped microcontrollers from Renesas Electronics.

The PG-FP5 flash memory programmer handles programming from a host machine or programming in stand-alone mode while the Renesas Flash Programmer only handles programming from a host machine.

NOTE

Refer to the *PG-FP5 Flash Memory Programmer User's Manual* for the details of PG-FP5, and *Renesas Flash Programmer Flash Programming Software User's Manual* for details of Renesas flash programmer flash programming software.

31.6 Selection of the Communication Method

With this product, 2-line UART connection or clock synchronous connection is selectable for serial communication method by the setting of the FLMODE pin. For details of the FLMODE pin setting, see **Section 5, Operating Mode**. For setting of programming environment in line with each communication method, see ***PG-FP5 Flash Memory Programmer User's Manual*** and ***Renesas Flash Programmer Flash Programming Software User's Manual***.

31.7 Self-Programming

31.7.1 Overview

This product supports programming of the flash memory by user program itself. The code flash and data flash memory can be programmed by using the commands of flash memory application command interface (FACI) for flash memory programming in user's applications. Therefore, update of the user program and programming of constant data fields can be possible.

When the data flash memory is programmed, the background operation facility makes it possible to execute a programming program from the code flash memory to program the data flash memory. Furthermore, the programming program can be copied to the local RAM or global RAM in advance and executed to program the data flash memory.

For comprehensive information on flash self-programming, see the *RH850/C1x Flash Memory User's Manual: Hardware Interface* which this product targets.

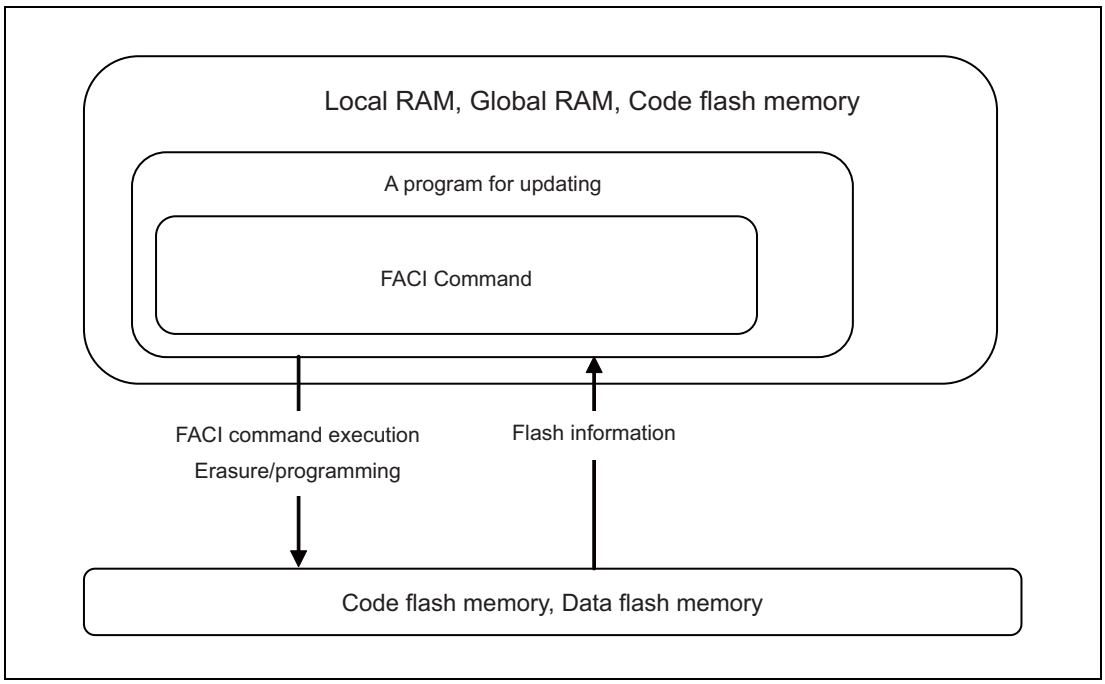


Figure 31.6 Concept of Self-Programming

31.7.2 Background Operation

Background operations can be used when the combination of the flash memory for writing and the flash memory for reading is any of those listed below.

Table 31.7 Conditions under which Background Operation is Usable

Range for Writing	Range for Reading
Data flash memory	Code flash memory

31.8 Reading Flash Memory

31.8.1 Reading Code Flash Memory

Special settings are not required to read code flash memory in normal mode and user boot mode. Data can simply be read out through access to addresses in the code flash memory.

Reading from an area of code flash memory that has been erased but not yet been programmed again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. Furthermore, since the values of data cannot be guaranteed when an ECC error has been generated, use blank checking when you need to confirm that an area is in the non-programmed state. See **Section 27, Functional Safety**, for the details on ECC.

31.8.2 Reading Data Flash Memory

Configure the number of read cycle in the EEPRDCYCL register prior to reading data from data flash memory in normal mode. Once this register is properly configured, data can be read by simply accessing addresses in the data flash memory. Once the setting for the number of cycles is made, data can simply be read out through access to addresses in the data flash memory.

Values read from data flash memory that has been erased but not yet been programming again are undefined. Use blank checking when you need to confirm that an area is in the non-programmed state.

31.9 Description of Registers

31.9.1 Registers Related to Data Flash Memory

Table 31.8 shows the list of registers related to data flash memory.

Table 31.8 List of Registers Related to Data Flash Memory

Module Name	Register Name	Abbreviation	R/W	Value after Reset	Address	Access Size
FLASH	Data flash memory read cycle setting register	FRDCYCLD	R/W	$0F_H$	FFC5 9810_H	8

31.9.1.1 FRDCYCLD — Data Flash Memory Read Cycle Setting Register

This register sets the read cycle of data flash memory.

Access: This register can be read/written in 8-bit units.

Address: FFC5 9810_H

Value after reset: $0F_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FRDCYCLD[3:0]			
Value after reset	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 31.9 FRDCYCLD Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved
3 to 0	FRDCYCLD[3:0]	Number of data flash memory read cycles Data flash memory is read in the setting value + 1 cycles. 0_H to 2_H: Setting prohibited 3_H: Read cycle 4 4_H: Read cycle 5 5_H: Read cycle 6 6_H: Read cycle 7 7_H: Read cycle 8 8_H: Read cycle 9 9_H to F_H: Read cycle 10

Note: One read cycle is CLK_LSB.

31.9.2 Registers Related to Write and Erase Protection of Flash Memory

Table 31.10 shows the list of registers related to write and erase protection of flash memory.

Table 31.10 Registers Related to Write and Erase Protection of Flash Memory

Module Name	Register Name	Abbreviation	R/W	Value after Reset	Address	Access Size
FLASH	FHVE15 control register	FHVE15	R/W	0000 0000_H	FFF8 $A430_H$	32
FLASH	FHVE3 control register	FHVE3	R/W	0000 0000_H	FFF8 2410_H	32

31.9.2.1 FHVE15 — FHVE15 Control Register

FHVE15 is a readable/writable register for software protection of flash memory against programming, erasure, and blank checking. Set both the FHVE15 and FHVE3 registers in the programmable, erasable, and blank-checkable state (0000 0001_H) to program, erase, or blank-check flash memory.

Access: This register can be read/written in 32-bit units.

Address: FFF8 $A430_H$

Value after reset: 0000 0000_H

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

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

Table 31.11 FHVE15 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE15CNT	0: Programming, erasure, and blank checking are disabled. 1: Programming, erasure, and blank checking are enabled.

31.9.2.2 FHVE3 — FHVE3 Control Register

FHVE3 is a readable/writable register for software protection of flash memory against programming, erasure, and blank checking. Set both the FHVE15 and FHVE3 registers in the programmable, erasable, and blank-checkable state ($0000\ 0001_H$) to program, erase, or blank-check flash memory.

Access: This register can be read/written in 32-bit units.

Address: $FFF8\ 2410_H$

Value after reset: $0000\ 0000_H$

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

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE3 CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 31.12 FHVE3 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE3CNT	0: Programming, erasure, and blank checking are disabled. 1: Programming, erasure, and blank checking are enabled.

31.9.3 Registers Related to Product Information

Table 31.13 shows the list of registers related to product information.

Table 31.13 List of Registers Related to Product Information

Module Name	Register Name	Abbreviation	R/W	Value after reset	Address	Access Size
FLASH	Product name storage register (1)	PRDNAME1	R	See **Table 31.14**	FFCD 00D0$_H$	32
FLASH	Product name storage register (2)	PRDNAME2	R	See **Table 31.14**	FFCD 00D4$_H$	32
FLASH	Product name storage register (3)	PRDNAME3	R	See **Table 31.14**	FFCD 00D8$_H$	32
FLASH	Product name storage register (4)	PRDNAME4	R	See **Table 31.14**	FFCD 00DC$_H$	32

Table 31.14 Relationship between Product Name and PRDNAME Initial Value (Value after Reset)

Product Name	PRDNAME4	PRDNAME3	PRDNAME2	PRDNAME1
R7F701270	2020 2020$_H$	2020 2030$_H$	3732 3130$_H$	3746 3752$_H$
R7F701271	2020 2020$_H$	2020 2031$_H$	3732 3130$_H$	3746 3752$_H$

31.9.3.1 PRDNAMEn[31:16] (n = 1 to 4) — Product Name Storage Register

This register stores the product name. The product model name is stored in 16-byte ASCII code, and PRDNAME1, PRDNAME2, PRDNAME3, and PRDNAME4 correspond to the fourth to first bytes, eighth to fifth bytes, twelfth to ninth bytes, and sixteenth to thirteenth bytes of the product model name respectively.

Access: This register can be read in 32-bit units.

Address: PRDNAME1: FFCD 00D0$_H$
PRDNAME2: FFCD 00D4$_H$
PRDNAME3: FFCD 00D8$_H$
PRDNAME4: FFCD 00DC$_H$

Value after reset:

C1x product name	Initial value (PRDNAMEn, n = 1 to 4)
R7F701270	37463752 37323130 20202030 20202020$_H$
R7F701271	37463752 37323130 20202031 20202020$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRDNAMEn															
Value after reset *1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRDNAMEn															
Value after reset *1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. This is the value indicated in **Table 31.14, Relationship between Product Name and PRDNAME Initial Value (Value after Reset)**.

Table 31.15 List of Registers Related to Product Information

Bit Position	Bit Name	Function
31 to 0	PRDNAMEn[31:0]	Product name: Indicate product model names by using 16-byte ASCII code. PRDNAME1[31:0]: Fourth to first bytes of product model name PRDNAME2[31:0]: Eighth to fifth bytes of product model name PRDNAME3[31:0]: Twelfth to ninth bytes of the product model name PRDNAME4[31:0]: Sixteenth to thirteenth bytes of the product model name

31.10 Option Bytes

The flash memory has the extended area (option bytes) to store data specified by the user for various purposes. Changes in settings such as initial setting of peripheral functions using option bytes become effective after release from the reset state. For how to set and read the option bytes, refer to the *PG-FP5 Flash Memory Programmer User's Manual* and *Renesas Flash Programmer Flash Programming Software User's Manual*, or the *RH850/C1x Flash Memory User's Manual: Hardware Interface*.

Table 31.16 Setting Area of Option Bytes

Option Byte Area (each 8 bits × 32 = Total 256 bits)	Setting Effective Area	Initial State of Shipped Product*1
Option bytes 4 to 1 ((OPBT0)	Enable	7FFF FFFE$_H$
Option bytes 8 to 5 (OPBT1)	Reserved	FFFF FFFF$_H$
Option bytes 12 to 9 (OPBT2)	Enable	FFFF FFFF$_H$
Option bytes 16 to 13 (OPBT3)	Reserved	FFFF FFFF$_H$
Option bytes 20 to 17 (OPBT4)	Reserved	FFFF FFFF$_H$
Option bytes 24 to 21 (OPBT5)	Reserved	FFFF FFFF$_H$
Option bytes 28 to 25 (OPBT6)	Reserved	FFFF FFFF$_H$
Option bytes 32 to 29 (OPBT7)	Reserved	FFFF FFFF$_H$

Note 1. This is the initial value of the shipped product. The value can be modified by setting the option bytes.

31.10.1 OPBT0 — Option Byte 0 Register

Access: 32-bit read and write accesses are available only when the on-chip debug function is used.

Address: FFCD 0030_H

Value after reset: Specified by the user (7FFF $FFFE_H$ at shipment)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPWD RUN	OPWD INT	OPWD WS1	OPWD WS0	OPWD OVF2	OPWD OVF1	OPWD OVF0	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AUDR EN	—	—	STM SEL1	STM SEL0
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. Only when an FACI command is used and when the PG-FP5 flash memory programmer and Renesas flash programmer flash programming software are used, reading and setting (R/W) are possible.

Table 31.17 OPBT0 Register Contents (1/2)

Bit Position	Bit Name	Function
31	OPWDRUN	This bit selects the start mode of WDTA0. 0: WDTA0 software trigger start mode 1: WDTA0 default start mode
30	OPWDINT	This bit enables or disables a 75% interrupt request of WDTA0 (WDTA0TIT). 0: WDTA0TIT is disabled. 1: WDTA0TIT is enabled.
29, 28	OPWDWS1, OPWDWS0	These bits select the window-open period of WDTA0.

OPWDWS1	OPWDWS0	Window-open period
0	0	25%
0	1	50%
1	0	75%
1	1	100%

Bit Position	Bit Name	Function
27 to 25	OPWDOVF2 to OPWDOVF0	These bits select the overflow interval time of WDTA0.

OPWDOVF2	OPWDOVF1	OPWDOVF0	Overflow interval time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

Bit Position	Bit Name	Function
24 to 5	—	Reserved (The write value should be 1.)
4	AUDREN	AUDRAM monitor enable 0: AUDRAM monitor is disabled. 1: AUDRAM monitor is enabled.
3, 2	—	Reserved (The write value should be 1.)

Table 31.17 OPBT0 Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	STMSEL1, STMSEL0	These bits select operating mode and startup area. When all of the MD0, MD1, and FLMODE pins are 0, the following operating mode and startup area are selected depending on the combination of the values of STMSEL1 and STMSEL0. For details, see **Section 5, Operating Mode**.

STMSEL1	STMSEL0	Operating mode	Startup area
0	0	User boot mode	User area
0	1	User boot mode	User boot area
1	x	Serial programming mode	Boot area

Note: x: Don’t care.

31.10.2 OPBT2 — Option Byte 2 Register

Access: 32-bit read and write accesses are available only when the on-chip debug function is used.

Address: FFCD 0038_H

Value after reset: Specified by the user (FFFF $FFFF_H$ at shipment)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	OPJTAG1	OPJTAG0	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

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

Note 1. Only when an FACI command is used and when the PG-FP5 flash memory programmer and Renesas flash programmer flash programming software are used, reading and setting (R/W) are possible.

Table 31.18 OPBT2 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved (The write value should be 1.)
30, 29	OPJTAG1, OPJTAG0	Switch the debug interfaces. The following debug interface is selected depending on the combination of the values of OPJTAG1 and OPJTAG0.
28 to 0	—	Reserved (The write value should be 1.)

OPJTAG1	OPJTAG0	Debug Interface
0	0	FLSCI3 (writer interface)
0	1	LPD (4-pin)
1	0	Setting prohibited
1	1	Nexus (JTAG)

31.11 Notes

(1) Reading areas where programming or erasure was interrupted

When programming or erasure of an area of flash memory is interrupted, the data stored in the area become undefined. To avoid undefined data that are read out becoming the source of faulty operation, take care not to fetch instructions or read data from areas where programming or erasure was interrupted.

(2) Reading the code flash memory that has been erased but not yet been programming again

Note that reading from an area of code flash memory that has been erased but not yet been programming again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. Use blank checking when you need to confirm that an area is in the non-programmed state.

(3) Prohibition of additional writing

Writing to a given area twice is not possible. If you want to overwrite data in an area of flash memory after writing to the area has been completed, erase the area first.

(4) Resets during programming and erasure

In the case of an external reset during programming and erasure, wait for at least the minimum width of the reset pulse once the operating voltage is within the range stipulated in the electrical characteristics after assertion of the reset signal before releasing the device from the reset state.

(5) Allocation of vectors for interrupts and other exceptions during programming and erasure

For generation of an interrupt or other exception during programming/erasure on the code flash memory or FCU firm transfer, assign the interrupt handler address table*1 and the exception handler to the space in which instructions can be fetched, other than the code flash memory, or mask interrupts and other exceptions in advance.

Note 1. It is applicable when the table reference method is used as the interrupt handler address selection method. For details, see *the RH850G3M User's Manual: Software for RH850G3M*.

(6) Abnormal termination of programming and erasure

Even if programming/erasure ends abnormally due to the generation of an external reset or power cutoff, the programming or erasure state of the flash memory with undefined data cannot be verified or checked. For the area where programming or erasure ends abnormally, the blank check function cannot judge whether the area is erased successfully or not. Erase the area again to prove that the corresponding area is completely erased before using.

If programming and erasure of code flash memory are not completed normally, the lock bit for the target area may be enabled (locked). In such cases, erase the block to erase the lock bit while the lock bit is in the disabled state (the area is not locked).

(7) Items prohibited during programming and erasure

Do not perform the following operations during programming and erasure.

- Have the operating voltage from the power supply go beyond the allowed range.
- Update the values of FHVE15 and FHVE3.
- Change the operating frequency of the peripheral clock.

(8) Flash memory commands prohibited before completion of clock gear-up sequence

Various commands during serial programming and self-programming must be executed after clock gear-up sequence is complete. For details of clock gear-up sequence, see **Section 10, Clock Controller**.

(9) Securing coherency of instruction cache and data buffer

After the code flash memory is programmed or erased, a reset or instruction cache and data buffer clearing are required to secure coherency of the instruction cache and data buffer. For details of instruction cache and data buffer, see **Section 3, CPU System**.

Section 32 Flash Security

To protect the code flash memory, data flash memory, and ID codes, this product has the functions described in **Section 31, Flash Memory**, and functions to restrict connection of the debug interface, which are described in this section.

For detailed descriptions of the security functions in serial programming mode and how to program the flash memory, see **Section 31, Flash Memory**.

In this section, the interfaces that may be used in on-chip debugging (the Nexus and LPD 4 pins) are referred to as “debug interfaces”. In addition, ID authentication to protect the code flash memory, data flash memory, and ID codes in user boot mode is referred to as “SELF ID authentication”, and ID authentication to protect against access through the on-chip debug functions is referred to as “OCD ID authentication”.

The data length of each ID code during SELF ID and OCD ID authentication as well as the ID code in the ID authentication stage of serial programming is 128 bits. In the initial state of the product at shipment, the ID codes are FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF$_H$.

32.1 Features

32.1.1 Protection of Code Flash Memory, Data Flash Memory, and ID Codes

This microcontroller includes the following security functions in user boot mode and serial programming mode to prevent any leaking of the user program programmed in the code flash memory.

32.1.1.1 Functions Unique to User Boot Mode

SELF ID authentication protects against programming and erasure of code flash memory and reading of ID code.

(1) Security States

This mode has two security states, protection locked and protection unlocked. Transition between these two states can be made by changing SELF ID authentication and the ID codes.

1. Protection unlocked
   Security functions have been unlocked by SELF ID authentication so that protection against programing and erasure of code flash memory and reading of ID codes is released.
2. Protection locked
   Security functions have been enabled by SELF ID authentication and protection against programing and erasure of code flash memory and ID code reading is in place.

32.1.1.2 Functions Unique to Serial Programming Mode

Three functions are provided as security functions unique to serial programming mode: ID authentication, prohibition of programming, erasure, and read commands, and prohibition of serial programmer connection. Parallel use of these functions is not allowed.

(1) ID Authentication

ID codes are checked for authenticity to protect the code flash memory and data flash memory. Programming, erasure, and reading of the code flash and data flash memory can proceed upon successful ID authentication.

(2) Prohibition of Programming, Erasure, and Read Commands

Issuing of programming, erasure, and read commands can be enabled or disabled individually for the code flash memory and data flash memory. All commands are permitted for both areas in the initial state of the product as shipped.

(3) Prohibition of Serial Programmer Connection

Issuing commands for programming, erasure, or reading to the code flash memory and data flash memory can be disabled in serial programming mode. Transitions to other functions are not possible once this state is set.

32.1.1.3 Common Function of User Boot Mode and Serial Programming Mode

(1) OTP (One Time Programming)

Setting an area of the code flash memory as OTP protects it against any further programming and erasure. For details of the setting, see **Section 31, Flash Memory**.

32.1.2 Connection Restriction Function of Debug Interface

This product is provided with unauthorized access protection via debug interfaces and two security levels are available.

- Security level 1:
  Debug interfaces can be used. At this level, the on-chip debugging function is protected by OCD ID authentication. For OCD to be used, it must be unlocked by using OCD ID authentication.
- Security level 2:
  At this level, the debug interfaces cannot be used.

As described above, restrictions on the connection of code flash and data flash memory, ID code protection, and availability of the debug interfaces differ according to the mode. **Table 32.1** summarizes security functions in each mode.

Table 32.1 Security Functions in Each Mode

Operation Mode	Code Flash and Data Flash, ID Code Protection	Restriction on Debug Interface Connection
User boot mode	• SELF ID authentication • OTP (parallel use possible)	• Security level 1 (OCD ID authentication) • Security level 2 (Debug interface connection is prohibited)
Serial programming mode	• ID authentication • Programming commands, block erasure commands, and read commands are prohibited. • Connection of serial programmers is prohibited. (The above three cannot be used in parallel.) • OTP (parallel use possible)	• No function (Debug interface connection is always prohibited.)

32.2 Security in User Boot Mode

32.2.1 SELF ID Authentication

To prevent leakage of the user program written to the code flash memory, this product has security functions to enable or disable the programming and erasure of code flash memory and the reading of ID codes. The setting of enabling or disabling of these functions can be switched through SELF ID authentication with the ID code the user has set as the expected value.

32.2.2 SELF ID Authentication and Security State

Table 32.2 and **Figure 32.1** show SELF ID authentication security state and conditions for transition.

Table 32.2 Security Setting State

State	SELF ID Authentication	Protection State of Code Flash Programming and Erasure and of Reading of ID Codes
Protection unlocked	Unlocked	Not protected
Protection locked	Locked	Protected

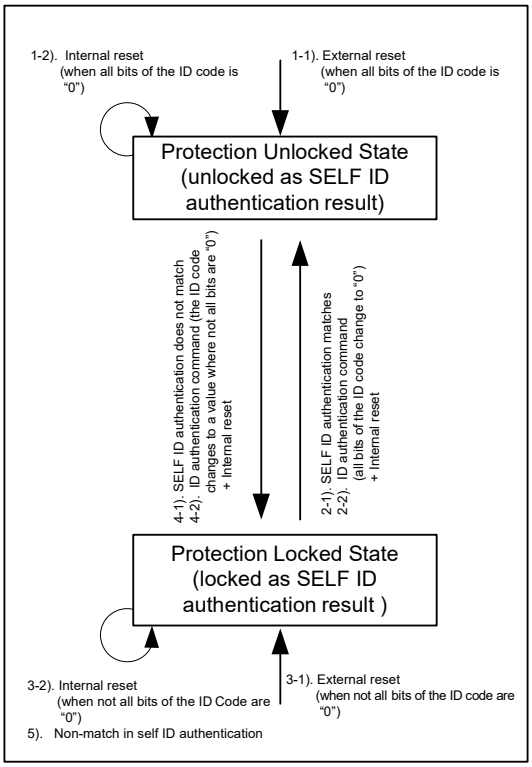


Figure 32.1 Security Setting State by SELF ID Authentication and Transition

The conditions for the transitions of security states shown in **Figure 32.1** are as follows.

- Conditions for Transitions to the Protection Unlocked State

1) Startup when protection is released

1-1) When an external reset is applied while the ID code is all 0s, startup proceeds with protection released.

1-2) When an internal reset is applied without modification of the ID code while all bits of the ID code are 0, startup proceeds with protection in place.

2). Transition from Protection Locked to Protection Released

2-1) A transition to the protection unlocked state follows a match in SELF ID authentication matches.

2-2) When an internal reset is applied after changing the ID code has been changed from other than “all bits 0” to “all bits 0”, startup proceeds with protection released.

- Conditions for Transitions to Protection Locked

3) Startup with protection in place

3-1) When an external reset is applied while the ID code is other than “all bits 0”, startup proceeds with protection in place.

3-2) When an internal reset is applied without modification of the ID code while not all bits of the ID code are 0, startup proceeds with protection in place.

4). Transition from protection unlocked to protection locked

4-1) A transition to the protection locked state follows a non-match in SELF ID authentication.

4-2) When an internal reset is applied after the ID code has been changed from “all bits 0” to “other than all bits 0”, startup proceeds with protection in place.

5) To retain protection locked

Protection remains in place in case of a non-match in SELF ID authentication.

32.3 Security Functions in Serial Programming Mode

For details of security functions in serial programming mode, see **Section 31, Flash Memory**.

32.4 Restricting Connection with Debug Interfaces

This product is capable of restricting connection of the debug interfaces to protect against unauthorized access via the debug interfaces. This feature has two security levels.

- Security level 1: Restrict access to OCD functions by using OCD ID authentication
- Security level 2: Prohibit all connection to the debug interfaces

These security levels can be changed by bits 30 and 29 (OPJTAG1 and OPJTAG0) in option byte 2 in the extended area of the flash memory.
In this section, OPJTAG0 and OPJTAG1 are referred to as the OPJTAG bits.

32.4.1 Security Levels and State of Restricting the Connection of Debug Interfaces

Table 32.3 show each security level and the corresponding security states and **Figure 32.2** shows the conditions for transitions.

Table 32.3 Security Levels and State of Restricting Connection to the Debug Interfaces

State	Result of OCD ID Authentication	OPJTAG Bit *1	Restriction on Debug Interface Connection
Security level 1	Unlocked	Other than 00_B	No restriction
	Locked	Other than 00_B	Restriction on access via the debug interfaces is in place.
Security level 2	—	00_B	Connection to the debug interfaces is prohibited.

Note 1. For a detailed description of the OPJTAG bits, see **Section 31, Flash Memory**.

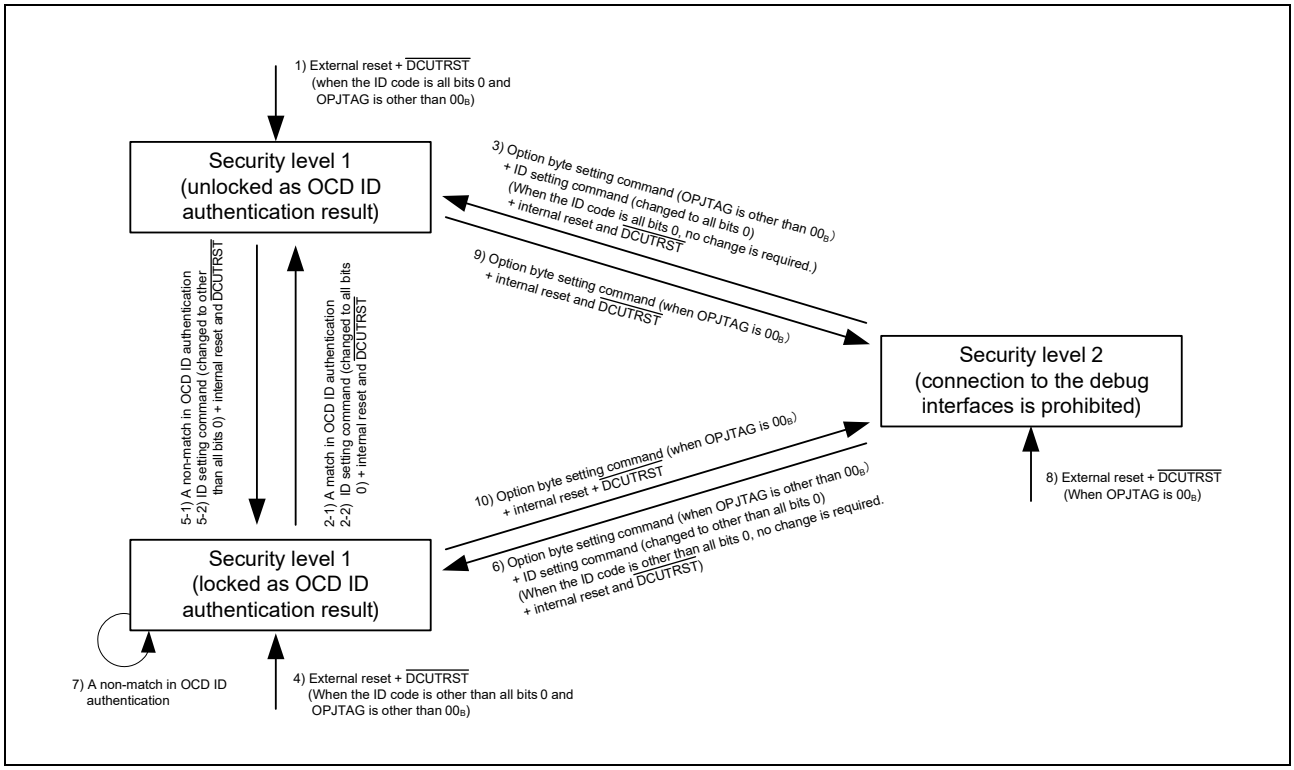


Figure 32.2 Transitions of Security Level

Conditions for transitions to each security level described in **Figure 32.2**.

- Conditions for transitions to security level 1 (unlocked as OCD ID authentication result)

  1) Startup in the state of security level 1 (unlocked as OCD ID authentication result)
  When an external reset and a reset by $\overline{\text{DCUTRST}}$ are applied while the ID code is all 0s and while the value of the OPJTAG is other than 00_B, startup proceeds with security level 1 (unlocked as OCD ID authentication result).

  2) Transition from security level 1 (locked as OCD ID authentication result) to security level 1 (unlocked as OCD ID authentication result)

    2-1) Transition to security level 1 (unlocked as OCD ID authentication result) follows a match in OCD ID authentication.

    2-2) When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “all bits 0”, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

  3) Transition from security level 2 to security level 1 (unlocked as OCD ID authentication result)
  When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to “all bits 0” and the value of the OPJTAG to other than 00_B, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

- Conditions for Transitions to Security Level 1 (locked as OCD ID Authentication result)

  4) Startup in the state of security level 1 (locked as OCD ID authentication result)
  When an external reset or a reset by $\overline{\text{DCUTRST}}$ is applied while the ID code is “other than all bits 0” and the value of the OPJTAG is other than 00_B, startup proceeds at security level 1 (locked as OCD ID authentication result).

5) Transition from security level 1 (unlocked as OCD ID authentication result) to security level 1 (locked as OCD ID authentication result)

5-1) A transition to security level 1 (locked as OCD ID authentication result) follows a non-match in OCD ID authentication.

5-2) When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to "other than all bits 0", startup proceeds at security level 1 (locked as OCD ID authentication result).

6) Transition from security level 2 to security level 1 (locked as OCD ID authentication result)
When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the ID code to "other than all bits 0" and the value of OPJTAG is other than 00_B, startup proceeds at security level 1 (locked as OCD ID authentication result).

7) Retaining security level 1 (locked as OCD ID authentication result)
In case of a non-match in OCD ID authentication, security level 1 (locked as OCD ID authentication result) remains in place.

- Transition to security level 2

8) Startup at security level 2
When an external reset or a reset by $\overline{\text{DCUTRST}}$ is applied while the value of OPJTAG is 00_B, startup proceeds at security level 2.

9) Transition from security level 1 (unlocked as OCD ID authentication result) to security level 2
When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

10) Transition from security level 1 (locked as OCD ID authentication result) to security level 2
When an internal reset or a reset by $\overline{\text{DCUTRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

Section 33 RAM

33.1 List of On-Chip RAM

RH850/C1x includes the following RAM.

- Local RAM (CPU1): 64 Kbytes
- Local RAM (CPU2): 64 Kbytes
- Global RAM: max. 112 Kbytes*1

Note 1. Refer to **Table 1.1, Overview of Products** for including size in each product.

33.2 Features

Access:

CPU1, CPU2 and DMAC can access the local RAM (CPU1, CPU2) and the global RAM.

CPU1 and CPU2 have the same latency for access to their respective areas of local RAM.

For details of address map and access availability, see **Section 4, Address Space**.

ECC:

The local RAM (CPU1), local RAM (CPU2), and global RAM include the ECC and address parity.

For details, see **Section 27, Functional Safety**.

33.3 Notes

(1) The local RAM and global RAM must be initialized with the maximum bit length of its access size before using the RAM.
If the RAM is accessed before its initialization, an ECC error may be detected. Also if initialization with the maximum bit length is not performed (e.g. if a 32-bit RAM is initialized by an 8-bit or 16-bit access), an ECC error may be detected.

(2) On path between Local RAM and CPU described follows, buffers are implemented to realize fast Local RAM access.

- Local RAM (CPU1), Local RAM (CPU2)

  When a load instruction is executed from the same address after a store instruction to Local RAM, the load instruction may read out data from buffers instead of data on Local RAM. Either of following procedures can be used to surely read data on Local RAM.

1. To read out the first written data after writing more than 32-byte data into Local RAM.
2. To execute SYNCM instruction before a load instruction is executed from the same address after a store instruction to Local RAM.

Section 34 Boundary Scan

This LSI has the JTAG interface and provides the boundary scan function conforming to the IEEE1149.1 standard.

Boundary Scan function is included only in BGA252.

34.1 Features

- Five test signals (DCUTCK, DCUTDI, DCUTDO, DCUTMS, and $\overline{\text{DCUTRST}}$)
- TAP controller
- Instruction register
- Bypass register
- Boundary scan register

The JTAG interface has six commands.

- BYPASS mode
  Test mode conforming to the IEEE 1149.1
- EXTEST mode
  Test mode conforming to the IEEE 1149.1
- SAMPLE/PRELOAD mode
  Test mode conforming to the IEEE 1149.1
- CLAMP mode
  Test mode conforming to the IEEE 1149.1
- HIGHZ mode
  Test mode conforming to the IEEE 1149.1
- IDCODE mode
  Test mode conforming to the IEEE 1149.1

Figure 34.1 shows a block diagram of the JTAG interface.

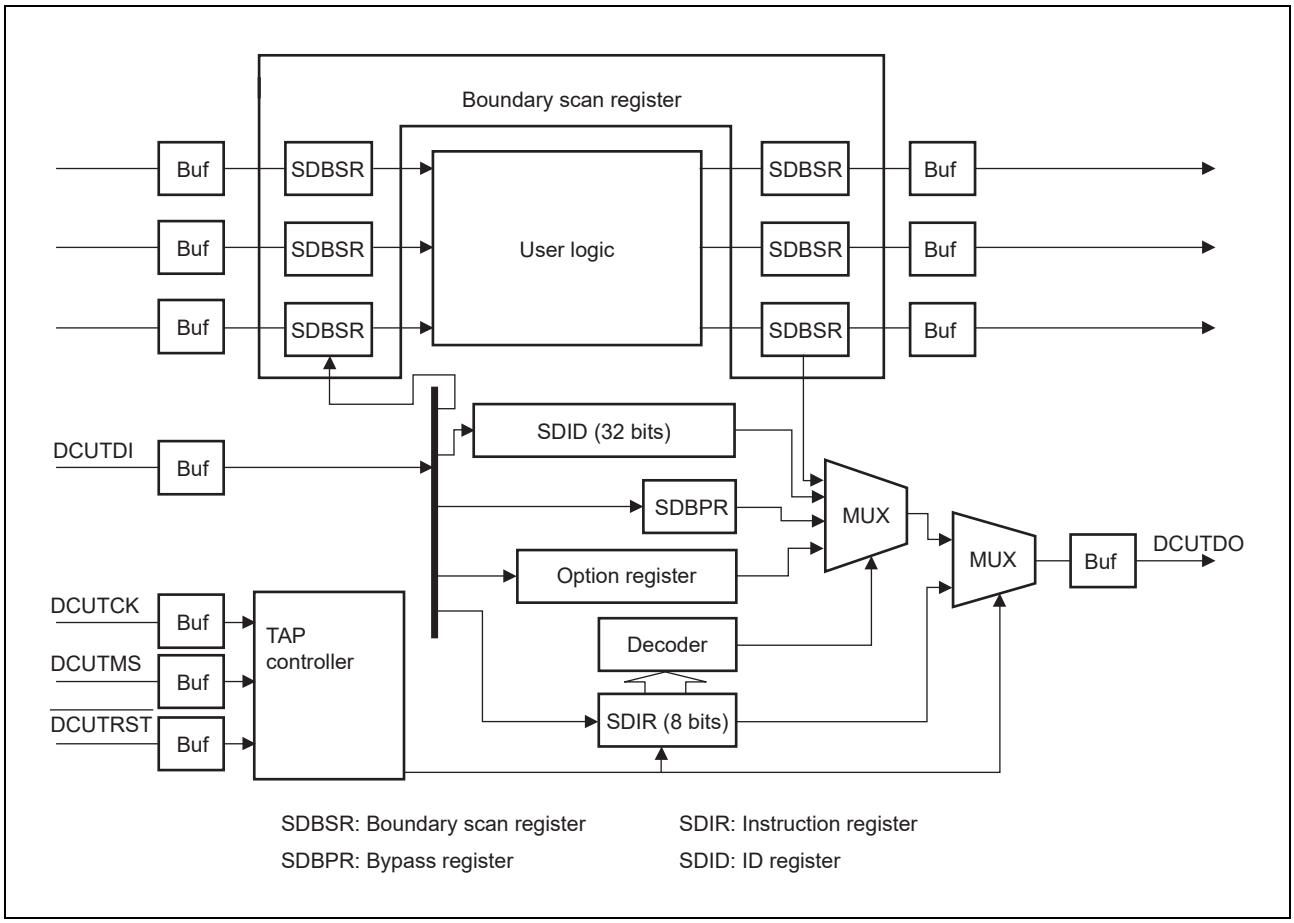


Figure 34.1 Block Diagram of JTAG Interface

34.2 Input/Output Pins

There are five JTAG control signals: DCUTCK, DCUTDI, DCUTMS, DCUTDO, and $\overline{\text{DCUTRST}}$.

Table 34.1 shows the pin configuration.

Table 34.1 Pin Configuration

Pin Name	Description
DCUTCK	Serial data input/output Clock pin Data is supplied via the data input pin (DCUTDI) and is output from the data output pin (DCUTDO) in synchronization with this clock signal.
DCUTMS	Mode select input pin Changing the level of this signal in synchronization with DCUTCK changes the state of the TAP controller. For the protocol, refer to **Figure 34.2, TAP Controller State Transition Diagram**.
$\overline{\text{DCUTRST}}$	Reset input pin A low-level input of this signal, which is accepted asynchronously with DCUTCK, resets the JTAG interface. Even if the JTAG interface is not used, $\overline{\text{DCUTRST}}$ must be held low for the specified time at a power on.
DCUTDI	Serial data input pin Changing the state of this pin in synchronization with DCUTCK allows data to be sent to the JTAG interface.
DCUTDO	Serial data output pin Reading the state of this pin in synchronization with DCUTCK allows data to be read from the JTAG interface.

34.3 Register Descriptions

The JTAG interface has the following registers. None of the registers can be accessed by the CPU.

- SDIR: Instruction register
- SDID: ID register
- SDBPR: Bypass register
- SDBSR: Boundary scan register

Table 34.2 Register Configuration

Register Name	Symbol	Access Size	Initial Value*1
Instruction register	SDIR	8	55_H
ID register	SDID	32	*2
Bypass register	SDBPR	1	Undefined
Boundary scan register	SDBSR	—	Undefined

Note 1. Registers are initialized when $\overline{\text{DCUTRST}}$ pin is 0 or when TAP is in the Test-Logic-Reset state.

Note 2. The initial value of the ID register is as follows.

C1x product name	Value after reset
R7F701270	1830D447

Commands can be serially transferred from the serial data input pin (DCUTDI) and input to the instruction register (SDIR). The bypass register (SDBPR) is a 1-bit register, to which DCUTDI and DCUTDO are connected in BYPASS mode, CLAMP mode, and HIGHZ mode. The boundary scan register (SDBSR) is connected DCUTDI and DCUTDO in SAMPLE/PRELOAD mode and EXTEST mode. The ID code register (SDID) is a 32-bit register, from which the ID code is output via DCUTDO in IDCODE mode.

Table 34.3 shows the serial transfer types possible with the JTAG interface registers.

Table 34.3 Serial Transfer Types Possible with JTAG Interface Registers

Register	Serial Input	Serial Output
SDIR	Possible	Impossible*1
SDBPR	Possible	Possible
SDBSR	Possible	Possible
SDID	Impossible	Possible

Note 1. A fixed value is read out.

34.3.1 Instruction Register (SDIR)

SDIR is an 8-bit register that holds a boundary scan command. SDIR is initialized by asserting $\overline{\text{DCUTRST}}$ or in the TAP Test-Logic-Reset state. Operation is not guaranteed when any reserved command is set in this register.

Table 34.4 Boundary Scan Commands

IR Code								Description
0	0	0	0	0	0	0	0	JTAG EXTEST
0	1	0	0	0	0	0	0	JTAG SAMPLE/PRELOAD
1	1	0	1	0	0	0	0	JTAG CLAMP
1	0	0	0	0	0	0	0	JTAG HIGHZ
0	1	0	1	0	1	0	1	JTAG IDCODE (value after reset)
1	1	1	1	1	1	1	1	JTAG BYPASS
Other than above								Reserved

34.3.2 SDID — ID Register

SDID is a 32-bit register that indicates an LSI-specific ID.

SDID can be read via the JTAG interface pin when the IDCODE command is set but cannot be written to.

For the read values, refer to **Table 34.2, Register Configuration**.

34.3.3 SDBPR — Bypass Register

SDBPR is a 1-bit register. When SDIR is set to BYPASS mode, SDBPR is connected between DCUTDI and DCUTDO. The value after reset is undefined. SDBPR is not initialized by a power-on reset or asserted $\overline{\text{DCUTRST}}$.

34.3.4 SDBSR — Boundary Scan Register

SDBSR is a shift register located on the PADs for controlling the external I/O pins. When SDIR is set to SAMPLE/PRELOAD or EXTEST mode, SDBSR is connected between DCUTDI and DCUTDO. The value after reset is undefined. SDBSR is not initialized by a power-on reset or asserted $\overline{\text{DCUTRST}}$.

34.4 Operation

34.4.1 TAP Controller

Figure 34.2 shows the TAP controller internal states.

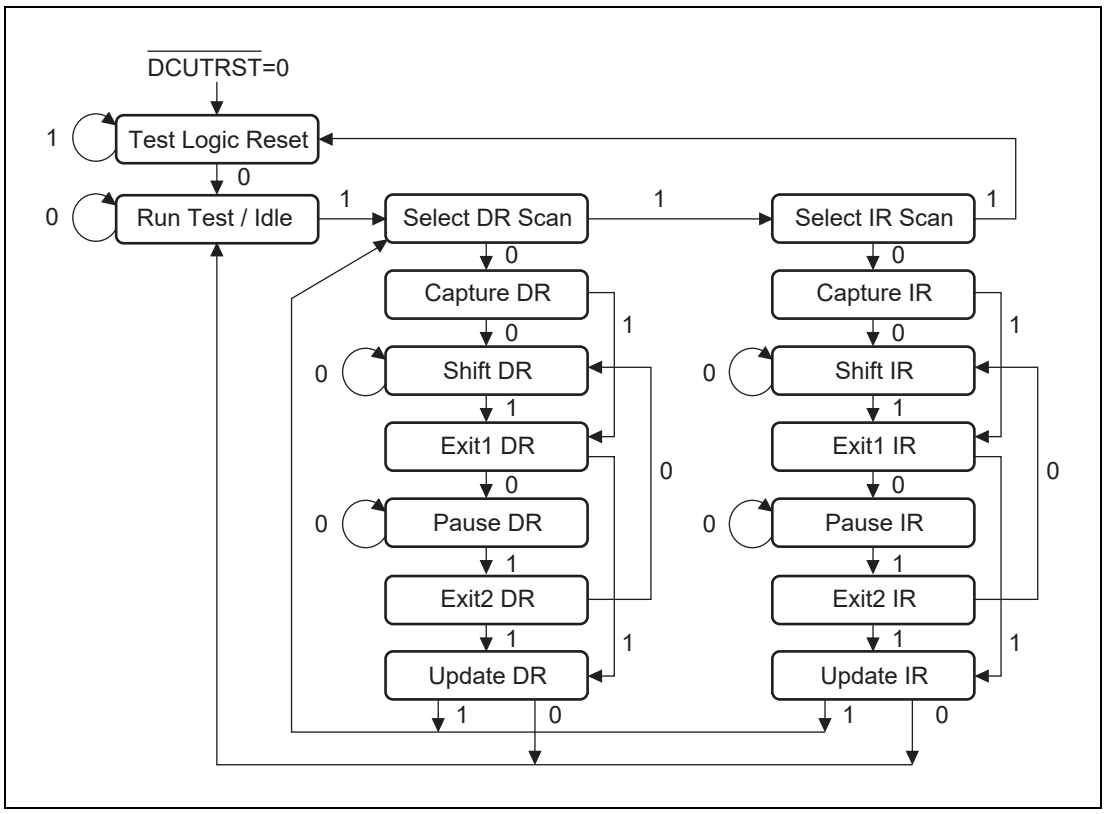


Figure 34.2 TAP Controller State Transition Diagram

Note 1. Transition is made according to the DCUTMS value at the rising edge of DCUTCK. The DCUTDI value is sampled at the rising edge of DCUTCK and is shifted at the falling edge. DCUTDO is in the high-impedance state in the states other than Shift-DR and Shift-IR. Asserting $\overline{\text{DCUTRST}}$ causes transition to Test-Logic-Reset state asynchronously with DCUTCK.

34.4.2 Supported Commands

34.4.2.1 BYPASS

The BYPASS command is a standard command indispensable to bypass register operation. This command shortens the shift path to achieve high-speed serial data transfer of other LSIs on the printed-circuit board. During execution of this command, the test circuit has no effect on the system circuit.

34.4.2.2 SAMPLE/PRELOAD

The SAMPLE/PRELOAD command is used to input the value to the boundary scan register from the internal circuits of this LSI; to output the value from the scan path; and to load data onto the scan path. During execution of this command, the level of the input pin of this LSI is sent to the internal circuits as is, and the value of the internal circuits is output to the outside via the output pin as is. Executing this command has no effect on the system circuit of this LSI.

The SAMPLE operation allows taking in the snapshots of the value to be transferred to the internal circuits from the input pin or the value to be transferred to the output pin from the internal circuits to the boundary scan register and allows reading the snapshots from the scan path. Snapshots can be taken in without preventing the normal operation of this LSI.

The PRELOAD operation allows setting the value after reset to the parallel output latch of the boundary scan register from the scan path prior to the EXTEST command. If the EXTEST command is executed without PRELOAD operation, the undefined value is output from the output pin until the first scan sequence is completed (transfer to the output latch) because the parallel output latch value is always output to the output pin with the EXTEST command.

34.4.2.3 EXTEST

The EXTEST command is used to test the external circuits when this LSI is mounted on the printed-circuit board. When this command is executed, the output pin is used to output the test data (previously set with the SAMPLE/PRELOAD command) from the boundary scan register to the printed-circuit board; whereas the input pin is used to take in the test result from the printed-circuit board to the boundary scan register. When the EXTEST command is executed N times for testing, the test data for the Nth execution is scanned in at the (N - 1)th scan-out.

If the data is loaded onto the boundary scan register of the output pin in the Capture-DR state of this command, it is not used for testing the external circuits (replaced through shift operation).

34.4.2.4 CLAMP

When the CLAMP command is selected, the output pin outputs the boundary scan register value that has been previously set with the SAMPLE/PRELOAD command. While the CLAMP command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between DCUTDI and DCUTDO and operates in the same manner as when the BYPASS command is selected.

34.4.2.5 HIGHZ

When the HIGHZ command is selected, all the output pins go to the high-impedance state. While the HIGHZ command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between DCUTDI and DCUTDO and operates in the same manner as when the BYPASS command is selected.

34.4.2.6 IDCODE

The IDCODE command sets the JTAG interface pins to IDCODE mode, which is defined by the JTAG standard. When the JTAG interface is initialized (by asserting $\overline{\text{DCUTRST}}$ or placing TAP in the Test-Logic-Reset state), IDCODE mode is set.

34.4.3 Notes

There are restrictions related to the JTAG interface as follows.

- The power supply/GND pins are not subjected to boundary scan.
- The reference voltage pins of the A/D converter (A0VREFH, and A1VREFH) are not subjected to boundary scan.
- The EPT control pin (EPTVOUT) is not subjected to boundary scan.
- The NC pins are not subjected to boundary scan.
- **Table 34.5** lists the pins not subjected to boundary scan.

Table 34.5 Pins not Subjected to Boundary Scan

Type	Pins
Analog input	ADCC000/RDC20SINMNT, ADCC001, ADCC002, ADCC003/RDC20COSMNT, ADCC010, ADCC011, ADCC012, ADCC013, ADCC020/RDC21SINMNT, ADCC021/RDC21COSMNT, ADCC022, ADCC023, ADCC030, ADCC031, ADCC032, ADCC033, ADCC100, ADCC101, ADCC102, ADCC110, ADCC111, ADCC112, ADCC120, ADCC121, ADCC122, ADCC130, ADCC131, ADCC132, ADCC140, ADCC141, ADCC142, ADCC150, ADCC151, ADCC152, ADCC160, ADCC161, ADCC162, RDC20COM, RDC21COM, RDC20RSO, RDC21RSO, RDC20S1, RDC20S2, RDC20S3, RDC20S4, RDC21S1, RDC21S2, RDC21S3, RDC21S4
Debug system	AUDCK, $\overline{\text{DCURDY}}$, DCUTCK, DCUTDI, DCUTDO, DCUTMS, $\overline{\text{DCUTRST}}$
Mode setting	FLMODE, MD0, MD1
Clock, reset	X1, X2, $\overline{\text{RESET}}$
Error output	$\overline{\text{ERROROUT\_M}}$

- The HIGHZ command is invalid for the pulled-down pins.

34.5 Usage Notes

1. Once a command is set, it is not modified until another command is issued again. To continuously issue the same commands, insert a command that has no effect on chip operation (such as BYPASS mode) between the desired commands.
2. To start the system in boundary scan mode, negate $\overline{\text{DCUTRST}}$ while $\overline{\text{RESET}}$ is high.
3. For the maximum clock frequency that can be input to DCUTCK, refer to **Section 35, Electrical Characteristics**.
4. If the number of serially transferred bits exceeds the number of bits of the register connected between DCUTDI and DCUTDO, the data that is input from DCUTDI is output from DCUTDO after the serial data equal to the number of register bits are output.
5. If the serial transfer sequence is corrupted, be sure to reset $\overline{\text{DCUTRST}}$. Here, start the transfer over again regardless of the point of transfer corruption.
6. Data is output via DCUTDO at the falling edge of DCUTCK.
7. To facilitate debugging, route $\overline{\text{DCUTRST}}$ on the board in such a way that patterns can be easily cut.

Section 35 Electrical Characteristics

35.1 Absolute Maximum Ratings

Table 35.1 lists absolute maximum ratings.

Table 35.1 Absolute Maximum Ratings

Item		Symbol	Rated Value	Unit	Note
Power voltage*1	PLLVCC, SYSVCC, VCC	VCC	–0.3 to +4.3	V	
	EVCC	EVCC	–0.3 to +6.5	V	
	VDD	VDD	–0.3 to +1.8	V	
Input voltage	SYSVCC power pin	Vin	–0.3 to SYSVCC +0.3	V	See **Table 35.2** for intended pins
	VCC power pin	Vin	–0.3 to VCC +0.3	V	
	EVCC power pin	Vin	–0.3 to EVCC +0.3	V	
	5V-tolerant pin*2	Vin	–0.3 to 5.8	V	
Analog power voltage		A0VCC, A1VCC	–0.3 to +6.5	V	
		RVCC	–0.3 to +6.5	V	
Analog reference voltage		A0VREFH	–0.3 to A0VCC +0.3	V	
		A1VREFH	–0.3 to A1VCC +0.3	V	
Analog input voltage		V_{AIN}	–0.3 to A0VCC +0.3 –0.3 to A1VCC +0.3	V	
		V_{RIN}	–0.3 to RVCC +0.3	V	
VSS differential voltage*3 (Condition: Between any two of VSS, A0VSS, A1VSS, RVSS, and PLLVSS)			–0.1 to +0.1	V	
Maximum input current (per pin)	Digital input pin	Imax	–25 to +25	mA	Only 1 pin simultaneously
	Analog input pin	Imax	–25 to +25	mA	Only 1 pin simultaneously
Junction temperature*1		Tj	–40 to +150	°C	
Storage temperature		Tstg	–55 to +150	°C	After installation

Note 1. Cumulative hours of operation of this LSI with Tj in the range from 125°C to 150°C must be kept within 3000 hours.

Note 2. Pins below described as “(5 V tol.)” in **Table 35.2, Relationship between Power Name and Pin**.
FLMODE, MD0, MD1,RESET, DCUTRST, LPDTRST

Note 3. Of the digital power supplies on the board, short-circuit PLLVSS to VSS.

NOTE

Using this LSI without observing these absolute maximum ratings may result in permanent breakdown of the LSI.

This product is used in combination of multiple power voltages simultaneously in some cases. Use this LSI conforming to power pin connections, conditions for combination of power voltages to be applied, voltages that can be applied to pins, and output voltage conditions, which are specified in the manual. Using this LSI with unspecified power connection or voltage may result in permanent breakdown of the LSI or damage to the system that contains this LSI.

Input voltage, analog reference voltage and analog input voltage must not exceed 6.5 V.

35.2 DC Characteristics

35.2.1 Relationship between Power Name and Pin

Table 35.2 shows the relationship between power name and pin.

Table 35.2 Relationship between Power Name and Pin

Pin Name (Initial Value)	Circuit Power Name	I/O	Input Buffer Type	Note
Px_x	EVCC	I/O	Schmitt B	Variable driving ability
ADCC0Ixx	A0VCC	I[*1]	Analog (ADC)	
ADCC1Ixx	A1VCC	I	Analog (ADC)	
RDC2nSx	RVCC	I	Analog (RDC)	
RDC2nRSO	RVCC	I/O	Analog (RDC)	
RDC2nCOM	RVCC	I/O	Analog (RDC)	
$\overline{\text{RESET}}$	SYSVCC	I	Schmitt A	5 V tolerant
FLMODE	SYSVCC	I	Schmitt A	5 V tolerant
MD0	SYSVCC	I	Schmitt A	5 V tolerant
MD1	SYSVCC	I	Schmitt A	5 V tolerant
NMI	SYSVCC	I	Schmitt A	
$\overline{\text{ERROROUT\_M}}$	EVCC	O	—	
X1	VCC	I	CMOS	
X2	VCC	O	—	
$\overline{\text{AUDRST}}$	VCC	I	Schmitt A	
AUDCK	VCC	I	TTL2	
$\overline{\text{AUDSYNC}}$	VCC	I	TTL2	
AUDATAx	VCC	I/O	TTL2	
$\overline{\text{DCUTRST}}$/$\overline{\text{LPDTRST}}$	SYSVCC	I	Schmitt A	5 V tolerant
DCUTDO/LPDO	VCC	O	—	
DCUTMS	VCC	I	TTL2	
DCUTCK/LPDCLK	VCC	I	TTL2	
DCUTDI/LPDI	VCC	I	TTL2	
$\overline{\text{DCURDY}}$/LPDCLKOUT	VCC	O	—	

Note 1. Some pins also serve as RDC2 pins. Following pins are I/O for both functions. ADCC0I00/RDC20SINMNT, ADCC0I03/RDC20COSMNT, ADCC0I20/RDC21SINMNT, ADCC0I21/RDC21COSMNT

35.2.2 Recommended Operating Conditions

Table 35.3 Recommended Operating Conditions

Symbol	Min.	Typ.	Max.	Unit	Note
SYSVCC	3.0	3.3	3.6	V	—
VCC*1	3.0	3.3	3.6	V	—
PLLVCC*1	3.0	3.3	3.6	V	—
VDD*2	1.15	1.25	1.35	V	—
EVCC	4.5	5.0	5.5	V	—
A0VCC, A1VCC*3	4.5	5.0	5.5	V	—
RVCC*3	4.5	5.0	5.5	V	—
A0VREFH, A1VREFH*4	4.5	5.0	5.5	V	—

Note: VSS = A0VSS = A1VSS = PLLVSS = RVSS = 0V must be kept.

Note 1. VCC and PLLVCC must be connected to the same electric potential.

Note 2. This specification must also be followed while the EPT is operating.

Note 3. A0VCC, A1VCC and RVCC must be connected to the same electric potential.

Note 4. Do not exceed A0VCC or A1VCC.

35.2.3 Input Voltage Characteristics

Table 35.4 DC Characteristics (Input Voltage)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Schmitt trigger input voltage (Buffer type A)	SYSVCC VCC	V_T^+ (V_{IH})	SYSVCC × 0.75 VCC × 0.75	—	SYSVCC + 0.3 VCC + 0.3	V	See **Table 35.2** (Item of Schmitt A input buffer type)
		V_T^- (V_{IL})	–0.3	—	SYSVCC × 0.25 VCC × 0.25	V	
		V_{HS}	SYSVCC × 0.2 VCC × 0.2	—	—	V	
Schmitt trigger input voltage (Buffer type B)	EVCC	V_T^+ (V_{IH})	EVCC × 0.7	—	EVCC + 0.3	V	See **Table 35.2** (Item of Schmitt B input buffer type)
		V_T^- (V_{IL})	–0.3	—	EVCC × 0.42	V	
		V_{HS}	EVCC × 0.082	—	—	V	
TTL input voltage	VCC	V_{IH}	2.2	—	VCC + 0.3	V	See **Table 35.2** (Item of TTL2 input buffer type)
		V_{IL}	–0.3	—	0.8	V	
CMOS input voltage	VCC	V_{IH}	VCC × 0.7	—	VCC + 0.3	V	See **Table 35.2** (Item of CMOS input buffer type)
		V_{IL}	–0.3	—	VCC × 0.2	V	

35.2.4 Input Leak Current Characteristics

Table 35.5 DC Characteristics (Input Leak Current)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input leak current	Other than A/D port and R/D port*1	I Iin I	—	—	1	µA	Vin = 0V to EVCC Vin = 0V to SYSVCC Vin = 0V to VCC
	A/D port	I Iin I	—	—	0.1	µA	Vin = 0 V to A0VCC, A1VCC
	R/D port	I Iin I	—	—	0.3	µA	Vin = 0 V to RVCC, and stopping RDC2

Note 1. X1 pin is not intended. Pull-up/pull-down pins are also not intended.

35.2.5 Pull-Up/Pull-Down MOS Current Characteristics

Table 35.6 DC Characteristics (Pull-Up/pull-Down MOS Current)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input pull-up MOS current	DCUTMS, DCUTCK, DCUTDI	I_{pu}	—	—	350	µA	Vin = 0 V VCC = 3.0 to 3.6 V
	AUDCK, $\overline{\text{AUDSYNC}}$, AUDATA3-0*1		—	—	350	µA	Vin = 0 V VCC = 3.0 to 3.6 V
	General port		—	—	350	µA	Vin = 0 V, EVCC = 5.5 V
Input pull-down MOS current	$\overline{\text{RESET}}$	I_{pd}	25	60	120	µA	Vin = SYSVCC = 3.6 V
			5	20	40	µA	Vin = SYSVCC = 2.0 V
	$\overline{\text{DCUTRST}}$		—	—	350	µA	Vin = SYSVCC = 3.0 to 3.6 V
	NMI, FLMODE, MD0, MD1		15	—	120	µA	Vin = SYSVCC = 3.0 to 3.6 V
	$\overline{\text{AUDRST}}$		—	—	350	µA	Vin = VCC = 3.0 to 3.6 V
	General port		—	—	350	µA	Vin = EVCC = 5.5 V

Note 1. The pull-up of AUDATA3-0 is valid not only in input but also in output.

35.2.6 Output Voltage Characteristics

Table 35.7 DC Characteristics (Output Voltage)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output high level voltage	EVCC power supply system pins	V_{OH}	EVCC - 0.5	—	—	V	I_{OH} = 200 µA EVCC = 4.5 to 5.5 V
			EVCC - 1.0	—	—	V	I_{OH} = 1 mA EVCC = 4.5 to 5.5 V
	VCC power supply system pins		VCC - 0.1	—	—	V	I_{OH} = 50 µA VCC = 3.0 to 3.6 V
			VCC - 1.0	—	—	V	I_{OH} = 200 µA VCC = 3.0 to 3.6 V
Output low level voltage	EVCC power supply system pins	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA EVCC = 4.5 to 5.5 V
			—	—	1.2	V	I_{OL} = 4 mA EVCC = 4.5 to 5.5 V
	VCC power supply system pins		—	—	0.1	V	I_{OL} = 50 µA VCC = 3.0 to 3.6 V
			—	—	0.4	V	I_{OL} = 1.6 mA VCC = 3.0 to 3.6 V

35.2.7 Allowable Output Current

Table 35.8 DC Characteristics (Allowable Output Current)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item	Symbol	Min.	Typ.	Max.	Unit
Output low-level allowable current (per pin)	I_{OL}	—	—	4.0	mA
Output low-level allowable current (total)	ΣI_{OL}	—	—	80.0	mA
Output high-level allowable current (per pin)	I_{OH}	—	—	2.0	mA
Output high-level allowable current (total)	ΣI_{OH}	—	—	25.0	mA

This item affects the calorific value and Tj of the chip. In addition to these restrictions, you also need to take thermal design into consideration.

35.2.8 Injection Current

Table 35.9 DC Characteristics (Injection Current)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit
DC injection current (per pin)	Logic pin	I_{IC}	–2.0	—	2.0	mA
	Analog pin*1		–3.0	—	3.0	mA
DC injection current (total)		$\Sigma\|I_{IC}\|$	–50.0	—	50.0	mA

This item affects the calorific value and Tj of the chip. In addition to these restrictions, you also need to take thermal design into consideration.

Note 1. The objects are ADCCn pins. However, the following pins are excluded:
ADCC0I00, ADCC0I03, ADCC0I20, ADCC0I21

35.2.9 Input Capacitance

Table 35.10 DC Characteristics (Input Capacitance)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input capacitance	All pins	C_{in}	—	10	20	pF	Vin = 0 V, f = 1 MHz, Tj = 25ºC

35.2.10 Supply Current Characteristics

Table 35.11 DC Characteristics (Supply Current: C1H)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Core supply current (VDD power supply)	Normal operation	I_{dd}	—	185	310	mA	
	Current during reset	I_{ddrst}	—	30	150	mA	
VCC power supply current (when EPT is not in use)*1	Normal operation (excluding erasure of code flash)	I_{CC}	—	10	15	mA	
	Erasure of code flash	$I_{cc\_cferase}$	—	—	40	mA	
	Current during reset	I_{ccrst}	3	7	15	mA	
System supply current (SYSVCC power supply)	Normal operation	I_{SYS}	—	—	1	mA	
	Current during reset	Isysrst	0.1	—	1	mA	
PLL supply current (PLLVCC power supply)		I_{PLL}	2	3.5	5	mA	
Analog power supply current (A0VCC, A1VCC power supply)		I_{AVCC}	—	3	5	mA	T&H operation disabled
			—	10	20	mA	T&H operation enabled
Analog power supply current (RVCC power supply)		I_{RVCC}	—	—	20	mA	
ADC reference power supply current (A0VREFH, A1VREFH)		I_{AVREF}	—	0.22	0.5	mA	

Note 1. When an EPT is to be used, calculate the current through the EPT from its current gain and the current from EPTVOUT, then add the result to the value for VCC.

CAUTIONS

1. **When the A/D converter is not used or it is in the standby state, do not open the A0VCC pin, A1VCC pin, A0VREFH pin, A1VREFH pin, A0VSS pin, and A1VSS pin.**
2. **Supply current values are those measured when VIHmin = VCC – 0.5 V/EVCC – 0.5 V and VIL = 0.5 V with no load applied to all output pins.**

Table 35.12 DC Characteristics (Supply Current: C1M)

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Core supply current (VDD power supply)	Normal operation	I_{dd}	—	135	250	mA	
	Current during reset	I_{ddrst}	—	30	150	mA	
VCC power supply current	Normal operation (excluding erasure of code flash)	I_{CC}	—	13.5	20	mA	
	Erasure of code flash	$I_{cc\_cferase}$	—	—	45	mA	
	Current during reset	I_{ccrst}	5	10.5	20	mA	
System supply current (SYSVCC power supply)	Normal operation	I_{SYS}	—	—	1	mA	
	Current during reset	Isysrst	0.1	—	1	mA	
Analog power supply current (A0VCC, A1VCC power supply)		I_{AVCC}	—	3	5	mA	T&H operation disabled
			—	10	20	mA	T&H operation enabled
Analog power supply current (RVCC power supply)		I_{RVCC}	—	—	10	mA	
ADC reference power supply current (A0VREFH, A1VREFH)		I_{AVREF}	—	0.22	0.5	mA	

CAUTIONS

1. **When the A/D converter is not used or it is in the standby state, do not open the A0VCC pin, A1VCC pin, A0VREFH pin, A1VREFH pin, A0VSS pin, and A1VSS pin.**
2. **Supply current values are those measured when VIHmin = VCC – 0.5 V/EVCC – 0.5 V and VIL = 0.5 V with no load applied to all output pins.**

35.3 AC Characteristics

Unless otherwise described, the following timing conditions are applied.

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5V to 5.5V, A0VREFH = 4.5 V to A0VCC,
A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

- In the port control register, conditions where all output pins of the module used in the same channel are set to the same driving ability are applied to output pins whose driving ability is selectable. Unless otherwise specified, all driving ability settings are included.
- Unless otherwise described, AC measurement conditions described in the figure below are applied.

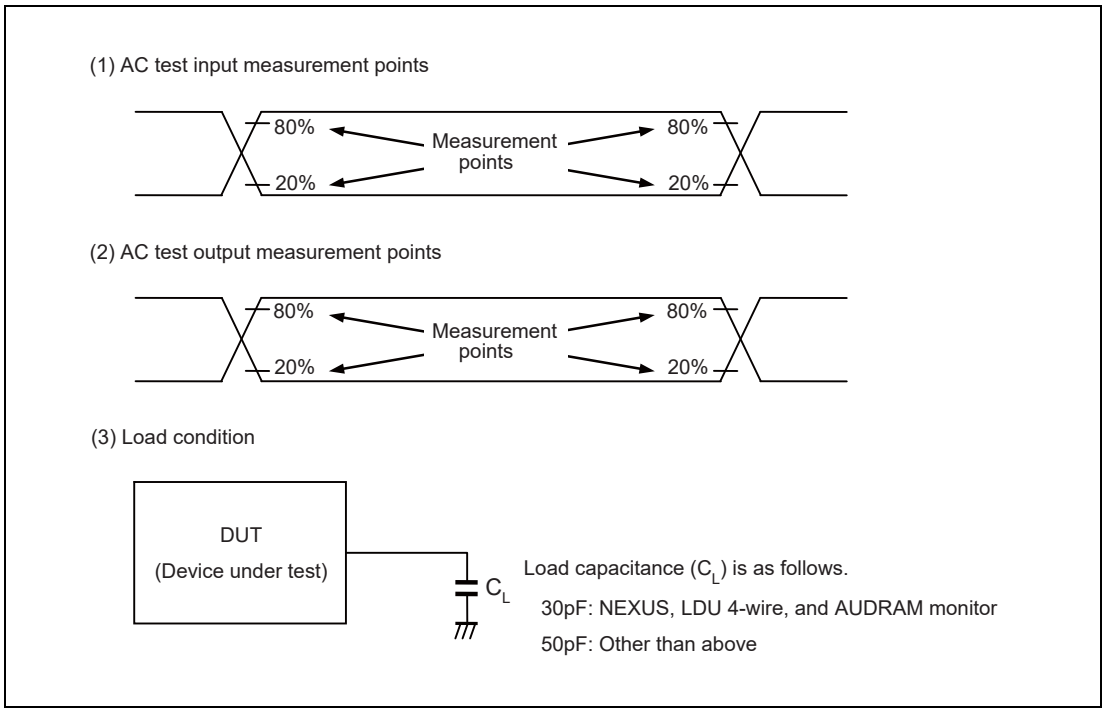


Figure 35.1 AC Measurement Conditions

35.3.1 Power On/Off Timings

(1) When EPT is not used

Table 35.13 Power On/Off Timings

Item	Symbol	Min.	Max.	Unit	Note
Pin reset L time at power-on	tRESW1	10	—	ms	*1
Pin reset L time at power-off	tRESW2	2	—	µs	*2
PLL1 lock-in time	tPLL1L0	—	1	ms	*3

Note 1. tRESW1 is the reset time required for the supply of internal clock signals to become stable after all power voltages are turned on.

Note 2. tRESW2 is the time from assertion of the reset signal until all of the power voltages have dropped below the lower-limit voltages.

Note 3. tPLL1L0 is the time required for PLL1 to lock in after MOSC (main oscillator) oscillation has become stable.

CAUTIONS

- **The states of I/O pins are not reset during the reset noise cancellation interval (max. 1.2 µs) following assertion of the reset signal while power is being turned off.**
- **If power is disconnected during programming or erasure of flash memory, data in the area of the flash memory that was being programmed or erased are not guaranteed.**

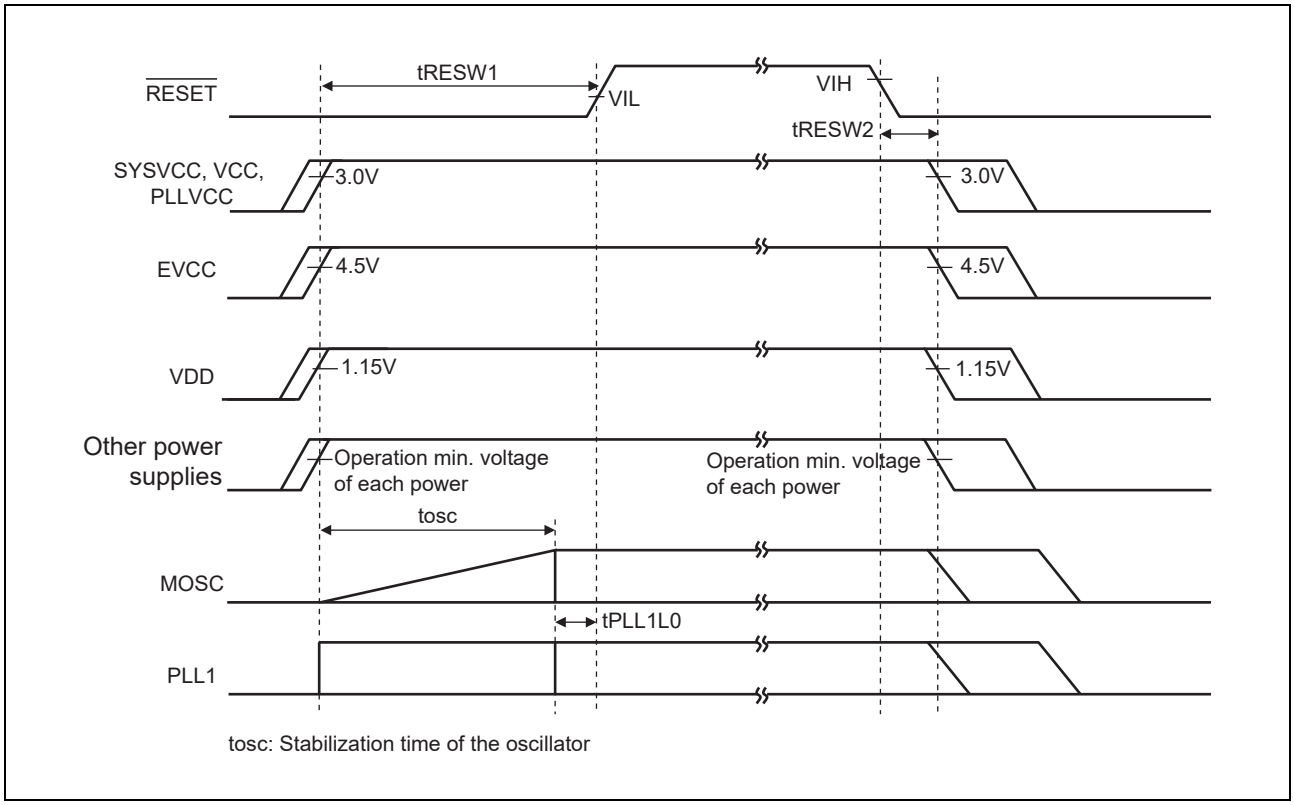


Figure 35.2 Power On/Off Timings

(2) When EPT is used

Table 35.14 Power On/Off Timings

Item	Symbol	Min.	Max.	Unit	Reference
Pin reset L time at power-on	tRESW4	10	—	ms	*1
Pin reset L time at power-off	tRESW5	2	—	µs	*2
PPL1 lock-in time	tPLL1L0	—	1	ms	*3

Note 1. tRESW4 is the reset time required for the supply of internal clock signals to become stable after all power voltages other than VDD are turned on.

Note 2. tRESW5 is the time from assertion of the reset signal until all of the power voltages have dropped below the lower-limit voltages.

Note 3. tPLL1L0 is the time required for PLL1 to lock in after MOSC (main oscillator) oscillation has become stable.

CAUTIONS

- **The states of I/O pins are not reset during the reset noise cancellation interval (max. 1.2 µs) following assertion of the reset signal while power is being turned off.**
- **If power is disconnected during programming or erasure of flash memory, data in the area of the flash memory that was being programmed or erased are not guaranteed.**

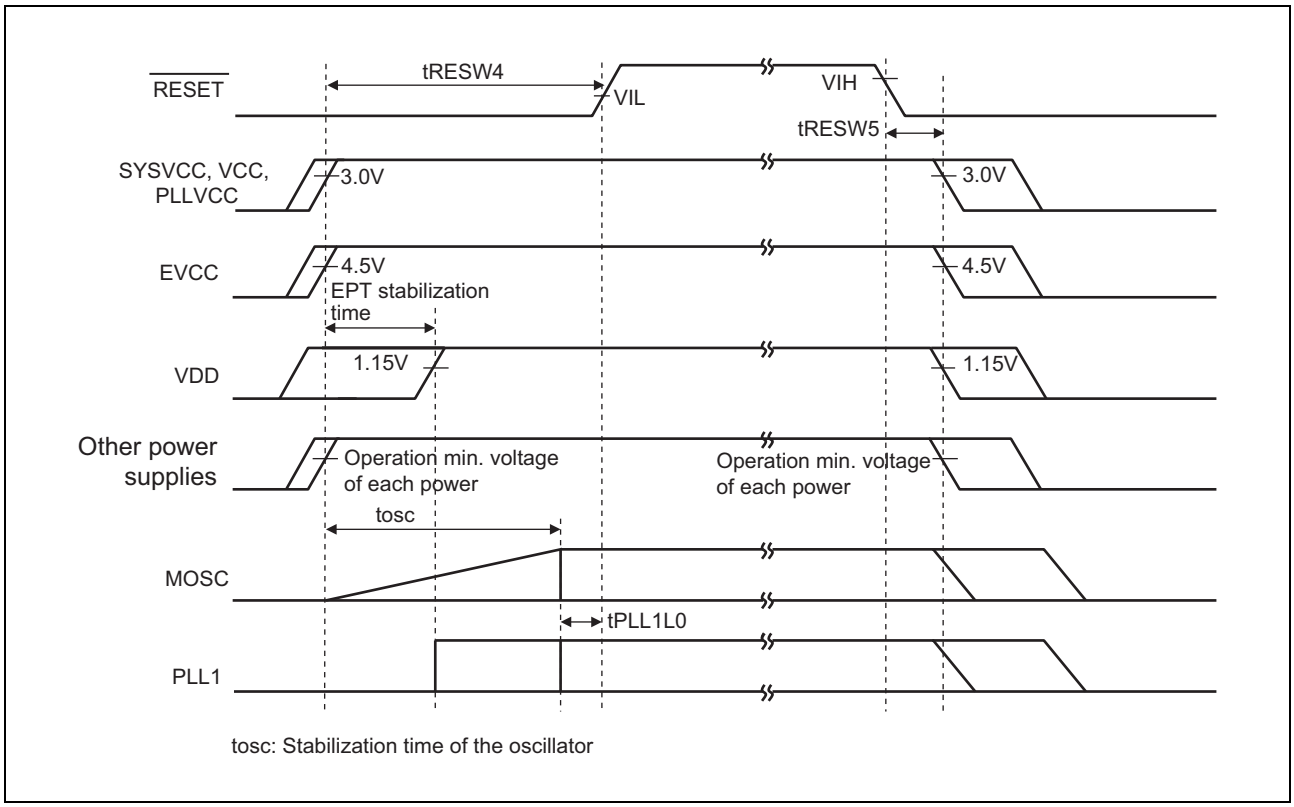


Figure 35.3 Power On/Off Timings when EPT is Used

35.3.2 Clock Timing

35.3.2.1 Spread Spectrum Clock Generator

Table 35.15 SSCG Timing

Item	Symbol	Min.	Typ.	Max.	Unit
Modulation frequency*1	f_{mod}	20	—	100	kHz
Frequency dithering range*1	f_{dit}	4.1	—	—	%
Frequency stabilization time (OFF → ON)		—	—	1.6	ms

Note 1. The modulation method is applied only to down spread.

35.3.2.2 Oscillation Frequency Accuracy of the On-Chip Oscillator

Table 35.16 Oscillation Frequency Accuracy of the On-Chip Oscillator

Item	Symbol	Min.	Typ.	Max.	Unit
CLK_LIOSC oscillation frequency	fLIOSC	160	240	360	kHz

35.3.3 Output Slew Rate

EVCC power supply pins

Table 35.17 Selection of Driving Ability = High

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	4	6	ns
		CL = 50 pF	—	6	12	ns
		CL = 75 pF	—	8	16	ns
		CL = 100 pF	—	10	20	ns

Table 35.18 Selection of Driving Ability = Mid

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	8	15	ns
		CL = 50 pF	—	15	30	ns
		CL = 75 pF	—	23	45	ns
		CL = 100 pF	—	30	60	ns

Table 35.19 Selection of Driving Ability = Low

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Output rising time/falling time slew rate	tR, tF	CL = 25 pF	—	25	50	ns
		CL = 50 pF	—	50	100	ns
		CL = 75 pF	—	70	120	ns
		CL = 100 pF	—	85	150	ns

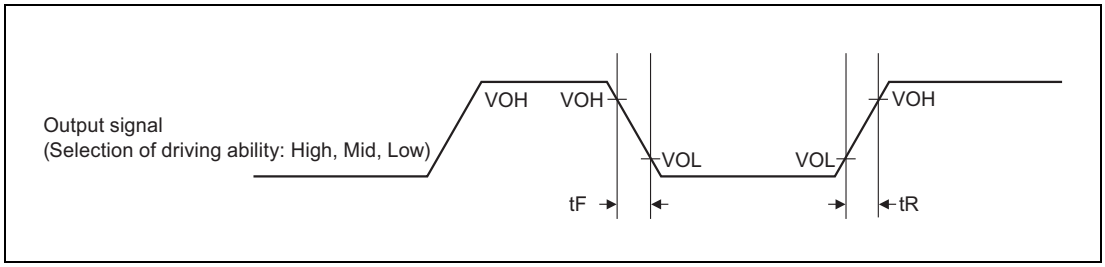


Figure 35.4 Output Signal Timing

35.3.4 Control Signal Timing

Table 35.20 Control Signals

Item	Symbol	Min.	Typ.	Max.	Unit
Reset pulse width*1	tRESW3	1.5	—	—	µs
Reset noise cancel width	tRESNCW	0.2	0.4	1.2	µs
NMI noise cancel width	tNC	0.2	0.4	1.2	µs
IRQ pulse width*2	tIRQ	50	—	—	ns
Operating mode setup time	tMDS	1	—	—	ms
Operating mode hold time	tMDH	1	—	—	ms

Note 1. The reset pulse width must be equal to or more than the minimum tRESW3 value.
If the reset pulse width is less than the minimum value of the reset noise cancel width, the reset cannot be accepted.

Note 2. In case noise removal is disabled by DNF.

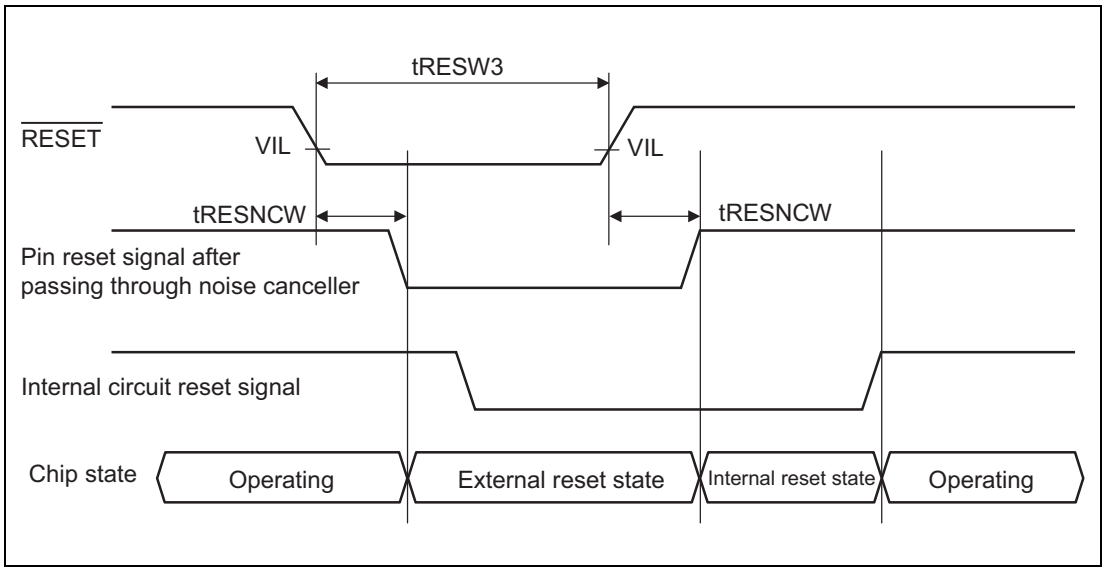


Figure 35.5 Reset Timing

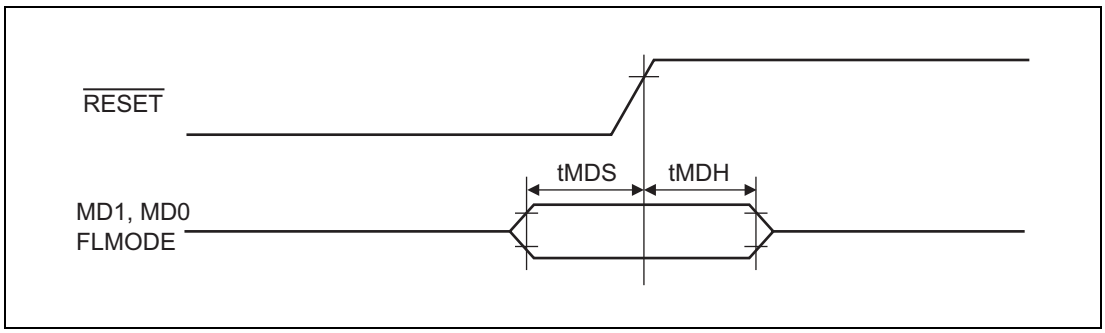


Figure 35.6 Control Signal Timing

35.3.5 CSIH Timing

35.3.5.1 Master Mode

Table 35.21 CSIH Timing in Master Mode

Condition: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
CSIHnSC cycle	tKCYM		100	—	ns
CSIHnSC high-level width	tKWHM		(tKCYM/2) – 20	—	ns
CSIHnSC low-level width	tKWLM		(tKCYM/2) – 20	—	ns
CSIHnSI setup time	tSSIM		18	—	ns
CSIHnSI hold time	tHSIM		10	—	ns
CSIHnSO output delay time	tDSOM		—	10	ns
CSIHnSO output hold time (vs. CSIHnSC)	tHSOM		tKWHM – 10	—	ns
CSIHnRYI setup time	tSRYI	HSE = 1	(2×tPAck) + 30	—	ns
CSIHnCSSx inactive level width	tWSCSB	*1	(CSidle + 0.5) × tKCYM – 20	—	ns
		Other than above	CSidle × tKCYM – 20	—	ns
CSIHnCSSx setup time	tSSCSB0	DAP = 0	CSsetup × tKCYM – 10	—	ns
	tSSCSB1	DAP = 1	(CSsetup + 0.5) × tKCYM – 10	—	ns
CSIHnCSSx hold time	tHSCSB0	SIT = 0	CShold × tKCYM – 10	—	ns
	tHSCSB1	SIT = 1	(CShold + 0.5) × tKCYM – 10	—	ns

Note 1. When the serial clock level is changed during communication and when the idle time is set to 0.5 transmission clock periods.

Note: tPAck is the operating clock cycle of CSIH (80 MHz SSCG).
n = 0 to 1, x = 0 to 3
CSsetup: CSIHnCFGx.CSIH0SPx3-0 set value
CShold: CSIHnCFGx.CSIH0HDx3-0 set value
CSidle: CSIHnCFGx.CSIH0IDLx2-0 set value
DAP: CSIHnCFGx.CSIHnDAP bit
SIT: CSIHnCTL1.CSIHnSIT bit
HSE: CSIHnCTL1.CSIHnHSE bit

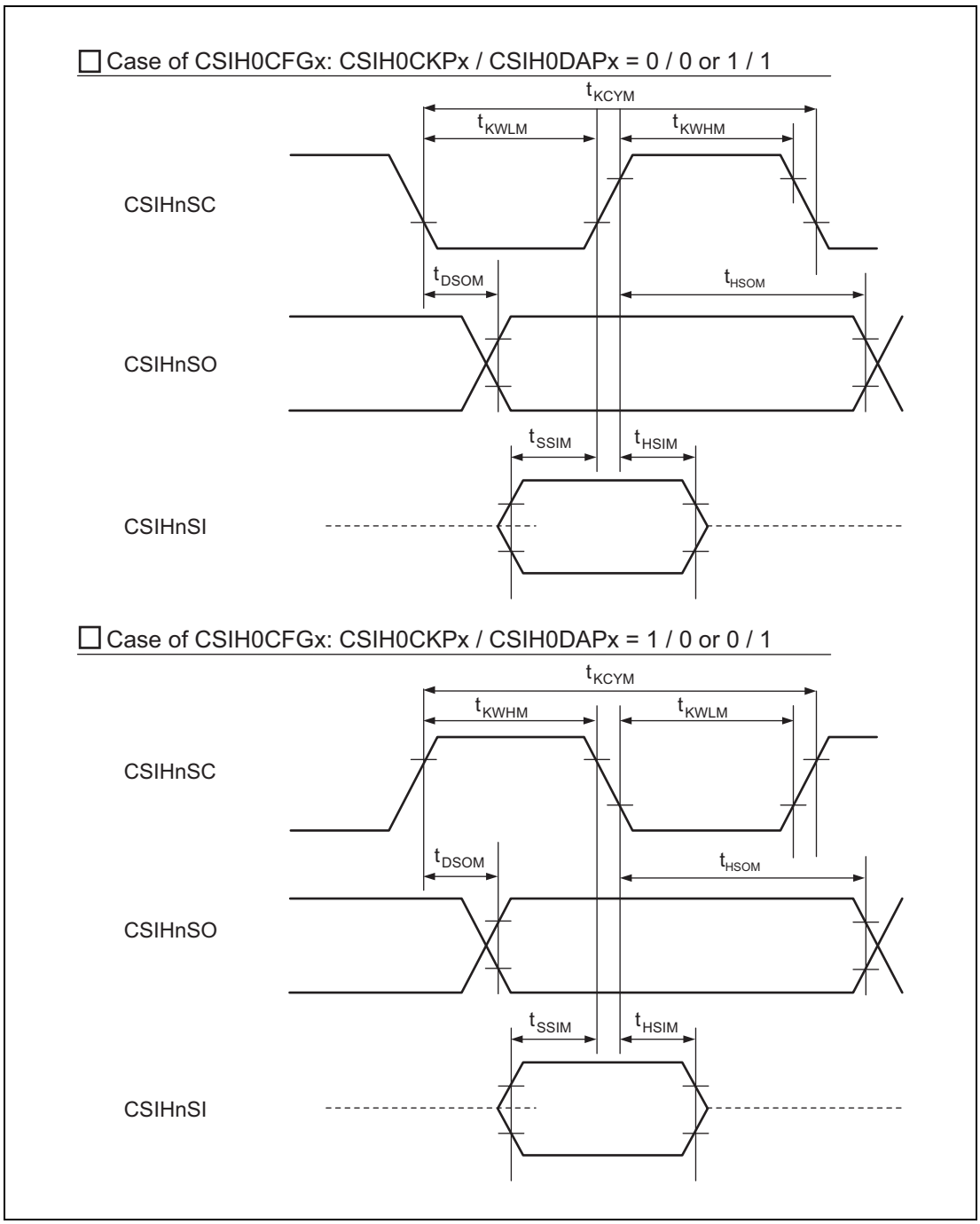


Figure 35.7 CSIH Timing (Master Mode) (1/4)

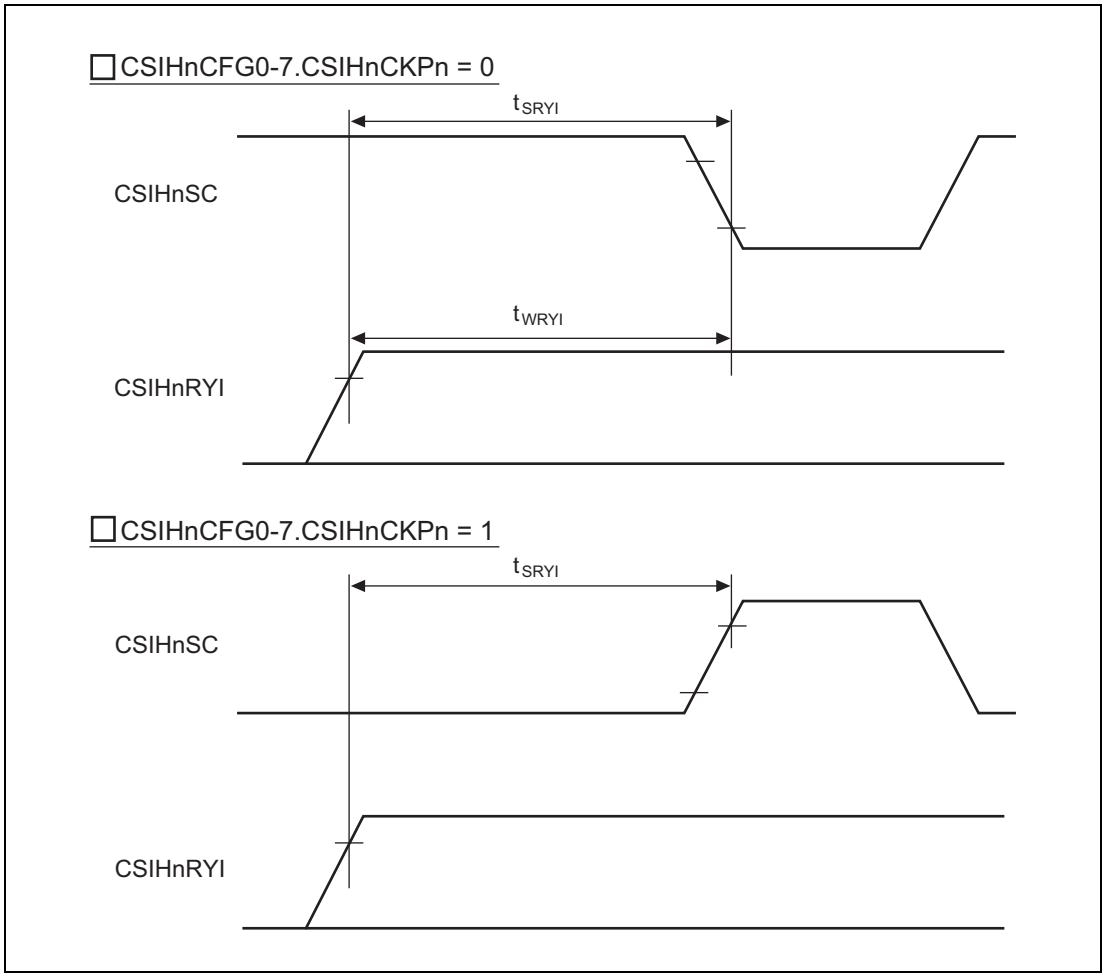


Figure 35.7 CSIH Timing (Master Mode) (2/4)

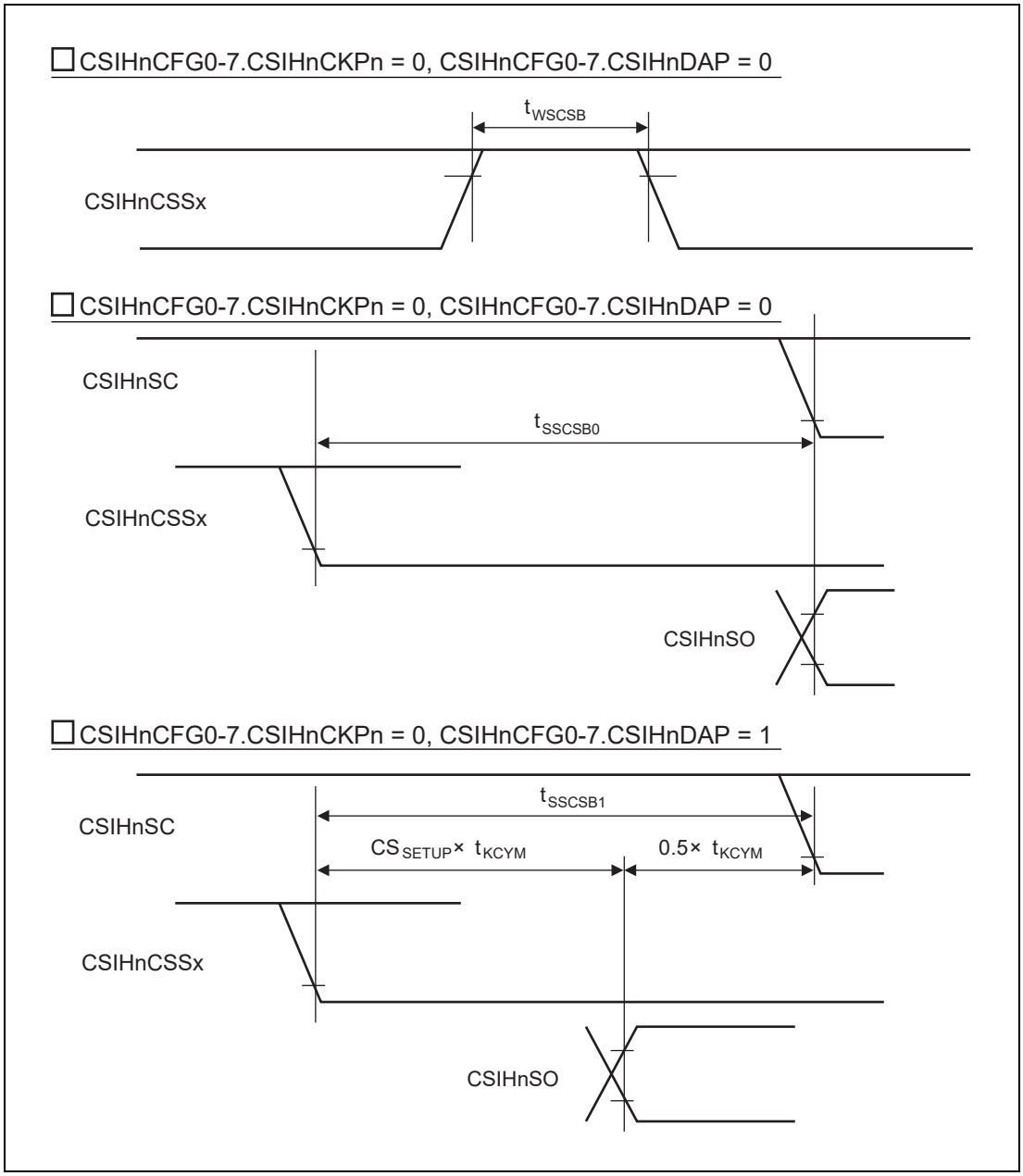


Figure 35.7 CSIH Timing (Master Mode) (3/4)

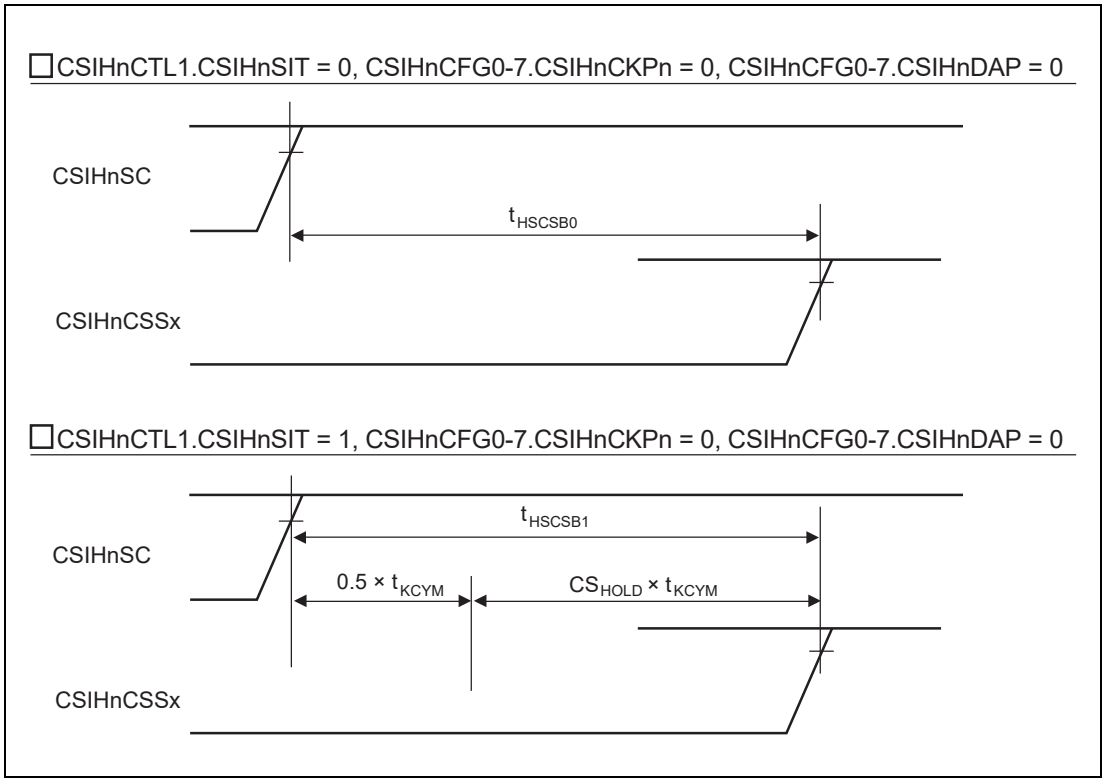


Figure 35.7 CSIH Timing (Master Mode) (4/4)

35.3.5.2 Slave Mode

Table 35.22 CSIH Timing in Slave Mode

Condition: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
CSIHnSC cycle	tKCYS		200	—	ns
CSIHnSC input high-level width	tKWHS		(tKCYS/2) – 30	—	ns
CSIHnSC input low-level width	tKWLS		(tKCYS/2) – 30	—	ns
CSIHnSI input setup time	tSSIS		15	—	ns
CSIHnSI input hold time	tHSIS		tPAck + 15	—	ns
CSIHnSO output delay time	tDSOS		—	30	ns
CSIHnSO output hold time (vs. CSIHnSC)	tHSOS		tKWHS	—	ns
CSIHnRYO output delay time	tSRYO		—	30	ns
$\overline{\text{CSIHnSSI}}$ setup time	tSSSIS		0.5 × tKCYS	—	ns
$\overline{\text{CSIHnSSI}}$ hold time	tHSSIS		tPAck + 30	—	ns
Slave output release time	tREL		—	100	ns

Note: tPAck is the operating clock cycle of CSIH (80 MHz SSCG).

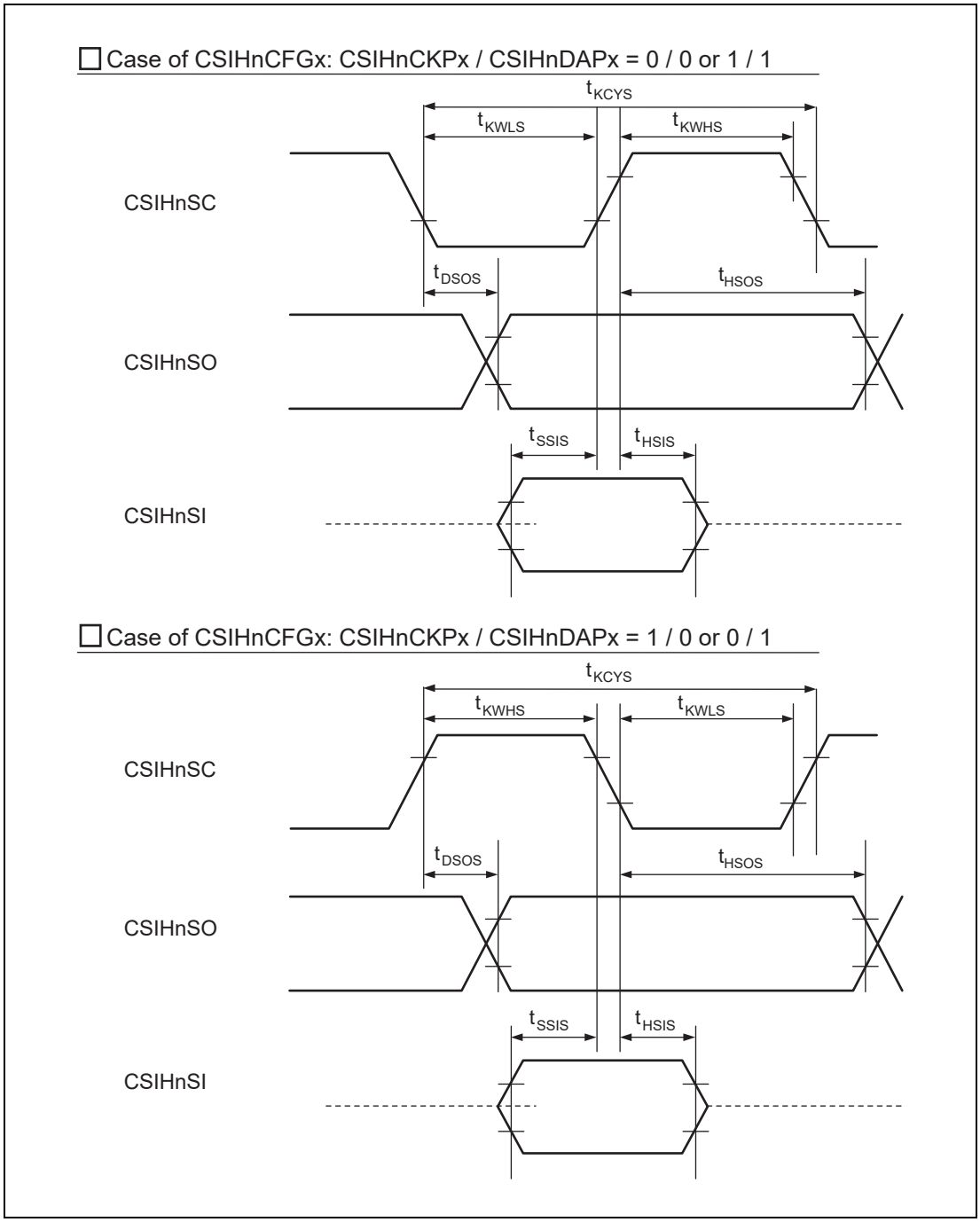


Figure 35.8 CSIH Timing (Slave Mode) (1/3)

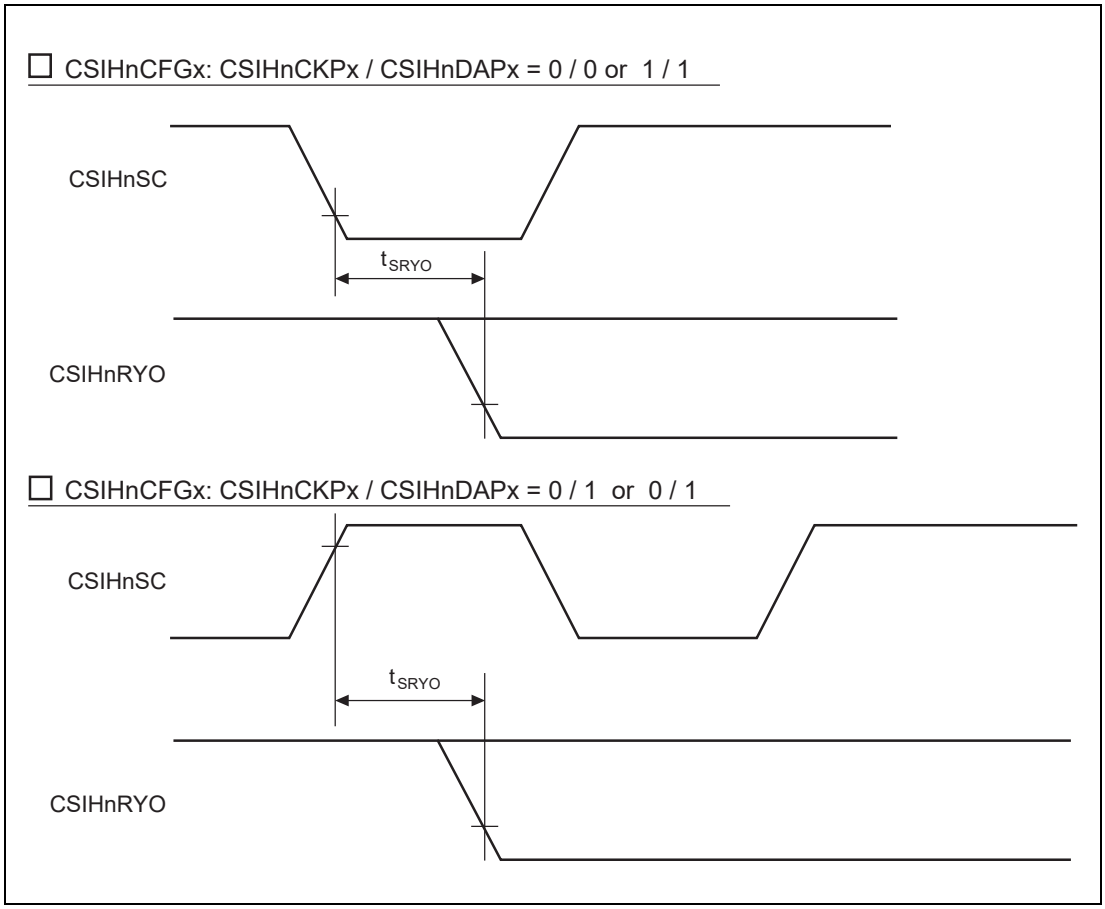


Figure 35.8 CSIH Timing (Slave Mode) (2/3)

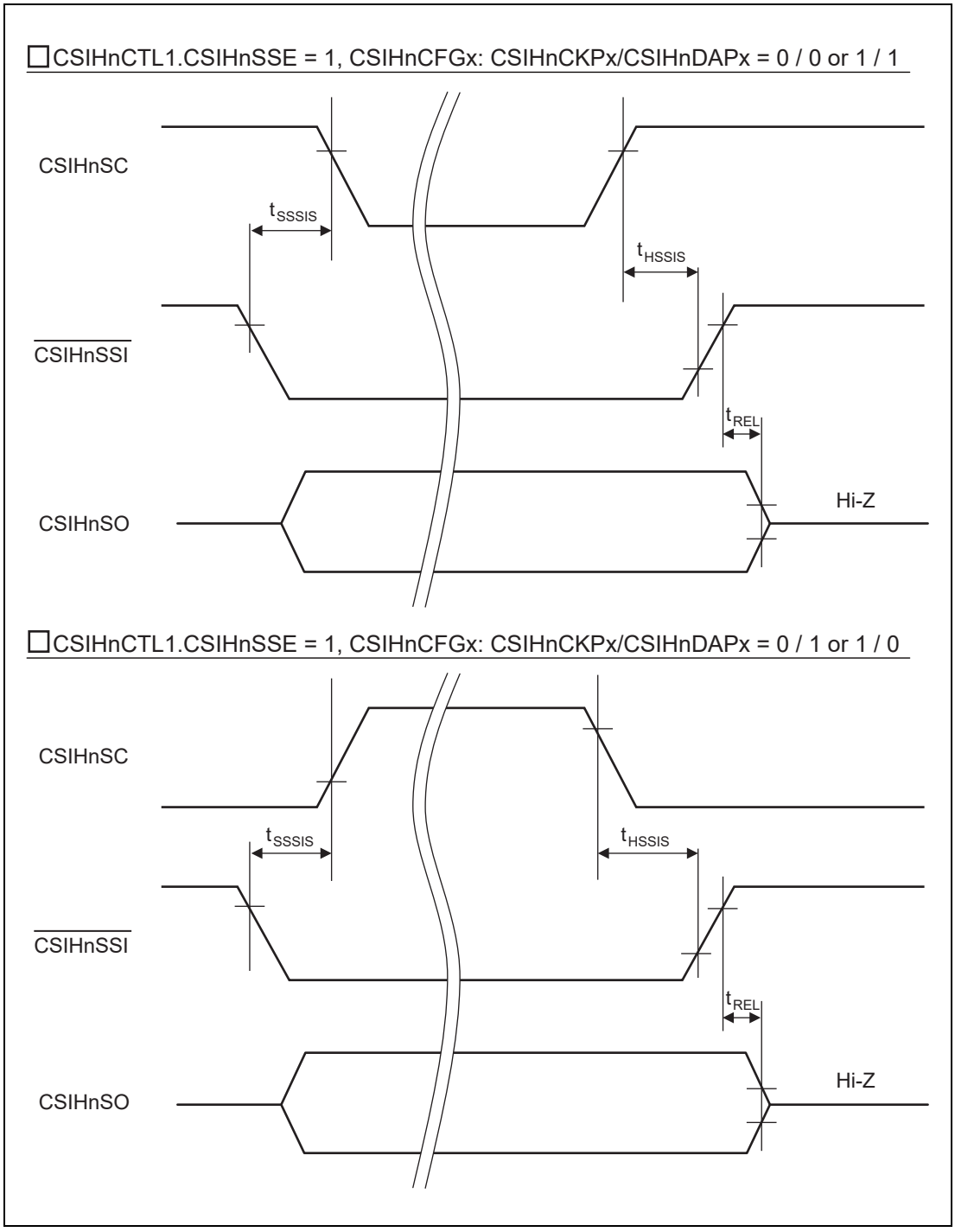


Figure 35.8 CSIH Timing (Slave Mode) (3/3)

35.3.6 SCI/FLSCI Timing

Table 35.23 SCI3 Timing (Master Mode)

Condition: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit
Output clock cycle	tScyc	Asynchronous	16 × tPck	—	ns
		Clock synchronous	8 × tPck	—	ns
Output clock pulse width	tSCKW		0.4 × tScyc	0.6 × tScyc	ns
Transmit data delay time	tTXD	Clock synchronous	—	40	ns
Receive data setup time	tRXS	Clock synchronous	2 × tPck	—	ns
Receive data hold time	tRXH	Clock synchronous	2 × tPck	—	ns

Note: tPck is the operating clock cycle of SCI (40 MHz clean clock).

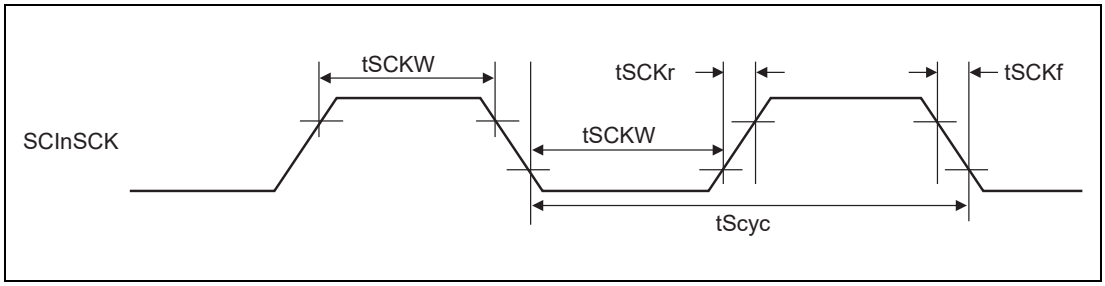


Figure 35.9 SCI Clock Input/Output Timing

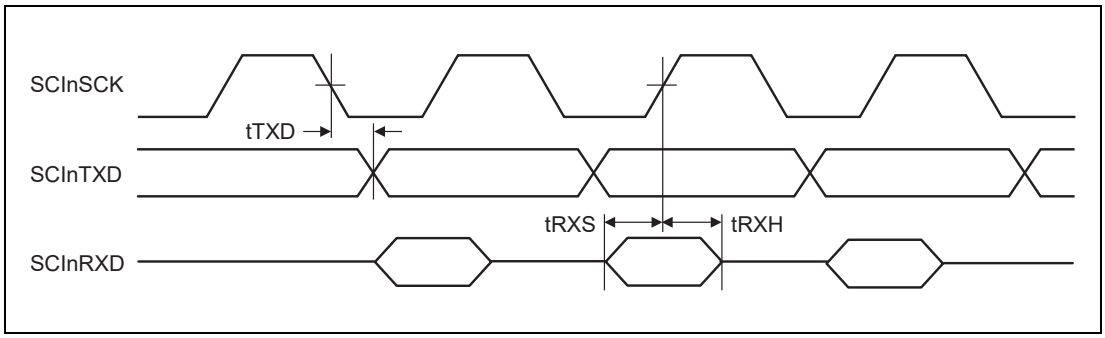


Figure 35.10 SCI Input/Output Timing, Clock Synchronous Mode

Table 35.24 SCI3 Timing (Slave Mode)

Condition: CL = 50 pF, selection of driving ability = High

Item	Symbol	Min.	Max.	Unit
Input clock cycle	tScyc	12 × tPck	—	ns
Input clock pulse width	tSCKW	0.4 × tScyc	0.6 × tScyc	ns
Input clock rising time	tSCKr	—	20	ns
Input clock falling time	tSCKf	—	20	ns
Transmit data delay time*1	tTXD	2 × tPck	50 + 3 × tPck	ns
Receive data setup time	tRXS	2 × tPck	—	ns
Receive data hold time	tRXH	2 × tPck	—	ns

Note 1. For bits other than for data 0 (1st bit) in discontinuous transfer. Data 0 (1st bit) in discontinuous transfer is transmitted at the same time when TDRE is cleared to 0.

Note: tPck is the operating clock cycle of SCI (40 MHz clean clock).
Asynchronous clock input mode is not supported.

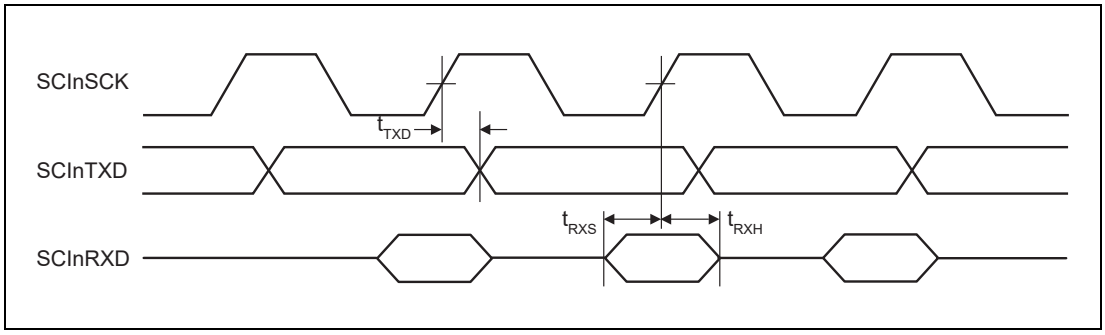


Figure 35.11 SCI Input/Output Timing in Clock Synchronous Mode (as a Slave)

35.3.7 RS-CAN Timing

Table 35.25 RS-CAN Timing

Condition: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Transfer rate			—	—	1	Mbps
Internal delay time	t_{NODE}		—	—	100	ns

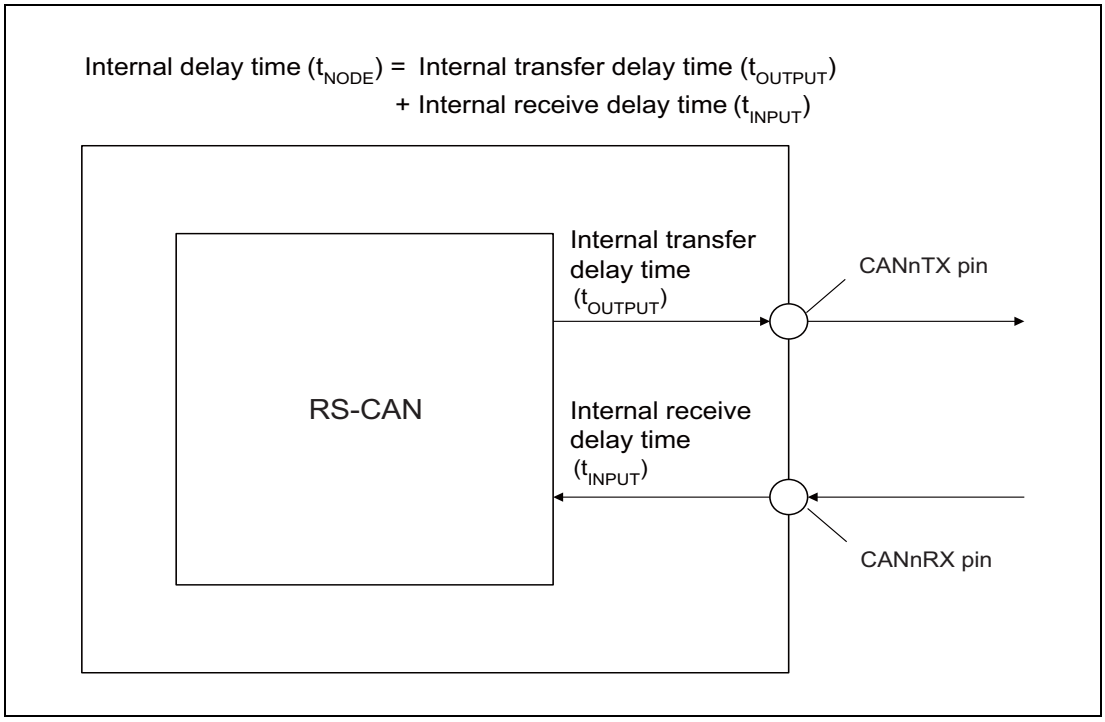


Figure 35.12 RS-CAN Timing

Definition of internal delay time of RS-CAN

Internal delay time (tNODE) = tOUTPUT + tINPUT

35.3.8 RLIN2 Timing

Table 35.26 RLIN2 Timing

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Transfer rate			—	—	20	kbps

35.3.9 Motor Control Signals Timing

Table 35.27 Motor Control Signals Timing

Item	Symbol	Condition	Min.	Max.	Unit
Input high-level width	tTIH	ENCAnE0-1, ENCAnEC *1, TAPAnESO	1.5 × tPck	—	ns
Input low-level width	tTIL	ENCAnE0-1, ENCAnEC *1, TAPAnESO	1.5 × tPck	—	ns

Note: In case noise removal is disabled by DNF.

Note 1. When used as hall sensor inputs (TSG3nPTSI0-I2).

Note: tPck is the operating clock cycle of TSG3 (80 MHz clean clock).

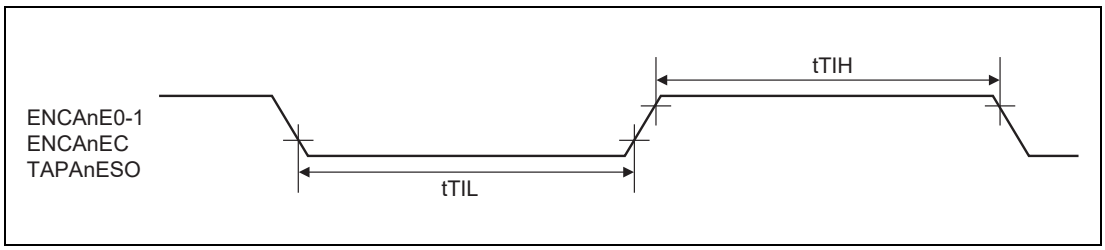


Figure 35.13 Motor Control Signals Timing

35.3.10 Timer Timing

Table 35.28 Timer Timing

Item	Symbol	Condition	Min.	Max.	Unit
Input high-level width	tTIH	TAUDnI0-15, TAUJ0I0-3, ENCAnI0-1, ENCAnE0-1, ENCAnEC	1.5 × tPck	—	ns
Input low-level width	tTIL	TAUDnI0-15, TAUJ0I0-3, ENCAnI0-1, ENCAnE0-1, ENCAnEC	1.5 × tPck	—	ns

Note: In case noise removal is disabled by DNF.

Note: tPck is the operating clock cycle of TSG3 (80 MHz clean clock).

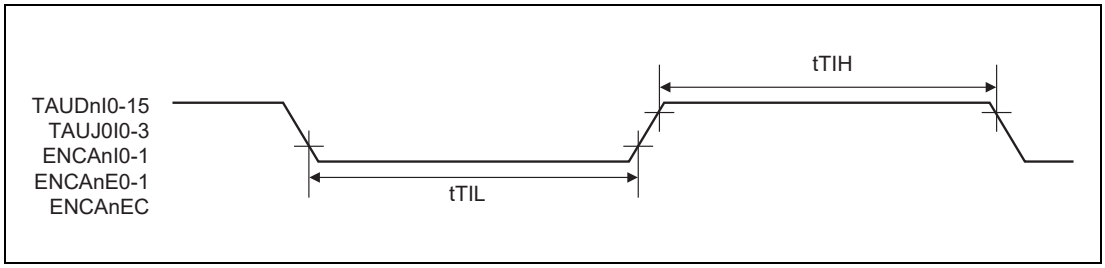


Figure 35.14 Timer Timing

35.3.11 JTAG/NEXUS Timing

Table 35.29 JTAG/NEXUS Timing

Condition: C_L = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit
DCUTCK cycle time	tTCKW		50	—	ns
DCUTCK high-level width	tTCKWH		21	—	ns
DCUTCK low-level width	tTCKWL		21	—	ns
DCUTMS/DCUTDI setup time (to DCUTCK ↑)	tTISU		12	—	ns
DCUTMS/DCUTDI hold time (from DCUTCK ↑)	tTIH		12	—	ns
DCUTDO output delay time (from DCUTCK ↓)	tTDOD		—	tTCKW – 20	ns
$\overline{\text{DCURDY}}$ output delay time (from DCUTCK ↓)	tRDYD			tTCKW – 20	ns
$\overline{\text{DCUTRST}}$ low-level width	tTRSTWL		1200	—	ns
$\overline{\text{DCUTRST}}$/DCUTCK/DCUTMS/DCUTDI input rising time	tTIR		—	12	ns
$\overline{\text{DCUTRST}}$/DCUTCK/DCUTMS/DCUTDI input falling time	tTIF		—	12	ns

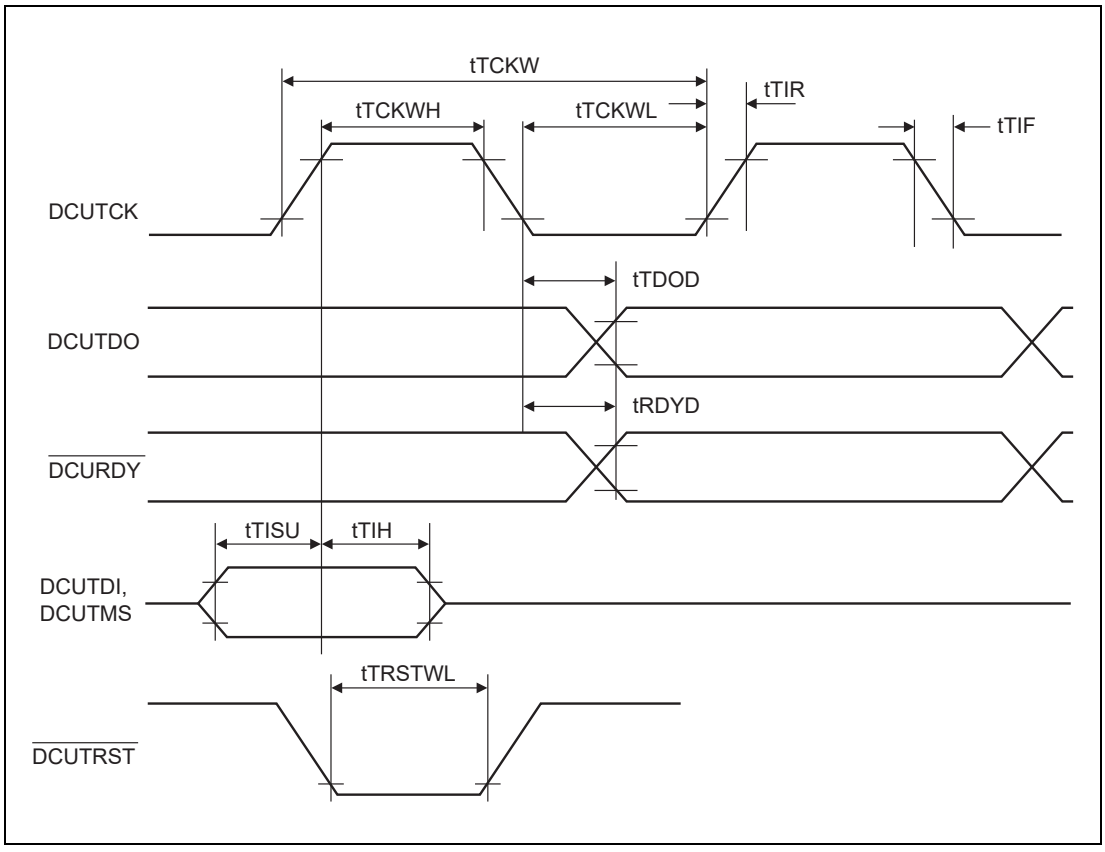


Figure 35.15 JTAG/NEXUS Timing

35.3.12 LPD (4-pin) Timing

Table 35.30 LPD (4-pin) Timing

Condition: Tj = –40ºC to 150ºC, C_L = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit
LPDCLK cycle	tLPDCKW		25	—	ns
LPDCLK high-level width	tLPDCKWH		4.5	—	ns
LPDCLK low-level width	tLPDCKWL		4.5	—	ns
LPDCLK input rising time	tLPDCKR		—	8	ns
LPDCLK input falling time	tLPDCKF		—	8	ns
LPDI setup time (to LPDCLK ↑)	tLPDSU		2	—	ns
LPDI hold time (from LPDCLK ↑)	tLPDH		2	—	ns
LPDCLKOUT cycle time	tLPDCKOW		25	—	ns
LPDCLKOUT high-level width	tLPDCKOWH		4.5	—	ns
LPDCLKOUT low-level width	tLPDCKOWL		4.5	—	ns
LPDCLKOUT rising time	tLPDCKOR		—	8	ns
LPDCLKOUT falling time	tLPDCKOF		—	8	ns
LPDO output delay (from LPDCLKOUT ↑)	tLPDOD		0	12	ns

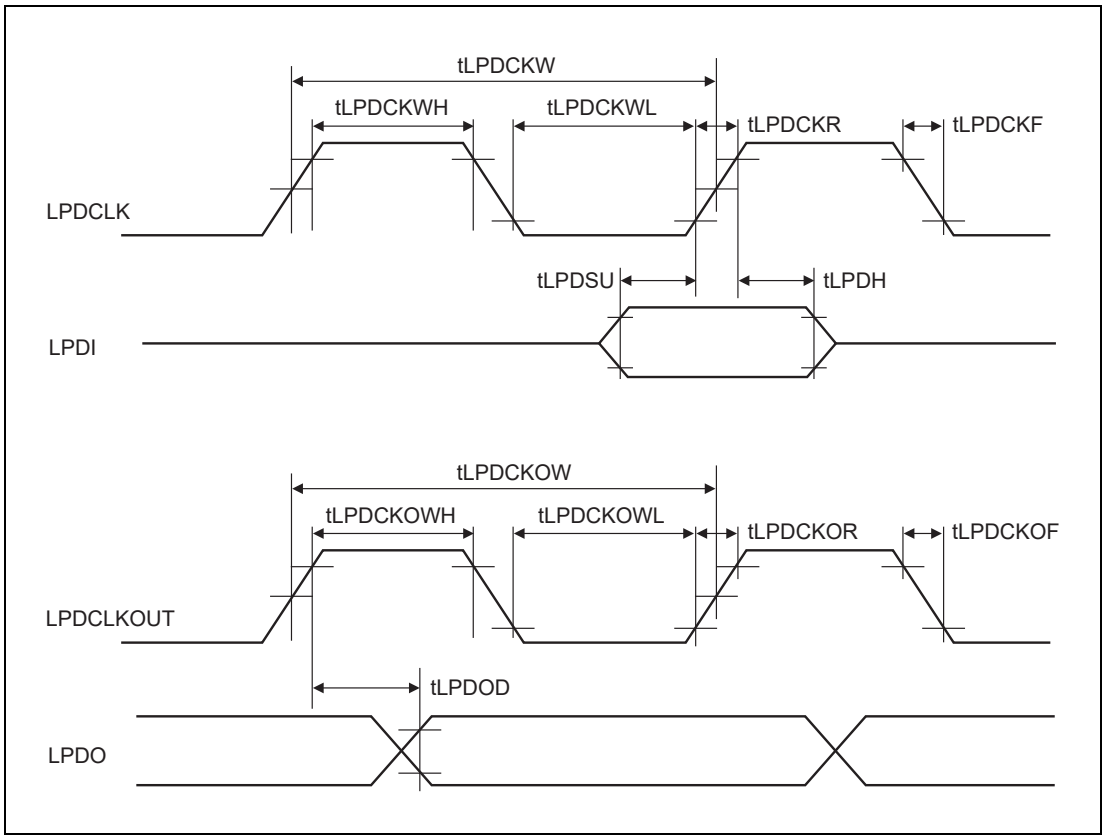


Figure 35.16 LDU 4-Wire Timing

35.3.13 AUD RAM Monitor

Table 35.31 AUD RAM Monitor Timing

Condition: Tj = –40ºC to 150ºC, C_L = 30 pF

Item	Symbol	Min.	Max.	Unit
AUDCK cycle time (monitor mode)	tAUCKMcyc	50	—	ns
AUDCK high-level width (monitor mode)	tAUCKMH	0.4 × tAUCKMcyc	—	ns
AUDCK low-level width (monitor mode)	tAUCKML	0.4 × tAUCKMcyc	—	ns
$\overline{\text{AUDRST}}$ setup time (monitor mode, to AUDCK ↑)	tAURSTMS	30	—	ns
$\overline{\text{AUDRST}}$ input pulse width (monitor mode)	tAURSTMW	5 × tAUCKMcyc	—	ns
Monitor data output delay time (to AUDCK ↑)	tAUDTMD	—	35	ns
Monitor data input setup time (to AUDCK ↑)	tAUDTMS	15	—	ns
Monitor data input hold time (from AUDCK ↑)	tAUDTMH	5	—	ns
$\overline{\text{AUDSYNC}}$ input setup time (to AUDCK ↑)	tAUDSYS	15	—	ns
$\overline{\text{AUDSYNC}}$ input hold time (from AUDCK ↑)	tAUDSYH	5	—	ns

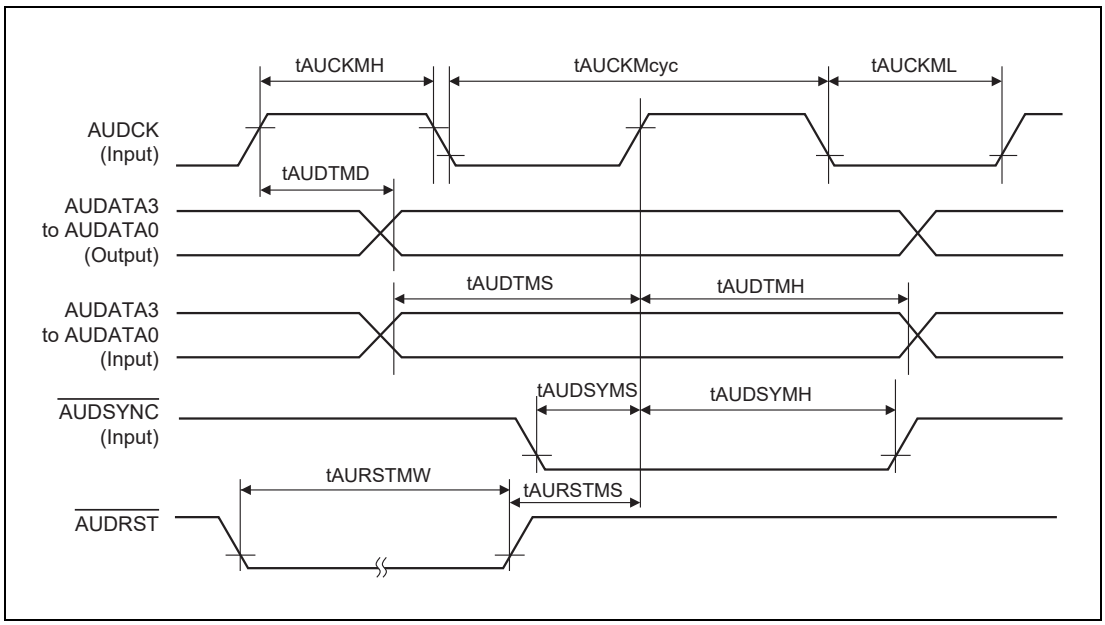


Figure 35.17 AUD RAM Monitor Timing

35.4 A/D Converter Characteristics

Table 35.32 A/D Converter Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Digital resolution	—	—	—	12	—	bit
A/D conversion time*1	—	—	—	1	—	µs
Integral nonlinear error	—	Using T&H amplifier	—	—	±4	LSB
Offset error	—	Using T&H amplifier	—	—	±7.5	LSB
Full-scale error	—	Using T&H amplifier	—	—	±7.5	LSB
Quantization error	—	—	—	—	±0.5	LSB
Absolute error	—	—	—	—	±8.0	LSB
Self-diagnosis absolute error	—	In A/D converter self-diagnosis	—	—	±8.0	LSB
	—	In pin level self-diagnosis	—	—	±80	LSB
Analog input capacitance	—	A/D conversion standby	—	—	10	pF
	—	In sampling	—	—	20	pF
Allowable analog signal source impedance	—	—	—	—	3	kΩ
Channel T&H hold time*2	—	—	—	—	10	µs
T&H sampling time	—	—	—	—	0.45	µs
Input voltage range	—	Not used T&H amplifier	0	—	A0VREFH A1VREFH	V
	—	Using T&H amplifier	0.2	—	A0VREFH-0.2 A1VREFH-0.2	V

Note 1. Conversion time for a channel, and not include T&H time.

Note 2. When the T&H circuit is in use, A/D conversion must be performed within the maximum time.

- Errors in the External Circuit of the A/D Converter

A formula for errors in sampled values due to the external circuit of the A/D converter is given below. These errors will depend on the input circuit and conversion cycle. The formula given below for the errors is simplified for the calculation of sampling errors based on internal stray capacitance, amplifier offset, resistance of the signal source, and conversion cycle. This formula can also be used to calculate the effects of the signal source resistance and conversion cycle on these errors.

The formula gives the error of analog input 2 as shown in the figure below when A/D conversion is performed in the order analog input 1 then 2.

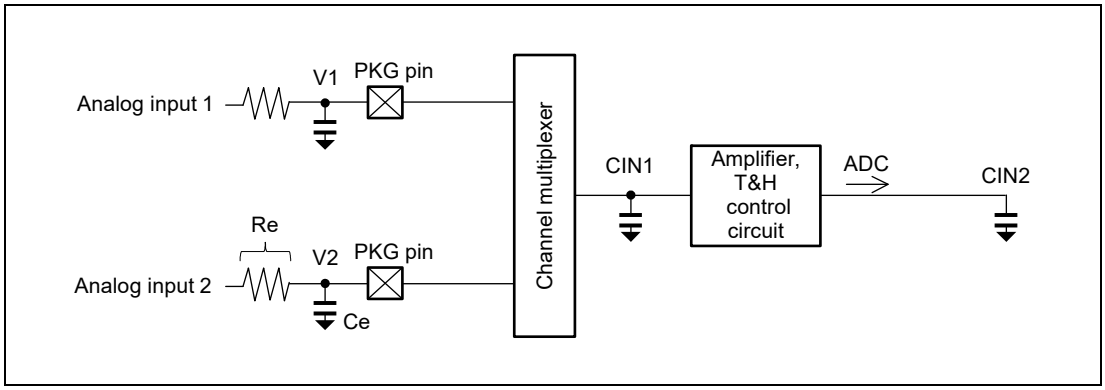
$$Sampling\ error\ (LSB) = \left[\left(\frac{|V2 - V1| \times CIN1}{Ce + CIN1} + \frac{|Vvfaerr| \times CIN2}{Ce + CIN2}\right) \times \frac{1}{1 - e^{-T1 / (Re \times Ce)}} + \left(\frac{1}{T1} \times C1 \times V3 \times Re\right)\right] \times \frac{4096}{Vavrefh}$$

Table 35.33 Parameters of C1M (R7F701271EAFP #0) and C1H (R7F701270EABG #**0)**

Item	Symbol	Reference	Unit
Common capacitance of the final stage of channel multiplexer (ADCC0)	CIN1	1.6	pF
Common capacitance of the final stage of channel multiplexer (ADCC1)	CIN1	2	pF
Common capacitance of the final state of the amplifier and T&H control circuit	CIN2	10	pF
External capacitor on analog input pin	Ce	Depends on user board	µF
Signal source impedance	Re		kΩ
Conversion cycle of conversion pins	T1		ms
AnVREFH voltage (n = 0, 1)	Vavrefh		V
Potential difference between V1 and V2	V2-V1	5	V
Offset voltage of amplifier and T&H control circuit	Vvfaerr	50	mV
Parasitic capacitance of the last stage of channel multiplexer	C1	10	pF
|AnVCC voltage /2 – measured pin voltage (V2)|	V3	Depends on user board	V

Table 35.34 Parameters of C1M (R7F701271EAFP #4) and C1H (R7F701270EABG-C #**4)**

Item	Symbol	Reference	Unit
Common capacitance of the final stage of channel multiplexer (ADCC0)	CIN1	1.6	pF
Common capacitance of the final stage of channel multiplexer (ADCC1)	CIN1	2	pF
Common capacitance of the final state of the amplifier and T&H control circuit	CIN2	10	pF
External capacitor on analog input pin	Ce	Depends on user board	µF
Signal source impedance	Re		kΩ
Conversion cycle of conversion pins	T1		ms
AnVREFH voltage (n = 0, 1)	Vavrefh		V
Potential difference between V1 and V2	V2-V1	5	V
Offset voltage of amplifier and T&H control circuit	Vvfaerr	50	mV
Parasitic capacitance of the last stage of channel multiplexer	C1	2	pF
|AnVCC voltage /2 – measured pin voltage (V2)|	V3	Depends on user board	V



- Values for conversion error calculated by using this formula do not include error (absolute error, etc.) specified in the A/D converter characteristics.
- This formula is a desktop formula and theoretical. When the signal source has an extremely high resistance or when the conversion cycle is too short, calculated and measured values may differ. Actual error depends on the capacitor, resistor, capacitance and resistance of board wiring, so please evaluate and verify the error on the user board is no greater than the value produced by this formula ($Re < 1.5\ M\Omega$ and $T1 \geq 10\ \mu s$, or $1.5\ M\Omega \leq Re \leq 2\ M\Omega$ and $T1 \geq 512\ \mu s$).

35.5 R/D Converter Characteristics

Conditions: EVCC = 4.5 V to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 V to 1.35 V
A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 V to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = –40ºC to 150ºC

35.5.1 RDC Conversion Performance

Table 35.35 RDC Conversion Performance

Item	Condition		Min.	Typ.	Max.	Unit
Resolution*1			—	—	16	bit
Conversion accuracy*2	Absolute error in electrical angle signal while operation is stopped (12-bit resolution)		—	—	±4	LSB
Settling time (Response for input of electric angle of 180°)	Settling range within ±8 LSB	Band 800 Hz	—	53	—	ms
		Band 1500 Hz	—	31	—	ms
		Band 1000 Hz	—	43	—	ms
		Band 500 Hz	—	85	—	ms
		Band 200 Hz	—	211	—	ms
		Automatic adjustment	—	1.9	—	ms
Maximum angular velocity*3 (The values in parentheses are applied to range setting as automatic adjustment)	16-bit resolution		15000 (7500)	—	—	min^{-1}
	14-bit resolution		60000 (30000)	—	—	min^{-1}
	13-bit resolution		120000 (60000)	—	—	min^{-1}
	12-bit resolution		240000 (120000)	—	—	min^{-1}
	11-bit resolution		480000 (240000)	—	—	min^{-1}
	10-bit resolution		960000 (480000)	—	—	min^{-1}
Maximum angular acceleration Following angular acceleration range (electrical angle)	Band 800 Hz		—	146000	—	rad/s^2
	Band 1500 Hz		—	513000	—	rad/s^2
	Band 1000 Hz		—	183000	—	rad/s^2
	Band 500 Hz		—	46000	—	rad/s^2
	Band 200 Hz		—	5000	—	rad/s^2
	Automatic adjustment		—	3000000	—	rad/s^2
Response delay*4	Electrical angle output response delay in fixed angular velocity		–0.2	—	0.20	°/10000 min^{-1}
BIST determination time*5	Angle conversion BIST (angle determination threshold is within ±8 LSB)		—	—	10	ms
	Resolver signal error detection BIST		—	—	1.5	ms
	Resolver signal cut off detection BIST		—	—	1	ms
	Conversion error BIST		—	—	10	ms
BIST recovery time*6	All kinds of BIST		—	—	10	ms

Note 1. The resolution is changed by the setting of the maximum angular velocity select bit in the RDC2n maximum angular velocity setting register. The angle can be read with maximum 16-bit width by register access.

Note 2. It is the ability, when the waveform of analog input to RDC is ideal sign wave. RDC conversion result will defer from resolver machine angle with distortion or slippage of analog input signal or power supply voltage.

Note 3. Following angular velocity range (electrical angle). It is changed by the setting of the maximum angular velocity select bit in the RDC2n maximum angular velocity setting register.

Note 4. PHI angle output from RDC is added accuracy error through analog circuit to this value. The read of PHI angle output register value with bus access needs access time. However, it does not need access time using PHI compare signal.

Note 5. It is the time to stabilize BIST determination.
The determination time of conversion error BIST is default value. The determination time of conversion error BIST depends on the value of the EDPS[1:0] bits in the RDC2n conversion condition select register RDC2nCONSEL. Thus, when the conversion error determination time is set to over 10 ms, BIST determination time is also set to over 10 ms.

Note 6. Recovery time from BIST operation to normal operation.
When the excitation frequency is under 9 kHz, BIST recovery time is maximum 15 ms.
When the conversion error determination time is set to over 10 ms, BIST recovery time is also set to over the setting value (10 ms).

35.5.2 RDC Analog Pin

Table 35.36 RDC Analog Pin Characteristics

Signal	Symbol	Item	Min.	Typ.	Max.	Unit
Signal source output*1 for resolver excitation power supply	RSO	Frequency	—	—	40	kHz
		Output voltage*2	0.38 × RVCC	0.4 × RVCC	0.42 × RVCC	V_{P-P}
		Load impedance	10	—	—	kΩ
		Output switching*3	-40	±0	+20	%
Common voltage output for resolver excitation power supply	COM	Output voltage	0.475 × RVCC	0.5 × RVCC	0.525 × RVCC	V
		Load impedance	10	—	—	kΩ
Resolver excitation signal external input	R1E, R2E	Frequency*4 *10	5	—	40	kHz
		Input voltage range	0	—	RVCC	V
		Input voltage differential amplitude	2	—	—	V_{P-P}
		Input impedance*11	32	40	48	kΩ
Resolver signal input	S1, S2, S3, S4	Frequency*10	5	—	40	kHz
		Input voltage range*5	—	—	—	V
		Input impedance*6	16	21	26	kΩ
		Input impedance switching*7	-40	±0	+40	%
Resolver signal monitor output	COSMNT, SINMNT	Frequency*8	5	—	40	kHz
		Output voltage*9	0.4 × VCC	—	0.6 × VCC	V_{P-P}
		Load impedance	100	—	—	kΩ

Note 1. Pseudo sign wave output, 7-bit D/A output.

Note 2. The center of amplitude is COM voltage. Described values are default (±0%) of output adjustment.

Note 3. The output voltage can be adjusted in four steps of -40, -20, ±0, and +20% by adjust function.

Note 4. When the excitation frequency is over 22 kH with resolver excitation signal external input (RDC2nCON.EXIO = 0b), set RDC2nCON.CVEDS = 1b, and do not set using RD conversion error detection circuit (high-speed rotation).

Note 5. Depends on external circuit.
Input voltage must be adjusted for COSMNT, SINMNT = 0.4 RVCC to 0.6 RVCC(V_{P-P}).

Note 6. Input impedance with on-chip feed-back resistor. This is the default value (±0%).

Note 7. Can be adjusted from -40 to +40% in step of 10% by adjust function.

Note 8. Same as the frequency of resolver signal input.

Note 9. Must be adjusted within this range to keep angle conversion resolution.

Note 10. The phase error of the excitation component of the resolver excitation signal external input and the resolver signal input must be within 45º.

Note 11. When RDC2nCON.EXIO = 0b (i.e. the external excitation signal input is set), this impedance pulls the RSO (R1E) and COM (R2E) pins down to RVSS.

35.5.3 Error Detect Characteristics

Table 35.37 Error Detect Characteristics

Error Detect Item		Set Threshold	Detect Time [ms]
Resolver signal error monitor output amplitude voltage*1	Set register RDC2nCON.REDTH = 0b	0.10 × (RVCC ± 5%) [V_{P-P}]	2 (max.)
	Set register RDC2nCON.REDTH = 1b	0.14 × (RVCC ± 5%) [V_{P-P}]	2 (max.)
	Relative declination between ranges	0.04 × (RVCC ± 5%) [V_{P-P}]	2 (max.)
Breaking of resolver signal (Direct current bias supply method) VSINMNT-VCOM or VCOSMNT-VCOM*2	Set register RDC2nCON.BDVTH = 1b	COM + 0.35 × (RVCC ± 5%) [VDC]	10 (max.)
	Set register RDC2nCON.BDVTH = 0b, RDC2nCON.EXIO = 1b, and RDC2nCON.SENS = 0b	COM + 0.35 × (RVCC ± 5%) [VDC]	10 (max.)
	Setting other than above	COM + 0.08 × (RVCC ± 5%) [VDC]	10 (max.)
R/D conversion error (over control declination) Recognition level for internal control declination (ε)*3	High side	0.55 × (RVCC ± 5%) [VDC]	*4
	Low side	0.45 × (RVCC ± 5%) [VDC]	*4

Note 1. Determined as an error, when both of SINMNT and COSMNT become under threshold.

Note 2. Determined as an error, when DC level changed over threshold.

Note 3. Determined as over, when control declination becomes over high side threshold, or under low side threshold.

Note 4. Determined as an error, when control declination recognition rate becomes over 50% as average of period set by the EDPS [1:0] bits in the RDC2nCONSEL register (default is about 7.4 ms). It might not be detected, when the error continuous period is shorter than detect period.

35.6 Code Flash Characteristics

Table 35.38 Code Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Number of programming times*1	CWRT	Retained for 20 years*2	1000	—	—	Times
Programming temperature	TPRG	Tj	–40	—	+150	°C
Reading temperature	TREAD	Tj	–40	—	+150	°C

Note 1. The number of programming times is the number of erasure of each block. If the number of programming times is n (n = 1000), each block can be erased n times. For example, if 256-byte data is written 128 times to different addresses of a 32KB block and then the bock is erased, the number of programming times is counted as 1. Note, however, that writing data to the same address multiple times for one erasure is not allowed (overwrite is prohibited).

Note 2. This is the case when the average Ta = 85°C. This retained period is from when the erasure of the code flash memory has been normally completed.

Table 35.39 Code Flash Programming Characteristics

Conditions: EVCC = 4.5 to 5.5V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 to 1.35 V
A0VCC, A1VCC = 4.5 to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = -40°C to 150°C

Item	Condition	Block size	Min.	Typ.	Max.	Unit
Programming time	Number of programming times < 100	256 B	—	0.4*1	6*1	ms
		32 KB	—	80	360	ms
	Number of programming times ≥ 100	256 B	—	0.5*1	7.2*1	ms
		32 KB	—	96	432	ms
Erasing time*1	Number of programming times < 100	8 K	—	39	120	ms
		32 KB	—	141	480	ms
	Number of programming times ≥ 100	8 K	—	47	144	ms
		32 KB	—	169	576	ms

Note 1. Only the hardware processing time is included. Software overhead is not included.

35.7 Data Flash Characteristics

Table 35.40 Data Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Number of programming times*1	CWRT	Retained for 20 years*2	125000	—	—	Times
		Retained for 3 years*2	250000	—	—	Times
Programming temperature	TPRG	Tj	-40	—	+150	°C
Reading temperature	TREAD	Tj	-40	—	+150	°C

Note 1. The number of programming times is the number of erasure of each block. If the number of programming times is n (n = 125000), each block can be erased n times. For example, if 4 byte data is written 16 times to different addresses of a 64B block and then the bock is erased, the number of programming times is counted as 1. Note, however, that writing data to the same address multiple times for one erasure is not allowed (overwrite is prohibited).

Note 2. This is the case when the average Ta = 85°C. This retained period is from when the erasure of the data flash memory has been normally completed.

Table 35.41 Data Flash Programming Characteristics

Conditions: EVCC = 4.5 to 5.5 V, SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V, VDD = 1.15 to 1.35 V
A0VCC, A1VCC = 4.5 to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC
RVCC = 4.5 to 5.5 V
VSS = PLLVSS = A0VSS = A1VSS = RVSS = 0 V
Tj = -40°C to 150°C

Item	Block size	Min.	Typ.	Max.	Unit
Programming time*1	4 B	—	0.16	1.7	ms
Erasing time*1	64 B	—	1.7	10	ms
Blank checking time*1	4 B	—	—	30	µs
	64 B	—	—	100	µs

Note 1. Only the hardware processing time is included. Software overhead is not included.

35.8 Thermal Characteristics

35.8.1 Parameters

Table 35.42 Thermal Resistance of RH850/C1x

Package	Parameter	Estimate	Unit	Note
FPBGA1717-252	4LΨjb	27	°C/W	L board (4 layers)
	4LTb_inc	9	°C/W	L board (4 layers)
	4LΨjt	1	°C/W	L board (4 layers)
LQFP2020-144	4LΨjb	30	°C/W	L board (4 layers)
	4LTb_inc	9	°C/W	L board (4 layers)
	4LΨjt	1	°C/W	L board (4 layers)

Note: Thermal resistance and thermal characteristics parameters will change according to the usage environment.

35.8.2 Assumed Board

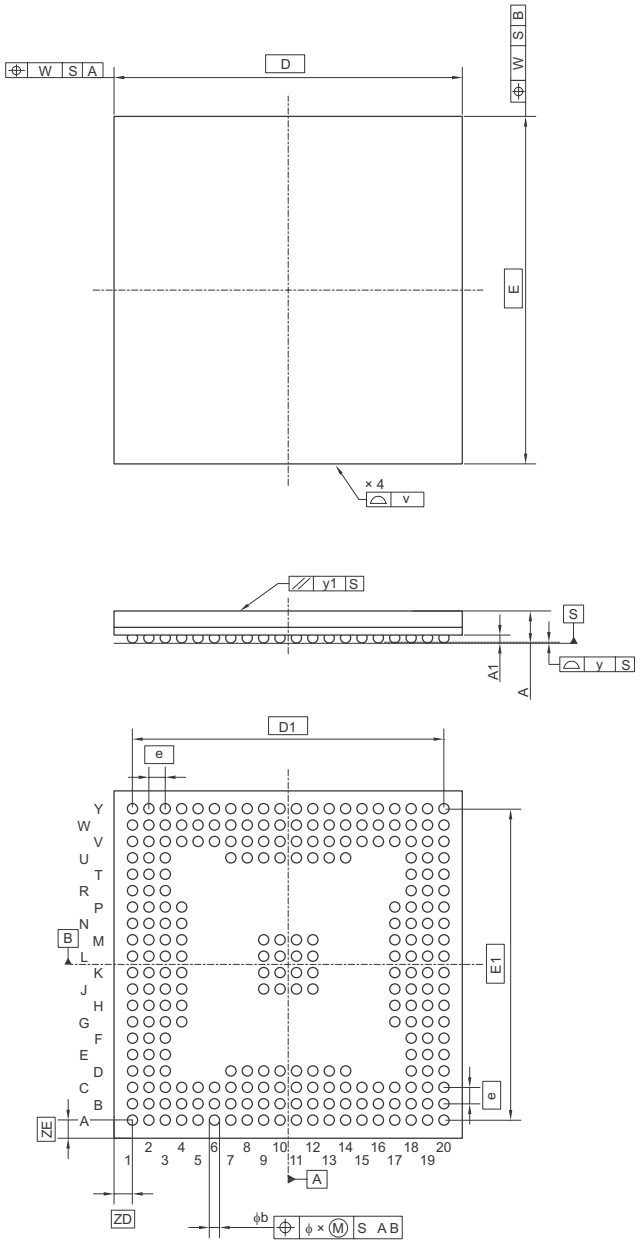
Table 35.43 L Board (4 layers)

Package	Board Size (mm)		Area (mm^2)
	X	Y	
L board	90	160	14400
Remaining copper rates		Thickness of conductors	
30-80-80-30%		35-35-35-35 µm	

Appendix Package Dimensions

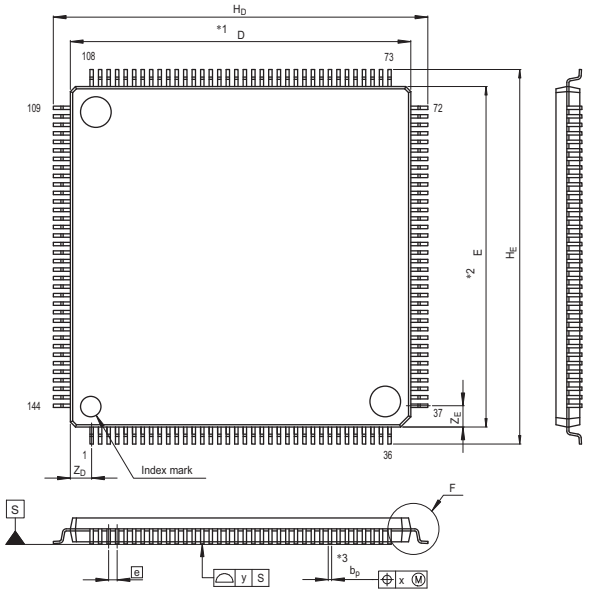
JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-FBGA252-17x17-0.80	PRBG0252GB-A	—	0.90

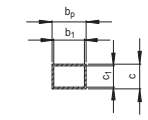
Unit: mm



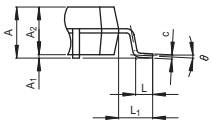
Reference Symbol	Dimensions in millimeters		
	Min	Nom	Max
D	—	17.00	—
D1	—	15.20	—
E	—	17.00	—
E1	—	15.20	—
v	—	—	0.15
w	—	—	0.20
e	—	0.80	—
A	—	1.58	2.00
A1	0.30	0.35	0.40
b	0.49	0.54	0.59
x	—	—	0.08
y	—	—	0.10
y1	—	—	0.20
ZD	—	0.90	—
ZE	—	0.90	—

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LFQFP144-20x20-0.50	PLQP0144KA-A	144P6Q-A / FP-144L / FP-144LV	1.2g





Terminal cross section



Detail F

NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	19.9	20.0	20.1
E	19.9	20.0	20.1
A_2	—	1.4	—
H_D	21.8	22.0	22.2
H_E	21.8	22.0	22.2
A	—	—	1.7
A_1	0.05	0.1	0.15
b_p	0.17	0.22	0.27
b_1	—	0.20	—
c	0.09	0.145	0.20
c_1	—	0.125	—
θ	0°	—	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.10
Z_D	—	1.25	—
Z_E	—	1.25	—
L	0.35	0.5	0.65
L_1	—	1.0	—

REVISION HISTORY	RH850/C1x User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
0.40	Jul 12, 2013	—	First Edition issued
0.41	Feb 05, 2014	Throughout	Renamed, user mat→user area, user boot mat→user boot area, and ICU-S2→ICU-S
		Section 1 Overview	
		50, 51	1.1.3 Internal Block Diagram, added introduction and changed figure (put peripheral circuits in orde in each peripheral group)
		Section 2 Pins	
		65	2.1.2.1 Terms, reviewed the definition of terms
		66	Table 2.3 Pin Function Configuration (Outline), changed Added "Note 2. When the pin is used as an input pin in alternative mode, set PIBCn_m to 0."
		72	Table 2.6 Registers for Port Group Configuration, corrected the address of PFCEn
		77	2.1.4.2 (6) Port Input Buffer Control Register, changed the description
		88	Table 2.24 List of the Pins which Require PIPC Register Setting, added Notes 1, 2 (expressed pins not assigned in RH850/C1M)
		120, 121	Table 2.57 DNF Insertion Targets, added Note 1 (expressed pins not assigned in RH850/C1M)
		123, 125	Table 2.58 C1H Pin Function (2/3), Table 2.59 C1M Pin Function (1/2), deleted (EVSS)
		128-129	2.4.4 Handling of Unused Pins, added the contents
		Section 3 CPU System	
		130	3.1.1 Block Configuration - Global RAM, changed
		134	3.2.1.2 (1) (b) PC, added Note 1
		135	3.2.1.2 (2) Basic System Registers, changed
		137	Table 3.5 EIPSW Register Contents, changed CU2-0 bits
		139	Table 3.7 FEPSW Register Contents, changed CU2-0 bits
		145	3.2.1.2 (2) (k) ASID - Address space ID register, Table 3.15, corrected description of bits 9 to 0
		147	Table 3.19 MEA Register Contents, changed
		151	Table 3.22 RBASE Register Contents, changed RINT bit
		152	Table 3.25 PID Register Contents, changed bits 23 to 8
		154	3.2.1.2 (2) (x) MCFG1, the register figure and Table 3.29 MCFG1 Register Contents, changed bits 15 to 8
		155	Table 3.30 MCTL Register Contents, changed Note 1
		160	Table 3.39 HTCTL Register Contents, changed EN bit
		162	3.2.1.2 (5) (d) VMPRT0, changed
		164	Table 3.46 MPU Function System Registers, changed Note 3
		165	3.2.1.2 (7) (a) MPM, changed the register figure
		166	Table 3.48 MPRC Register Contents, changed E15-E12 bits and Note 1 3.2.1.2 (7) (c) MPBRGN, changed the register figure and Table 3.49 MPBRGN Register Contents
		167	3.2.1.2 (7) (d) MPTRGN, changed the register figure, Table 3.50 MPTRGN Register Contents and Note 1
		168	Table 3.53 MCC Register Contents, changed bit names
		169	3.2.1.2 (7) (i) MPPRT0, changed
		169-171	Table 3.56 MPLAn Register Contents to Table 3.58 MPATn Register Contents, changed Note 1
		173, 174	3.2.1.2 (8) (c) ICDATL, (d) ICDATH, changed
		176	Table 3.66 ICERR Register Contents, changed
		181	3.2.3 Inter-Processor Interrupts, changed interrupt channels
		-	3.2.4.1 (1) (4) Others, deleted
		183	3.2.4.1 (3) List of PEG Protection Setting Registers and Table 3.71 Base Address of PEG Register, changed and deleted Note 1
		184	3.2.4.1 (4) (b) PEGGnMK, changed
		185	3.2.4.1 (4) (c) PEGGnBA, changed the register figure
		186	3.2.4.2 (1) (4) Notifying violation, changed Notes, and (5) Invalidating subsequent accesses, added Notes 1, 2
		187	3.2.4.2 (3) IPG Protection Setting Registers for Illegal Users, changed

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	188	3.2.4.2 (4) IPG Protection Setting Registers for Illegal Virtual Machines, changed
		189-194	3.2.4.2 (5) Register Set, corrected the descriptions and CPU core names
		195	3.2.4.3 System Error Generator Function, changed, Table 3.85 Base Address of SEG Register, corrected -32 of the Operable Bit, changed Note 1, and deleted Note 2
		196-197	Table 3.86 SEGCONT Register Contents, corrected descriptions of bits 9, 4
		200	3.2.4.3 (3) (a) SEG Function, changed
		201	3.2.4.3 (3) (c) Supplementary notes on SYSERR exception, and 3.2.4.4 Checker Core, changed
		202	3.3.3 Exclusive Function, and Table 3.89 List of Registers, changed
		203	3.3.3.2 Operations of the LDL.W and STC.W Instructions, added
		204	3.4.1 Synchronization of Store Instruction Completion and Subsequent Instruction Generation - When updated results in the control registers are reflected in the implementation of a subsequent instruction, changed
		205	3.4.2 Register Access Using a Bit Operation Instruction, and Ensure Coherency after Rewriting the Code Flash, added
		Section 4 Address Space	
		206	4.1.1 Address Space, corrected the description and Note 2, and added Note 3
		207	4.1.2.2 Data Space Accessible by CPU1, and 4.1.2.3 Data Space Accessible by CPU2, corrected
		208	Figure 4.1 Address Space Viewed from Each Bus Master (C1H) - Deleted needless addresses - Corrected accessible areas
		210	4.2.1 Address Space, corrected the description and added Note 1
		211	4.2.2.2 Data Space Accessible by CPU1, corrected
		212	Figure 4.3 Address Space Viewed from Each Bus Master (C1M) - Deleted needless addresses - Corrected accessible areas
		Section 6 Interrupt	
		215	6.1 Overview - Sensing of external interrupts, changed
		216	6.2.1 Configuration of Registers, changed the description
		221	Table 6.5 EIBD register contents, corrected the PEID bit
		223	6.2.7 NMICTL – NMI Interrupt Control Register, corrected the description
		229-230	6.2.12 PINT0 to PINT7, PINTCLR0 to PINTCLR7 - PINTCLRn + x, corrected Value after reset (R→W)
		233	6.3.1 NMI Interrupts, corrected about sensing
			6.3.2 IRQ Interrupts, corrected the description
		236, 242	Table 6.13 Interrupt Exception Handlers and Orders of Priority - Corrected the interrupt names Error control module FE level interrupts→Error control module NMI level interrupts Flash sequencer completion interrupt→Flash sequencer processing completion interrupt - Corrected CAUTIONS
		244	6.5.4.1 Flow of Processing for NMI, corrected about detection
		Section 7 DMA Controller	
		269	7.3.4 Masking and Clearing a Hardware DMA Transfer Request by the DTFR, added description
		272	7.4.3 DTSRAM Error, corrected the description
		273	7.5.2.1 Identifying the Accessing Master to 7.5.2.3 Channel Assignment, corrected the description
		276	Table 7.7 Channel Assignment, corrected the description
		277	7.6.2 Setting Up the Overall DMA Operation, corrected the description
		295-299	7.9.2.7 DTSER2 to 7.9.2.11 DTSCMV, corrected bit names and descriptions of functions
		306	7.9.2.15 DTRERINT, corrected bit name and descriptions of functions
		310	7.9.2.19 DMnnCM - Value after reset, corrected
		314	Table 7.39 DSAn Register Contents, changed table of CAUTION 3
		315	Table 7.40 DDAn Register Contents, changed table of CAUTION 4
		320	Table 7.43 DRSAn Register Contents, changed table of CAUTION
		321	Table 7.44 DRDAn Register Contents, changed table of CAUTION
		322	Table 7.45 DRTCn Register Contents, changed table

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	334	7.11.1.4 Caution about accessing the TI, corrected the register name (DTTCCnnn)
		336	Table 7.54 DTSAnnn Register Contents, changed table of CAUTION
		337	Table 7.55 DTDAnnn Register Contents, changed table of CAUTION 2
		342	Table 7.58 DTRSAnnn Register Contents, changed table of CAUTION
		343	Table 7.59 DTRDAnnn Register Contents, changed table of CAUTION
		344	Table 7.60 DTRTCnnn Register Contents, changed
		Section 9 Power Supply Circuit	
		Throughout	Changed function of reserved bits in register tables
		357	9.1 Features of the RH850/C1x Power Supply Circuit, changed the second table
		358	Figure 9.1 and Figure 9.2 Power: Examples of Connection of Power Management Ics changed the pin name ($\overline{\text{RESET}}$)
		355	Table 9.1 List of Registers, changed the register names
		360	9.3.3 PROT0PHCMD, changed the register figure
		Section 10 Clock Controller	
		363	10.1 Features of the RH850/C1x Clock Controller, changed Note 1
		364	Table 10.2 Clocks and Functional Modules, corrected
		365	Figure 10.1 Clock Controller Block Diagram, corrected (CKSC0 → CKSC0CTL.CKSC[5:0])
		368	Table 10.6 PLL0CLKC1 Register Contents, corrected (SELMPRECENT[2:0] → SELMPERCENT[2:0])
		369	10.2.4 CKSC0CTL, corrected the register figure (CKSC0 → CKSC[5:0])
		Section 11 Clocked Serial Interface H (CSIH)	
		Throughout	Reviewed the structure
		390	Table 11.12 CSIHnCTL1 Register Contents (2/2), corrected functions of CSIHnJE and CSIHnSSE
		392	11.3.4 CSIHnCTL2, corrected names of baud rate setting bits to be selected by CSIHnCFGx.CSIHnBRSS[1:0]
		395-396	Table 11.16 CSIHnSTR0 Register Contents, corrected functions of CSIHnTMOE, CSIHnOFE, CSIHnTSF and CSIHnDCE
		398	Table 11.18 CSIHnSTCR0 register contents, corrected function of CSIHnPCT
		407-408	Table 11.23 CSIHnCFGx Register Contents, corrected functions of CSIHnIDLx and CSIHnIDx[2:0]
		409	Table 11.24 CSIHnTX0W Register Contents, corrected CAUTION of CSIHnEDL
		412	Table 11.26 CSIHnRX0W Register contents, corrected function of CSIHnCSx (x = 3 - 0)
		415	Table 11.29 Interrupt Generation, corrected Cause of interrupt
		417	Table 11.30 INTCSIHTIC Interrupt Generation, corrected Cause of interrupt
		423	Table 11.33 Data Error Types, corrected Communication status after error interrupt and Comment
		424	Table 11.34 INTCSIHTIJC Interrupt Generation, corrected Cause of interrupt
		437	Table 11.36 Dual Buffer Mode, corrected and added the description Table 11.37 Transmit-only Buffer Mode, corrected and added the description
		441	Figure 11.24 EDL Timing Diagram, corrected the description
		443	11.4.16.1 SPI Communication Timing Using SS Function, corrected the description
		445-446	11.4.17.1 Slave Mode, corrected the description
		447	11.4.17.2 Master Mode, added CAUTION
		454	11.4.18.5 Overrun Error, added description
		456	11.4.18.5 (2) FIFO Mode - 2, corrected the description
		457	11.4.19 Loop-back Mode, corrected the description
		458-460	11.4.20 CPU-controlled High Priority Communication Function, corrected Figure 11.47 (PCLK×4 → PCLK×3) and added description
		462-463	11.5.1.1 Transmit/Receive in Master Mode when job mode is Disabled, corrected the description
		465	11.5.1.2 Transmission/Reception in Master Mode when job mode is Enabled, corrected the description
		466-467	11.5.2.1 Transmit/Receive in Master Mode when job mode is Disabled, corrected the description
		468-469	11.5.2.2 Transmit/Receive in Master Mode When job mode is Enabled, corrected the description
		470-471	11.5.3.1 Transmit/Receive in Master Mode when job mode is Disabled, corrected the description

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	472-473	11.5.3.2 Transmit/Receive in Master Mode when job mode is Enabled, corrected the description
		474-475	11.5.3.3 Transmit/Receive in Slave Mode when job mode is Disabled, corrected the description
		476-477	11.5.4.1 Transmit/Receive in Master Mode when job mode is Disabled, corrected the description
		478-479	11.5.4.2 Master in Transmit/Receive Mode when job mode is Enabled, corrected the description
		480-481	Table 11.39 Notes on Setting Registers, corrected Content
		Section 12 Serial Communication Interface 3 (SCI3)	
		482	12.1.3 Clock Supply, added description and Note 1
		483	Table 12.7 External Input/Output Signals, added Note 1
		Section 13 LIN Master Interface (RLIN2)	
		Throughout	Changed function of reserved bits in register tables
		531	13.1.3 Clock Supply, changed Reset Source Table 13.7 Reset Sources, corrected the Reset Source
		533	Figure 13.1 LIN Master Interface Block Diagram, corrected the signal names (LRX0→RLIN20RX, LTX0→RLIN20TX, LRX1→RLIN21RX, LTX1→RLIN21TX, LRX2→RLIN22RX, LTX2→RLIN22TX)
		538	Table 13.14 RLN21nGLSTC register contents, corrected bit description
		549	13.3.3.10 RLN21nmList, corrected Access and R/W (read/written→read, R/W→R)
		551	13.3.3.11 RLN21nmLiEST, corrected Access and R/W (read/written→read, R/W→R)
		Section 14 CAN Interface (RS-CAN)	
		584	Table 14.5 Clock Supply - clk_xincan, corrected (Main clock (external clock input)→Main OSC)
		624	14.3.5 RSCAN0CmERFL - BEF Flag, added NOTE
		633	14.3.9 RSCAN0GERFL - DEF Flag, added description
		634-635	Table 14.23 RSCAN0GTINTSTS0 register contents, corrected description of function
		638	14.3.12 RSCAN0GAFLECTR, corrected the page numbers of Receive Rule Table (page 23 (10111_B)→page 15 (01111_B), "10111_B"→"01111_B")
		646	14.3.17 RSCAN0GAFLP1j, corrected the bit name and description of function
		648	14.3.19 RSCAN0RMNDy - RMNSq Flags, corrected the description
		656	14.3.25 RSCAN0RFSTSx - RFIF Flag and RFMLT Flag, added the description
		667-668	14.3.32 RSCAN0CFSTSk - CFTXIF Flag, CFRXIF Flag and CFMLT Flag, added the description
		670	14.3.33 RSCAN0CFPCTRk, corrected the description of CFPC[7:0] bit
		706-707	14.3.55 RSCAN0TXQCCm - Address, corrected ((10_H X m)→(4_H X m))
		714	14.3.59 RSCAN0THLSTSm - THLIF Flag, and THLELT Flag, added the description
		719	14.3.63 RSCAN0GTSTCTR - RTME bit, changed the format
		734	Figure 14.6 Entry of Receive Rules, corrected the page number (Page 19 →Page15)
		751	Figure 14.16 CAN Setting Procedure after the MCU is Reset, corrected Note
		754	14.5.1.4 Receive Rule Setting, corrected the page number (pages 0 to 19 →pages 0 to 15)
		757-758	14.5.2.1 Receive Buffer Reading Procedure, corrected the description
		Section 16 OS Timer (OSTM)	
		Throughout	Changed function of reserved bits in register tables, and corrected siganl names OTSMnTSST, OSTMnCNT and OSTMnTINT
		791	16.1.1 Units, changed the description
		792	16.1.5 Reset Sources, changed
		794	16.2.3 Counter clock, changed
		796-800	16.3.3 OSTMnCNT to 16.3.9 OSTMnCTL, changed
		801	16.4.1 Starting and stopping the timer, changed
		802-804	16.4.2.1 Basic operation in interval timer mode, changed
		803, 804	16.4.2.1 Figure 16.3 and Figure 16.4, added OSTMnTTOUT signal
		805	16.4.2.2 Operation when OSTMnCMP = 0000 0000_H, changed
		805, 807, 808	16.4.2.2 Figure 16.5 to 16.4.3.2 Figure 16., added OSTMnTTOUT signal
		Section 17 Timer Array Unit D (TAUD)	
		814	Table 17.9 Functional List of TAUD Operations, changed the item name
		816	17.2.5 Block Diagram, changed
		825	Table 17.15 TAUDnCNTm Read Values after Re-Enabling Counter, changed
		831	Table 17.18 TAUDnCSRm Register Contents, changed

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	852	17.4.3.4 (1) Simultaneous Rewrite when the Master Channel (Re)starts Counting - changed Description (1)
		856	Figure 17.7 Simultaneous Rewrite When INTTAUDnIm Is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm, changed
		858	Figure 17.8 Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal, changed
		861	Table 17.43 Channel Output Modes, corrected TAUDnTME.TAUDnTMEm of Synchronous channel output mode 1 with non-complementary modulation output (1 → X)
		867	Figure 17.12 Set/Reset Conditions for Synchronous Channel Output Mode 2 with One-Phase PWM Output, changed
		873	17.4.7.1 Count Capture Mode, changed
		877	Table 17.45 TAUDnCMORm Settings for Interval Timer Function, changed
		884	Table 17.50 TAUDnCMORm Settings for TAUDnTTINm Input Interval Timer Function, changed
		890	Table 17.55 TAUDnCMORm Settings for Clock Divide Function, changed
		897	Table 17.60 TAUDnCMORm Settings for External Event Count Function, changed
		903	Table 17.64 TAUDnCMORm Settings for Delay Count Function, changed
		907	Table 17.68 TAUDnCMORm Settings for One-Pulse Output Function, changed
		910-911	17.4.9.7 (1) Overview - Functional description, changed
		913	Table 17.74 TAUDnCMORm Settings for TAUDnTTINm Input Pulse Interval Measurement Function, changed
		920	Table 17.79 TAUDnCMORm Settings for TAUDnTTINm Input Signal Width Measurement Function, changed
		923	Figure 17.54 TAUDnCMORm.TAUDnCOS[1:0] = 10_B, TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 11_B, changed the title
		925	17.4.9.9 (1) Overview - Functional description and (2) Equations, changed
		926	Figure 17.57 General Timing Diagram of TAUDnTTINm Input Position Detection Function, changed
		927	Table 17.83 TAUDnCMORm Settings for TAUDnTTINm Input Position Detection Function, changed
		928	Table 17.86 Operating Procedure for TAUDnTTINm Input Position Detection Function, changed
		929	Figure 17.58 Operation Stop and Restart (TAUDnCMORm.TAUDnMD0 = 0, TAUDnCMURm.TAUDnTIS[1:0] = 00_B), changed
		930	17.4.9.10 (1) Overview - Functional description, and (2) Equations, changed
		931	Figure 17.59 Block Diagram of TAUDnTTINm Input Period Count Detection Function, changed Figure 17.60 General Timing Diagram of TAUDnTTINm Input Period Count Detection Function, changed
		932	Table 17.87 TAUDnCMORm Settings for TAUDnTTINm Input Period Count Detection Function, changed
		933	Table 17.90 Operating Procedure for TAUDnTTINm Input Period Count Detection Function, changed
		934	Figure 17.61 Operation Stop and Restart (TAUDnCMURm.TAUDnTIS[1:0] = 11_B), changed
		937	Table 17.91 TAUDnCMORm Settings for TAUDnTTINm Input Pulse Interval Judgment Function, changed
		941	Table 17.95 TAUDnCMORm Settings for TAUDnTTINm Input Signal Width Judgment Function, changed
		946	Table 17.99 TAUDnCMORm Settings for Real-Time Output Function Type 1, changed
		953	Table 17.105 TAUDnCMORm Settings for Real-Time Output Function Type 2, changed
		961	Figure 17.73 Block Diagram of Simultaneous Rewrite Trigger Generation Function Type 1, changed
		962	Table 17.111 TAUDnCMORm Settings for Simultaneous Rewrite Trigger Generation Function Type 1, changed
		970	17.4.12.1 (4) (d) Simultaneous rewrite for master channels, changed
		990	17.4.12.3 (2) Equations, NOTE 2, changed
		1012	Table 17.154 Operating Procedure for Offset Trigger Output Function, changed
		1016	17.4.12.5 (3) General Timing Diagram, deleted the title
		1044	17.4.12.9 (1) Overview - Functional description, changed
		1054	Figure 17.109 General Timing Diagram of One-Phase PWM Output Function, changed

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	1055	Table 17.186 TAUDnCMORm Settings for One-Phase PWM Output Function, changed
		1060	17.4.13.1 (1) Overview - Conditions, changed
		1071	Table 17.203 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (2/2), changed
		1073	17.4.13.2 (1) Overview - Prerequisites, changed
		1074	17.4.13.2 (1) Overview - Conditions, changed
		1077	Figure 17.114 General Timing Diagram of Non-Complementary Modulation Output Function Type 2, changed
		1083	Table 17.214 Control Bit Settings for Slave Channels 2 to 7 in Synchronous Channel Output Mode 2 with Non-Complementary Modulation Output, changed
		1085	Table 17.216 Operating Procedure for Non-Complementary Modulation Output Function Type 2 (2/2), changed
		1086	Figure 17.115 Specific Timing Diagram of Non-Complementary Modulation Output Function Type 2, changed
		1088	17.4.13.3 (1) Overview - Functional description, changed
		1099	Table 17.230 TAUDnCMORm Settings for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function, changed
		1102	Table 17.234 Operating Procedure for Complementary Modulation Output Function (2/2), changed
		1103	17.4.13.3 (9) Specific Timing Diagrams, changed
		Section 18 Timer Array Unit J (TAUJ)	
		1105	18.1.3 Clock supply, changed
		1110	18.2.2 Block Diagram, changed
		1119	Table 18.14 TAUJnCNTm Read Values after Re-Enabling Counter, added
		1123	Table 18.17 TAUJnCSRm register contents, changed
		1149	Table 18.37 Operating procedure for interval timer function, changed
		1158	18.4.8.3 (1) Overview - Description, changed
		1172	18.4.8.5 (1) Overview - Description, and (2) Equations, changed
		1173	Figure 18.34 General timing diagram for TAUJnTTINm input position detection function, changed
		1174	Table 18.53 TAUJnCMORm register contents, changed
		1175	Table 18.56 Operating procedure for TAUJnTTINm input position detection function, changed
		1176	Figure 18.35 Operation stop and restart, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B, changed
		1177	18.4.8.6 (1) Overview - Description, changed
		1178	18.4.8.6 (2) Equations, and Figure 18.37 General timing diagram for TAUJnTTINm input period count detection function, changed
		1179	Table 18.57 TAUJnCMORm register contents, changed
		1180	Table 18.60 Operating procedure for TAUJnTTINm input period count detection function, changed
		1181	Figure 18.38 Operation stop and restart, TAUJnCMURm.TAUJnTIS[1:0] = 11_B, changed
		Section 19 Motor Control Timer (TSG3)	
		Throughout	Reviewed overall expressions and forms of descriptions
		1196	19.2.1 Functional Overview, changed the description (TOP function→ timer option (TAPA) function)
		1198	Figure 19.1 TSG3n Block Diagram, corrected
		1228	Table 19.27 TSG3nOPT2 Register Contents, changed the description
		1229	Table 19.28 TSG3nOPT2BF Register Contents, changed the description
		1234	Table 19.34 TSG3nCMP0E Register Setting, corrected Maximum Value of HT-PWM mode (TSG3nDTC0+TSG3nDTC1+TSG3nCMP0E → $3FFFE_H$)
		1264	Table 19.48 Shift Width and Duty Setting in HSP-PWM Mode, corrected addresses of TSG3nHSPCMVE and TSG3nHSPCMUE registers
		1274	Figure 19.16 Interrupt Generation Example (Example of HT-PWM Mode), corrected (INTTSG3nI1, 5 and 9 generation at count up → at count down, INTTSG3nI2, 6, 10 generation at count down → at count up)

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	1298	Figure 19.39 When TSG3nRMC = 1 in TSG3nCTL3 (Anytime Rewrite Mode), deleted CAUTION below
		1302	Figure 19.42 When TSG3nPIE = 1, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 00_B in TSG3nCTL4, and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode), corrected
		1303	Figure 19.43 When TSG3nPIE = 0, TSG3nVIE = 1, and TSG3nRCC04 to TSG3nRCC00 = 02_B in TSG3nCTL4 and TSG3nACC01 and TSG3nACC00 = 00_B in TSG3nCTL5 (HT-PWM Mode), corrected bit name
		1305	Figure 19.46 Example of Operation of A/D Conversion Trigger Skipping Function, corrected a setting value
		1344	19.4.7.3 Data Transfer from EMU2, changed description and figure
		1347-1349	19.4.7.4 ESW Function, changed description and figures
		1354	Table 19.77 Example of Setting Each Timer Output Condition in SP-PWM Mode, changed Setting Condition of TSG3nO2, TSG3nO4, TSG3nO6
		1399	Table 19.89 Functions of Compare Registers, Sift Width Setting Register, and Dead Time Setting Register in HSP-PWM, changed Rewrite during Operation of SG3nDTC0W, TSG3nDTC1W (Possible→Prohibited)
		Section 20 Timer Option Module (TAPA)	
		1419	20.2.1 Functional Overview, changed description (analog inputs→ TAPA input signals)
		1424	Table 20.9 TAPAnFLG Register Contents, deleted needless term Table 20.10 TAPAnACWE Register Contents, changed description (to TAPA0ACTS and TAPA0ACTT→to TAPA0ACTS or TAPA0ACTT)
		1425	Table 20.12 TAPAnACTT Register Contents, corrected (if TAPAnACWE = 0 →if TAPAnACWE = 1)
		1427	20.4.1.1 Overview, reviewed the expression, and cleared intended timer output pins
		1428	Figure 20.4 An Example of System Configuration of Asynchronous Hi-Z Control for Analog Inputs, changed
		1433	Figure 20.8 Example of Operations in Response to Master Channel 0 of TAUD, changed
		1435	Table 20.17 Operation of TAPAnCTL1.TAPAnATS[3:2] and TAPAnTADOUT1, added
		Section 21 Timer Pattern Baffer (TPBA)	
		1442	Figure 21.1 Block Diagram of TPBA, corrected signal name and added description -TPBAnSST → TPBAnTSST -TPBAnTSST: simultaneousl atart trriger (from PIC1A)
		1448	21.3.7 TPBAnTO — TPBAn Timer Output Register, changed the description
		Section 22 Encoder Timer (ENCA)	
		Throughout	Corrected register names ENCANCCR0 → ENCAnCCR0 ENCANCCR1 → ENCAnCCR1 ENCASST → ENCAnTSST
		1467	22.1.1 Units, added channel description
		1468	Table 22.6 ENCAn Reset Source, changed the title
		1470	Figure 22.1 Encoder Timer block diagram, corrected the description ENCAnCCR0 (capture/compare) → ENCAnCCR1 (capture/compare)
		-	22.2.3 Preliminary knowledge for understanding basic specifications, deleted
		1473	Table 22.9 ENCAnCTL register contents (2/2) Corrected function of ENCAnUDS[1:0] bits (counter up/down control register→ counter up/down control bits)
		1477	Table 22.12 ENCAnFLG register contents, changed the description of ENCAnCSF bit, and corrected function (ENCASST → ENCAnTSST)
		1482	Table 22.17 ENCAnTE register contents, changed the description of ENCAnTE bit
		1489	22.4.2.3 When ENCAnUDS1 and ENCAnUDS0 = {1, 0}, corrected the description (ENCAnUDS1 and ENCAnUDS0 = {0, 1} → ENCAnUDS1 and ENCAnUDS0 = {1, 0})
		1495	Table 22.28 Compare match interrupt detection mask function, changed the item name
		1497	22.4.6.1 Startup of timer, changed the description, and 22.4.6.2 Starting the Timers in a Dual Encoder-Timer Configuration, corrected (ENCAnTE → ENCAnTS)
		1498	22.4.6.3 Stop of timer, deleted Figure 22.9 and Figure 22.10 22.4.6.4 Stopping the Timers in a Dual Encoder-Timer Configuration - Example 1 - Restart procedure, changed

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	1500	Table 22.31 Encoder Timer Setting Procedure, changed
		1501	Figure 22.13 Setting procedure for ENCAnCCR1 register, changed
		1510	Figure 22.24 Conflict between overflow occurrence and clear operation by phase Z input, corrected signal name (ENCAnUOVF → ENCAnOVF)
		1517	22.6.15 (5), deleted "by APB" from description
		1521	Figure 22.34 Capture operation between count clocks, corrected signal name (ENCAATZIN1 → ENCATTIN1)
		1522	Figure 22.35 Capture operation between count clocks, corrected signal name (ENCAATZIN0, ENCAATZIN1 → ENCATTIN0)
		1523	22.6.20 and Figure 22.36, changed the title (Encoder operation when ENCAnECM1-0 = {0, 1} and ENCAnCTS = 0 → Encoder operation when compare match clear control is enabled and ENCAnCTS = 0)
			Figure 22.36 Encoder operation when compare match clear control is enabled and ENCAnCTS = 0, corrected signal names (ENCAATZIN1 → ENCATTIN1, ENCATINT1 → INTENCAnI1)
		1524	22.6.21 and Figure 22.37, changed the title (Encoder operation when ENCAnECM1-0 = {0, 1} and ENCAnCTS = 1 → Encoder operation when compare match clear control is enabled and ENCAnCTS = 1)
		1525	22.6.22 and Figure 22.38, changed the title (Encoder operation when ENCAnECM1-0 = {0, 0} → Encoder operation when compare match clear control is disabled)
			Figure 22.38 Encoder operation when compare match clear control is disabled, corrected signal name (ENCATTIN0 → ENCAnI0)
			22.6.22 (1), corrected description (ENCAnCRM1=1 → ENCAnCRM1=0)
		1530	Figure 22.44 Capture operation performed upon clearing by ENCAnEC when ENCAnSCE = 0, corrected signal name (ENCAnCN → ENCAnCNT)
		Section 23 Peripheral Interconnection (PIC)	
		1534, 1535	Table 23.9 Registers and Table 23.10 Registers Used by Each Function, deleted control register PIC1AEN
		-	23.2.2.2 PIC1AEN, deleted
		1540	23.2.2.8 PIC1AINIn0, corrected the description
			23.2.2.9 PIC1AINIn1, corrected the description
		1557	23.2.2.25 PIC1AREG2n1 - Address, corrected: FF81 C0C4$_H$ (n=0), FF81 C0D8$_H$ (n=1) →FFDD 00C4$_H$ (n=0), FFDD 00D8$_H$ (n=1)
		1565	Table 23.38 PIC1AREG50 register contents, corrected the description of bits 6, 5
		1566	Table 23.39 PIC1AREG51 register contents, corrected the description of bits 6, 5
		1613	Figure 23.27 Timing Diagram of Two-Phase Encoder Control Function (Control Method 1) at Up Count (Normal Rotation), corrected the signal name (TSnOPCI0→TSG3nOPCI0)
		1620-1622	23.2.3.8 (5), changed flow charts
		1628-1629	23.2.3.9 (5), changed flow charts
		1652	23.2.3.14 (3) Registers, corrected the description
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1692	Table 24.38 EMU2nIPTRG Register Contents, corrected Note 2
		1753	Table 24.104 EMU2nSUMIDM Register Contents, changed the description
		1780-1781	24.3.110 EMU2nPTNCD and 24.3.111 EMU2nPTNEF, changed the description
		1787-1791	24.3.121 EMU2nIRUCPPN0 to 24.3.129 EMU2nIRWCPPN2, added description
		1798	Figure 24.3 Initial Setting Flow for EMU2, corrected setting value for protection
		1806	24.4.4 (3) Generating Electrical Angle, reviewed the description
		1813	24.4.6 PI Control IP, changed the description Figure 24.11 Process Flow of PI Control IP, corrected the register names
		1825	24.4.10 Batch Rectangle IP, reviewed the description and added Note 5
		1826	Figure 24.19 Overview of Rectangular Wave Output, deleted Note 1
		1827	24.4.11 Independent Rectangle IP, deleted repeat description
		Section 25 R/D Converter (RDC2)	
		1837	Table 25.4 Clock Supply, corrected the clock name (CLK_LSB → CLKC_LSB)
		1839	25.1.6 External Input/Output Signals, added description below to Note 1; These pins must not be used for input amplifier monitor output pins and for A/D conversion pins simultaneously.

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	1860	25.3.15 RDC2nBISTC - BSTF Bit, corrected the setting value
		1861	25.3.17 RDC2nINGR - Value after reset, changed expression from 2-adic number to 16-adic number
		1863	25.3.19 RDC2nEXAAT - Value after reset, changed expression from 2-adic number to 16-adic number
		1867	25.3.21 RDC2nCMINT - Value after reset, changed expression from 2-adic number to 16-adic number Changed bit names (INTC[2:0] → INTCLR[2:0], INTF[2:0] → INTFLG[2:0])
		1882	25.4.3.1 Error Detection Function, corrected table number to 25.53 in the description
		1883	25.4.4.1 (1), corrected
		1885	25.4.5.1 Period Measurement Timer, added calculation of the cycle of excitation signal
		Section 26 A/D Converter (ADCC)	
		Throughout	Reviewed the overall structure
		Section 27 Functional Safety	
		1963	27.1 Overview - Lockstep, added CAUTION
		1965	Table 27.2 Address Parity Overview, changed
		1966	27.2.2.1 Overview, changed
		1967	Figure 27.1 ECC and Address Parity of Code Flash, added
		1968	Table 27.3 List of Registers, changed
		1974	27.2.2.3 (6) CF1STERSTR_VCI/PE1/PE2, changed
		1976	27.2.2.3 (8) CFSTSTCTL_VCI/PE1/PE2, added
		1977	27.2.2.4 Test Function, added
		1978	27.2.3.1 Overview, changed
		1979	27.2.3.2 List of Registers, changed
		1985	27.2.3.2 (10) DFTSTCTL, added
		1985-1986	27.2.3.3 Test Function, added
		1987-1988	27.2.4.1 Overview, changed
		1989	Table 27.23 List of Registers, changed
		1991	27.2.4.3 (2) LRTSTCTL_PE1/PE2, added
		1992	27.2.4.3 (3) LRTDATBFn_PE1/PE2, added
		1999-2000	27.2.4.4 Test Function, added
		2001-2002	27.2.5.1 Overview, changed
		2003	27.2.5.2 List of Registers, changed
		2005-2006	27.2.5.3 (2) GRTSTCTL, added
		2007	27.2.5.3 (3) GRTDATBFn, added
		2009	27.2.5.3 (5) GRDECINBF0, added
		2010	27.2.5.3 (6) GRDECINBF1, added
		2016-2018	27.2.5.4 Test Function, added
		2019	27.2.6.1 Overview, changed
		2020	Table 27.45 List of Registers, changed
		2032	27.2.6.4 Test Function, added
		2033	27.2.8.1 Overview - Error Detection and Correction, changed CAUTION
		2045-2047	27.2.8.5 Test Function, added
		2066	27.4.3.2 (1) FSGDxxDPROTn, changed
		2068	27.4.3.2 (4) ERRSLVxxADDR, changed
		2069	Table 27.81 ERRSLVxxTYPE Contents, changed
		2070	27.5.1 Overview, changed
		2071-2073	27.5.2 Block Diagram, added
		2074-2076	27.5.3 Functional Specification, added
		2079-2092	27.5.4.2 MISRCDRL_PE1/PE2 to 27.5.4.15 MISRERRCTL, added
		2093-2094	27.5.5 Usage, added
		Section 28 Error Control Module (ECM)	
		2104	Table 28.3 Clock Supply, corrected the clock name (CLKC_LSB → CLK_LSB)
		2106	Table 28.7 Function Overview, changed description of Error status

Rev.	Date	Description	
		Page	Summary
0.41	Feb 05, 2014	2107	28.2.2 Block Diagram, changed the description Figure 28.1 Block Diagram of ECM, added the ERROROUTZ output
		2108-2109	Table 28.8 List of Error Inputs, changed No.22, No.23, No.26, No.42, No.44, No.46 and No.62
		2110	Table 28.9 Error Source Aggregation, changed No.23, and added No.22, No.46 and No.48
		2111	28.3.1 List of Registers, changed the description
		2115, 2116	28.3.4 ECMmESSTR0 and 28.3.5 ECMmESSTR1, changed the description
		2118, 2119	28.3.7 ECMmESSTR0n and 28.3.8 ECMmESSTR1n, changed the description
		2121-2124	28.3.10 ECMMICFG0 to 28.3.13 ECMNMICFG1, changed the description
		2125	28.3.14 ECMIRCFG0, changed Value after reset of bits 3 to 0 in register figure (0 → 1)
		2134	28.3.25 ECMDTMR, changed the description
		2140	28.4.1 Operations for Error Output, added Note 1
		2141	28.4.4 Error Status and 28.4.5.1 Protection Unlock Sequence, changed the description
		Section 29 Data CRC (DCRA)	
		2143	Changed the introduction
		2144	29.2.1 Functional Overview, changed the description
			Figure 29.1 Block diagram of Data CRC Function A, changed
		2147	29.3.3 DCRAnCOUT, changed
		2148	29.3.4 DCRAnCTL, changed
		2149	29.4 Function, changed NOTES
		Section 31 On-Chip Debugging Unit (OCD)	
		2153	31.3 Peripheral Break Control, added Note 2
		2157	Table 31.3 List of Registers, written AUDMBRC with AUDMB
		2159	31.4.3.2 AUDMBR/AUDMBRC, changed (added AUDMBRC)
		2168	31.4.4.4 Enabling and Disabling the RAM Monitor Function, added NOTE
		Section 32 Flash Memory	
		2170, 2171	Figure 32.1 Code Flash Memory Mapping (8 Kbytes × 8 + 32 Kbytes × 126 configuration) _C1H, corrected addresses: 001E 8FFF$_H$ → 001F 8000$_H$, 001E FFFF$_H$ → 001F FFFF$_H$ Figure 32.2 Code Flash Memory Mapping (8 Kbytes × 8 + 32 Kbytes × 62 configuration) _C1M, corrected addresses: 0080 0000$_H$ → 001F 8000$_H$, 001E FFFF$_H$ → 001F FFFF$_H$
		2172	Figure 32.3 Data Flash Memory Mapping (64 bytes × 512 configuration), corrected the title: 64 Kbytes → 64 bytes
		2179	32.6.1 Overview, corrected the description
		2182	Table 32.11 FRDCYCLD register contents, corrected function (0x0$_H$ to 0x2$_H$:Setting prohibited → 0x0$_H$ to 6$_H$:Setting prohibited)
		2185	32.9.3.1 PRDNAMEn - Value after reset, corrected
			32.10.1 OPBT0 - Access ,Value after reset and R/W, corrected
		2187	32.10.2 OPBT2 - Access ,Value after reset and R/W, corrected Table 32.18 OPBT2 register contents, corrected Debug interface of bits 30, 29: FLSCI3 → FLSCI3 (writer interface)
		2188	32.11 Notes (4), (8), corrected the description
		Section 35 Boundary Scan	
		2197	35.3 Register Descriptions, reviewed the description, and corrected Value after reset of ID register
		Section 36 Electrical Characteristics	
		2203	Table 36.1 Absolute Maximum Ratings, added 5V-torelant pin to Input voltage, and added Note 4
		2212	Table 36.12 Power On/Off Timings, changed the description of Note 1
		2213	Table 36.13 Power On/Off Timings, reviewed the item, and deleted Note 1 Figure 36.3 Power On/Off Timings when EPT is Used, changed following above
		2214	36.3.2.2 Oscillation Frequency Accuracy of the Low Speed On-Chip Oscillator, corrected the title
		2219, 2222	Figure 36.8 CSIH Timing (Master Mode) (1/4) and (4/4), corrected a pin name
		2227	Table 36.22 SCI3 Timing (Master Mode), added lacking items
		2229	36.3.7 RS-CAN Timing, corrected the title and pin names
		2234	Table 36.31 A/D Converter Characteristics, corrected Input voltage range
		2238, 2239	36.6 Code Flash Characteristics, and 36.7 Data Flash Characteristics, reviewed the items

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	All	Name changed: ICU-S→ICUSB
		Section 1 Overview	
		50	Table 1.1 Overview of Products (2/2): Items of the A/D converter and I/O power supply of power supply voltage changed (Number of input pins → deleted, 3.3V ± 0.3V→ deleted)
		52	Figure 1.1 Internal Block Diagram of RH850/C1H: Changed (Arrowed lines added between CPU1/CPU2 and P-Bus)
		53	Figure 1.2 Internal Block Diagram of RH850/C1M: Changed (An arrowed line added between CPU1 and P-Bus)
		Section 2 Pins	
		All	Name changed: Initial value → value after reset
		85	Table 2.23 PCRn_m Register Contents: Notation and functional description of the reserved bits (b31 to 26, b23 to 20, b15 to 13, b11 to 9, b7, and b3) changed
		92 to 98	Table 2.25 to 2.31, List of Registers in C1H Port Group 0 to 6: Erroneous description of the bitmap corrected
		108 to 111	Table 2.42 to 2.45, List of Registers in C1M Port Group 1 to 4: Erroneous description of the bitmap corrected
		109	Table 2.43 List of Registers in C1M Port Group 2: Changed
		121	2.3.2.3 DNFP01nCTLm — Digital Noise Elimination Control Register: Name bit changed (b6, b5, and b2 to 0) and functional description (b4 and 3) changed
		124	Table 2.58 C1H Pin Function (1/3): Changed (FLSCI3TX (FPDT), FLSCI3RX (FPDR), and FLSCI3SCK (FPCK))
		126	Table 2.58 C1H Pin Function (3/3): Functional description of pin X1 and X2 changed (Main OSC → Crystal oscillator)
		127	Table 2.59 C1M Pin Function (1/2): Changed (FLSCI3TX (FPDT), FLSCI3RX (FPDR), and FLSCI3SCK (FPCK))
		128	Table 2.59 C1M Pin Function (2/2): Functional description of the Pin X1 and X2 changed (Main OSC→ Crystal oscillator)
		130	Table 2.61 Pin State (2/2): Debug system added, Note 1 added
		Section 3 CPU System	
		All	Notation of bit name unified: ASID→ASID9-0, PEID→PEID2-0, VCID→VCID2-0, TCID→TCID7-0, MEA→MEA31-0, DATAECC→DATAECC7-0, and TAGECC→TAGECC7-0 Term unification: SysErr → SYSERR, peripheral function → peripheral device, CodeFlash→Code Flash, the external master → the bus master, Operable Bit in the table header → Access Size
		133	3.1.1 Block Configuration: Description of local RAM and code flash changed
		134	Slave Guard (1) to (3): Description changed
		135	Table 3.1 Features of the RH850 G3M Core: Description of the CPU added
		137	3.2.1.2 (1) (a) General-purpose registers: Description changed
		139	3.2.1.2 (2) (a) EIPC — Status save register when acknowledging EI level exception: Description changed
		141	3.2.1.2 (2) (c) FEPC — Status save register when acknowledging FE level exception: Description changed
		144	Table 3.9 PSW Register Contents (2/3): Description of b16 changed (CU0 bit 16 → CU0 bit)
		151	U bit is added to bit 8 in the bit assignment chart
		154	Table 3.22 RBASE Register Contents: Description of b31 to 9 changed.
		154	Table 3.22 RBASE Register Contents: Description of b0 changed
		154	Table 3.23 EBASE Register Contents: Description of b31 to 9 changed
		154	Table 3.23 EBASE Register Contents: Description of b0 changed
		155	Table 3.24 INTBP Register Contents: Description of b31 to 9 changed
		157	3.2.1.2 (2) (w) MCFG0 — Machine configuration register: Notation of b4 deleted
		157	Table 3.28 MCFG0 Register Contents: Description of b1 and b0 changed
		157	3.2.1.2 (x) MCFG1 — Machine configuration register: Value after reset changed
		158	3.2.1.2 (2) (y) MCTL — Machine control register: Note 1 deleted
		159	Table 3.32 FPIPR Register Contents: Description of b4 to 0 changed, Note deleted
		162	Table 3.35 ICSR Register Contents: Description of b1 and b0 changed
		166	3.2.1.2 (6) FPU Function Registers: Description changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	169	3.2.1.2 (7) (c) MPBRGN — MPU base region register: Value after reset changed
		170	3.2.1.2 (7) (d) MPTRGN — MPU end region register: Value after reset changed
		171	3.2.1.2 (7) (h) MCR — Memory protection setting check result register: Description changed
		172	Table 3.56 MPLAn Register Contents: Description of b31 to 2 changed
		173	Table 3.57 MPUAn Register Contents: Description of bits 31 to 2 changed (implicitly set to 0 → implicitly set to 1)
		181	3.2.2.1 Features: Description changed (CPU1 → CPU)
		182	3.2.2.2 Instruction Cache Function: Caution in LRU added
		185	3.2.4.1 (1) Overview of the PEG Function: Description changed
		186	3.2.4.1 (3) List of PEG Setting Registers: Description changed
		187	3.2.4.1 (4) (a) PEGSP — PE guard SPID master decision control register: Description changed
		187	Table 3.72 PEGSP Register Contents: Description of b0 changed
		188	3.2.4.1 (4) (b) PEGGnMK — PE guard protection area n mask setting register: Description changed
		189	3.2.4.1 (4) (c) PEGGnBA — PE guard protection area n base setting register (n = 0 to 3): Description changed
		190	3.2.4.2 Internal Peripheral Guard Function (IPG): Title changed
		190	3.2.4.2 (1) Overview of the IPG Function: Description changed, Note changed
		191	3.2.4.2 (2) IPG Function: Description changed
		196	Table 3.81 Register Contents of IPGPMTUM1, IPG0PMT1, IPG0PMTUM1: Description of b6 and b2 changed (GRAM → Global RAM)
		198	Table 3.83 Register Contents of IPGPMTUM3, IPG0PMT3, IPG0PMTUM3: Description of b6 to b4 changed (SysErrGen → SEG)
		199	3.2.4.3 System Error Generator Function (SEG): Body changed
		200, 201	Table 3.86 SEGCONT Register Contents: Description of b9 and b4 changed, Note added
		202	Table 3.87 Error Factor Codes and Handling of G3M Core SYSERR Exceptions: added
		203	3.2.4.3 (2) (b) SEGFLAG — Error occurrence retention register: Description changed
		204	Table 3.89 SEGSIDE Register Contents: Description changed
		206	3.2.4.4 Checker Core: Caution added
		207	3.3.2 Inter-Processor Interrupt Function: Description changed
		209	3.4.1 (4) Subsequent instruction (EI): Description of example 2 changed (function → device)
		Section 4 Address Space	
		211	Table 4.1 Address Space (C1H): Address and address space type changed
		213	Figure 4.1 Address Space Viewed from Each Bus Master (C1H): Changed
		215	Table 4.2 Address Space (C1M): Address and address space type changed
		217	Figure 4.3 Address Space Viewed from Each Bus Master (C1M): Changed
		Section 6 Interrupt	
		220, 221	6.1 Overview: Description changed and added
		222	Table 6.1 Interrupt Control: Changed
		222	6.2.1 Configuration of Registers: Body changed
		223	Table 6.3 Merging of Interrupts: Register name and R/W property changed (Interrupt clear register n → peripheral interrupt status clear register n)
		224	6.2.2 EI Level Interrupt Control Registers 0 to 255 (EIC0 to EIC255): Body changed and deleted, Caution added, name of b3 to b0 changed
		224, 225	Table 6.4 EIC register contents: Description of b12 changed, Note added
		226	6.2.3 EI Level Interrupt Mask Registers 0 to 7 (IMR0 to IMR7): Note added
		227	6.2.4 EI Level Interrupt Bind Registers 0 to 255 (EIBD0 to EIBD255): Description changed
		227	Table 6.5 EIBD register contents: Note and Caution added
		—	6.2.5 FNC — FE Level NMI Control Register: Deleted
		—	6.2.6 FIC — FE Level Interrupt Control Register: Deleted
		228	6.2.7 NMICTL — NMI Interrupt Control Register: Bit name changed (b1, 0)
		229	6.2.8 EXINTCTL — External Interrupt Control Register: Bit name changed
		232	6.2.11 SINTR0 to SINTR3 — Software Interrupt Register: Bit name changed
		233	6.2.12 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers: Title changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	235, 236	Bit position of PINT0 to PINT7 and PINTCLR0 to PINTCLR7 changed
		238	6.3 Interrupt Sources: Body changed
		238	6.3.1 NMI Interrupts: Body changed
		238	6.3.2 IRQ Interrupts: Description changed
		238	6.3.3 ECM Interrupts: Title and body changed
		238	6.3.4 Inter-Processor Interrupts: Added
		239	6.3.5 Software Interrupts: Body changed
		239	6.3.6 Internal Peripheral Module Interrupts: Module names and symbols changed, description changed
		240	6.4 Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority: Description changed
		241	Table 6.13 Interrupt Exception Handlers and Orders of Priority (1/6): Changed
		245	Table 6.13 Interrupt Exception Handlers and Orders of Priority (5/6): Changed (ICU-S → ICUSB)
		246	Table 6.13 Interrupt Exception Handlers and Orders of Priority (6/6): Caution changed
		248	6.5.1 External Interrupts (NMI and IRQ): Body changed
		248	6.5.2 Inter-Processor Interrupts: Added
		249	6.5.5.1 Flow of Processing for NMI: Description added
		249	Figure 6.1 Flow of Processing for NMI: Changed
		251	Figure 6.2 Flow of Processing for External Interrupts: Changed
		252	6.5.5.3 Inter-Processor Interrupt Flow: Added
		253	6.5.5.4 Software Interrupt Processing Flow: Added
		254	Figure 6.4 Software Interrupt Processing Flow: Changed
		256	Figure 6.5 Flow of Processing for DTS Interrupts: Changed
		257	6.6 Interrupt Response Times: Added
		Section 7 DMA Controller	
		261	7.2.1.5 Transfer Completion Interrupt and Transfer Count Match Interrupt Outputs: Description changed (transfer completion → DMAC transfer completion)
		277	Figure 7.10 Example of Suspension, Resume, and Transfer Abort of a DMAC Channel: Changed (Transfer aborted added to Time 11 of channel 0)
		278	Figure 7.11 Example of Suspension, Resume, and Transfer Abort of a DTS: Changed
		292	Table 7.8 List of DMA Trigger Sources (3/4): Changed (ICU-S → ICUSB)
		296	Table 7.9 List of DTS Trigger Sources (3/4): Changed (ICU-S → ICUSB)
		317	Table 7.33 DTRTSCTL Register Contents: Description of b0 changed
		342	Figure 7.12 Structure of the TI: TI-D changed
		Section 8 Reset	
		363	8.4.2 RESF — Reset Source Determination Register: Description added.
		364	8.4.3 RESFC — Reset Source Clear Register: Description added
		Section 9 Power Supply Circuit	
		373	9.5 Notes, changed
		373	9.5.1 Usage with an EPT, added
		373	9.5.2 Example of Connections between Power-Supply Pins and External Capacitors, added
		373	Figure 9.4 Example of Connections between Power-Supply Pins and External Capacitors, changed
		Section 10 Clock Controller	
		374	Features of the RH850/C1x Clock Controller, changed
		376	Table 10.3 Pins Related to the Clock Controller, changed (external clock → Crystal input, Crystal → Crystal output)
		376	10.1.3 How to Connect a Crystal Oscillator, title and description changed (resonator → Oscillator)
		376	Figure 10.2 Connection Example of Crystal Oscillator, title changed
		380	10.2.4 CKSC0CTL — Clock 0 Selection Control Register: Value after reset changed
		380	Table 10.7 CKSC0CTL Register Contents, b5 to 0, binary number added
		382	Table 10.9 CLKD0DIV Register Contents, b2 to 0, binary number added
		384	Table 10.11 CKSC1CTL Register Contents, b5 to 0, binary number added
		385	Table 10.12 CKSC1ACT Register Contents, b5 to 0, representation of the settings changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	387	10.2.11 PROT1PS — Protect 1 Status Register: Access changed
		Section 11 Clocked Serial Interface H (CSIH)	
		All	Unification of the description of the reserved bits: - Read/write register: When read, the value after reset is read. When writing, write the value after reset. - Read-only register: When read, the value after reset is read. - Write-only register: When writing, write the value after reset.
		395	11.2.1 Functional Overview: Description changed
		396	11.2.2 Functional Overview Description, Table: Description of the CSIHnMCTL2 register changed
		399	Table 11.11 CSIHnCTL0 Register Contents: Functional description of b7 changed
		400	Table 11.12 CSIHnCTL1 Register Contents: Functional description of b16
		405 to 407	Table 11.16 CSIHnSTR0 Register Contents: Functional description of b31 to 24, b23 to 16, b15, b14, and b3 changed
		409	Table 11.18 CSIHnSTCR0 register contents: Functional description of b8 changed
		411	Table 11.20 CSIHnMCTL1 Register Contents: Functional description in b22 to 16 and b6 to 0 deleted
		413	Table 11.21 CSIHnMCTL2 Register Contents: Functional description of b6 to 0 changed
		418	Table 11.23 CSIHnCFGx Register Contents (3/5): Functional description of b17 and b16 changed, description in b15 deleted
		419	Table 11.23 CSIHnCFGx Register Contents (4/5): Functional description of b14 to 12 (110_B) and b11 to 8 changed
		420	Table 11.23 CSIHnCFGx Register Contents (5/5): Functional description of b7 to 4 and b3 to 0 changed
		421	11.3.12 CSIHnTX0W — CSIHn Transmit Data Register 0 for Word Access: Bit assignment chart changed (b19 to 16)
		422	Table 11.24 CSIHnTX0W Register Contents (2/2): Functional description of b19 to 16 changed
		424	11.3.14 CSIHnRX0W — CSIHn Receive Data Register 0 for Word Access: Bit assignment chart changed (b19 to 16)
		427	Table 11.29 Interrupt Generation: Description in cause of interrupt changed, Note 4 changed
		428	11.4.2 Interrupt Delay: Title changed
		430	11.4.3.1 INTCSIHTIC in Direct Access Mode: Description changed (No general interrupt delay → No interrupt delay)
		430	Figure 11.3 Generation of INTCSIHTIC After Transfer (CSIHnCTL1.CSIHnSLIT = 0): Changed
		430	Figure 11.4 Immediate Generation of INTCSIHTIC (CSIHnCTL1.CSIHnSLIT = 1): Changed
		430	11.4.3.1 INTCSIHTIC in Direct Access Mode: Note changed
		431	11.4.3.2 INTCSIHTIC in FIFO Mode: Description changed (No general interrupt delay → No interrupt delay)
		432	11.4.3.3 INTCSIHTIC in job mode: Description changed (No general interrupt delay → No interrupt delay)
		433	Table 11.32 INTCSIHTIR Interrupt Generation: Description in Cause of interrupt of FIFO mode changed
		433	11.4.4.1 INTCSIHTIR in Direct Access Mode: Description changed (No general interrupt delay → No interrupt delay)
		434	11.4.4.2 INTCSIHTIR in Dual Buffer Mode: Description changed (No general interrupt delay → No interrupt delay)
		437	Figure 11.9 Transmit/Receive in Master Mode: Changed
		438	11.4.7.2 Slave Mode: Note changed
		439	11.4.8.2 One Master and Multiple Slaves: Description changed
		441	11.4.9.1 Configuration Registers: Description changed
		442	Figure 11.13 Chip Select Timings: Changed
		442	11.4.9.1 Configuration Registers: Caution for Chip Select Timings added
		443	Figure 11.14 Chip Select and RCB Example: Changed
		443	11.4.9.2 CS Example: Body changed
		443	11.4.9.3 Job Concept: Body changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	444	11.4.10.1 Changing the Clock Phase: Description changed (data packet → data)
		445	11.4.10.2 Changing the Data Phase: Body changed
		446	11.4.11 Transmission Clock Selection: Description changed (CSIHnTSCK → CSIHTSCK, Clearing CSIHnBRSy.CSIHnBRSy[11:0] → Setting CSIHnBRSy.CSIHnBRSy[11:0] to 000_H)
		447	11.4.11 Transfer clock frequency upper and lower limits: Description deleted
		448	11.4.12.1 FIFO Mode: Description changed
		448	Table 11.35 FIFO Mode: Description and Note 1 changed
		449	Table 11.36 Dual Buffer Mode: Pointer description changed
		449	Table 11.37 Transmit-only Buffer Mode: Pointer description and Note 1 changed
		450	11.4.13.2 Receive-only Mode: Description changed (the clock from the master is received → the transmission clock CSIHTSCK from the master is received)
		451	11.4.14.1 Data Length between 2 and 16 Bits: Description changed (using → by using)
		452	11.4.14.2 Data Length Greater than 16 Bits: Description changed
		453	11.4.14.2 Data Length Greater than 16 Bits: Note 4 and 6 added
		454	11.4.15 Serial Data Direction Selection: Description changed (using → by using)
		454	Figure 11.25 Serial Data Direction Select Function - MSB First (CSIHnDIRx = 0): Figure title changed
		454	Figure 11.26 Serial Data Direction Select Function - LSB First (CSIHnDIRx = 1): Figure title changed
		455	11.4.16 SS (Slave Select) Function: Title changed
		455	11.4.16.1 Communication Timing Using SS Function: Title and description of (3) and (6) changed
		455	Figure 11.27 Tx/Rx Timing of Communication Using SS Function: Title changed
		457	Table 11.39 Memory Mode and Slave Transfer State: Added
		458	Table 11.40 Memory Mode and Slave Transfer State: Added
		458	11.4.17.1 Slave Mode: Description 2 changed
		460	Figure 11.34 Master's Reaction on CSIHTRYI (CSIHnCFGx.CSIHnDAPx = 1): Changed
		465, 466	11.4.18.4 Overflow Error, Example: Description changed
		468	11.4.18.5 (1) Direct Access/Transmit-only Buffer: Note added
		468	11.4.18.5 (2) FIFO Mode 1: Note added
		469	Figure 11.42 Overrun Error Detection in FIFO Mode (no data): Changed (Overrun error → INTCSIHTIRE)
		—	11.4.18.5 (2) FIFO Mode 2. The CPU attempts to read non existing reception data: Note deleted
		470	11.4.19 Loop-back Mode: Description changed
		470	Table 11.41 Pin Output Level in Loop-back Mode: Added
		470	Figure 11.44 Loop-back Operation: Changed
		474	11.4.21 Enforced Chip Select Idle Setting: Caution changed (IDLE state → idle state)
		475	11.5.1.1 Transmit/Receive in master mode when job mode is disabled: Changed (No general interrupt delay → No interrupt delay)
		475	Figure 11.49 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 0: Changed
		475, 476	11.5.1.1 Transmit/Receive in master mode when job mode is disabled: Procedure changed
		477	11.5.1.2 Transmission/Reception in Master Mode when job mode is Enabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		477	Figure 11.50 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 1: Changed
		478	11.5.1.2 Transmission/Reception in Master Mode when job mode is Enabled: Procedure changed
		479	11.5.2.1 Transmit/Receive in Master Mode when job mode is Disabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		479	Figure 11.51 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Changed (CSIHTCSS → CSIHTCSSx)
		479	11.5.2.1 Transmit/Receive in Master Mode when job mode is Disabled: Procedure changed
		481	11.5.2.2 Transmit/Receive in Master Mode When job mode is Enabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		481	Figure 11.52 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 1: Changed
		481	11.5.2.2 Transmit/Receive in Master Mode When job mode is Enabled: Note changed
		482	11.5.2.2 Transmit/Receive in Master Mode When job mode is Enabled: Procedure changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	483	11.5.3.1 Transmit/Receive in Master Mode when job mode is Disabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		483	Figure 11.53 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Changed
		484	11.5.3.1 Transmit/Receive in Master Mode when job mode is Disabled: Procedure changed
		485	11.5.3.2 Transmit/Receive in Master Mode when job mode is Enabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		485	Figure 11.54 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 1: Changed
		485	11.5.3.2 Transmit/Receive in Master Mode when job mode is Enabled: Notes changed
		486	11.5.3.2 Transmit/Receive in Master Mode when job mode is Enabled: Procedure changed
		487	11.5.3.3 Transmit/Receive in Slave Mode when job mode is Disabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		487	Figure 11.55 Slave in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Changed
		488	11.5.3.3 Transmit/Receive in Slave Mode when job mode is Disabled: Procedure 6 changed
		489	11.5.4.1 Transmit/Receive in Master Mode when job mode is Disabled: Changed
		489	Figure 11.56 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 0: Changed (CSIHTCSS → CSIHTCSSx)
		490	11.5.4.1 Transmit/Receive in Master Mode when job mode is Disabled: Procedure changed
		491	11.5.4.2 Master in Transmit/Receive Mode when job mode is Enabled: Changed (No general interrupt delay → No interrupt delay, data packet → data)
		491	Figure 11.57 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 1: Changed
		492	11.5.4.2 Master in Transmit/Receive Mode when job mode is Enabled: Procedure changed
		494	Table 11.42 Notes on Setting Registers (2/2): Content changed
		Section 12 Serial Communication Interface 3 (SCI3)	
		All	Term unification: f→PCLK, bit/s→bps, TxDi/RxDi→TxDn/RxDn
		496	Table 12.5 Interrupt Requests: Interrupt names changed
		497	12.2.1 Functional Overview: Body changed, table added
		—	12.2.2 Functions: Deleted
		—	12.2.3 Serial Communication Modes: Deleted
		499	Table 12.8 Register Configuration: Description changed, Note 1 added
		500	12.3.3 SCI3nRDR — Receive Data Register: Changed
		501	12.3.4 SCI3nTDR — Transmit Data Register: Changed
		—	12.3.2 SCI3nRSR — Receive Shift Register and 12.3.5 SCI3nTSR — Transmit Shift Register: Deleted
		508	Table 12.17 Relationship between Setting N in SCI3nBRR and Bit Rate B: Changed
		509	Table 12.18 Examples of BRR Settings for Bit Rates (Asynchronous Mode), Example of setting baud rate 4800bps: Changed
		510	Description above Table 12.20 Examples of Bit Rate Settings for Clock Synchronous Mode: Changed
		515	12.4.1.4 Double-Speed Operation: Description deleted
		520	12.4.1.7 Serial Data Reception (Asynchronous Mode): Description in 5. changed (RDRF flag → SCI3nRDR)
		530	12.4.3 Operation in Clock Synchronous Mode: Description changed (φ → PCLK)
		Section 13 LIN Master Interface (RLIN2)	
		544	Table 13.5 Clock Supply: Changed (pclk → PCLK, CLKC_LSB → CLK_LSB, unmodulated low-speed peripheral clock → low-speed peripheral clock)
		545	13.2.1 Functional Overview: Description changed
		545	Table 13.9 LIN Master Interface Specifications: Specifications changed
		546	Figure 13.1 LIN Master Interface Block Diagram: Changed (Notation of the unit channel changed)
		547	Table 13.10 LIN Master Interface Registers Overview: Note added
		548	Table 13.11 RLN21nGLWBR Register Contents: Functional description of b0 changed
		549	13.3.2.2 RLN21nGLBRP0 — LIN Baud Rate Prescaler 0 Register: Description changed (the clock that is selected by LPRS bits (prescaler clock select bits) → LIN communication clock source)

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	550	13.3.2.3 RLN21nGLBRP1 — LIN Baud Rate Prescaler 1 Register: Description changed (the clock that is selected by LPRS bits (prescaler clock select bits) → LIN communication clock source)
		551	13.3.2.4 RLN21nGLSTC — LIN Self-Test Control Register: Changed
		556	13.3.3.5 RLN21nmLiIE — LIN Interrupt Enable Register: Description in ERRIE, FRCIE, and FTCIE bits changed (interrupt → interrupt request)
		560	13.3.3.8 RLN21nmLiTRC — LIN Transmission Control Register: Description of the RTS bit changed (MOV instruction → store instruction)
		561	13.3.3.9 RLN21nmLiMST — LIN Mode Status Register: Description in OMM1 bit changed
		562	13.3.3.10 RLN21nmLiST — LIN Status Register: Access, R/W property, and description changed, description of the HTRC flag changed
		563	13.3.3.10 RLN21nmLiST — LIN Status Register: Description in D1RC to FTC flags changed (interrupt → interrupt request, TER → FTER)
		564	13.3.3.11 RLN21nmLiEST — LIN Error Status Register: Access, R/W property, and description changed
		566	13.3.3.12 RLN21nmLiDFC — LIN Data Field Configuration Register: Description changed
		570	13.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register: Title changed
		573	13.5 Modes: Body changed
		578	Figure 13.5 Operation in Response Transmission: Position of (1) and (7) changed
		578	Table 13.33 Processing in Response Transmission: Description of LIN Master Interface Processing in (1) and (4) changed
		579	Figure 13.6 Operation in Response Reception: Position of (2), (3), and (6) changed
		579	Table 13.34 Processing in Response Reception: Description changed
		580	Figure 13.7 Example of Data Transmission Timing: Changed
		584	13.12.1 Wake-up Transmission: Body changed
		589	13.15 LIN Self-Test Mode: Changed (Frame time-out error deleted)
		590	13.15.1 Change to LIN Self-Test Mode: Description changed
		593	13.15.4 Terminating LIN Self-Test Mode: Description changed
		594	Figure 13.16 Block Diagram of Baud Rate Generation in LIN Master Mode: Description of Note 3 changed
		594	Table 13.37 Examples of Baud Rate (19200, 10417, 9600, and 2400 bps) Generation: 32MHz to 8MHz deleted
		Section 14 CAN Interface (RS-CAN)	
		595	Table 14.2 Index, index n, x, q, description changed
		599	Table 14.9 RS-CAN Module Specifications (2/2), Receive FIFO interrupt changed ((1 source) deleted)
		625	Table 14.15 RSCAN0CmCFG Register Contents (1/2), b22 to 20, note 1 added
		637	Table 14.19 RSCAN0GCFG Register Contents, b31 to 16, note 3 added
		644	14.3.9 RSCAN0GERFL — Global Error Flag Register, Access, description changed
		648	14.3.10 RSCAN0GTINTSTS0 — Global TX Interrupt Status Register 0, description of the CFTIFn bits changed (FIFO buffer → FIFO)
		661	Table 14.33 RSCAN0RMIDq Register Contents, RMIDE bit changed (GAFLIDE → RMIDE)
		676	Table 14.44 RSCAN0CFCCk Register Contents, description of the CFITR bit changed (the CFITSS bit is 1 → the CFITSS bit is 0)
		678	14.3.32 RSCAN0CFSTSk — Transmit/Receive FIFO Buffer Status Register (k = 0 to 11), Access and Value after Reset changed
		678	14.3.32 RSCAN0CFSTSk — Transmit/Receive FIFO Buffer Status Register (k = 0 to 11), CFTXIF flag, description changed
		683	14.3.34 RSCAN0CFIDk — Transmit/Receive FIFO Buffer Access ID Register (k = 0 to 11), b28 to 0, note added
		722	14.3.57 RSCAN0TXQPCTRm — Transmit Queue Pointer Control Register (m = 0 to 3), Value after reset changed
		725	14.3.59 RSCAN0THLSTSm — Transmit History Status Register (m = 0 to 3), Access, description changed
		726	Table 14.77 RSCAN0THLSTSm Register Contents, note added
		727	Table 14.78 RSCAN0THLACCm Register Contents, b2 to 0, description changed
		732	14.3.64 RSCAN0GLOCKK — Global Lock Key Register, b11, R/W changed (R/W → W)

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	734	14.4.1 Interrupt Sources, changed (Receive FIFO interrupt (1 source) → Receive FIFO interrupt)
		735	Table 14.84 List of CAN Interrupt Sources, changed (FIFO transmission → FIFO transmission complete, FIFO reception → FIFO reception complete)
		737	Figure 14.3 CAN Channel Interrupt Block Diagram, changed (CANm transmit/receive FIFO reception interrupt → CANm transmit/receive FIFO reception complete interrupt)
		739	14.4.2.1 (2) Global Reset Mode, changed
		Section 15 Window Watchdog Timer A (WDTA)	
		790	Table 15.5 Interrupt Requests, changed, note 1 added
		790	Table 15.7 WDTA0 Start-Up Options, changed
		791	15.2.1 Functional Overview, changed
		791	Figure 15.1 Block Diagram of the WDTA, changed
		793	15.3.2 WDTAnWDTE — WDTA Enable Register, changed
		793	Table 15.9 WDTAnWDTE Register Contents, changed
		793	Table 15.10 WDTAnRUN[7] Values after Reset, added
		794	15.3.3 WDTAnMD — WDTA Mode Register, description added, note 1 deleted
		794	Table 15.11 WDTAnMD Register Contents (1/2), b6 to 4, b3, description changed
		795	15.4 Interrupt Sources, added
		796	15.5.1.1 Start modes, changed
		796	15.5.1.2 Start mode selection (only for WDTA0), title changed
		796	15.5.1.3 WDTA settings after reset release, table contents changed, Change WDTA settings, changed
		797	Figure 15.2 Timing Diagram of WDTA Start in Default Start Mode, changed
		797	15.5.1.4 Default start mode timing (only for WDTA0), description changed
		798	Figure 15.3 Timing Diagram of WDTA Start in Software Trigger Start Mode, changed
		798	15.5.1.5 Software trigger start mode timing (common to WDTA0 and WDTA1), changed
		799	15.5.2 WDTA Trigger, description changed
		799	15.5.3 Error Detection, description changed
		800	15.5.4 WDTA Error Mode, added
		800	Figure 15.4 Timing Diagram of WDTA Internal Reset Generation, title changed, description below the figure changed
		801	15.5.5 75% Interrupt Request Signals, description changed, description of (7) changed
		801	Figure 15.5 Timing Diagram of WDTA 75% Interrupt Request Signals, changed, description below the figure changed
		802	15.5.6 Window Function, changed
		802	Figure 15.6 Timing Diagram of WDTA Window Function, changed
		Section 16 OS Timer (OSTM)	
		All	Term unification: OSTMn → OSTM, OS timer → OSTM
		803	16.1.1 Number of Units, description changed
		803	Table 16.1 Number of Units, title changed
		804	Table 16.5 Interrupt Requests, names in table header changed
		805	16.2.1 Functional Overview, description changed
		805	Figure 16.1 Block Diagram of the OSTM, title and contents changed
		806	16.2.3 Counter Clock, changed
		806	Figure 16.2 Generating an Interrupt when Counting Starts (in Interval Timer Mode), title changed
		807	16.3.2 OSTMnCMP — OSTMn Compare Register, description changed, Value after reset, description deleted
		807	Table 16.8 OSTMnCMP Register Contents, changed (down-counter → counter)
		808	16.3.3 OSTMnCNT — OSTMn Counter Register, Value after reset, description changed
		808	Table 16.9 OSTMnCNT Register Contents, changed (value after reset → initial value)
		808	Table 16.10 Correspondence between Operating Mode, Counting Direction and Initial Value, changed (value after reset → initial value)

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	809	16.3.4 OSTMnTO — OSTMn Output Register, Value after reset, description deleted and changed
		809	Table 16.11 OSTMnTO Register Contents, b0, description changed
		809	16.3.5 OSTMnTO — OSTMn Output Enable Register, Value after reset, description deleted
		810	16.3.6 OSTMnTE — OSTMn Count Enable Status Register, Value after reset, description deleted and changed
		810	Table 16.13 OSTMnTE Register Contents, b7 to 1 and b0, description changed
		811	16.3.7 OSTMnTS — OSTMn Count Start Trigger Register, Value after reset, description deleted and changed
		811	Table 16.14 OSTMnTS Register Contents, b0, description changed
		811	16.3.8 OSTMnTT — OSTMn Count Stop Trigger Register, Value after reset, description deleted and changed
		811	Table 16.15 OSTMnTT Register Contents, b0, description changed
		812	16.3.9 OSTMnCTL — OSTMn Control Register, Value after reset, description deleted
		813	16.4.1 Starting and Stopping the Timer, description changed
		813	16.4.1 Starting and Stopping the Timer, Triggering of simultaneous start, title changed
		815	16.4.2.1 Basic Operation in Interval Timer Mode, OSTMnTINT period, description changed, description of (1) and (4) changed
		816	16.4.2.1 Basic Operation in Interval Timer Mode, Forced restart, description changed, description of (1) changed
		817	16.4.2.2 Operation when OSTMnCMP = 0000 0000_H, description changed
		818	16.4.2.3 Setting Procedure for Interval Timer Mode, title and description changed
		819	16.4.3.1 Basic Operation in Free-Run Compare Mode, changed
		820	16.4.3.1 Basic Operation in Free-Run Compare Mode, OSTMTINT period, title added
		820	16.4.3.2 Operation when OSTMnCMP = 0000 0000_H, description changed
		821	16.4.3.3 Setting Procedure for Free-Run Compare Mode, title and description changed
		Section 17 Timer Array Unit D (TAUD)	
		All	Notation unified: TAUDnTTINm → TAUDTTINm, TAUDnTTOUTm→TAUDTTOUTm, TAUDnTSSTm→TAUDTSSTm Term unification: Gate count mode → deleted, inverted logic → negative logic, realtime trigger → real time output trigger, INT→INTTAUDnIm, 0 before address xxx_H deleted Interrupt Request Signals Skipping Function→Interrupt Request Signals Culling Function
		828	Figure 17.2 Block Diagram of the TAUD: Changed
		840	Table 17.16 TAUDnCMORm Register Contents: Description of the reserved bit b5 changed
		842	Table 17.17 TAUDnCMURm Register Contents: Description of the reserved bits b7 to 2 changed
		843	Table 17.18 TAUDnCSRm Register Contents: Description of the reserved bits b7to 2 changed
		844	Table 17.19 TAUDnCSCm Register Contents: Description of the reserved bits b7to 1 changed
		846	Table 17.24 TAUDnRDS Register Contents: Functional description changed
		848	17.3.17 TAUDnRDT — TAUDn Channel Reload Data Trigger Register: Description changed
		848	Table 17.27 TAUDnRDT Register Contents: Functional description changed
		848	Table 17.28 TAUDnRSF Register Contents: Functional description changed
		849	Table 17.29 TAUDnTOE Register Contents: Functional description changed
		856	17.4.1 General Operating Procedure: Description of (3) and (4) changed
		857	17.4.2 Concepts of Synchronous Channel Operation: Body changed
		860	Table 17.41 Simultaneous Rewrite Methods and when They are Triggered: Description of B method and description of C1 and C2 changed
		861	Figure 17.4 General Procedure for Simultaneous Rewrite: (pending flag → enabling flag)
		862	17.4.3.2 (3) Simultaneous Rewrite: Description changed
		866	17.4.3.4 (2), Setting: Changed (up and down → down)
		868	Figure 17.7 Simultaneous Rewrite When INTTAUDnIm Is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm: Changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	—	17.4.3.4 (4) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal (Method C2): Deleted
		879	17.4.5.1 Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode: Title changed
		879	Figure 17.12 Start Timing in Interval Timer Mode, Judge Mode, Capture Mode, Up/Down Count Mode, and Count Capture Mode: Title changed
		879	17.4.5.1: Description in Note changed
		887	Table 17.45 Contents of TAUDnCMORm Register for Interval Timer Function: Changed
		887	Table 17.46 Contents of TAUDnCMURm Register for Interval Timer Function: Changed
		888	Table 17.47 Control Bit Settings in Independent Channel Output Mode 1: Changed
		891	17.4.9.1 (6) (d) Forced restart (TAUDnCMORm.TAUDnMD0 = 1): Title and description changed
		892	17.4.9.1 (7) Forced restart (TAUDnCMORm.TAUDnMD0 = 0): Added
		895	Table 17.50 Contents of TAUDnCMORm Register for TAUDTTINm Input Interval Timer Function: Title and functional description changed
		895	Table 17.51 Contents of TAUDnCMURm Register for TAUDTTINm Input Interval Timer Function: Title and functional description changed
		896	Table 17.52 Control Bit Settings in Independent Channel Output Mode 1: Changed
		901	Table 17.55 Contents of TAUDnCMORm Register for Clock Divide Function: Changed
		901	Table 17.56 Contents of TAUDnCMURm Register for Clock Divide Function: Changed
		902	Table 17.57 Control Bit Settings in Independent Channel Output Mode 1: changed
		908	Table 17.60 Contents of TAUDnCMORm Register for External Event Count Function: Changed
		908	Table 17.61 Contents of TAUDnCMURm Register for External Event Count Function: Changed
		911	17.4.9.4 (6) (c) Forced restart: Description changed
		914	Table 17.64 Contents of TAUDnCMORm Register for Delay Count Function: Changed
		914	Table 17.65 Contents of TAUDnCMURm Register for Delay Count Function: Changed
		918	Table 17.68 Contents of TAUDnCMORm Register for One-Pulse Output Function: Changed
		918	Table 17.69 Contents of TAUDnCMURm Register for One-Pulse Output Function: Changed
		919	Table 17.70 Control Bit Settings in Independent Channel Output Mode 2: Title changed
		921	Table 17.73 Effects of Overflow: 10 and 11 of TAUDnCSRm.TAUDnOVF changed (0 → Unchanged)
		922	17.4.9.7 (1): Description in Note changed
		924	Table 17.74 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Measurement Function: Changed
		924	Table 17.75 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Measurement Function: Changed
		925	Table 17.77 Operating Procedure for TAUDTTINm Input Pulse Interval Measurement Function: Description changed
		929	Table 17.78 Effects or Overflow: 10 and 11 of TAUDnCSRm.TAUDnOVF changed (0 → Unchanged)
		930	17.4.9.8 (1) Functional description: Description in Note changed
		931	Table 17.79 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Measurement Function: Changed
		931	Table 17.80 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Measurement Function: Changed
		932	Table 17.82 Operating Procedure for TAUDTTINm Input Signal Width Measurement Function: Description changed
		938	Table 17.83 Contents of TAUDnCMORm Register for TAUDTTINm Input Position Detection Function: Changed
		938	Table 17.84 Contents of TAUDnCMURm Register for TAUDTTINm Input Position Detection Function: Changed
		939	Table 17.86 Operating Procedure for TAUDTTINm Input Position Detection Function: Changed
		941	17.4.9.10 TAUDTTINm Input Period Count Detection Function: Description in Note added

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	943	Table 17.87 Contents of TAUDnCMORm Register for TAUDnTTINm Input Period Count Detection Function: Changed
		943	Table 17.88 Contents of TAUDnCMURm Register for TAUDTTINm Input Period Count Detection Function: Changed
		944	Table 17.90 Operating Procedure for TAUDTTINm Input Period Count Detection Function: Description changed
		948	Table 17.91 Contents of TAUDnCMORm Register for TAUDTTINm Input Pulse Interval Judgment Function: Changed
		948	Table 17.92 Contents of TAUDnCMURm Register for TAUDTTINm Input Pulse Interval Judgment Function: Changed
		949	Table 17.94 Operating Procedure for TAUDTTINm Input Pulse Interval Judgment Function: Changed
		952	Table 17.95 Contents of TAUDnCMORm Register for TAUDTTINm Input Signal Width Judgment Function: Changed
		952	Table 17.96 Contents of TAUDnCMURm Register for TAUDTTINm Input Signal Width Judgment Function: Changed
		953	Table 17.98 Operating Procedure for TAUDTTINm Input Signal Width Judgment Function: Changed
		957	Table 17.99 Contents of TAUDnCMORm Register for Real-Time Output Function Type 1: Changed
		957	Table 17.100 Contents of TAUDnCMURm Register for Real-Time Output Function Type 1: Changed
		958	Table 17.101 Control Bit Settings in Independent Channel Output Mode 1 with Real-Time Output: Changed
		959	Table 17.103 Control Bit Settings for the Lower Channels in Independent Channel Output Mode 1 with Real-Time Output: Changed
		961	17.4.10.1 (7) Specific Timing Diagrams: Description changed
		965	Table 17.105 Contents of TAUDnCMORm Register for Real-Time Output Function Type 2: Changed
		965	Table 17.106 Contents of TAUDnCMURm Register for Real-Time Output Function Type 2: Changed
		966	Table 17.107 Control Bit Settings in Independent Channel Output Mode 2 with Real-Time Output: Changed
		967	Table 17.109 Control Bit Settings for Lower Channels in Independent Channel Output Mode 2 with Real-Time Output: Changed
		968	Table 17.110 Operating Procedure for Real-Time Output Function Type 2: Description changed
		973	Figure 17.73 General Timing Diagram of Simultaneous Rewrite Trigger Generation Function Type 1: Changed
		—	17.4.11.1 (3) Block Diagram and General Timing Diagram: Description deleted
		974	Table 17.111 Contents of TAUDnCMORm Register for Simultaneous Rewrite Trigger Generation Function Type 1: Changed
		974	Table 17.112 Contents of TAUDnCMURm Register for Simultaneous Rewrite Trigger Generation Function Type 1: Changed
		975	Table 17.113 Simultaneous Rewrite Settings for Simultaneous Rewrite Trigger Generation Function Type 1: Changed
		975	17.4.11.1(5)(c) Channel output mode for lower channels: Description added
		975	Table 17.114 Simultaneous Rewrite Settings for Lower Channels in Simultaneous Rewrite Trigger Generation Function Type 1: Changed
		981	Table 17.116 Contents of TAUDnCMORm Register for Master Channels of PWM Output Function: Changed
		981	Table 17.117 Contents of TAUDnCMURm Register for Master Channels of PWM Output Function:Changed
		982	Table 17.118 Simultaneous Rewrite Settings for Master Channels of the PWM Output Function: Changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	983	Table 17.119 Contents of TAUDnCMORm Register for Slave Channels of PWM Output Function: Changed
		983	Table 17.120 Contents of TAUDnCMURm Register for Slave Channels of PWM Output Function: Changed
		984	Table 17.122 Simultaneous Rewrite Settings for Slave Channels of PWM Output Function: Changed
		985	Table 17.123 Operating Procedure for PWM Output Function: Description changed
		990	17.4.12.2 (2) Equations: Description changed
		992	Table 17.124 Contents of TAUDnCMORm Register for Master Channels of One-Shot Pulse Output Function:Changed
		992	Table 17.125 Contents of TAUDnCMURm Register for Master Channels of One-Shot Pulse Output Function: Changed
		993	Table 17.126 Simultaneous Rewrite Settings for Master Channels of One-Shot Pulse Output Function: Changed
		994	Table 17.127 Contents of TAUDnCMORm Register for Slave Channels of One-Shot Pulse Output Function: Changed
		994	Table 17.128 Contents of TAUDnCMURm Register for Slave Channels of One-Shot Pulse Output Function: Changed
		995	Table 17.129 Control Bit Settings in Synchronous Channel Output Mode 2: Changed
		995	Table 17.130 Simultaneous Rewrite Settings for Slave Channels of One-Shot Pulse Output Function: Changed
		997	Figure 17.81 TAUDnCDRm (Master) = 0000_H: Changed
		998	Figure 17.82 TAUDnCDRm (Slave) = 0000_H: Changed
		998	17.4.12.2 (7) (b) TAUDnCDRm (slave) = 0000_H: Description deleted
		1000	Figure 17.84 Interval of TAUDTTINm ≤ delay time + pulse width + 1: changed
		1004	17.4.12.3 (3) Block Diagram and General Timing Diagram: Description and Note added
		1005	Table 17.132 Contents of TAUDnCMORm Register for Master Channels of Delay Pulse Output Function:Changed
		1005	Table 17.133 Contents of TAUDnCMURm Register for Master Channels of Delay Pulse Output Function: Changed
		1005	17.4.12.3 (4) (c) Channel output mode for master channels: Description deleted
		1006	Table 17.134 Simultaneous Rewrite Settings for Master Channels of Delay Pulse Output Function: Changed
		1007	Table 17.135 Contents of TAUDnCMORm Register for Slave Channel 1 of Delay Pulse Output Function: Changed
		1007	Table 17.136 Contents of TAUDnCMURm Register for Slave Channel 1 of Delay Pulse Output Function: Changed
		1008	Table 17.137 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1:Changed
		1008	Table 17.138 Simultaneous Rewrite Settings for Slave Channel 1 of Delay Pulse Output Function: Changed
		1009	Table 17.139 Contents of TAUDnCMORm Register for Slave Channel 2 of Delay Pulse Output Function: Changed
		1009	Table 17.140 Contents of TAUDnCMURm Register for Slave Channel 2 of Delay Pulse Output Function: Changed
		1010	17.4.12.3 (6) (c) Channel output mode for slave channel 2: Description deleted
		1010	Table 17.141 Simultaneous Rewrite Settings for Slave Channel 2 of Delay Pulse Output Function: Changed
		1011	Table 17.142 Contents of TAUDnCMORm Register for Slave Channel 3 of Delay Pulse Output Function:Changed
		1011	Table 17.143 Contents of TAUDnCMURm Register for Slave Channel 3 of Delay Pulse Output Function:Changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1012	Table 17.145 Simultaneous Rewrite Settings for Slave channel 3 of Delay Pulse Output Function: Changed
		1015	Figure 17.87 Duty Cycle (slave 3) = 100%: Changed
		1016	Figure 17.88 TAUDTTOUTm (Slave 1) = TAUDTTOUTm (Slave 3): Changed
		1018	17.4.12.4 (2) Equations: Description Changed
		1019	17.4.12.4 (3) Block Diagram and General Timing Diagram: Note added
		1020	Table 17.147 Contents of TAUDnCMORm Register for Master Channels of Offset Trigger Output Function: Changed
		1020	Table 17.148 Contents of TAUDnCMURm Register for Master Channels of Offset Trigger Output Function: Changed
		1021	17.4.12.4 (4) (c) Channel output mode for master channels: Description deleted
		1022	Table 17.150 Contents of TAUDnCMORm Register for Slave Channels of Offset Trigger Output Function: Changed
		1022	Table 17.151 Contents of TAUDnCMURm Register for Slave Channels of Offset Trigger Output Function: Changed
		1023	Table 17.152 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1: Title and desciption changed
		1024	Table 17.154 Operating Procedure for Offset Trigger Output Function: Changed
		1030	17.4.12.6 (2) Equations: Description Changed
		1033	Table 17.155 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function: Changed
		1033	Table 17.156 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function:Changed
		1034	Table 17.157 Control Bit Settings in Independent Channel Output Mode 1: Changed
		1034	Table 17.158 Simultaneous Rewrite Settings for Master Channels of Triangle PWM Output Function:Changed
		1035	Table 17.159 Contents of TAUDnCMORm Register for Slave Channels of Triangle PWM Output Function: Changed
		1035	Table 17.160 Contents of TAUDnCMURm Register for Slave Channels of Triangle PWM Output Function: Changed
		1036	Table 17.162 Simultaneous Rewrite Settings for Slave Channels of Triangle PWM Output Function: Changed
		1038	Figure 17.97 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1: changed (CDRn → TAUDnCDRm)
		1039	Figure 17.98 TAUDnCDRm (Slave) = 0000_H: Changed (CDRn → TAUDnCDRm)
		1044	17.4.12.7 (3) • Slave channel 3: Description changed
		1044	Figure 17.100 General Timing Diagram of Triangle PWM Output Function with Dead Time: Changed
		1045	Table 17.165 Contents of TAUDnCMORm Register for Master Channels of Triangle PWM Output Function with Dead Time: Changed
		1045	Table 17.166 Contents of TAUDnCMURm Register for Master Channels of Triangle PWM Output Function with Dead Time: Changed
		1046	Table 17.167 Control Bit Settings in Independent Channel Output Mode 1: Changed
		1046	Table 17.168 Simultaneous Rewrite Setting for Master Channels of Triangle PWM Output Function with Dead Time: Changed
		1047	Table 17.169 Contents of TAUDnCMORm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time: Changed
		1047	Table 17.170 Contents of TAUDnCMURm Register for Slave Channel 2 of Triangle PWM Output Function with Dead Time: Changed
		1048	Table 17.172 Simultaneous Rewrite Settings for Slave Channel 2 of Triangle PWM Output Function:Changed
		1049	Table 17.173 Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time: Changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1049	Table 17.174 Contents of TAUDnCMURm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time: Changed
		1050	Table 17.176 Simultaneous Rewrite Settings for Slave Channel 3 of Triangle PWM Output Function: Changed
		1052	Figure 17.101 TAUDnCDRm (Slave) ≥ TAUDnCDRm (Master) + 1: Changed (CDRn → TAUDnCDRm)
		1053	Figure 17.102 TAUDnCDRm (Slave) = 0000_H: Changed
		—	17.4.12.9 (3) Block Diagram and General Timing Diagram: Description deleted
		1059	Table 17.178 Contents of TAUDnCMORm Register for Master Channels of Interrupt Request Signals Skipping Function: Changed
		1059	Table 17.179 Contents of TAUDnCMURm Register for Master Channels of Interrupt Request Signals Skipping Function: Changed
		1059	17.4.12.9 (4) (c) Channel output mode for master channels: Description deleted
		1060	Table 17.180 Simultaneous Rewrite Settings for Master channels of Interrupt Request Signals Skipping Function: Changed
		1061	Table 17.181 Contents of TAUDnCMORm Register for Slave Channels of Interrupt Request Signals Skipping Function: Changed
		1061	Table 17.182 Contents of TAUDnCMURm Register for Slave Channels of Interrupt Request Signals Skipping Function: Changed
		1061	17.4.12.9 (5) (c) Channel output mode for the slave channel: Description deleted
		1062	Table 17.183 Simultaneous Rewrite Settings for Slave Channels of Interrupt Request Signals Skipping Function: Changed
		1067	Table 17.186 Contents of TAUDnCMORm Register for One-Phase PWM Output Function: Changed
		1067	Table 17.187 Contents of TAUDnCMURm Register for One-Phase PWM Output Function: Changed
		1072	Table 17.192 TAUDTTOUTm Output from One Pair of Slave Channels of Non-Complementary Modulation Output Function Type 1: Added
		1074	Figure 17.110 Block Diagram of Non-Complementary Modulation Output Function Type 1: Changed
		1075	17.4.13.1 (3) Block Diagram and General Timing Diagram: Note added
		1076	Table 17.193 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 1: Changed
		1076	Table 17.194 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 1: Changed
		1076	17.4.13.1 (4) (c) Channel output mode for master channels: Description deleted
		1077	Table 17.195 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 1: Changed
		1077	17.4.13.1 (4) (d) Simultaneous rewrite for master channels: Description in Note added
		1078	Table 17.196 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1: Changed
		1078	Table 17.197 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1: Changed
		1079	Table 17.198 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 1: Changed
		1080	Table 17.199 Contents of TAUDnCMORm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1: Changed
		1080	Table 17.200 Contents of TAUDnCMURm Register for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1: Changed
		1081	Table 17.202 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 1: Changed
		1085	17.4.13.2 (1) Functional description, Slave channel 1: Description changed
		1086	17.4.13.2 (1) Conditions: Description changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1086	Table 17.204 TAUDTTOUTm Output of a Pair of Slave Channels in Non-Complementary Modulation Output Function Type 2 (TAUDnTOL.TAUDnTOLm = 0): Title changed and description added
		1088	Figure 17.113 Block Diagram of Non-Complementary Modulation Output Function Type 2: Changed
		1089	Figure 17.114 General Timing Diagram of Non-Complementary Modulation Output Function Type 2: Changed
		1090	Table 17.205 Contents of TAUDnCMORm Register for Master Channels of Non-Complementary Modulation Output Function Type 2: Changed
		1090	Table 17.206 Contents of TAUDnCMURm Register for Master Channels of Non-Complementary Modulation Output Function Type 2: Changed
		1091	Table 17.207 Control Bit Settings for Master Channels in Non-ComplemEntary Modulation Output Function Type 2: Changed
		1091	Table 17.208 Simultaneous Rewrite Settings for Master Channels of Non-Complementary Modulation Output Function Type 2: Changed
		1092	Table 17.209 Contents of TAUDnCMORm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2: Changed
		1092	Table 17.210 Contents of TAUDnCMURm Register for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2: Changed
		1093	Table 17.211 Simultaneous Rewrite Settings for Slave Channel 1 of Non-Complementary Modulation Output Function Type 2: Changed
		1094	Table 17.212 Contents of TAUDnCMORm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2: Changed
		1094	Table 17.213 Contents of TAUDnCMURm Register for slave channels 2 to 7 of Non-Complementary Modulation Output Function Type 2: Changed
		1095	Table 17.215 Simultaneous Rewrite Settings for Slave Channels 2 to 7 of Non-Complementary Modulation Output Function Type 2:Changed
		1099	17.4.13.3 (1) Summary, Functional description, and Conditions: Description changed
		1100	17.4.13.3 (1) Conditions: Description changed
		1101, 1102	17.4.13.3 (1) summary: Note changed, description added
		1101	Table 17.217 TAUDTTOUTm Output (TAUDnTOL.TAUDnTOLm = 0) for a Pair of Slave Channels of Complementary Modulation Output Function: Changed (Columns for TAUDnTDL.TAUDnTDL2 and 3 added)
		1103	Figure 17.116 Block Diagram of Complementary Modulation Output Function: Changed
		1103	17.4.13.3 (3) Block Diagram and General Timing Diagram: Description added
		1105	Table 17.219 Contents of TAUDnCMORm Register for Master Channels of Complementary Modulation Output Function: Changed
		1105	Table 17.220 Contents of TAUDnCMURm Register for Master Channels of Complementary Modulation Output Function: Changed
		1106	Table 17.221 Control Bit Settings in Independent Channel Output Mode 1: Changed
		1106	Table 17.222 Simultaneous Rewrite Settings for Master Channels of Complementary Modulation Output Function: Changed
		1107	Table 17.223 Contents of TAUDnCMORm Register for Slave Channel 1 of Complementary Modulation Output Function: Changed
		1107	Table 17.224 Contents of TAUDnCMURm Register for Slave Channel 1 of Complementary Modulation Output Function: Changed
		1108	Table 17.225 Simultaneous Rewrite Settings for Slave Channel 1 of Complementary Modulation Output Function: Changed
		1109	Table 17.226 Contents of TAUDnCMORm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function: Changed
		1109	Table 17.227 Contents of TAUDnCMURm Register for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function: Changed
		1110	Table 17.228 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output: Changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1110	Table 17.229 Simultaneous Rewrite Settings for Slave Channels 2, 4, and 6 of Complementary Modulation Output Function: Changed
		1111	Table 17.230 Contents of TAUDnCMORm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function: Changed
		1111	Table 17.231 Contents of TAUDnCMURm Register for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function: changed
		1112	Table 17.232 Control Bit Settings in Synchronous Channel Output Mode 2 with Complementary Modulation Output: Changed
		1112	Table 17.233 Simultaneous Rewrite Settings for Slave Channels 3, 5, and 7 of Complementary Modulation Output Function: Changed
		1115	17.4.13.3 (9) Specific Timing Diagrams: Description: Description changed
		Section 18 Timer Array Unit J (TAUJ)	
		All	Notation unified: TAUJnTTOUT→TAUJTTOUT, TAUJnTTINm→TAUJTTINm 0 before address xx_H deleted, inverted logic → negative logic, gate count mode → deleted
		1122	Figure 18.2 Block diagram of the TAUJ: Changed
		1123	18.2.2.1 Description of Blocks, Clock and count clock selection: Description changed
		1130	18.3.5 TAUJnCNTm — TAUJn channel counter register: The value after reset changed
		1132	Table 18.15 TAUJnCMORm Register Contents (1/3): Changed
		1133	Table 18.15 TAUJnCMORm Register Contents (2/3): Changed
		1134	Table 18.15 TAUJnCMORm Register Contents (3/3): Changed
		1135	Table 18.16 TAUJnCMURm Register Contents: Changed
		1136	Table 18.17 TAUJnCSRm Register Contents: Changed
		1137	18.3.10 TAUJnTS — TAUJn channel start trigger register: The value after reset changed
		1137	Table 18.19 TAUJnTS Register Contents: Changed
		1137	Table 18.20 TAUJnTE Register Contents: Changed
		1138	18.3.12 TAUJnTT — TAUJn channel stop trigger register: The value after reset changed
		1138	Table 18.21 TAUJnTT Register Contents: Changed
		1138	Table 18.22 TAUJnTOE Register Contents: Changed
		1139	Table 18.23 TAUJnTO Register Contents: Changed
		1139	Table 18.24 TAUJnTOM Register Contents: Changed
		1140	Table 18.25 TAUJnTOC Register Contents: Changed
		1140	Table 18.26 TAUJnTOL Register Contents: Changed
		1141	18.3.18 TAUJnRDE — TAUJn channel reload data enable register: Description deleted
		1141	Table 18.27 TAUJnRDE Register Contents: Changed
		1141	Table 18.28 TAUJnRDM Register Contents: Changed
		1142	18.3.20 TAUJnRDT — TAUJn channel reload data trigger register: Description and the value after reset changed
		1142	Table 18.29 TAUJnRDT Register Contents: Changed
		1142	Table 18.30 TAUJnRSF Register Contents: Changed
		1143	18.4.1 General Operating Procedure: Description of (3) and (4) changed
		1149	18.4.3.3 Simultaneous rewrite procedure: Description changed
		1149	18.4.3.3: Description in Setting changed
		1150	18.4.3.3: Description (6) changed
		1152	18.4.4 Output logic: Description changed
		1155	18.4.5.1 Interval Timer Mode, Capture Mode, and Count Capture Mode: Title changed
		1155	Figure 18.7 Start Timing in Interval Timer Mode, Capture Mode, and Count Capture Mode: Title changed
		1157	Figure 18.11 Basic edge detection timing: Changed
		1160	Table 18.33 Contents of TAUJnCMORm Register for Interval Timer Function: Title and functional description changed
		1160	Table 18.34 Contents of TAUJnCMURm Register for Interval Timer Function: Title and functional description changed
		1167	Table 18.38 Contents of TAUJnCMORm Register for TAUJTTINm input Interval Timer Function: Title and functional description changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1167	Table 18.39 Contents of TAUJnCMURm Register for TAUJTTINm input Interval Timer Function: Title and functional description changed
		1171	Table 18.43 Effects of an overflow: Changed(0 → Unchanged)
		1174	Table 18.44 Contents of TAUJnCMORm Register for TAUJTTINm Input Pulse Interval Measurement Function: Title and functional description changed
		1174	Table 18.45 Contents of TAUJnCMURm Register for TAUJTTINm Input Pulse Interval Measurement Function: Title and functional description changed
		1175	Table 18.47 Operating procedure for TAUJTTINm input pulse interval measurement function: Description in Operation of Initial channel setting changed
		1179	Table 18.48 Effects of an overflow: Changed(0 → Unchanged)
		1181	Table 18.49 Contents of TAUJnCMORm Register for TAUJTTINm Input Signal Width Measurement Function: Title and functional description changed
		1181	Table 18.50 Contents of TAUJnCMURm Register for TAUJTTINm Input Signal Width Measurement Function: Title and functional description changed
		1182	Table 18.52 Operating procedure for TAUJTTINm input signal width measurement function: Description in Operation of Initial channel setting changed
		1187	Table 18.53 Contents of TAUJnCMORm Register for TAUJTTINm Input Position Detection Function: Title and functional description changed
		1187	Table 18.54 Contents of TAUJnCMURm Register for TAUJTTINm Input Position Detection Function: Title and functional description changed
		1188	Table 18.56 Operating procedure for TAUJTTINm input position detection function: Description in Operation of Initial channel setting changed
		1192	Table 18.57 Contents of TAUJnCMORm Register for TAUJTTINm Input Period Count Detection Function: Title and functional description changed
		1192	Table 18.58 Contents of TAUJnCMURm Register for TAUJTTINm Input Period Count Detection Function: Title and functional description changed
		1193	Table 18.60 Operating procedure for TAUJTTINm input period count detection function: Description in Operation of Initial channel setting changed
		1197	18.4.9.1 (3) Block diagram and general timing diagram: Note added
		1198	Table 18.61 Contents of TAUJnCMORm Register of Master Channel for PWM Output Function: Title and functional description changed
		1198	Table 18.62 Contents of TAUJnCMURm Register of Master Channel for PWM Output Function: Title and functional description changed
		1200	Table 18.64 Contents of TAUJnCMORm Register of Slave Channel for PWM Output Function: Title and functional description changed
		1200	Table 18.65 Contents of TAUJnCMURm Register of Slave Channel for PWM Output Function: Title and functional description changed
		1203	Figure 18.41 TAUJnCDRm (slave) = 0000 0000_H, positive logic (TAUJnTOL.TAUJnTOLm (slave) = 0): Timing of INTTAUJnIm changed
		Section 19 Motor Control Timer (TSG3)	
		—	Bit name changed: TSG3n120DCMC → TSG3nS120DCO Description of the reserved bits unified: - When read, the value after reset is read. When writing, write the value after reset. - When read, the value after reset is read. - When writing, write the value after reset (or the fixed value).
		1210	19.2.1 Functional Overview: Note added
		1216	Table 19.10 TSG3nCTL1 Register Contents (1/2): Description of b1 and b0 changed, Caution changed
		1218	Table 19.11 TSG3nCTL3 Register Contents: Description of b0 changed
		1227	19.3.8 TSG3nCTL7 — TSG3n Control Register 7: Description deleted
		1227	Table 19.15 TSG3nCTL7 Register Contents: Note added
		1228	19.3.9 TSG3nCTL8 — TSG3n Control Register 8: b0 name changed
		1228	Table 19.16 TSG3nCTL8 Register Contents: b0 name changed, Caution added
		1230	Table 19.18 TSG3nIOC1 Register Contents: Description of b4 changed
		1233	19.3.14 TSG3nSTR0 — TSG3n Status Register 0, description changed
		1233	Table 19.21 TSG3nSTR0 Register Contents: Description of b1 changed
		1234	19.3.15 TSG3nSTR1 — TSG3n Status Register 1, description changed
		1234	Table 19.22 TSG3nSTR1 Register Contents: Description of b7 to 4 changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1235 to 1237	Table 19.23 TSG3nSTR2 Register Contents: Description of b9 to 0 changed
		1238, 1239	Table 19.24 TSG3nSTC Register Contents: Description of b15 to 10 and b0 changed
		1240	Table 19.25 TSG3nOPT0 Register Contents: Description of b3 changed
		1244	Table 19.29 TSG3nTRG0 Register Contents: Description of b7 to 1 and b0 changed
		1244	Table 19.30 TSG3nTRG1 Register Contents: Description of b7 to 1 changed
		1245	Table 19.31 TSG3nTRG2 Register Contents: Description of b7 to 1 changed
		1272	Table 19.39 List of Modes: Description of 100 changed
		1272	19.4.1.1 Basic Operation of 18-Bit Counter: Body changed
		1274	Table 19.40 Compare Register Functions in Each Mode (2/7): Changed
		1274	Table 19.40 Compare Register Functions in Each Mode (3/7): Changed
		1274	Table 19.40 Compare Register Functions in Each Mode (4/7): Changed
		1275	Table 19.40 Compare Register Functions in Each Mode (7/7): Changed
		1276	Table 19.41 Updating Timing of Compare Registers by Mode: Title added and description changed
		1276	19.4.1.3 Anytime Rewrite Mode: Changed
		1279	Figure 19.8 Anytime Rewrite Timing (Example in PWM Mode): Changed
		1280	Figure 19.9 Update Timing of TSG3nCMP1E, 2E, 5E, 6E, 9E,and 10E at Anytime Rewrite Operation in HT-PWM Mode: Changed
		1281	19.4.1.3 (2) Example of Operation in Reload Mode (Simultaneous Rewrite Function): Description changed
		1283	Table 19.47 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 0): Table header changed
		1284	Table 19.48 List of Reload Settings (when TSG3nCTL3.TSG3nRIA = 1): Table header changed
		1285	19.4.1.4 (a) TSG3nO7 Pin Output Control: Body changed
		1286	Figure 19.14 Example of TSG3nO7 Pin Diagnostic Pulse Output Timing (2) (with Pulse Output Width Overlapped): Changed
		1288	Figure 19.15 Interrupt Generation Example (Example of HT-PWM Mode): Changed
		1295	Figure 19.19 Example of Positive Phase and Inverse Phase Simultaneous Active State Detection Flag Operation: Changed
		1298	Table 19.52 Pattern Order Detection Flag and Pattern Input Order: Title added
		1303	Figure 19.29 Operation of TSG3nPTSI2 to TSG3nPTSI0 Pin Abnormal Toggle Detection Flag Operation: Changed
		1305	Figure 19.31 Example of Pattern Difference Detection Flag Operation: Changed
		1307	Table 19.54 Change Timing of Pattern Switch Detection Signal: Title added
		1314	Figure 19.40 A/D Conversion Trigger and Diagnostic Output Control Circuit: Changed
		1318	Figure 19.44 Example of Operation of A/D Conversion Trigger Skipping Function: Changed
		1319	Figure 19.45 Example of Operation of A/D Conversion Trigger Skipping Function: Changed
		1319	19.4.5.1 (3) Notes on A/D Conversion Trigger: Changed
		1320	Figure 19.46 Error Interrupt (INTTSG3nIER) Generation Control Circuit: Changed
		1320	19.4.6.1 Error Interrupt Function: Caution changed
		1322	Figure 19.49 Example of Error Interrupt Operation: Changed
		1331	19.4.7.1 (3) Controlling Dead Time in PWM Mode: Note added
		1339	Table 19.64 Timer Output Function in HT-PWM Mode: Functional description of TSG3nO1, 3, and 5 changed, Notes changed
		1339	Table 19.66 Compare Match Timing in HT-PWM Mode: Changed
		1340	Table 19.67 Example of Setting Each Timer Output Condition in HT-PWM Mode: Setting Condition changed
		1346	19.4.7.2 (5) (a) Example of Timer Output Immediately after the Start of the TSG3n Timer Operation: Changed
		1351	19.4.7.2 (7) Dead Time Control in HT-PWM Mode: Changed
		1351	Figure 19.65 Example of Output Waveform near Trough after Reloading: Changed
		1352	19.4.7.2 (8) Notes Concerning Dead Time Control in HT-PWM Mode: Cautions changed
		1354	Figure 19.69 Flow of Software Output Control in HT-PWM Mode: Changed
		1355	19.4.7.2 (9) Software Output Control Function in HT-PWM Mode: Description (2) changed
		1354	Figure 19.69 Flow of Software Output Control in HT-PWM Mode: Title added
		1364	19.4.7.5 Prerequisites: Description changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1371	19.4.7.5 (4) Dead Time Control in SP-PWM mode: Changed
		1372	19.4.7.5 (5) Software Output Control Function in SP-PWM Mode: Procedure (1) and (3) changed
		1376	Figure 19.83 Block Diagram in 120-DC Mode: Changed
		1377	Figure 19.84 Basic Operation Flow in 120-DC Mode: Changed
		1378	Table 19.75 Counter Functions in 120-DC Mode: Description of Clear changed
		1378	Table 19.78 Timer Output Function in 120-DC Mode: Changed
		1382	Table 19.82 Control Method in 120-DC Mode: Title changed, functional description changed
		1382	Table 19.83 Setting of TSG3nS120DCO and Operation in 120-DC Mode: Title added and functional description changed, Cautions deleted
		1382	19.4.7.6 (3) Setting software output control method: Changed
		1382	19.4.7.6 (3) Operation of Software Output Control Method: Changed
		1383	19.4.7.6 (3) Setting pattern switch method: Changed
		1383	19.4.7.6 (3) Operation of pattern switch method: Changed
		1384	19.4.7.6 (3) Operation of trigger switch method: Changed
		1387	19.4.7.6 (5) Operation in 120-DC Mode: Changed
		1388	Figure 19.87 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 0): Notes changed
		1389	Figure 19.88 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 1): Notes changed
		1390	Figure 19.89 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 0): Notes changed
		1391	Figure 19.90 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 1): Notes changed
		1392	19.4.7.6 (6) List of Output Patterns in 120-DC Mode: Description changed
		1392	Table 19.86 Selection of Pattern Order Direction in 120-DC Mode: Changed
		1392	Table 19.87 List of Output Patterns in 120-DC Mode (1/4): Changed
		1392	Table 19.87 List of Output Patterns in 120-DC Mode (2/4): Changed
		1393	Table 19.87 List of Output Patterns in 120-DC Mode (3/4): Changed
		1393	Table 19.87 List of Output Patterns in 120-DC Mode (4/4): Changed
		1393	19.4.7.6 (7) Operation Start Timing in 120-DC Mode: Changed
		1394	Figure 19.92 Control when Timer Output Starts in Reverse Rotation (when Normal Pattern is Input): Changed
		1395	Figure 19.93 Control when Timer Output Starts in Normal Rotation (when Error Pattern is Input): Changed
		1395	Figure 19.94 Control when Timer Output Starts in Reverse Rotation (when Error Pattern is Input): Changed
		1396	19.4.7.6 (8) Output Switch Timing in 120-DC Mode (TSG3nS120DCO = 0): Body changed
		1397	Figure 19.97 Output Switch Example (Switched by TSG3nOPT0.TSG3nPSC): The condition above the figure changed
		1397	Figure 19.98 Output Switch Example (Switched by TSG3nOPT1.TSG3nSPC2 to TSG3nSPC0): The condition above the figure changed
		1399	Figure 19.100 Output Switch Example: Changed
		1401	Table 19.88 TSG3nPTE Toggle Operation when TSG3nPSC is 0: Title added
		1401	Table 19.89 TSG3nPTE Toggle Operation when TSG3nPSC is 1: Title added
		1402	Table 19.90 TSG3nPEC Toggle Operation: Title added
		1402	19.4.7.6 (12) (c) Change Timing of TSG3nO1 to TSG3nO6 Pins: Changed
		1402	Table 19.91 Setting and Clearing of TSG3nTSF: Title added
		1404	19.4.7.6 (13) Basic Control Flow in 120-DC Mode: Body changed
		1407	Figure 19.106 Process Flow of Software Output Control: Changed
		1407	Procedure (1) and (3) below figure 106: changed
		1409	19.4.7.7 Prerequisites: Description changed
		1410	Figure 19.107 Basic Timing in HSP-PWM Mode: Title added and figure changed
		1411	Figure 19.108 Basic Operation Flow in HSP-PWM Mode: Cautions added below the figure
		1416	Table 19.98 Setting Prohibited in HSP-PWM Mode: Title and Caution added

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1417	Table 19.99 Algorithm for Compare Setting in HSP-PWM Mode: Title added
		1421	Figure 19.112 Waveform in HSP-PWM Mode (TSG3nHSPCMUE is set to TSG3nCMP0E < TSG3nHSPCMUE < TSG3nCMP0E + TSG3nDTC1 + 1): Changed
		—	19.4.7.7 (7) Formula for reload adjustment operation: Deleted
		1425	Table 19.100 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is 0 (PWM sift width is set to 0): Title added
		1425	Table 19.101 Formula for Reload Adjustment Operation When TSG3nHSPSHUE/VE/WE is not 0 (PWM sift width is not set to 0): Title added
		1427	Figure 19.116 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 0 (shift width is set to 0)): Setting values changed
		1428	Figure 19.117 Timing Diagram of Operation Start in HSP-PWM Mode (TSG3nHSPSHUE = 5 (shift width is set to five clock cycles)): Changed
		1429	19.4.7.10 Software Output Control Function: Changed
		1429	Table 19.102 Registers Associated with Software Output Control Function: Changed (direction of rotation → electric current direction)
		Section 20 Timer Option Module (TAPA)	
		1436	Table 20.7 TAPAnCTL0 Register Contents, b4 changed (TAPAnOPHS0 deleted)
		Section 21 Timer Pattern Buffer (TPBA)	
		1465	Table 21.17 TPBAnBUFm Register Contents: Description of Caution changed
		1475	21.4.3.2 Access to TPBAnBUFm, In 8 bits × 128 patterns mode (TPBAnDPS = 1): Bit number added
		1478	Table 21.24 Functions of Compare Registers and Buffer Registers: Title and contents changed
		1478	Buffer Registers added after Table 21.24
		1478	Table 21.27 Compare Match Timing: Changed (00_H → 01_H)
		1480	Figure 21.5 Example of Basic Timing: Caution added
		1481	Figure 21.6 Example of Basic Timing: Caution added
		Section 22 Encoder Timer (ENCA)	
		All	Term unification: Encoder timer → ENCA
		1482	22.1.1 Number of Units, ENCA deleted
		1482	Table 22.4 Clock Supply: Description changed (Internal Clock Signal → Supply Clock Name)
		1483	Table 22.6 Reset Sources, title changed
		1483	Table 22.7 External Input/Output Signals, note 1 added
		1484	22.2.1 Functional Overview, changed
		1487 to 1498	22.3.2 ENCAnCTL — ENCAn Control Register to 22.3.12 ENCAnTT — ENCAn Timer Stop Trigger Register, Value after reset, description deleted
		1487	Table 22.9 ENCAnCTL Register Contents (1/2), b14 and b7, description changed
		1488	Table 22.9 ENCAnCTL Register Contents (2/2), b4 to b2, description changed (register → bit)
		1489	22.3.3 ENCAnIOC0 — ENCAn I/O Control Register 0, Access, description changed
		1490	Table 22.11 ENCAnIOC1 Register Contents, description changed
		1491	22.3.5 ENCAnFLG — ENCAn Status Flag Register, Access, description changed
		1491	Table 22.12 ENCAnFLG Register Contents, changed
		1492	22.3.6 ENCAnFGC — ENCAn Status Flag Clear Register, Access, description changed
		1493	22.3.7 ENCAnCCR0 — ENCAn Capture Compare Register 0, Access, description changed
		1493	Table 22.14 ENCAnCCR0 Register Contents, changed
		1494	22.3.8 ENCAnCCR1 — ENCAn Capture Compare Register 1, Access, description changed
		1496	22.3.10 ENCAnTE — ENCAn Timer Enable Status Register, Access, description changed
		1496	Table 22.17 ENCAnTE Register Contents, changed
		1497	22.3.11 ENCAnTS — ENCAn Timer Start Trigger Register, Access, description changed
		1498	22.3.12 ENCAnTT — ENCAn Timer Stop Trigger Register, Access, description changed
		1499	22.4 Functions, changed (phase A, phase B, phase Z → deleted)
		1499	Figure 22.2 Timer Counter Initial Value Setting/Start/Stop, changed
		1499, 1500	22.4 (1) to (4), description changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1501	22.4.2.1 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B, title and description changed
		1501	Table 22.20 When ENCAnUDS1 and ENCAnUDS0 Bits = 00_B, changed
		1501	Figure 22.3 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B, changed
		1502	22.4.2.2 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B, changed
		1502	Table 22.21 When ENCAnUDS1 and ENCAnUDS0 Bits = 01_B, changed
		1502	Figure 22.4 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B, changed
		1503	22.4.2.3 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B, changed
		1503	Table 22.22 When ENCAnUDS1 and ENCAnUDS0 Bits = 10_B, changed
		1503	Figure 22.5 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 10_B, changed
		1504	22.4.2.4 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B, changed
		1504	Table 22.23 When ENCAnUDS1 and ENCAnUDS0 Bits = 11_B, changed
		1504	Figure 22.6 Counter Operation when ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 11_B, changed
		1505	22.4.3 Timer Counter Clear Control by Encoder Input, changed (phase Z encoder input → encoder clearing input signal)
		1505	22.4.3.1 Clearing Method when ENCAnSCE = 0, changed (clear interrupt → encoder clear interrupt)
		1506	Table 22.25 Clearing Conditions of the Timer Counter, changed
		1507	22.4.4.1 Compare Function, note changed
		1507	22.4.4.2 Capture Function, changed (ENCATTIN0 → ENCAnI0)
		1508	22.4.5.1 Compare Function, Compare match interrupt mask function, changed (interrupt detection mask → interrupt mask)
		1509	Table 22.28 Compare Match Interrupt Mask Function, changed
		1509	Table 22.29 Capture Function of ENCAnCTS, contents changed, note changed
		1510	22.4.5.3 Timer Counter Clearing upon Compare Register Match, changed
		1511	22.4.6.1 Starting the Timers, changed
		1511	22.4.6.2 Stopping the Timers, changed
		1512	22.4.6.3 Example of Connection when Two ENCAn Units are Used, changed
		1512	Figure 22.7 Example of Connection when Two ENCAn Units are Used, title and contents changed
		1513	Table 22.31 ENCAn Setting Procedure, changed
		1513	Figure 22.8 Initial Setting Procedure for the Counter, changed
		1514	Figure 22.9 Initial Setting Procedure for Counter Clearing, changed
		1514	Figure 22.10 Setting Procedure for ENCAnCCR0 Register, changed
		1514	Figure 22.11 Setting Procedure for ENCAnCCR1 Register, changed
		1515	22.6.1 Timing of Basic Encoder Operation 1 (Encoder Comparison Mode 1), descriptions of 1 and 2 changed
		1516	22.6.2 Timing of Basic Encoder Operation 2 (Encoder Comparison Mode 2), descriptions of 1 and 2 changed
		1517	22.6.3 Timing of Basic Encoder Operation 3 (Encoder Comparison Mode 3), descriptions of 1 and 2 changed
		1518	22.6.4 Timing of Basic Encoder Operation 4 (Encoder Capture Mode), descriptions of 1 and 2 changed
		1519	22.6.5 Timing of Basic Encoder Operation 5 (Encoder Capture and Comparison Mode, descriptions of 1 and 2 changed
		—	Figure 22.19 Overflow occurrence, deleted
		1520	22.6.6 Overflow Occurrence and Overflow Flag Clear Operation, changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1521	22.6.7 Underflow Occurrence and Underflow Flag Clear Operation, changed
		—	Figure 22.21 Underflow occurrence, deleted
		1522	22.6.8 Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin), title and description changed
		1522	Figure 22.19 Timing Chart of Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin), changed
		1523	22.6.9 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), title and description changed
		1523	Figure 22.20 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), changed
		1524	22.6.10 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), title and description changed
		1524	Figure 22.21 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin), changed
		1525	22.6.11 Overflow Operation Immediately after Startup, description changed
		1525	Figure 22.22 Overflow Operation Immediately after Startup, changed
		1526	22.6.12 Underflow Operation Immediately after Startup, description changed
		1526	Figure 22.23 Underflow Operation Immediately after Startup, changed
		1527	22.6.13 Using the ENCAnLDE Function Immediately after Startup, description changed
		1527	Figure 22.24 Using the ENCAnLDE Function Immediately after Startup, changed
		1528	22.6.14 (1) <When ENCAnLDE = 0>, description of (6) changed
		1528	Figure 22.25 ENCAnLDE Function (when ENCAnLDE = 0), changed
		1529	Figure 22.26 ENCAnLDE Function (when ENCAnLDE = 1), changed
		1530	22.6.15 Conflict between ENCAnLDE Function (loading counter value) and Rewriting of ENCAnCCR0 Register, description changed
		1530	Figure 22.27 Conflict between ENCAnLDE Function and Rewriting of ENCAnCCR0 Register, changed
		1531	22.6.16 Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input (ENCAnEC pin), description changed
		1531	Figure 22.28 Conflict between ENCAnLDE Function and Clear Operation by Encoder Clear Input, changed
		1533	22.6.17 Up-counting after Conflict between ENCAnLDE Function (loading counter value) and Clear Operation by Encoder Clear Input, description changed
		1533	Figure 22.29 Up-counting after Conflict between ENCAnLDE Function and Clearing of the Encoder Counter, changed
		1534	22.6.18 Capture Operation between Counter Clocks (ENCAnCCR1), description changed
		1534	Figure 22.30 Capture Operation between Counter Clocks (ENCAnCCR1), changed
		1534	22.6.19 Capture Operation between Counter Clocks (ENCAnCCR0), description changed
		1534	Figure 22.31 Capture Operation between Counter Clocks (ENCAnCCR0), changed
		1535	22.6.20 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0, description changed
		1535	Figure 22.32 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0, changed
		1536	22.6.21 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1, description changed
		1536	Figure 22.33 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1, changed
		1537	22.6.22 Encoder Operation when Compare Match Clear Control is Disabled, description changed
		1537	Figure 22.34 Encoder Operation when Compare Match Clear Control is Disabled, changed
		1538	22.6.23.1 Accompanying Capture Operation, description changed
		1538	Figure 22.35 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 1, changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1539	22.6.23.2 When the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count, description changed
		1539	Figure 22.36 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Up-count, changed
		1540	Figure 22.37 Clearing for when the Timing of the ENCAnEC Input is the Same as that of the ENCAnE1 Input during Up-count, changed
		1540	Figure 22.38 Clearing for when the Timing of the ENCAnEC Input is Earlier than that of the ENCAnE1 Input during Up-count, changed
		1541	Figure 22.39 Clearing for when the Timing of the ENCAnEC Input is Later than that of the ENCAnE1 Input during Down-count, changed
		1542	22.6.24 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0, description changed
		1542	Figure 22.40 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0, changed
		Section 23 Peripheral Interconnection (PIC)	
		1545	23.2 Peripheral Interconnection 1 (PIC1A), title changed
		1545	23.2.1.1 Functional Overview, cautions changed
		1552	23.2.2.9 PIC1ALHSEL0 — TSG30 Output Low/High Level Select Register, b6 to 1 changed (TSG output → TSG30 output)
		1553	23.2.2.11 PIC1ALHSEL1 — TSG31 Output Low/High Level Select Register, b6 to 1 changed (TSG output → TSG31 output)
		1573	Table 23.36 PIC1AREG30 Register Contents: b22, description changed
		1577	23.2.2.29 Timer Input/Output Control Register 50 (PIC1AREG50), b7 to 5 and b0, description changed
		1590	23.2.3.3 (1) Overview, description changed
		1623	Figure 23.26 Block Diagram of Two-Phase Encoder Control Function (Control Method 1), changed
		1623	23.2.3.7 (2) Configuration, • [PIC1A_ENCAn input selector], description changed
		1624	23.2.3.7 (3) Registers, PIC1A_ENCAn input selector, description changed
		1635	Figure 23.31 Block Diagram of Two-Phase Encoder Control Function (Control Method 3), changed
		1635	23.2.3.9 (2) Configuration, description changed
		1636	Figure 23.32 Block Diagram of PIC1A, changed
		1646	Figure 23.38 Three-Phase Pulse Input Control Function, pin names changed
		1648	Figure 23.39 An Example of Switch Operation from Fixed Phase Control 1 to Variable Phase Control, pin names changed
		1649	Figure 23.40 An Example of Switch Operation from Variable Phase Control to Fixed Phase Control 1, pin names changed
		1652	Figure 23.41 Block Diagram of Three-Phase Encoder Control Function, changed
		1652	23.2.3.11 (2) Configuration, description changed
		1653	23.2.3.11 (3) Registers, Unit (2): PIC1A_ENCAn input selector, description changed
		1654	Figure 23.43 Timing Diagram of Three-Phase Encoder Function_ENCAnUDS1 and ENCAnUDS0 = 00_B, pin and signal names changed
		1654	Figure 23.44 Timing Diagram of Three-Phase Encoder Function (Normal Rotation/Reverse Rotation), pin and signal names changed
		1656	Figure 23.45 Block Diagram of ENCA Input Select Function, changed
		1657	Figure 23.46 Block Diagram of PIC1A, signal names changed
		1657	23.2.3.12 (3) Registers, PIC1A_input selector, description changed
		1659	Figure 23.47 Example of Connection Route 1 (ENCA0 Pin Connected to ENCA0 Timer), signal names changed
		1660	Figure 23.48 Example of Connection Route 2 (ENCA1 Pin and RDC0 Connected to ENCA0 and ENCA1, Respectively), signal names changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1660	Figure 23.49 Example of Connection Route 2 (ENCA1 Pin Connected to ENCA0 and ENCA1), signal names changed
		1662	23.2.3.13 (1) Overview, description changed
		1662	Figure 23.50 Block Diagram of TAUD Input Select Function, pin names changed
		1663	23.2.3.13 (4) Function, description changed
		1665	23.2.3.14 (3) Registers, (2) Switch between low level output and high level output, changed (TSG output → TSG3n output)
		1669	23.3 Peripheral Interconnection 2 (PIC2B), title changed
		1677	Figure 23.58 Block Diagram of TAUD Trigger Output Function, signal names changed
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1682	Figure 24.1 Overall Configuration of EMU2: Changed
		1694	Table 24.26 List of EMU2 Register Functions (General): EMU2nAHBPRT register deleted
		1702	Table 24.35 EMU2nCTR Register Contents: Note1 description changed
		1729	Table 24.57 EMU2nCBCTR1 Register Contents: Description in b0 changed
		1731	Table 24.59 EMU2nANGCTR Register Contents: Description in b1 and 0 changed (Angle data → Angle data/Z-phase signal)
		1731	24.3.27 EMU2nCPJUD0 — EMU2n Compare Judgment Correction Register 0 (n = 0, 1): Description in value after reset changed
		1733	24.3.31 EMU2nPXR — EMU2n Electrical Angle Generation Coefficient Register (n = 0, 1): Description in value after reset changed
		1738	24.3.38 EMU2nVMTCAP — EMU2n Speed Measurement Timer Capture Register (n = 0, 1): Description added
		1741	24.3.41 EMU2nADmk — EMU2n ADm Data Register k (n = 0, 1) (m = 0, 1) (k = 0 to 2): Description added
		1776	24.3.80 EMU2nSR23 — EMU2n Three-phase Voltage Transformation Coefficient Register (n = 0, 1): Description in value after reset changed
		1783	24.3.95 EMU2nUPWM — EMU2n U-phase PWM Register (n = 0, 1): Description changed
		1784	24.3.96 EMU2nVPWM — EMU2n V-phase PWM Register (n = 0, 1): Description changed
		1784	24.3.97 EMU2nWPWM — EMU2n W-phase PWM Register (n = 0, 1): Description changed
		1796	24.3.112 EMU2nCMP0 — EMU2n Compare Register 0 (n = 0, 1): Description changed
		1796	24.3.113 EMU2nCMP1 — EMU2n Compare Register 1 (n = 0, 1): Description changed
		1798	24.3.117 EMU2nIPCMP0 — EMU2n IP Compare Value 0 Register (n = 0, 1): Description deleted
		1808	24.3.134 EMU2nCBIDFIX — EMU2n d-Axis Current Value Verification Buffer Register (n = 0, 1): Description in b31 changed
		1809	24.3.135 EMU2nCBIQFIX — EMU2n q-Axis Current Value Verification Buffer Register (n = 0, 1): Description in b31 changed
		1820	Figure 24.9 Example of Electrical Angle Output: Added
		1820	Description below figure 24.9 Example of Electrical Angle Output: Changed
		1822	(4) Determining Angle Compare 0 Match: Description for positive rotation changed
		1823	(5) Determining Angle Compare 1 Match: Description for positive rotation changed
		1825	24.4.5 (2) Selecting Electrical Angle Source and Obtaining A/D Conversion Results: Description changed
		1826	24.4.5 (3) Calculating Motor Current: Description changed
		1828	24.4.6 PI Control IP: Description changed
		1838	24.4.8 (2) Generating Switching Instruction: Description changed
		1849	24.5.2 Equivalence Check Function: Description changed
		1850	24.5.3.2 Verification of Agreement of Equivalence Check Results: Description changed and added
		Section 25 R/D Converter (RDC2)	
		1870	25.3.5 RDC2nANGDAT — RDC2n Angle Data Register (n = 0, 1), value after reset changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1871	25.3.7 RDC2nMNTC — RDC2n Monitor Pin Setting Register (n = 0, 1), value after reset changed
		1871	25.3.8 RDC2nDATSTR — RDC2n Data Storage Register, value after reset changed
		1873	25.3.12 RDC2nINIT — RDC2n Initializing Register 2 (n = 0, 1), value after reset changed
		1882	25.3.20 RDC2nERDET — RDC2n Error Detection Register (n = 0, 1), value after reset changed
		1883	Table 25.40 Errors Indicated by EID[2:0], description added
		1884	25.3.21 RDC2nCMINT — RDC2n Compare Match Interrupt Register (n = 0, 1), value after reset changed
		1886	Table 25.42 RDC2nENCP Register Contents, description of the HYSS bit added, and description in b7 to b5 changed
		1891	25.3.30 RDC2nEXSQR — RDC2n Excitation Amplitude Integral Square-Sum Monitor Register (n = 0, 1), value after reset changed
		1895	25.4.1.8 Encoder Pulse Output Function, description changed
		Section 26 A/D Converter (ADCC)	
		1920	26.2.1 Functional Overview, A/D conversion start trigger: Note added
		1935	26.3.11 ADCCnTHSTPCR — T&H Stop Control Register: Caution added
		1937	26.3.15 ADCCnTHACR — T&H Group A Control Register: Caution changed
		1938	26.3.16 ADCCnTHBCR — T&H Group B Control Register: Caution changed
		1940	Table 26.26 ADCCnSFTCR Register Contents: Description in b1 changed (parity error interrupts → parity errors)
		1950	Table 26.38 ADCCnSGCRx Register Contents (x = 0 to 2): Description in b0 and Caution changed (H/W trigger → A/D conversion start trigger)
		1951	Table 26.39 ADCCnSGCRx Register Contents (x = 3 or 4): Description in b0 changed (H/W trigger → A/D conversion start trigger)
		1952	26.3.32 ADCCnSGVCSPx — Scan Group x Start Virtual Channel Pointer: Cautions added
		1952	26.3.33 ADCCnSGVCEPx — Scan Group x End Virtual Channel Pointer: Caution added
		1962	26.4.3.2 (2) Starting Scan Groups by A/D Timer Triggers: Description changed
		1970	26.4.5.3 A/D Parity Error Trigger: Description changed
		1971	Figure 26.21 Functional Diagram of Pin-Level Self-Diagnosis: Changed
		1978	26.5.5 Procedure for Setting Wiring-Break Detection Self-Diagnosis: Description changed
		1978	Figure 26.27 Flow of Wiring-Break Detection Self-Diagnostic Settings: Changed
		1980	26.7 Notes: Note 3. changed, Note 4. added
		1980	Table 26.49 Notes on Setting Registers: Title added
		Section 27 Functional Safety	
		All	Expression of terms and bit names unified: PROT1, PROT0 → PROT[1:0], (PROT1, PROT0) = (x, x) → PROT[1:0] bits to xx_B, LRAM → local RAM, GRAM → global RAM, D-Parity → data parity
		1982	Table 27.1 ECC Overview, discription changed
		1983	27.2.1.2 Address Parity, changed (address EDC (parity) → address parity)
		1983	27.2.1.3 Data Parity, changed (data EDC (parity) → data parity)
		1984	Table 27.3 Overview of the Code Flash ECC, table title added, description changed, description below the table changed
		1986	Table 27.4 List of Registers, changed (Initial Value → Value after Reset)
		1987	Table 27.5 CFAPCTL Register Contents, changed
		1988	Table 27.6 CFECCCTL Register Contents, changed
		1990	(4) CFSTCLR_VCI/PE1/PE2 — Code Flash Status Clear Register, title changed
		1990	(4) CFSTCLR_VCI/PE1/PE2 — Code Flash Status Clear Register, description changed
		1991	(5) CFOVFSTR_VCI/PEI/PE2 — Code Flash Error Count Overflow Status Register, title changed
		1991	(5) CFOVFSTR_VCI/PEI/PE2 — Code Flash Error Count Overflow Status Register, description changed
		1992	(6) CF1STERSTR_VCI/PE1/PE2 — Code Flash 1st Error Status Register, title changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	1992	(6) CF1STERSTR_VCI/PE1/PE2 — Code Flash 1st Error Status Register, description changed
		1994	(8) CFSTSTCTL_VCI/PE1/PE2 — Code Flash Sub-Test Control Register, description changed
		1994	Table 27.12 CFSTSTCTL Register Contents, changed
		1995	Table 27.13 Results of Code Flash Reading, table title added
		1995	27.2.2.4 (3) Self-diagnosis, changed
		1996	Table 27.14 Overview of the Data Flash ECC, table title added, description changed
		1997	Table 27.16 List of Registers, changed (Initial Value → Value after Reset)
		1998	Table 27.17 DFECCCTL Register Contents, changed
		1999	(4) DFERSTR — Data Flash Error Status Register, description changed
		2003	Table 27.24 DFTSTCTL Register Contents, changed
		2004	27.2.3.3 (3) Self-diagnosis, changed
		2005	Table 27.25 Overview of the Local RAM ECC of CPU1 and CPU2, table title added, description changed
		2006	Table 27.27 Relationship between Addresses and Banks, table title added
		2007	Table 27.28 List of Registers, changed (Initial Value → Value after Reset)
		2008	Table 27.29 LRAPCTL Register Contents, changed
		2009	Table 27.30 LRTSTCTL Register Contents, b15 and b14, description changed
		2011	Table 27.32 LRECCCTL Register Contents, changed
		2013	27.2.4.3 (6) LRSTCLR_PE1/PE2 — Local RAM Status Clear Register, description changed
		2015	Table 27.36 LR1STERSTR Register Contents, description of the DEDFn flag changed
		2016	(9) LR1STEADRn_PE1/PE2 — Local RAM 1st Error Address Register n (n = 0 to 3), description changed, EADR[12:0] changed to EADR[14:0]
		2016	Table 27.37 LR1STEADRn Register Contents, b14 to b0 changed (EADR[12:0] → EADR[14:0])
		2017	27.2.4.4 (3) Writing to the ECC and address parity bits, description of (c) changed
		2019	Table 27.38 Overview of the Global RAM ECC, table title added, description changed
		2020	Figure 27.2 ECC of Global RAM and Address Parity, changed (PWM → RWM)
		2021	Table 27.39 List of Registers, changed (Initial Value → Value after Reset)
		2022	27.2.5.3 (1) GRECCCTL_GRAMC — Global RAM ECC Control Register (GRAMC), title and description changed
		2022	Table 27.40 GRECCCTL_GRAMC Register Contents, changed
		2023	27.2.5.3 (2) GRTSTCTL — Global RAM Test Control Register, description changed
		2023, 2024	Table 27.41 GRTSTCTL Register Contents, description changed
		2025	Table 27.42 GRTDATBFn Register Contents, description changed
		2026	27.2.5.3 (4) GRECCCTL_VCI/PE1/PE2 — Global RAM ECC Control Registerr (VCI/PE1/PE2), title and description changed
		2026	Table 27.43 GRECCCTL_VCI/PE1/PE2 Register Contents, description changed
		2030	Table 27.47 GRSTCLR Register Contents, b1 and b0, description added
		2031	Table 27.48 GROVFSTR Register Contents, b1 and b0, description added
		2033	27.2.5.3 (11) GR1STEADRn_VCI/PE1/PE2 — Global RAM 1st Error Address Register n (n = 0, 1), description changed
		2034	27.2.5.4 (3) Writing to the ECC and address parity bits, description of (c) changed
		2037	Table 27.51 Overview of the Instruction Cache ECC, table title added, contents changed
		2038	Table 27.52 List of Registers, changed (Initial Value → Value after Reset)
		2039	Table 27.53 IDECCCTL Register Contents, changed
		2041	27.2.6.3 (3) IDSTCLR_PE1/PE2 — Instruction Cache Data RAM Error Status Clear Register, changed (error overflow → error count overflow)
		2045	(7) ITECCCTL_PE1/PE2 — Instruction Cache Tag RAM ECC Control Register, description changed, b1 changed
		2045	Table 27.59 ITECCCTL Register Contents, changed
		2047	(9) ITSTCLR_PE1/PE2 — Instruction Cache Tag RAM Error Status Clear Register, changed (error overflow → error count overflow)

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	2050	Table 27.64 IT1STEADR0 Register Contents, b8 to b0 changed
		2051	27.2.8.1 Overview, Error Notification, title and descriptions changed
		2051	27.2.8.1 Overview, Address Capture, changed
		2053	Table 27.65 Modules Provided with the ECC and Base Addresses, table title added
		2053	Table 27.66 List of Registers, changed (Initial Value → Value after Reset)
		2053	Table 27.67 Register Map, table title added
		2054	27.2.8.3 (1) E710CTL — ECC Control Register, changed (ECC macro → ECC module, EMCA1, 0 → EMCA[1:0])
		2054, 2055	Table 27.68 E710CTL Register Contents, changed
		2056	(2) E710TMC — ECC Test Mode Control Register, changed ETMA1, 0 → ETMA[1:0])
		2056, 2057	Table 27.69 E710TMC Register Contents, changed
		2058	(3) E710TED — ECC Encoder and Decoder Data Test Register, description below NOTE changed
		2059	(5) ECSYND — ECC Decoder Syndrome Data Register, b6 to b0 and bit number, changed
		2060	(6) ECHORD — ECC 7-Bit Redundant Data Holding Test Register, b6 to b0 and bit number, changed
		2060	(7) ECECRD — ECC Encoder Test Register, b6 to b0 and bit number, changed
		2061	(8) ECERDB — ECC Redundant Bit Input and Output Substitution Buffer Register, b6 to b0 and description, changed
		2062	27.2.8.4 Notification to ECM, changed
		2064	27.2.8.5 Test Function, changed (figures deleted, descriptions added)
		2064	Table 27.70 Transfer Paths to which Data Parity is Applied, table title added, contents changed (DNF deleted, description below the table changed)
		2064	Table 27.71 List of Registers, changed (Initial Value → Value after Reset)
		2064	Table 27.72 List of Data Parity Control Modules, changed (DNF0 and DNF1 deleted)
		2068	27.2.9.2 (4) APDPERRADR_xx — P-Bus Data Parity Error Address Register, R/W of b15 to b0 changed, cautions added
		2069	27.3 Lockstep, description changed
		2069	Table 27.77 List of Registers, changed (Initial Value → Value after Reset)
		2073	Table 27.80 Identifiers for Slave Guard, table title added
		2074	Table 27.81 List of Registers, changed (Initial Value → Value after Reset)
		2075	27.4.2.2 (1) MGDGRPROTn — GRG Protection Setting Register n (n = 0 to 3), b31, 29 to 27, and 3 to 0 changed to reserved bits
		2075	Table 27.82 MGDGRPROTn Register Contents, b31, 29 to 27, and 3 to 0 changed to reserved bits, description of b30 changed
		2076	(2) MGDGRBADn — GRG Compare Base Address Register n (n = 0 to 3) and Table 27.83 MGDGRBADn Register Contents, changed (RFU bits → reserved bits)
		2076	(3) MGDGRADVn — GRG Valid Compare Address Register n (n = 0 to 3) and Table 27.84 MGDGRADVn Register Contents, changed (RFU bits → reserved bits), description of b20 to b9 changed, description below the table added
		2078	(4) MGDGRSCTL_VCI/PE1/PE2 — GRG Control Register n and Table 27.85 MGDGRSCTL Register Contents, title and description changed (RFU bits → reserved bits, descriptions of b1 and b0 changed)
		2078	Table 27.86 Settings of the ERRCLO and ERRCLE Bits, description changed, table title added
		2079	(5) MGDGRSSTAT_VCI/PE1/PE2 — GRG Error Status Register and Table 27.87 MGDGRS-STAT Register Contents, title and description changed (RFU bits → reserved bits, descriptions of b1 and b0 changed)
		2079	(6) MGDGRSAD_VCI/PE1/PE2 — GRG Error Address Register and Table 27.88 MGDGRSAD Register Contents, title and description changed (RFU bits → reserved bits)
		2080	(7) MGDGRSTYPE_VCI/PE1/PE2 — GRG Error Access Type Register, title changed
		2080	Table 27.89 MGDGRSTYPE Register Contents, b17, 16 and b4 to 0, description changed
		2081	Table 27.90 Peripheral Modules to be Protected and the Corresponding PBG Channel Numbers, table title added

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	2082	Table 27.91 List of Registers, table title added, header changed (Initial Value → Value after Reset)
		2082	Table 27.92 List of Registers (for PBG Group), table title added, header changed (Initial Value → Value after Reset)
		2082	Table 27.93 Base Address Values for the PBG Channel Numbers, table title changed
		2083	Table 27.94 FSGDxxDPROTn Register Contents, b3 to b0, description changed
		2084	Table 27.95 ERRSLVxxCTL Register Contents, b1 and b0, description changed
		2085	Table 27.96 ERRSLVxxSTAT Contents, b1 and b0, description changed
		2085	Table 27.97 ERRSLVxxADDR Register Contents, b31 to 24, description of reserved bits added
		2087	Table 27.99 Specification Overview, Automatic signature comparison, Error notification, description changed
		2088	27.5.2.1 MISG, changed (three signature generation units → two signature generation units)
		2088	Figure 27.3 MISG Block Diagram (RH850/C1H), changed (block→unit)
		2090	Figure 27.5 Block Diagram of Signature Generation in MISR1H and MISR1L and the Polynomials, figure title added
		2090	Figure 27.6 Block Diagram of Signature Generation in MISR2H and MISR2L and the Polynomials, figure title added
		2091	Table 27.100 Signature Generation Conditions in MISRi, figure title added
		2092	27.5.3.2 Automatic Signature Comparison, description changed
		2094	Table 27.101 Registers of the Signature Generation Units, table title added, contents changed (Initial Value → Value after Reset, entries in Access Size column for 8 and 16 changed)
		2095	Table 27.102 Registers of the Signature Comparison Unit (MSD Sub-Block), table title added, header changed (Initial Value → Value after Reset)
		2096	Table 27.103 MISRCDRL Register Contents, changed (MISR → MISR1)
		2101	27.5.4.7 MISRCR_PE1/PE2 — MISR Control Register, changed (MISR1CND2 → MISR2CND)
		2101	Table 27.108 MISRCR Register Contents, b7 to 4, description of reserved bits added
		2104	Table 27.111 MISRDCNTCTL Register Contents, b7 to 2, description of reserved bits added
		2112	Table 27.117 List of Clocks Monitored by Each Clock Monitor and Sampling Clocks Used, changed (WDT → WDTA)
		2112	Table 27.118 List of Registers, changed (Initial Value → Value after Reset)
		2112	Table 27.119 Register Base Addresses, table title added
		2113	Table 27.120 Common Registers, table title added, header changed (Initial Value → Value after Reset)
		2113	Table 27.121 CLMAnCTL0 Register Contents, b7 to 1, description of reserved bits added
		2113	Table 27.122 CLMAnCMPL Register Contents, b15 to 12, description of reserved bits added
		2114	Table 27.123 CLMAnCMPH Register Contents, b15 to 12, description of reserved bits added
		2114	27.6.3.5 CLMAnPS — CLMAn Protection Command Status Register, b7 to 1, description of reserved bits added, • Operating conditions of the CLMAnPRERR bit added
		2114	Table 27.124 CLMAnPS Register Contents, b7 to 1, description of reserved bits added
		2116, 2117	27.6.3.6 CLMATEST — CLMA Self-Test Register, (1) Calculation method of the thresholds CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0] and (2) Definition of the input of initial values to the threshold registers, added
		2119	Figure 27.8 Operation when Clock Frequency is within the Specified Range, changed,
		2119	Figure 27.9 Operation when Clock Frequency is Lower than the Specified Range, changed
		2120	Figure 27.10 Operation when Clock Frequency is Higher than the Specified Range, added
		2123	Table 27.127 Failure Detection, table title added
		2123	27.7 BIST, discription changed
		2123	Figure 27.11 Lock Step Configuration, added
		Section 28 Error Control Module (ECM)	
		All	Term unification: Error pin, error output → ERROROUT pin, ERROROUT output
		2125	28.1.5 Reset Sources, description changed
		2125	Table 28.6 External Input/Output Signals, entries in “Description” column changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	2126	Table 28.7 Function Overview, changed
		2127	28.2.2 Block Diagram, changed (PIC1 → PIC1A)
		2127	Figure 28.1 Block Diagram of ECM, changed
		2128	28.2.3 Error Sources and Safety Processing, title and description changed
		2128, 2129	Table 28.8 List of Error Sources and Safety Processing, table contents changed, notes added
		2130	Table 28.9 Error Source Aggregation, table contents changed, note added
		2131	Table 28.10 List of Registers (ECM Master), table title and contents changed
		2131	Table 28.11 List of Registers (ECM Checker), table title and contents changed
		2132	Table 28.12 List of Registers (ECM Common Part), table title and ECMESSTCx register names changed, description below the table deleted
		2133	28.3.2 ECMmESET (m = M/C) — ECM Master/Checker Error Set Trigger Register, description changed, cautions changed
		2133	Table 28.13 ECMmESET Register Contents, functional description of b0 changed, cautions changed
		2134	28.3.3 ECMmECLR (m = M/C) — ECM Master/Checker Error Clear Trigger Register, title and description changed, cautions changed
		2134	Table 28.14 ECMmECLR Register Contents, functional description of b0 changed, cautions changed
		2135	28.3.4 ECMmESSTR0 (m = M/C) — ECM Master/Checker Error Source Status Register 0, description changed
		2135	Table 28.15 ECMmESSTR0 Register Contents, changed
		2136	28.3.5 ECMmESSTR1 (m = M/C) — ECM Master/Checker Error Source Status Register 1, description changed
		2136	Table 28.16 ECMmESSTR1 Register Contents, functional descriptions of b31, b15 to b9, and b7 to b5 changed
		2137	28.3.6 ECMmPCMD0 (m = M/C) — ECM Master/Checker Protection Command Register, description changed, R/W changed
		—	28.3.7 ECMmESSTR0n (m = M/C, n = A/B/C) — ECM Master/Checker Error Source Status Register 0n, deleted
		2138	28.3.7 ECMEPCFG — ECM Error Pulse Configuration Register, description changed
		—	28.3.8 ECMmESSTR1n (m = M/C, n = A/B/C) — ECM Master/Checker Error Source Status Register 1n, deleted
		2139 to 2146	28.3.8 ECMMICFG0 — ECM Maskable Interrupt Configuration Register 0 to 28.3.15 ECMEMK1 — ECM Error Mask Register 1, description and access changed
		2139 to 2157	Table 28.19 ECMMICFG0 Register Contents to Table 28.37 ECMDTMCFG3 Register Contents, error source numbers changed
		2141	Table 28.21 ECMNMICFG0 Register Contents, description of b31and b30 changed
		2143	Table 28.23 ECMIRCFG0 Register Contents, functional description of b3 to b1 changed, error source numbers changed
		2144	Table 28.24 ECMIRCFG1 Register Contents, error source numbers changed
		2147 to 2151	28.3.16 ECMESSTC0 — ECM Error Source Status Clear Trigger Register 0 to 28.3.21 ECMPE1 — ECM Pseudo Error Trigger Register 1, description changed
		2148	Table 28.28 ECMESSTC1 Register Contents, description of b30,b29 changed
		2152	28.3.22 ECMDTMCTL — ECM Delay Timer Control Register, description and access changed
		2152	Table 28.33 ECMDTMCTL Register Contents, functional description of b0 changed
		2152, 2153	28.3.23 ECMDTMR — ECM Delay Timer Register to 28.3.24 ECMDTMCMP — ECM Delay Timer Compare Register, description changed
		2154 to 2157	28.3.25 ECMDTMCFG0 — ECM Delay Timer Configuration Register 0 to 28.3.28 ECMDTMCFG3 — ECM Delay Timer Configuration Register 3, description and access changed
		2158	28.4.1 Operations for ERROROUT Output, changed
		2158	28.4.1.1 Enabling Dynamic Mode, changed
		2158	28.4.1.2 Disabling Dynamic Mode, changed
		2158	28.4.2 Loop-Back Function, changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	2159	28.4.5.1 Protection Unlock Sequence, changed (No.5 deleted)
		2160	28.4.6 Timeout Function for Interrupt Processing, description changed
		Section 29 Data CRC (DCRA)	
		All	Notation unified: DCRAn → DCRA
		2161	Table 29.3 Register Base Addresses, table header changed
		2161	Table 29.4 DCRAn Clock Supply, table header changed
		2161	29.1.4 Reset Source, discription changed
		2161	Table 29.5 Reset Source, table header and discription changed
		2163	29.2.3 Operation Circuit, added
		2165	29.3.2 DCRAnCIN — CRC Input Register, description deleted, value after reset changed
		2167	29.3.4 DCRAnCTL — CRC Control Register, value after reset changed, note deleted, caution changed
		2168	Table 29.10 Setting Example of Initial Start Values (When Read at a Reset), table title changed
		Section 30 Intelligent Cryptographic Unit (ICUSB)	
		All	Notation unified: ICU-S → ICUSB
		Section 31 On-Chip Debugging Unit (OCD)	
		2171	31.1 (10) Mask Function, changed (pin reset → external reset)
		2173	31.4.1 Overview, changed (this LSI → this product)
		2175	Table 31.2 I/O Pins of AUDR, changed (H → high level), and description of AUDCK pinchanged
		2177	Table 31.4 AUDISR Register Contents, functional descriptions of b15 to 4 and reserved bits changed
		Section 32 Flash Memory	
		2188	Introductory sentence changed
		2188	32.1 Features, changed
		2189	32.2 Structure of Memory, changed
		2191	Figure 32.3 Data Flash Memory Mapping (64 bytes × 512 configuration), description above the figure changed
		2193	Table 32.2 Programming Methods, changed (data read → data access, local RAM → local RAM or global RAM)
		2194	Table 32.3 Basic Functions at a Glance, description above the figure changed
		2194	Table 32.3 Basic Functions at a Glance, changed
		2195	32.4 Functional Overview. changed
		2195	Table 32.4 Summary of Security Functions, description above the figure changed
		2195	Table 32.4 Summary of Security Functions, changed
		2196	Table 32.5 Available Operations and Security Settings, note 1 added, description below note 1 changed
		2196	Table 32.6 Summary of Protection Functions, changed
		2197	32.5 Serial Programming, title changed
		2198	32.6 Selection of the Communication Method, added
		2198	32.7 Self-Programming, title changed
		2198	32.7.1 Overview, changed
		2198	Figure 32.6 Concept of Self-Programming, changed
		2199	32.8.1 Reading Code Flash Memory, description added
		—	32.9 Detection and Correction of Errors in Flash Memory, deleted
		2200	Table 32.9 FRDCYCLD Register Contents, changed
		2203	Table 32.13 List of Registers Related to Product Information, value after reset changed
		2203	Table 32.14 Relationship between Product Name and PRDNAME Initial Value (Value after Reset), added
		2204	32.9.3.1 PRDNAMEn — Product Name Storage Register (n = 1 to 4), value after reset changed
		2204	Table 32.15 List of Registers Related to Product Information, changed

Rev.	Date	Description	
		Page	Summary
0.50	Aug 28, 2014	2205	32.10 Option Bytes, description added
		2205	32.10.1 OPBT0 — Option Byte 0 Register, value after reset changed, note 1 added
		2207	32.10.2 OPBT2 — Option Byte 2 Register, note 1 changed
		2208	32.11 Notes, (4) and (6) changed, (7) to (9) changed and added
		Section 33 Flash Security	
		All	Whole section revised
		Section 34 RAM	
		2217	34.2 Features, description changed
		Section 36 Electrical Characteristics	
		2233	Table 36.10 DC Characteristics (Input Capacitance), measurement conditions changed
		2236	Table 36.12 Power On/Off Timings, note 4 changed (2 µs → 1.2 µs)
		2237	Table 36.13 Power On/Off Timings, note 4 changed (2 µs → 1.2 µs)
		2239	Table 36.17 Reset Signals, max. values changed (— → TBD)
		2240	Table 36.18 Interrupt Signals, max. value changed (— → TBD)
		2243	Figure 36.8 CSIH Timing (Master Mode), changed
		2258	Table 36.31 A/D Converter Characteristics, differential nonlinear error item deleted
		2259	Table 36.32 RDC Conversion Performance, Startup stabilization time, deleted
		2261	Table 36.33 RDC Analog Pin Characteristics, notes 10 and 11 added
1.00	Mar 04, 2015	All	Intelligent Cryptographic Unit (ICUSB): Description and section deleted
		All	Bit name and functional description of reserved bits unified
		All	Term unification: valley → trough, valid edge → effective edge, odd channel / even channel → odd-numbered channel / even-numbered channel, ECC 1-bit error / ECC 2-bit error → 1-bit ECC error / 2-bit ECC error, reversed value → inverse
		How to Use This Manual	
		4	Reference manual (software user's manual) in "Organization" and "How to read this manual" changed
		Section 1 Overview	
		50	Table 1.2 List of Products: Part number of C1H and C1M corrected
		55	Figure 1.4 Pin Connections of C1M: Pin names of pin number 80 and 83 corrected
		63	Table 1.5 Pin Assignments of C1M: Pin names of pin number 80 and 83 corrected
		Section 2 Pins	
		All	Table 2.25 List of Registers in C1H Port Group 0 to Table 2.32 List of Registers in C1H Port Group 7, Table 2.41 List of Registers in C1M Port Group 0 to Table 2.47 List of Registers in C1M Port Group 7: Module name column added
		All	Term unification: DNF macro number → DNF group number
		67	2.1.2.1 Terms: description of port group corrected
		68	Table 2.3 Pin Function Configuration (Outline): Corrected
		70	Table 2.5 PPRn_m Read Values: Description in the PPRn_m Read Value column corrected
		70	2.1.2.3 Pin Data Input/Output: Description of PBDCn.PBDCn_m corrected
		72	Figure 2.1 Block Diagram of Pin Configuration: Note deleted
		74	2.1.4 Port Group Configuration Register: Pin-unit register added
		76	2.1.4.2 (2) PMCSRn — Port Mode Control Set/Reset Register: Description corrected
		77	Table 2.9 PIPCn Register Contents: Functional description corrected
		78	2.1.4.2 (5) PMSRn — Port Mode Set/Reset Register: Description corrected
		84	2.1.4.3 (5) PSRn — Port Set/Reset Register: Description corrected
		86	2.1.4.5 (1) PCRn_m — Port Control Register: Corrected (PCR → PCRn_m)
		87	2.1.4.6 Example Port Settings: Description corrected
		92	2.1.5.2 Alternative Function to be Used in Direct I/O Control Alternative Mode: Description added
		93	2.1.5.4 Selecting Function for JTAG Port: Added
		129	Table 2.58 C1H Pin Function (3/3): Corrected (entry in the I/O column for TSG3n channel output m corrected)

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	131	Table 2.59 C1M Pin Function (2/2): Corrected (entry in the I/O column for TSG3n channel output m corrected)
		Section 3 CPU System	
		136	Figure 3.1 Block Diagram of the C1H Configuration: Corrected
		138	Table 3.1 Features of the RH850 G3M Core: CPU, CPU operating modes corrected
		139	Table 3.2 List of Program Registers: Context column and notes deleted
		139	Table 3.2 List of Program Registers: Description in NOTE corrected
		140	3.2.1.2 (1) (a) General-purpose registers: Description corrected
		141	3.2.1.2 (2) Basic System Registers: Description corrected
		141	Table 3.3 Basic System Registers: Context column, SR1, 1, and Note 1 and Note 2 deleted
		142	3.2.1.2 (2) (a) EIPC — Status save register when acknowledging EI level exception: Reference corrected
		143	3.2.1.2 (2) (b) EIPSW — Status save register when acknowledging EI level exception: b31 and b19 in the bit chart corrected
		143	Table 3.5 EIPSW Register Contents: b31 and b19 in the bit chart corrected
		144	3.2.1.2 (2) (c) FEPC — Status save register when acknowledging FE level exception: Reference corrected
		145	3.2.1.2 (2) (d) FEPSW — Status save register when acknowledging FE level exception: b31 and b19 in the bit chart corrected
		145	Table 3.7 FEPSW Register Contents: b31 and b19 corrected
		146	3.2.1.2 (2) (e) PSW — Program status word: b31 and b19 in the bit chart corrected
		146	Table 3.8 Access Permission for PSW Register: b31 and b19 deleted
		146, 147	Table 3.9 PSW Register Contents: b31, b19, b7, and b5 corrected, Note 1 to Note 4 deleted
		149	Table 3.10 EIIC Register Contents: Functional description corrected
		149	Table 3.11 FEIC Register Contents: Functional description corrected
		152	3.2.1.2 (2) (n) HTCFG0 — Thread configuration register: b15 and b10 to 0 in the bit chart and value after reset corrected
		152	Table 3.18 HTCFG0 Register Contents: Description of b15 to b0 corrected
		153	Table 3.20 MEI Register Contents: R/W column corrected
		154	Table 3.21 Instructions Causing Exceptions and Values of MEI Register: HVCALL instruction deleted
		155	3.2.1.2 (2) (q) RBASE — Reset vector base address register: Value after reset in the bit chart corrected
		155	Table 3.22 RBASE Register Contents: R/W column and R/W of b31 to 9 and b0 corrected
		156	3.2.1.2 (2) (t) PID — Processor ID register: Value after reset in the bit chart corrected
		156	Table 3.25 PID Register Contents: Description of b23 deleted, value after reset corrected
		158	3.2.1.2 (2) (w) MCFG0 — Machine configuration register: b2 to 0 in the bit chart corrected, value after reset corrected
		158	Table 3.28 MCFG0 Register Contents: b2 to 0, R/W, and value after reset corrected, Note 1 added
		—	3.2.1.2 (2) (x) MCFG1 — Machine configuration register: Deleted
		158	3.2.1.2 (2) (x) MCTL — Machine control register: b31 and b21 to 16 corrected, value after reset corrected
		158	Table 3.29 MCTL Register Contents: Description of b31 and b21 to 16 and R/W corrected, value after reset corrected, table header corrected
		159	3.2.1.2 (3) Interrupt Function Registers: Description corrected
		159	Table 3.30 Interrupt Function System Registers: Context column deleted, Note 1 deleted
		—	3.2.1.2 (4) Hardware thread function registers: Deleted
		—	3.2.1.2 (5) Virtualization support function registers: Deleted
		163	3.2.1.2 (4) FPU Function Registers: Description corrected
		163	Table 3.36 FPU System Registers: Context column deleted, Note 1 deleted
		163	3.2.1.2 (5) MPU function registers: Description corrected
		164	Table 3.37 MPU Function System Registers: Context column deleted, SR20, 5 deleted, Note 1 deleted
		165	3.2.1.2 (5) (a) MPM — Memory protection operation mode register: Description corrected, b2 in the bit chart corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	165	Table 3.38 MPM Register Contents: Functional description of b2 corrected
		166	Table 3.39 MPRC Register Contents: Table header corrected, description of b11 to 0 and R/W corrected
		166	3.2.1.2 (5) (c) MPBRGN — MPU base region register: Value after reset corrected
		166	Table 3.40 MPBRGN Register Contents: Functional description of b4 to 0, value after reset corrected, Note 1 deleted
		167	3.2.1.2 (5) (d) MPTRGN — MPU end region register: Functional description of b4 to 0, value after reset corrected, Note 1 deleted
		—	3.2.1.2 (5) (i) MPPRT0 — Protection area allocation register: Deleted
		169	Table 3.46 MPLAn Register Contents: R/W column and R/W of b31 to 2 corrected, Note 1 deleted
		169	3.2.1.2 (5) (j) MPUAn — Protection area upper limit address register: R/W column and R/W of b31 to 2 corrected, Note 1 deleted
		169	Table 3.48 MPATn Register Contents: R/W column and functional description of b7 to 0 corrected, Note 1 deleted
		171	3.2.1.2 (6) Cache Operation Function Registers: Description corrected
		171	Table 3.49 Cache Operation Function Registers: Context column deleted, Note 1 deleted
		172	3.2.1.2 (6) (b) ICTAGH — Instruction cache tag Hi access register: b6, b1, and b0 in the bit chart corrected
		172	Table 3.51 ICTAGH Register Contents: Functional description of b30, b23 to 16, b6, b1, and b0 corrected, R/W, value after reset corrected
		177	3.2.1.2 (7) Data Buffer Operation Function Registers: Description corrected, context column and Note 1 deleted
		178	3.2.2.1 Features: Corrected (CPU1 → CPU)
		178	Figure 3.2 Instruction Cache and Data Buffer: Corrected (CPU1 → CPU)
		179	Figure 3.3 Instruction Cache Configuration: Corrected
		179	3.2.2.2 Instruction Cache Function: Tag Array, description of VM bit and VCID deleted
		180	3.2.2.2 Instruction Cache Function: LRU, description in CAUTIONS corrected
		181	Table 3.59 List of Registers: Corrected
		181	Table 3.60 IPIR_CHn Interrupt Register Contents: Table title corrected
		182	3.2.4.1(2) Protection Made by SPID: • Setting PEG corrected (• Setting PEG protection → • Setting)
		183	Figure 3.4 Access Permission by the System Protection Identifier (SPID): Corrected (Local ROM area → Local RAM area)
		183	3.2.4.1(3) List of PEG Setting Registers: Description corrected
		183	Table 3.61 List of Registers: Corrected
		186	3.2.4.1(4) (c) PEGGnBA — PE guard area n base setting register (n = 0 to 3): Description corrected
		187	3.2.4.2 (1) (4) Notifying violation: Description corrected, NOTE deleted
		187	3.2.4.2 (1) (5) Invalidating subsequent accesses: NOTE 2 deleted
		188	3.2.4.2 (3) IPG Setting Registers for Illegal Users: Description corrected
		188	Table 3.65 List of Registers: Corrected
		—	3.2.4.2 (4) IPG Protection Setting Registers for Illegal Virtual Machines: Deleted
		189	3.2.4.2 (4) (a) IPGECRUM — Peripheral device protection violation access information register: Title corrected
		189	Table 3.66 IPGECRUM Register Contents: Table title and NOTE corrected
		190	3.2.4.2 (4) (b) IPGADRUM — Peripheral device protection violation access address register: Title corrected
		190	Table 3.67 IPGADRUM Register Contents: Table title and NOTE corrected
		190	3.2.4.2 (4) (c) IPGENUM — Peripheral device protection enable register: Title corrected
		190	Table 3.68 IPGENUM Register Contents: Table title corrected, functional description of b0 corrected (value after reset deleted)
		191	3.2.4.2 (4) (d) IPGPMTUM0 — Peripheral device protection setting register: Title corrected
		191	Table 3.69 IPGPMTUM0 Register Contents: Table title corrected, functional description of b6 to 4 corrected (value after reset deleted)
		192	3.2.4.2 (4) (e) IPGPMTUM1 — Peripheral device protection setting register1: Title corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	192	Table 3.70 IPGPMTUM1 Register Contents: Table title corrected, functional description of b6 and b2 corrected (value after reset deleted)
		193	3.2.4.2 (4) (f) IPGPMTUM2 — Peripheral device protection setting register2: Title corrected, b6 and b2 corrected
		193	Table 3.71 IPGPMTUM2 Register Contents: Table title corrected, functional description of b6 to 4, b2 to 0, b6, and b2 corrected (MECNT → MEV, value after reset deleted, b6 and b2 changed to reserved)
		194	3.2.4.2 (4) (g) IPGPMTUM3 — Peripheral device protection setting register3: Title corrected, b6 corrected
		194	Table 3.72 IPGPMTUM3 Register Contents: Table title corrected, functional description of b6 to 4 corrected (value after reset deleted, b6 changed to reserved)
		194	3.2.4.2 (4) (h) IPGPMTUM4 — Peripheral device protection setting register4: Title corrected
		194	Table 3.73 IPGPMTUM4 Register Contents: Table title corrected, functional description of b1 and b0 corrected (value after reset deleted)
		195	3.2.4.3 System Error Notification Control Function (SEG): Title corrected
		195	3.2.4.3 (1) List of SEG Function Control Registers, Note 1 deleted, description in NOTES corrected
		195	Table 3.74 List of Registers: Corrected
		197	Table 3.75 SEGCONT Register Contents: Note 1 added to b6 to 4
		200	3.2.4.3 (2) (c) SEGADDR — Error factor retention register (address): Title corrected
		203	Table 3.79 List of Registers: Corrected
		205	3.4.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction: Section number added to the title, description corrected, (4) corrected (Store instruction → Instruction), Note 1 added
		205	3.4.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction: Section number added to the title
		206	3.4.2 Access to Registers by Bit-Manipulation Instructions: Title corrected, description corrected
		206	3.4.4 Overwriting Context when Acknowledging Multiple Exceptions: Added
		206	3.4.5 Usage Notes on Prefetching: Added
		Section 4 Address Space	
		208	Table 4.1 Address Space (C1H): Reference manual in Note 3 corrected
		212	Table 4.2 Address Space (C1M): Reference manual in Note 1 corrected
		Section 6 Interrupt	
		All	Description of reserved bits unified: Bit name → Reserved, Function → When read, the value after reset is returned. When writing to these bits, write the value after reset. Term unification: When the detection in synchronization with an edge is selected → Edge detection, When the detection of the high level is selected → Level detection
		217	6.1 Overview, High-speed interrupts (EIINT0 to EIINT31), Low-speed interrupts (EIINT32 to EIINT255): Description corrected
		219	Table 6.1 Interrupt Control: Module name column added, table header corrected (Initial Value → Value after Reset)
		219	6.2.1 Configuration of Registers: Description corrected
		220	Table 6.2 External Interrupts, Software Interrupts, and NMI: Module name column added
		220	Table 6.3 Merging of Interrupts: Module name column added
		221	Table 6.4 EIC Register Contents: description of b15 corrected
		230	6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers: Description corrected
		235	6.3.2 IRQ Interrupts: Description corrected
		238 to 244	Table 6.11 Interrupt Exception Handlers and Orders of Priority: Note 1 and Note 2 added to the table header, description of direct branching corrected
		238	Table 6.11 Interrupt Exception Handlers and Orders of Priority (1/7): Corrected (entries in the level interrupt column for EIINT interrupt channel No.32, 34, and 37 deleted, OSTM interrupt names corrected, Note 3 added to “Flash sequencer processing completion interrupt” of code flash and data flash
		239	Table 6.11 Interrupt Exception Handlers and Orders of Priority (2/7): WDTA interrupt names corrected
		241, 242	Table 6.11 Interrupt Exception Handlers and Orders of Priority (4/7)(5/7): Corrected (TSG3 interrupt names corrected, ICUSB deleted)

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	244	Table 6.11 Interrupt Exception Handlers and Orders of Priority (7/7): CSIH and DTS interrupt names corrected
		244	Table 6.11 Interrupt Exception Handlers and Orders of Priority: Notes added, CAUTION deleted
		245	6.5.4 Merging of DTS Interrupts: Description corrected
		246	Figure 6.1 Example of NMI Processing Flow: Title corrected
		247	6.5.5.2 Flow of Processing for External Interrupts: Description corrected
		248	Figure 6.2 Example of External Interrupt Processing Flow: Title and entries corrected
		249	Figure 6.3 Example of Inter-Processor Interrupt Processing Flow: Title and entries corrected
		250	6.5.5.4 Software Interrupt Processing Flow: Description corrected
		250	Figure 6.4 Example of Software Interrupt Processing Flow: Title and entries corrected
		251	6.5.5.5 Flow of Processing for DTS Interrupts: Title corrected, description corrected
		252	Figure 6.5 Example of DTS Interrupt Processing Flow: Title and entries corrected
		Section 7 DMA Controller	
		All	Description of reserved bits added: Bit name → Reserved, Function → “When read, the value after reset is returned. When writing to these bits, write the value after reset.” or “When read, the value after reset is returned.” Notation unified: Bit chart and bit names unified to field notation (xx1, xx0 bits → xx[1:0] bits)
		254	7.1.1 Overview: Description corrected (external RAM → dedicated RAM)
		257	7.2.1.5 Transfer Completion Interrupt and Transfer Count Match Interrupt Outputs: Description corrected (“to external device” deleted)
		258	7.2.1.6 Continuous Transfer: Description corrected (DCEN.DTE → DCENn.DTE)
		259	Figure 7.2 Operation of Continuous Transfer by a DMAC: Corrected (DCEN.DTE → DCENn.DTE)
		264	Figure 7.5 Operation of Reload Function 1: Corrected (hex notation corrected)
		265	Figure 7.6 Operation of Reload Function 2: Corrected (hex notation corrected)
		265	Figure 7.7 Operation when Combining Reload Function 1 and Reload Function 2: Corrected (hex notation corrected)
		275	7.3.5 Masking and Clearing a Hardware DMA Transfer Request by the DTSFSL: Description corrected (“from outside” deleted)
		279	7.5.2.1 Identifying the Accessing Master: Corrected
		279	7.5.2.2 Master Access: Corrected
		279	7.5.2.3 Channel Assignment: Corrected
		280	Table 7.6 Master Information Output from DMA: Corrected (VM and VCID deleted)
		281	7.5.4.1 Restriction on the Next Channel in the Chain: Corrected (VM and VCID deleted)
		282, 283	Table 7.7 Channel Assignment: Title corrected, entries corrected (entry in the “Master that configures the setting” column corrected)
		283	7.6.2 Setting Up the Overall DMA Operation: Body corrected
		288	Table 7.8 List of DMA Trigger Sources (2/4): Entries in the Function/Module column for DMACTRG50 and 51 corrected, interrupt names for 56 to 63 and 74 to 79 corrected
		289	Table 7.8 List of DMA Trigger Sources (3/4): DMACTRG115 and 116, ICUSB deleted
		292	Table 7.9 List of DTS Trigger Sources (2/4): Entries in the Function/Module column for DTSTRG48 and 49 corrected
		293	Table 7.9 List of DTS Trigger Sources (3/4): Interrupt names for DTSTRG92 to 105 corrected, ICUSB for DTSTRG118 and 119 deleted
		295, 296	Table 7.10 List of Global Register Addresses: Module name column added, entries in the Meaning column for DTSPR0 to 7, DM00CM to DM17CMD, and DTSnnnCM corrected
		302	7.9.2.6 DTSER1 — DTS Error Register 1, R/W of b15 to b0 corrected
		305	7.9.2.8 DTSERC — DTS Error Clear Register: R/W of b30 and b14 corrected
		306	7.9.2.9 DM0CMV — DMAC0 Register Access Protection Violation Register: b16 changed to reserved
		306	Table 7.19 DM0CMV Register Contents: Description of b22 to 17 corrected
		307	7.9.2.10 DM1CMV — DMAC1 Register Access Protection Violation Register: b16 changed to reserved
		307	Table 7.20 DM1CMV Register Contents: Description of b22 to 17 corrected
		308	7.9.2.11 DTSCMV — DTS Register Access Protection Violation Register: b16 changed to reserved

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	308	Table 7.21 DTSCMV Register Contents: Description of b22 to 17 corrected
		315	Table 7.32 DTRERINT Register Contents: Description of b1 and b0 corrected (external notification enable → notification enable)
		319	7.9.2.19 DMnnCM — DMAC Channel Master Setting Register (nn = 00 to 07, 10 to17): b9 to 7 and b0 changed to reserved
		320	7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127): b25 to 23 and b16 changed to reserved, Value after reset corrected
		320	Table 7.37 DTS Channel Master Setting Register Contents: b16, description of reserved bit added
		322	Table 7.38 DMAC Channel Register Address: Module name column added
		327	Table 7.42 DTCTn Register Contents (2/3): Description of b10, 9 corrected
		339	Table 7.52 DTFRRQn Register Contents: Description of b0 corrected (“from the outside” deleted)
		343, 344	7.11.1.4 Caution about accessing the TI: Corrected (DTTCTnnn → DTTCCnnn)
		344	7.11.2 DTS Channel Register Address: Module name column added to the table
		345	7.11.3.1 DTSAnnn — DTS Source Address Register: Value after reset corrected
		346	7.11.3.2 DTDAnnn — DTS Destination Address Register: Value after reset corrected
		347	7.11.3.3 DTTCnnn — DTS Transfer Count Register: Value after reset corrected
		348	7.11.3.4 DTTCTnnn — DTS Transfer Control Register: Value after reset corrected
		350	Table 7.57 DTTCTnnn Register Contents (3/3): Description in CAUTIONS corrected
		351	7.11.3.5 DTRSAnnn — DTS Reload Source Address Register: Value after reset corrected
		352	7.11.3.6 DTRDAnnn — DTS Reload Destination Address Register: Value after reset corrected
		353	7.11.3.7 DTRTCnnn — DTS Reload Transfer Count Register: Value after reset corrected
		354	7.11.3.8 DTTCCnnn — DTS Transfer Count Compare Register: Value after reset corrected
		355	7.11.3.9 DTFSLnnn — DTSFSL Operation Setting Register: Value after reset corrected
		356	Table 7.63 DTFSTnnn Register Contents: Description of b0 corrected
		Section 8 Resets	
		361	Table 8.1 Registers: Module name column added
		364	8.4.4 SWRESA — Software Reset Request Register: Description below the table corrected
		365	Figure 8.2 Flow of Software Reset: Corrected (Note added)
		Section 9 Power Supply Circuit	
		368	Table 9.1 List of Registers: Module name column added
		372	Figure 9.4 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1H (BGA)): Title corrected, Note 1 corrected
		373	Figure 9.5 Example of Connections between Power-Supply Pins and External Capacitors (RH850/C1M (QFP)): Added
		Section 10 Clock Controller	
		379	10.2.3 PLL0CLKC1 — PLL0 Control Register 1: Description corrected
		379, 380	Table 10.6 PLL0CLKC1 Register Contents: Corrected, note changed
		381	Table 10.7 CKSC0CTL Register Contents: Description of b31 to 6 added
		383	Table 10.9 CLKD0DIV Register Contents: Description of b31 to 3 added
		385	Table 10.11 CKSC1CTL Register Contents: Description of b31 to 6 added, b5 to 0 corrected
		387, 388	10.2.10 PROT1PHCMD — Protect 1 Command Register: Description corrected
		389	Table 10.14 PROT1PS Register Contents: [Clearing Condition], description corrected
		Section 11 Clocked Serial Interface H (CSIH)	
		All	Term unification: Bit xxx → xxx or xxx bit, TBD deleted
		395	Table 11.6 Interrupt Requests: Interrupt names corrected
		397	11.2 Overview, Title corrected
		397	11.2.1 Functional Overview: • RCB (Recessive Configuration for Broadcasting) bit is included, added
		398	Table 11.10 Main Registers of CSIH: Table title added, CSIHnBRSy added
		403	Table 11.13 CSIHnCTL1 Register Contents: Functional description of b1 and b0 corrected (mode → function, disabled/enabled)
		405	11.3.4 CSIHnCTL2 — CSIHn Control Register 2: Description added
		408	Table 11.17 CSIHnSTR0 Register Contents: Description of b7 corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	410	Table 11.18 Behavior in Memory Mode: Description of b7 corrected
		422	Table 11.24 CSIHnCFGx Register Contents: Description of b7 to 4 and table header corrected (delay time → time)
		429	Table 11.30 Interrupt Generation: Corrected (entry in the “Job mode disabled” column for FIFO corrected, the “Transmit-only buffer, dual buffer” column split into separate columns)
		432	11.4.3.1 INTCSIHTIC in Direct Access Mode: Interrupt name in the description corrected (INTCSIHTIC → INTCSIHTIJC)
		434	Figure 11.6 Generation of INTCSIHTIC in job mode: Note 1 added
		439	11.4.7.1(2) Clock Default Setting: Description corrected
		440	11.4.7.2 Slave Mode: Description in NOTE corrected (CSIHnBRSy.CSIHnBRSy[11:0] → CSHInBRSy.CSIHnBRS[11:0])
		441, 442	11.4.8.2 One Master and Multiple Slaves: Description corrected
		444	Figure 11.13 Chip Select Timings: Corrected (CS2 added)
		445	Figure 11.14 Chip Select and RCB Example: Corrected (CSIHnIDx[2:0] → CSIHnID0[2:0])
		448	11.4.11 Transmission Clock Selection: Corrected
		451	Table 11.37 Dual Buffer Mode: Entries in the Range column corrected
		451	Table 11.38 Transmit-only Buffer Mode: Entries in the Range column corrected
		454	11.4.14.2 Data Length Greater than 16 Bits: Description of NOTES 1, 2, and 5 corrected
		456	Figure 11.25 Serial Data Direction Select Function - MSB First (CSIHnDIRx = 0): Corrected
		456	Figure 11.26 Serial Data Direction Select Function - LSB First (CSIHnDIRx = 1): Corrected
		457	11.4.16.1 Communication Timing Using SS Function: (3) and (4), description corrected
		459	11.4.17 Handshake Function: Description corrected
		462	11.4.17.2 Master Mode: Description corrected (“while CSIHnCTL1.CSIHnHSE = 1” added)
		462	11.4.17.2 Master Mode: Description of CAUTION 1 corrected
		466	11.4.18.3 Time-out Error: Description added
		472	11.4.19 Loop-back Mode: Description corrected
		472	Figure 11.44 Loop-back Operation: Corrected (Inactive level → High level)
		481	Figure 11.51 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Settings of CSIHnMCTL2.CSIHnSOP[6:0] corrected
		482	11.5.2.1 Transmission/Reception in Master Mode when Job Mode is Disabled Procedure: 2 to 4, description corrected
		483	Figure 11.52 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 1: Settings of CSIHnMCTL2.CSIHnSOP[6:0] corrected
		485	Figure 11.53 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Settings and timing corrected
		486	11.5.3.1 Transmission/Reception in Master Mode when Job Mode is Disabled: Procedure: 1, 2, 4, and 6, description corrected
		487	Figure 11.54 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 1: Corrected
		489	Figure 11.55 Slave in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: Settings and timing corrected
		492	11.5.4.1 Transmission/Reception in Master Mode when Job Mode is Disabled: Procedure: 5 and 6, description corrected
		493	Figure 11.57 Master in FIFO Mode, CSIHnCTL1.CSIHnJE = 1: Corrected (CSIHTSOE timing deleted)
		494	11.5.4.2 Transmission/Reception in Master Mode when Job Mode is Enabled: Title corrected, Procedure: 6, 8, and 9, description corrected
		495, 496	Table 11.43 Notes on Setting Registers: Description of bit names CSIHnTXE/CSIHnRXE and CSIHnMBS corrected, notation of register names CSIHnCFGx and CSIHnBRSy corrected)
		Section 12 Serial Communication Interface 3 (SCI3)	
		All	Notation of address of SCI3 registers corrected: <SCI3n_base> + $00xx_H$ → <SCI3n_base> + xx_H
		502	12.3.3 SCI3nTDR — Transmit Data Register: Description corrected (SCI3TDR → SCI3nTDR)
		503	12.3.4 SCI3nSMR — Serial Mode Register: b1, 0 corrected (CKS1, CKS0 → CKS[1:0])
		503	Table 12.11 SCI3nSMR Register Contents: Functional description of b1, b0 corrected (bit name corrected and n → m)
		505	12.3.5 SCI3nSCR — Serial Control Register: b1, 0 corrected (CKE1, CKE0 → CKE[1:0])

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	505	Table 12.12 SCI3nSCR Register Contents: Bit name of b1, b0 corrected, NOTE added
		513	Table 12.23 Relationship between SCI3nMDDR Setting and Bit Rate B when Bit Rate Modulation Function Is Used: Note corrected (SCI3nMDDR → MDDR)
		542	Table 12.26 SCI3 Interrupt Sources: Entries in the DST Activation column for RXI and TXI corrected (Not possible → Possible)
		Section 13 LIN Master Interface (RLIN2)	
		545	13.1.1 Units and Channels: Description corrected (RLIN21nGLWBR → RLN21nGLWBR, RLIN21nmLiMD → RLN21nmLiMD)
		553	13.3.2.4 RLN21nGLSTC — LIN Self-Test Control Register: Term corrected (release → unlock)
		556	13.3.3.3 RLN21nmLiSC — LIN Space Configuration Register: Description corrected (RLIN21nmLiSC → RLN21nmLiSC)
		561	13.3.3.7 RLN21nmLiCUC — LIN Control Register: Description corrected
		561	Table 13.21 RLN21nmLiCUC Register Contents: Description corrected
		570	13.3.3.13 RLN21nmLiIDB — LIN ID Buffer Register: Value after reset corrected
		571	13.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register: Value after reset corrected
		572	13.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register: Value after reset corrected
		570	13.3.3.13 RLN21nmLiIDB — LIN ID Buffer Register: Value after reset corrected, description corrected
		571	13.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register: Value after reset corrected, description corrected
		572	13.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register: Value after reset corrected
		583	Figure 13.8 Example of Data Reception Timing: Corrected
		586	13.12.1 Wake-up Transmission: Description corrected
		589	13.14.1 Types of Error Status: Description corrected
		589	Table 13.36 Types of Error Status: Note 1 corrected
		592	13.15.1 Change to LIN Self-Test Mode: Description corrected
		593	13.15.2 Transmission in LIN Self-Test Mode: • Set interrupt enable register and error enable related registers, RLN21nmLiEDE register corrected
		593	13.15.2 Transmission in LIN Self-Test Mode: • Release from LIN reset mode, description corrected; • Header transmission → response transmission started, description added
		594	13.15.3 Reception in LIN Self-Test Mode: • Set interrupt enable register and error enable related registers, • Release from LIN reset mode, description corrected; • Header transmission → response reception started, description added
		595	13.15.4 Terminating LIN Self-Test Mode: • Verify release from LIN self-test mode, description corrected.
		Section 14 CAN Interface (RS-CAN)	
		601	Table 14.9 RS-CAN Module Specifications (2/2): Description in specification of CAN clock source corrected
		602	Figure 14.1 RS-CAN Module Block Diagram: Corrected (location of clkc and clk_xincan corrected)
		605	Table 14.10 Registers (3/23): Notation of register name corrected (FIFO Msg lost status register → FIFO message lost status register)
		631	14.3.3 RSCAN0CmCTR — Channel Control Register (m = 0 to 3): Description of BOM[1:0] bits corrected (“at the time of return from the bus off state” deleted)
		669	14.3.25 RSCAN0RFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7): Access, description corrected
		676	14.3.31 RSCAN0CFCCk — Transmit/Receive FIFO Buffer Configuration and Control Register k (k = 0 to 11): Description of CFTML[3:0] bits corrected
		735	14.3.65 RSCAN0RPGACCr — RAM Test Page Access Register (r = 0 to 63): Address notation corrected (n → r)
		741	14.4.2.1(1) Global Stop Mode: Description corrected
		745	14.4.2.2 (5) Bus Off State: • When BOM[1:0] = 01_B: Description corrected
		759	14.4.5 Gateway Function: Description corrected (“transmit/receive FIFO buffer” → “transmit/receive FIFO buffer of a channel being used for transmission”)
		789	14.6 Notes: Corrected (CAN0TMTRSTS2 → RSCAN0TMTRSTS2)
		Section 15 Window Watchdog Timer A (WDTA)	
		794	15.3.2 WDTAnWDTE — WDTA Enable Register: Reference section corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	795	15.3.3 WDTAnMD — WDTA Mode Register: Description corrected
		Section 16 OS Timer (OSTM)	
		810	16.3.5 OSTMnTOE — OSTMn Output Enable Register: Title corrected (OSTMnTO → OSTMnTOE)
		818	16.4.2.2 Operation when OSTMnCMP = 0000 0000_H: (3), description corrected
		820	16.4.3.1 Basic Operation in Free-Run Compare Mode: OSTMnTINT period, title corrected
		822	16.4.3.2 Operation when OSTMnCMP = 0000 0000_H: (4), description corrected
		Section 17 Timer Array Unit D (TAUD)	
		824	17.1.6 External Input/Output Signals: CAUTION added
		827	Table 17.9 Functional List of TAUD Operations: Corrected (One-Phase PWM Output Function, Trigger Start PWM Output Function added)
		837	17.3.5 TAUDnCNTm — TAUDn Channel Counter Register: Value after reset corrected
		840	Table 17.16 TAUDnCMORm Register Contents (2/3): Functional description of 100_B and 101_B in the table of b10 to 8 corrected
		840	Table 17.16 TAUDnCMORm Register Contents (2/3): Description of 00_B in the table of b7, 6 corrected
		847	Table 17.26 TAUDnRDC Register Contents: Functional description corrected
		849	Table 17.29 TAUDnTOE Register Contents: Functional description corrected
		851	Table 17.33 TAUDnTOL Register Contents: Functional description added
		853	Table 17.37 TAUDnTRE Register Contents: Functional description corrected (TAUTnTOE → TAUDnTOE)
		860	Table 17.42 Channel Operation Functions and Methods They Use: Corrected (entries in A to C2, Trigger Start PWM Output Function added
		870, 871	17.4.3.4 (4) Simultaneous Rewrite when INTTAUDnIm is Generated on an Upper Channel Specified by TAUDnRDC.TAUDnRDCm that in Turn is Triggered by an External Signal (Method C2): Added
		878	Figure 17.11 Set/Reset Conditions for Synchronous Channel Output Mode 2 with Dead Time Output: Corrected (TAUDnTOL.TAUDnTOL → TAUDnTOL.TAUDnTOLm)
		883	Figure 17.17 INTTAUDnIm not Generated when Counter Starts: Title corrected (INTTAUDnIm → INTTAUDnIm)
		884	17.4.7 Interrupt Generation upon Overflow: Description deleted (• Both channels are triggered by the same TAUDTTINm input., • The trigger detection settings (TAUDnCMORm.TAUDnSTS[2:0] and TAUDnCMURm.TAUDnTIS[1:0]) must be identical for both channels. → deleted)
		896	17.4.9.2 (3) Block Diagram and General Timing Diagram: Corrected (TAUDnMD = 1 → TAUDnMD0 = 1)
		900	17.4.9.2 (6) Specific Timing Diagrams: Description corrected
		907	Figure 17.35 Forced Restart Operation (TAUDnCMORm.TAUDnMD0 = 1, TAUDnCMURm.TAUDnTIS[1:0] = 01_B): Title corrected (01 → 01_B)
		911	Table 17.63 Operating Procedure for External Event Count Function: TAUDn Status during Operation, description corrected
		931	17.4.9.8 (1) Overview, Summary: Corrected (TAUDTTINmT → TAUDTTINm)
		934	Table 17.82 Operating Procedure for TAUDTTINm Input Signal Width Measurement Function: Operation during Operation, description corrected
		938	17.4.9.9 (1) Overview: NOTE added
		949	Figure 17.64 General Timing Diagram of TAUDTTINm Input Pulse Interval Judgment Function: Corrected
		958	Figure 17.68 General Timing Diagram of Real-Time Output Function Type 1: Corrected (TAUDnTRO.TRO1/2 → TAUDnTRO.TAUDnTRO1/2)
		965	Figure 17.71 General Timing Diagram of Real-Time Output Function Type 2: Corrected (TAUDnCMOR1 → TAUDnCSR1)
		976	Table 17.113 Simultaneous Rewrite Settings for Simultaneous Rewrite Trigger Generation Function Type 1: Description of TAUDnRDM.TAUDnRDMm corrected
		976	Table 17.114 Simultaneous Rewrite Settings for Lower Channels in Simultaneous Rewrite Trigger Generation Function Type 1: Description of TAUDnRDM.TAUDnRDMm corrected
		978	17.4.12 Synchronous Channel Operation Functions: • Section 17.4.12.3, Trigger Start PWM Output Function, added

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	1002 to 1013	17.4.12.3 Trigger Start PWM Output Function: Added
		1028	17.4.12.4 (9) Specific Timing Diagrams: Description corrected (slave channel → slave channel 1)
		1037	Table 17.162 Operating Procedure for Offset Trigger Output Function: Operation during Operation, description corrected
		1051	17.4.12.7 (7) (a) Duty cycle = 0%: Description below figure 17.103 corrected
		1052	17.4.12.7 (7) (b) Duty cycle = 100%: Description below figure 17.104 corrected
		1053	17.4.12.8 (1) Overview: Prerequisites, description corrected (“Slave channel 1 can be used as a separate timer (independent function)” added)
		1053	17.4.12.8 (1) Overview: Prerequisites, NOTE corrected
		1054	17.4.12.8 (1) Overview: Functional description, description of Slave channel 2 corrected (“When the counter value of slave channel 2 reaches 0001_H, INTTAUDnIm is generated” added)
		1054	17.4.12.8 (1) Overview: Conditions, description corrected
		1055	17.4.12.8 (2) Equations: Corrected
		1056	17.4.12.8 (3) Block Diagram and General Timing Diagram: Description of Slave channel 3 corrected
		1057	Figure 17.106 General Timing Diagram of Triangle PWM Output Function with Dead Time: Corrected (entry for TAUDnCDRm of slave 3 corrected)
		1062	Table 17.181 Contents of TAUDnCMORm Register for Slave Channel 3 of Triangle PWM Output Function with Dead Time: Functional description of b10 to 8 corrected
		1073	Table 17.188 Simultaneous Rewrite Settings for Master channels of Interrupt Request Signals Culling Function: Description of TAUDnRDS.TAUDnRDSm corrected
		1075	Table 17.191 Simultaneous Rewrite Settings for Slave Channels of Interrupt Request Signals Culling Function: Description of TAUDnRDS.TAUDnRDSm corrected
		1076	17.4.12.10 (7) Specific Timing Diagrams: Description corrected (INTTAUDnIm → INTTAUDnIm)
		1077	Table 17.193 TAUDTTOUTm to which Dead Time is Added and State of TAUDTTINm: Table header corrected (TAUDnCMUR.TAUDnTIS → TAUDnCMURm.TAUDnTIS[1:0])
		1081	Table 17.196 Control Bit Settings in Synchronous Channel Output Mode 2 with One-Phase PWM Output: Corrected (description of TAUDnTDL.TAUDnTDLm corrected)
		1128	Figure 17.124 Specific Timing Diagram of Complementary Modulation Output Function: Corrected (TAUDnTDL.TAUDnTDL → TAUDnTDL.TAUDnTDLm)
		Section 18 Timer Array Unit J (TAUJ)	
		1136	18.2.2.1 Description of Blocks: Clock and count clock selection, description deleted
		1136	18.2.2.1 Description of Blocks: Controller, description added
		1143	18.3.5 TAUJnCNTm — TAUJn Channel Counter Register: Value after reset, description corrected
		1146	Table 18.15 TAUJnCMORm Register Contents (2/3): b7, 6, functional description corrected
		1149	Table 18.17 TAUJnCSRm Register Contents: b7 to 1 corrected
		1151	Table 18.22 TAUJnTOE Register Contents: b3 to 0, description corrected
		1153	Table 18.25 TAUJnTOC Register Contents: b3 to 0, description added
		1157	18.4.1 General Operatin Procedure: NOTE corrected
		1164	18.4.3.3 Simultaneous rewrite procedure: Description: (6), description corrected
		1166	Table 18.31 Channel Output Modes: CAUTION 3, corrected
		1170	Figure 18.10 INTTAUJnIm Not Generated when Counter Starts: Corrected (TAUJCNTm → TAUJnCNTm)
		1171, 1172	18.4.7 Interrupt Generation upon Overflow: Added
		1175, 1181, 1188, 1196, 1202, 1207, 1212	Figure 18.14 Block Diagram for Interval Timer Function, Figure 18.20 Block Diagram for TAUJTTINm Input Interval Timer Function, Figure 18.23 Block Diagram for TAUJTTINm Input Pulse Interval Measurement Function, Figure 18.29 Block diagram for TAUJTTINm input signal width measurement function, Figure 18.35 Block Diagram of TAUJTTINm Input Period Count Detection Function, Figure 18.38 Block Diagram for TAUJTTINm Input Period Count Detection Function, Figure 18.41 Block Diagram for PWM Output Function: Term unification (Upper → upper channel; Lower → lower channel)
		1198	Table 18.52 Operating procedure for TAUJTTINm input signal width measurement function: Operation and TAUJn Status during Operation, description corrected
		1199	18.4.9.2 (6) Specific timing diagrams: Description corrected
		1201	18.4.9.5 (1) Overview: NOTE added

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	1202	Figure 18.36 General Timing Diagram for TAUJTTINm Input Position Detection Function: Corrected ($FFFF_H$ → FFFF $FFFF_{H,}$ 0000_H → 0000 0000_H)
		1205	Figure 18.37 Operation Stop and Restart, TAUJnCMORm.TAUJnMD0 = 0, TAUJnCMURm.TAUJnTIS[1:0] = 00_B): Corrected ($FFFF_H$ → FFFF $FFFF_{H,}$ 0000_H → 0000 0000_H)
		Section 19 Motor Control Timer (TSG3)	
		1225	19.2.1 Functional Overview: • Reload mode, description corrected
		1270	Table 19.35 TSG3nCMP1E-TSG3nCMP12E Register Setting: Description in NOTE corrected
		1290	Table 19.40 Compare Register Functions in Each Mode (3/7): Table header corrected
		1320	19.4.3.9 TSG3nOPCI0, TSG3nOPCI1 Signal Simultaneous Trigger Detection Flag (TSG3nTDF): Title corrected
		1326	Figure 19.33 Interrupt Skipping Operation when TSG3nPIE = 1 and TSG3nVIE = 1 in TSG3nCTL4 (Peak and Trough Interrupt Generation in HT-PWM Mode): Corrected
		1345	Figure 19.52 Example of Basic Operation Timing of PWM Mode (1/2): Description of the condition above the table corrected
		1346	Figure 19.52 Example of Basic Operation Timing of PWM Mode (2/2) : Description of the condition above the table corrected
		1359	19.4.7.1 PWM Mode: (3) Controlling Dead Time in PWM Mode, description corrected
		1355	Table 19.66 Compare Match Timing in HT-PWM Mode: Description in the Timing column corrected
		1356	Table 19.67 Example of Setting Each Timer Output Condition in HT-PWM Mode: Setting condition of TSG3nO1,TSG3nO3,TSG3nO5 pins, description corrected
		1404	Figure 19.88 Example of Operation in 120-DC Mode (Normal Rotation: TSG3nSTR1.TSG3nTSF = 0 and TSG3nOPT0.TSG3nIDC = 1): Corrected
		1405	Figure 19.89 Example of Operation in 120-DC Mode (Reverse Rotation: TSG3nSTR1.TSG3nTSF = 1 and TSG3nOPT0.TSG3nIDC = 0): Corrected
		1413	19.4.7.6 120-DC Mode: (8) Output Switch Timing in 120-DC Mode (TSG3nS120DCO = 0), NOTE, unnecessary description deleted
		1414	19.4.7.6 120-DC Mode: (9) Compare Register Rewrite Timing in 120-DC Mode, Reference figure number corrected
		1419	19.4.7.6 120-DC Mode: (12) (g) Set Timing of TSG3nPEF Flag, expression unified
		Section 20 Timer Option Module (TAPA)	
		1458	20.4.1.2 An Example of System Configuration: Description corrected
		Section 21 Timer Pattern Buffer (TPBA)	
		1470	Table 21.5 Interrupt Requests: DTS Trigger Number column added
		1472	21.2.2 Block Diagram: Description corrected (TPBAnTSST → TPBAnSST)
		1497	Figure 21.5 Example of Basic Timing (2/2), Figure number corrected
		Section 22 Encoder Timer (ENCA)	
		All	Term unification: ENCAnCNT counter → ENCAnCNT register
		1501	Figure 22.1 Block Diagram of ENCA: Corrected (ENCAnUFD → ENCAnUDF)
		1503	Table 22.9 ENCAnCTL Register Contents (1/2): Description of b9, b8, and b7 corrected
		1507	Table 22.12 ENCAnFLG Register Contents: Functional description of b1 and b0 corrected
		1510	Table 22.15 ENCAnCCR1 Register Contents: Functional description of b15 to0 corrected
		1517	22.4.2.1 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 00_B: Description corrected
		1512	Table 22.17 ENCAnTE Register Contents: Functional description of b0 corrected
		1514	Table 22.19 ENCAnTT Register Contents: Functional description of b0 corrected
		1515, 1516	22.4.1 Timer Counter Operation: Items (3), (4), (5) corrected
		1518	22.4.2.2 When ENCAnUDS1 and ENCAnUDS0 Bits in ENCAnCTL = 01_B: Description corrected
		1524	22.4.5.1 Compare Function, Compare match interrupt mask function: Description corrected
		1525	Table 22.28 Compare Match Interrupt Mask Function: Description of Interrupt Mask Cancel Trigger corrected (ENCAnIDE → ENCAnLDE)
		1525	Table 22.29 Capture Function of ENCAnCTS: Description of Capture Trigger Signal when ENCAnCTS is 1 corrected (ENCAnEC pin →set with ENCAnSCE)
		1528	Figure 22.7 Example of Connection when Two ENCAn Units are Used: Corrected (ECM → ECM1, ENCAnCNT bits → ENCAnCNT registers)

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	1530	Figure 22.11 Setting Procedure for ENCAnCCR1 Register: Corrected (ENCAnEC pin input → ENCAnSCE setting, ENCAnT1 input → ENCAnI1 input)
		1539	Figure 22.20 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin): Corrected (xxx → xxx bit(s))
		1539	22.6.9 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin): Item (4) corrected
		1540	Figure 22.21 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin): Corrected (xxx → xxx bit(s), ENCA0CNT → ENCAnCNT)
		1544	Figure 22.25 ENCAnLDE Function (when ENCAnLDE = 0): Corrected (ENCAnCCR0 → ENCAnCNT)
		1545	Figure 22.26 ENCAnLDE Function (when ENCAnLDE = 1): Corrected (ENCAnCCR0 → ENCAnCNT)
		1546	Figure 22.27 Conflict between ENCAnLDE Function and Rewriting of ENCAnCCR0 Register: Corrected (ENCAnCCR0 counter → ENCAnCCR0 register)
		1547	Figure 22.28 Conflict between ENCAnLDE Function and Clear Operation by Encoder Clear Input: Corrected
		1549	Figure 22.29 Up-counting after Conflict between ENCAnLDE Function and Clearing of the Encoder Counter: Corrected
		1550	Figure 22.30 Capture Operation between Counter Clocks (ENCAnCCR1): Corrected
		1550	Figure 22.31 Capture Operation between Counter Clocks (ENCAnCCR0): Corrected
		1551	Figure 22.32 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 0: Corrected
		1552	Figure 22.33 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS = 1: Corrected
		1553	Figure 22.34 Encoder Operation when Compare Match Clear Control is Disabled: Corrected
		1554	22.6.23 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1: Title corrected
		1554	Figure 22.35 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1: Figure title corrected
		Section 23 Peripheral Interconnection (PIC)	
		All	Term unification: Trigger pulse → Trigger and pulse
		1565	Table 23.12 PIC1ASSER0 Register Contents: Bit name corrected
		1565	Table 23.13 PIC1ASSER1 Register Contents: Bit name corrected
		1682	23.2.3.14(5) Flow Chart: Note1 corrected
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1751	24.3.34 EMU2nRESRLD — EMU2n Resolver Pole Number Setting Register (n = 0, 1): Description added
		1751	24.3.35 EMU2nRESCNT — EMU2n Resolver Pole Count Register (n = 0, 1): Description added
		1754	Table 24.71 EMU2nVMTCAP Register Contents: b24 to 0, functional description corrected
		1760	24.3.45 EMU2nEARD — EMU2n Electrical Angle Response Delay Correction Variable Register (n = 0, 1): Description added
		1761	24.3.48 EMU2nSR2 — EMU2n dq-Axis Current Transformation Coefficient Register (n = 0, 1): Description added
		1792	24.3.80 EMU2nSR23 — EMU2n Three-phase Voltage Transformation Coefficient Register (n = 0, 1): Description added
		1793	24.3.81 EMU2nUVOFS — EMU2n U-phase Voltage Offset Register (n = 0, 1): Description added
		1793	24.3.82 EMU2nWVOFS — EMU2n W-phase Voltage Offset Register (n = 0, 1): Description added
		1813	24.3.112 EMU2nCMP0 — EMU2n Compare Register 0 (n = 0, 1): Description below the table corrected
		1813	24.3.113 EMU2nCMP1 — EMU2n Compare Register 1 (n = 0, 1): Description below the table corrected
		1818	24.3.122 EMU2nIRUCPPN1 — EMU2n Independent Rectangle IP U-phase Compare/ Pattern Setting Register 1 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRUCPPN1)

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	1819	24.3.123 EMU2nIRUCPPN2 — EMU2n Independent Rectangle IP U-phase Compare/ Pattern Setting Register 2 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRUCPPN2)
		1820	24.3.124 EMU2nIRVCPPN0 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 0 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRVCPPN0)
		1820	24.3.125 EMU2nIRVCPPN1 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 1 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRVCPPN1)
		1821	24.3.126 EMU2nIRVCPPN2 — EMU2n Independent Rectangle IP V-phase Compare/ Pattern Setting Register 2 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRVCPPN2)
		1821	24.3.127 EMU2nIRWCPPN0 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 0 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRWCPPN0)
		1822	24.3.128 EMU2nIRWCPPN1 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 1 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRWCPPN1)
		1822	24.3.129 EMU2nIRWCPPN2 — EMU2n Independent Rectangle IP W-phase Compare/ Pattern Setting Register 2 (n = 0, 1): Description added (EMU2nIRUCPPN0 → EMU2nIRWCPPN2)
		1844	24.4.5 (4) dq-axis Current Transformation: Reference sections corrected
		Section 25 R/D Converter (RDC2)	
		1879	25.3.2 RDC2nCONSEL — RDC2n Conversion Condition Select Register (n = 0, 1): Note 5 corrected ("01_B" → "10_B")
		1887	25.3.5 RDC2nANGDAT — RDC2n Angle Data Register (n = 0, 1): Value after reset in the bit chart corrected
		1888	25.3.7 RDC2nMNTC — RDC2n Monitor Pin Setting Register (n = 0, 1) : Value after reset in the bit chart corrected
		1888	25.3.8 RDC2nDATSTR — RDC2n Data Storage Register (n = 0, 1): Value after reset in the bit chart corrected
		1890	25.3.12 RDC2nINIT — RDC2n Initializing Register 2 (n = 0, 1) : Value after reset in the bit chart corrected
		1893	Table 25.32 BIST Settings and Results (BCON[3:0] and BRLT[3:0] Contents): Entries corrected (NG → NO)
		1895	25.3.17 RDC2nINGR — RDC2n Input Gain Resistance Value Register (n = 0, 1) : Value after reset in the bit chart corrected
		1897	25.3.19 RDC2nEXAAT — RDC2n Excitation Amplitude Automatic Adjustment Circuit Setting Register (n = 0, 1) : Value after reset in the bit chart corrected
		1901	25.3.21 RDC2nCMINT — RDC2n Compare Match Interrupt Register (n = 0, 1) : Value after reset in the bit chart corrected
		1909	25.4.1.2 Forced Gain Control Function: Description corrected
		1916	25.4.3.3 Resolver Signal Disconnect Error Detection Function: Reference section corrected
		1916	25.4.3.4 R/D Conversion Error Detection Function: Reference section corrected
		1917	Figure 25.6 Flowchart of Built-In Self-Test (BIST) Process: Reference section corrected
		1922	Figure 25.11 RDC2 Initial Operation Flow: Reference section corrected
		1924	25.6.1 Resolver Signal Input (Differential Input) Circuit: RIN: Resolver signal level, TBD changed
		Section 26 A/D Converter (ADCC)	
		1978	Table 26.48 List of Trigger Supports: Table header corrected
		1991	Figure 26.23 Flow of Initial Settings: Corrected
		1997	Table 26.49 Notes on Setting Registers: Entries in the Resisters column corrected
		Section 27 Functional Safety	
		1998	27.1 Overview: Description corrected
		2001	Table 27.3 Overview of the Code Flash ECC: Others, description corrected
		2003	Table 27.4 List of Registers: Module name column added
		2004	27.2.2.3 (1) CFAPCTL — Code Flash Address Parity Control Register: R/W of b15, b14 corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	2005	27.2.2.3 (2) CFECCCTL_VCI/PE1/PE2 — Code Flash ECC Control Register: R/W of b15, b14 corrected
		2010	27.2.2.3 (7) CF1STEADR0_VCI/PE1/PE2 — Code Flash 1st Error Address Register: Description corrected
		2011	27.2.2.3 (8) CFSTSTCTL_VCI/PE1/PE2 — Code Flash Sub-Test Control Register: R/W of b15, b14 corrected
		2014	Table 27.16 List of Registers: Module name column added
		2015	27.2.3.2 (3) DFECCCTL — Data Flash ECC Control Register: R/W of b15, b14 corrected
		2020	27.2.3.2 (10) DFTSTCTL - Data Flash Test Control Register: R/W of b15, b14 corrected
		2024	Table 27.28 List of Registers: Module name column added, Local RAM 1st error address registers0 to 3 and Local RAM 1st error address registers0 to 3 (PE2) deleted
		2025	27.2.4.3 (1) LRAPCTL_PE1/PE2 — Local RAM Address Parity Control Register: R/W of b15, b14 corrected
		2026	27.2.4.3 (2) LRTSTCTL_PE1/PE2 — Local RAM Test Control Register: R/W of b15, b14 corrected
		2028	27.2.4.3 (4) LRECCCTL_PE1/PE2 — Local RAM ECC Control Register: R/W of b15, b14 corrected
		2030	27.2.4.3 (6) LRSTCLR_PE1/PE2 — Local RAM Status Clear Register: Description corrected
		2030	Table 27.34 LRSTCLR Register Contents: Functional description of b3, b2, b1, and b0 corrected
		—	27.2.4.3 (9) LR1STEADRn_PE1/PE2 — Local RAM 1st Error Address Register n (n = 0 to 3): Deleted
		2036	Table 27.38 Address and Corresponding ECC Circuit: Title added
		2037	Table 27.39 List of Registers: Module name column added
		2038	27.2.5.3 (1) GRECCCTL_GRAMC — Global RAM ECC Control Register (GRAMC): Title corrected
		2038	Table 27.40 GRECCCTL_GRAMC Register Contents: Table title corrected
		2039	27.2.5.3 (2) GRTSTCTL — Global RAM Test Control Register: R/W of b15, b14 corrected
		2042	27.2.5.3 (4) GRECCCTL_VCI/PE1/PE2 — Global RAM ECC Control Register (VCI/PE1/PE2): Title corrected, R/W of b15, b14 corrected
		2042	Table 27.43 GRECCCTL_VCI/PE1/PE2 Register Contents: Table title corrected
		2049	27.2.5.3 (11) GR1STEADRn_VCI/PE1/PE2 — Global RAM 1st Error Address Register n (n = 0, 1): Description added
		2053	Table 27.51 Overview of the Instruction Cache ECC: Description of error notification corrected
		2054, 2055	Table 27.52 List of Registers: Module name column added
		2056	27.2.6.3 (1) IDECCCTL_PE1/PE2 — Instruction Cache Data RAM ECC Control Register: R/W of b15, b14 corrected
		2061	27.2.6.3 (6) ID1STEADRn_PE1/PE2 — Instruction Cache Data RAM (Bank n) 1st Error Address Register (n = 0, 1): Description corrected
		2061	Table 27.58 ID1STEADRn Register Contents: Description of b8 to 0 corrected
		2062	27.2.6.3 (7) ITECCCTL_PE1/PE2 — Instruction Cache Tag RAM ECC Control Register: R/W of b15, b14 corrected
		2063	Table 27.60 ITERRINT Register Contents: Description of b1, b0 corrected (detection and correction → detection)
		2067	27.2.6.3 (12) IT1STEADR0_PE1/PE2 — Instruction Cache Tag RAM 1st Error Address Register: Description corrected
		2067	27.2.6.4 Test Function: Description corrected
		2068	27.2.8.1 Overview: Error Notification, description corrected
		2070	Table 27.66 List of Registers: Module name column added
		2071	Table 27.68 E710CTL Register Contents: Functional description of bits 4 and 3 corrected
		2071	27.2.8.3 (1) E710CTL — ECC Control Register: R/W of b15, b14, b10, b9 corrected
		2071	Table 27.68 E710CTL Register Contents (1/2): Description of b7 corrected
		2073	27.2.8.3 (2) E710TMC — ECC Test Mode Control Register: R/W of b15, b14 corrected
		2079	27.2.8.4 Notification to ECM: Description corrected
		2081	Table 27.71 List of Registers: Module name column added
		2084	27.2.9.2 (3) APDPTMC_xx — P-Bus Data Parity Test Mode Control Register: R/W of b15, b14 corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	2086	Table 27.77 List of Registers: Module name column added
		2089	27.4.1 Overview: MPU, description corrected
		2090	Table 27.80 Identifiers for Slave Guard: Identifier VM, VCID, and TCID deleted
		2091	Table 27.81 List of Registers: Module name column added
		2092	27.4.2.2 (1) MGDGRPROTn — GRG Protection Setting Register n (n = 0 to 3): b26 and b16 to b9 changed to reserved, R/W corrected
		2092	Table 27.82 MGDGRPROTn Register Contents: b26 and b16 to b9 changed to reserved
		2097	27.4.2.2 (7) MGDGRSTYPE_VCI/PE1/PE2 — GRG Error Access Type Register: b12 to b10 and b7 changed to reserved
		2097	Table 27.89 MGDGRSTYPE Register Contents: b12 to b10 and b7 changed to reserved
		2099	Table 27.91 List of Registers: Module name column added
		2099	Table 27.92 List of Registers (for PBG Group): Module name column added
		2100	27.4.3.2 (1) FSGDxxDPROTn — PBGxx Protection Register n: b26 and b16 to b9 changed to reserved, R/W corrected
		2100	Table 27.94 FSGDxxDPROTn Register Contents (1/2): b26 and b16 to b9 changed to reserved
		2103	27.4.3.2 (5) ERRSLVxxTYPE — PBGxx Error Type Register: b12 to b10 and b7 changed to reserved
		2103	Table 27.98 ERRSLVxxTYPE Register Contents: b12 to b10 and b7 changed to reserved
		2111, 2112	Table 27.101 Registers of the Signature Generation Units: Module name column added
		2113	Table 27.102 Registers of the Signature Comparison Unit (MSD Sub-Block): Module name column added
		2130	Table 27.118 List of Registers: Module name column added
		2131	Table 27.120 Common Registers: Module name column added
		2131	Table 27.121 CLMAnCTL0 Register Contents: Functional description of b7 to 1 corrected
		2133	27.6.3.4 CLMAnPCMD — CLMAn Protection Command Register: Notation of bits corrected
		2133	Table 27.124 CLMAnPS Register Contents: Functional description of b7 to 1 corrected
		2133	27.6.3.5 CLMAnPS — CLMAn Protection Command Status Register: Operating conditions of the CLMAnPRERR bit, corrected (protection canceling sequence → protection unlock sequence)
		2134	Table 27.125 CLMATEST Register Contents: Functional description of b31 to 6 corrected
		2134	27.6.3.6 CLMATEST — CLMA Self-Test Register: (1) Calculation method of the thresholds CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0], (2) Definition of the input of initial values to the threshold registers, deleted
		2135	Table 27.126 CLMATESTS Register Contents: Functional description of b31 to 3 corrected
		2136 to 2138	27.6.4 Detection of Abnormal Clock Frequency: (1) Calculation method of the thresholds CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0] and (2) Definition of the input of initial values to the threshold registers, Description added
		2140	27.6.6.1 Writing to Protected Registers: Description corrected
		Section 28 Error Control Module (ECM)	
		All	Description of reserved bits unified
		2145	Figure 28.1 Outline of ECM: Added
		2145	Figure 28.2 Connection of ECM: Added
		2148	Table 28.9 Merging of Error Sources: Title corrected, entries corrected (description of error sources No.0, No.12, and No.13 corrected, error source No.41 corrected), term corrected (aggregated → merged)
		2151	28.3.2 ECMmESET (m = M/C) — ECM Master/Checker Error Set Trigger Register: Description corrected, R/W of b7 to 1 corrected
		2152	28.3.3 ECMmECLR (m = M/C) — ECM Master/Checker Error Clear Trigger Register: Description corrected, R/W of b7 to 1 corrected
		2153	Table 28.15 ECMmESSTR0 Register Contents: Expression of functional description of reserved bits corrected
		2155	28.3.6 ECMmPCMD0 (m = M/C) — ECM Master/Checker Protection Command Register: Description corrected
		2155	Table 28.17 ECMmPCMD0 Register Contents: Functional description of b31 to 8 corrected
		2156	28.3.7 ECMEPCFG — ECM Error Pulse Configuration Register to 28.3.28 28.3.28 ECMDTMCFG3 — ECM Delay Timer Configuration Register 3: Description of the sequence for writing data corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	2169	Table 28.30 ECMPS Register Contents: Functional description of b7 to 1 corrected
		2172	Table 28.33 ECMDTMCTL Register Contents: Functional description of b0 corrected
		2178	Table 28.38 ERROROUT Output Operation: Note 2 corrected
		2179	28.4.5 Write Protected Registers: Title corrected
		2179	28.4.5.1 Sequence of Writing to the Write-Protected Registers: Description corrected
		Section 29 Data CRC (DCRA)	
		2187	29.3.4 DCRAnCTL — CRC Control Register: CAUTIONS, description corrected (CRC input bit width)
		2188	Table 29.10 Setting Example of Initial Start Values (when read at a reset): Corrected (DCRA0COUT Read Value → DCRAnCOUT Read Value)
		Section 30 Intelligent Cryptographic Unit (ICUSB) → Section deleted	
		Section 30 On-Chip Debugging Unit (OCD)	
		2189	30.1 (1) Debug Interface: Corrected (4-pin → 4 pins)
		2189	30.1 (4) Software Break Function: Description corrected
		2191	30.2 (1) Trace RAM, (2) Software Trace: Corrected (4-pin → 4 pins)
		2194	Table 30.2 I/O Pins of AUDR: Corrected (AUDISR → AUDISR, AUDMBR and AUDMBRC)
		2196	30.4.3.1 AUDISR — AUDR Configuration Information Retention Register: • Retention of configuration information, description added
		2197	30.4.3.2 AUDMBR/AUDMBRC — AUDR Message Board Register: • Synchronous communications (message board) function, description added
		2199	30.4.4.2 (1) Single Transfer: Description corrected
		2201	30.4.4.2 (2) Continuous Transfer: Description corrected
		2207	30.4.4.3 Usage Notes on the AUDR Function: Title and entries corrected
		2207	30.5 Cautions on Using On-Chip Debugger: Title added, items (2) to (4) added
		Section 31 Flash Memory	
		2213	Table 31.2 Programming Methods: Description below the table added
		2214	Table 31.3 Basic Functions at a Glance: Entries in the Self-programming column for blank checking and verification and checksum corrected
		2215	Table 31.3 Basic Functions at a Glance: Description below the table added
		2215	Table 31.4 Summary of Security Functions: Description above the table corrected
		2218	31.7.1 Overview: Corrected
		2220	Table 31.8 List of Registers Related to Data Flash Memory: Module name column added
		2221	Table 31.10 Registers Related to Write and Erase Protection of Flash Memory: Module name column added
		2223	Table 31.13 List of Registers Related to Product Information: Module name column added
		2223	Table 31.14 Relationship between Product Name and PRDNAME Initial Value (Value after Reset): Corrected
		2224	31.9.3.1 PRDNAMEn[31:16] (n = 1 to 4) — Product Name Storage Register: Title, value after reset, and Note 1 corrected
		2224	Table 31.15 List of Registers Related to Product Information: Bit name and functional description corrected
		2225	31.10 Option Bytes: Corrected
		2225	31.10.1 OPBT0 — Option Byte 0 Register: b27 to 25 in the bit chart and Note 1 corrected
		2225, 2226	Table 31.16 OPBT0 Register Contents: Entries corrected
		2227	31.10.2 OPBT2 — Option Byte 2 Register: Note 1 corrected
		2227	Table 31.17 OPBT2 Register Contents: Entries corrected
		Section 34 Boundary Scan	
		2241	Table 34.2 Register Configuration: Note 2, C1x product name and value after reset corrected
		2243	34.4.1 TAP Controller: Corrected (Table 35.2 → Figure 34.2)
		2243	Figure 34.2 TAP Controller State Transition Diagram: Note 1 moved outside the figure
		Section 35 Electrical Characteristics	
		All	TBD corrected
		All	Term unification: Driveability → Driving ability
		2247	Table 35.1 Absolute Maximum Ratings: Notes corrected, storage temperature corrected

Rev.	Date	Description	
		Page	Summary
1.00	Mar 04, 2015	2248	Table 35.2 Relationship between Power Name and Pin: Entries in the Input Buffer Type column corrected
		2249	Table 35.3 Recommended Operating Conditions: Notes added, CAUTION deleted
		2250	Table 35.4 DC Characteristics (Input Voltage): Corrected
		2250	Table 35.5 DC Characteristics (Input Leak Current): Corrected
		2251	Table 35.6 DC Characteristics (Pull-Up/pull-Down MOS Current): Corrected, Note 1 added
		2252	Table 35.7 DC Characteristics (Output Voltage): Corrected
		2252	Table 35.8 DC Characteristics (Allowable Output Current): Corrected, description below the table corrected
		2253	Table 35.9 DC Characteristics (Injection Current): Corrected, description below the table corrected
		2253	Table 35.10 DC Characteristics (Input Capacitance): Corrected
		2254	Table 35.11 DC Characteristics (Supply Current: C1H): Title, entries, Note 1, and CAUTION 2 corrected
		2255	Table 35.12 DC Characteristics (Supply Current: C1M): Added
		2256	35.3 AC Characteristics: Body corrected
		2256	Figure 35.1 AC Measurement Conditions: Title corrected
		2257	Table 35.13 Power On/Off Timings: Entries changed, Notes changed, CAUTIONS added
		2258	Table 35.14 Power On/Off Timings: Entries changed, Notes changed, CAUTIONS added
		2259	Table 35.15 SSCG Timing: Entries corrected, Note 1 added
		2259	Table 35.16 Oscillation Frequency Accuracy of the On-Chip Oscillator: Title changed, entries corrected, CAUTION deleted
		2259	35.3.3 Output Slew Rate: Corrected
		—	35.3.4.1 Reset to 35.3.4.3 Mode: Deleted
		2261	Table 35.20 Control Signals: Added
		2261	Figure 35.5 Reset Timing: Added
		2261	Figure 35.6 Control Signal Timing: Added
		2262	Table 35.21 CSIH Timing in Master Mode: Entries corrected, Note 1 added, Note corrected
		2264	Figure 35.7 CSIH Timing (Master Mode) (2/4): tWRYI deleted
		2266	Figure 35.7 CSIH Timing (Master Mode) (4/4): Corrected
		2267	Table 35.22 CSIH Timing in Slave Mode: Entries corrected, Note added
		2268	Figure 35.8 CSIH Timing (Slave Mode) (1/3): Condition corrected
		2269	Figure 35.8 CSIH Timing (Slave Mode) (2/3): Condition corrected
		2270	Figure 35.8 CSIH Timing (Slave Mode) (3/3): Condition corrected
		2271	Table 35.23 SCI3 Timing (Master Mode): Entries corrected, Note added
		2272	Table 35.24 SCI3 Timing (Slave Mode): Entries and notes corrected
		2273	Table 35.25 RS-CAN Timing: Condition added
		2273	35.3.7 RS-CAN Timing: Definition of internal delay time of RS-CAN added
		2274	Table 35.27 Motor Control Signals Timing: Entries corrected, notes added
		2274	Table 35.28 Timer Timing: Entries corrected, notes added
		2275	Table 35.29 JTAG/NEXUS Timing: Corrected
		2276	Table 35.30 LPD (4-pin) Timing: Condition and entries corrected
		2277	Table 35.31 AUD RAM Monitor Timing: Condition and entries corrected
		2278	Table 35.32 A/D Converter Characteristics: Entries corrected, Note 2 added
		2279, 2280	Table 35.33 RDC Conversion Performance: Corrected, Note 1, Note 3, and Note 5 corrected
		2280, 2281	Table 35.34 RDC Analog Pin Characteristics: Entries corrected, Note 4 and Note11 corrected
		2282	Table 35.35 Error Detect Characteristics: Entries corrected, Note 4 corrected
		2283	Table 35.36 Code Flash Basic Characteristics: Entries corrected, Note 2 corrected
		2283	Table 35.37 Code Flash Programming Characteristics: Corrected
		2284	Table 35.38 Data Flash Basic Characteristics: Entries corrected, Note 2corrected
		2284	Table 35.39 Data Flash Programming Characteristics: Entries corrected
		2285	35.8 Thermal Characteristics: Added

Rev.	Date	Description	
		Page	Summary
1.10	Jan 20, 2016	All	Pin name corrected: ERROROUT_M → $\overline{\text{ERROROUT\_M}}$, ERROROUT_C → $\overline{\text{ERROROUT\_C}}$
		Section 1 Overview	
		50	Table 1.1 Overview of Product Enhanced motor control unit (EMU2) in Motor control: (2 channels) added
		54	Figure 1.3 Pin Connections of C1H: 1A, 1Y, 20A, and 20Y pins labelled “power-supply name (N.C.)”, CAUTION added
		Section 2 Pins	
		129	2.4.2 List of Pin Functions: CAUTION added
		131	Table 2.59 C1M Pin Function: PLLVCC and PLLVSS rows deleted
		Section 3 CPU System	
		183	Table 3.61 List of Registers: Address of the PEG corrected
		143	(b) EIPSW — Status save register when acknowledging EI level exception: CAUTION added
		143	Table 3.5 EIPSW Register Contents: Functional description on bit 11 to bit 9 (Debug) corrected
		145	(d) FEPSW — Status save register when acknowledging FE level exception: CAUTION added
		145	Table 3.7 FEPSW Register Contents: Functional description on bit 11 to bit 9 (Debug) corrected
		147	Table 3.9 PSW Register Contents: Functional description 0 of bit 6 (EP) corrected
		153	(p) MEI — Memory error information register: Register description corrected
		168	(h) MCR — Memory protection setting check result register: CAUTIONS added
		200	(c) SEGADDR — Error factor retention register (address): Register description corrected
		Section 4 Address Space	
		209	4.1.2.1 Space in which Instructions can be Fetched: (Local RAM self area → Local RAM area)
		213	4.2.2.1 Space in which Instructions can be Fetched: (Local RAM self area → Local RAM area)
		Section 5 Operating Mode	
		216	Table 5.1 Selection of Operating Mode: “Types of I/F” column added, “Remarks” corrected, Note 1 added
		Section 6 Interrupt	
		224	Table 6.5 EIBD Register Contents: Note 1 corrected
		242	Table 6.11 Interrupt Exception Handlers and Orders of Priority: Reserved EIINT interrupt channels 170 and 171, corrected (values in “Reference to a Table” and “Interrupt Priority Order (Value after Reset)” deleted)
		Section 8 Resets	
		364, 365	8.4.4 SWRESA — Software Reset Request Register: Description on the sequence of writing to the SWRESA register, corrected
		Section 9 Power Supply Circuit	
		367	Description of the power supply (Table): Note 1 added
		Section 10 Clock Controller	
		376	Table 10.2 Clocks and Functional Modules: Functional module name of the low speed peripheral clock (peripheral clock), corrected (FSC → FLSCI3)
		377	Table 10.3 Pins Related to the Clock Controller: Note 1 added
		380, 381	Table 10.6 PLL0CLKC1 Register Contents: Functional description on bit 7 to bit 3 (SELMFREQ[4:0]), and bit 2 to bit 0 (SELMPERCENT[2:0]), corrected
		Section 11 Clocked Serial Interface H (CSIH)	
		474	Figure 11.45 Example of CPU-Controlled High-Priority Communications, when CSIHnCTL1.CSIHnSLIT = 1: Title corrected
		Section 12 Serial Communication Interface 3 (SCI3)	
		522	Figure 12.8 Example Flowchart for Stopping the SCI3 after Serial Transmission: Functional block diagram, corrected (Clear DR to 0 and set DDR to 1 → Set the output on the TxDn pin to the low level by setting the corresponding bit in the port register). Description of [6] corrected
		534	Figure 12.19 Example of SCI3 Initialization Flowchart: Description of [5] corrected
		Section 13 LIN Master Interface (RLIN2)	
		572	13.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register: Register description of the LIN self-test mode, corrected
		Section 15 Window Watchdog Timer A (WDTA)	
		796	Table 15.11 WDTAnMD Register Contents: Functional description 0 of the WDTAnWIE bit, corrected

Rev.	Date	Description	
		Page	Summary
1.10	Jan 20, 2016	798	15.5.1.3 WDTA settings after reset release: Table corrected (“Setting after WDTA1 is Reset” of 75% interrupt mode, corrected (75% interrupt disabled → 75% interrupt enabled))
		Section 17 Timer Array Unit D (TAUD)	
		828	Table 17.9 Functional List of TAUD Operations: Corrected (Trigger Start PWM Output Function (Section 17.4.12.3) → Delay Pulse Output Function (Section 17.4.12.4))
		883	17.4.5.2 Event Count Mode: Description corrected (increments →decrements)
		1018	Figure 17.92 General Timing Diagram of Delay Pulse Output Function: Position of arrows “c” and “d” corrected
		Section 19 Motor Control Timer (TSG3)	
		1228	Figure 19.1 TSG3n Block Diagram: TSG3nCKS register deleted
		1246	Table 19.17 TSG3nIOC0 Register Contents: Functional description on TSG3nTOE6 to TSG3nTOE1 bit corrected
		1248	Table 19.19 TSG3nIOC2 Register Contents: NOTE corrected
		1276	19.3.45 TSG3nDTC1W — TSG3n Dead Time Setting Register 1: Description corrected (bit 30 to bit 16 → bit 14 to bit 0)
		1357	Table 19.67 Example of Setting Each Timer Output Condition in HT-PWM Mode: “Output an inactive level throughout one period (0% duty)” of “Output Condition” in TSG3nO2, TSG3nO4, TSG3nO6: Setting condition corrected
		Section 21 Timer Pattern Buffer (TPBA)	
		1498	Figure 21.5 Example of Basic Timing: Corrected
		Section 23 Peripheral Interconnection (PIC)	
		1564	Table 23.10 Registers Used by Each Function: Registers that are used by the simultaneous start trigger function, two-phase encoder control function (control method 3), three-phase pulse input control function, ENCA input select function, TAUD input select function, switch function between TSG output and low/high level output, and Hi-Z control function, corrected
		1572	Table 23.23 PIC1ATAUD0SEL Register Contents: Title corrected (register name)
		1595	Table 23.39 PIC1AREG51 Register Contents: Functional description on PIC1AREG5107 bit corrected (TS0OPCI1 signal → TS1OPCI1 signal)
		1687	23.3.2.2 PIC2BADCCnTSELx — A/D Converter n Trigger Select Control Register x: Description corrected (channel group → scan group)
		1687, 1688	Table 23.41 PIC2BADCCnTSELx Register Contents: Functional description corrected (channel group → scan group)
		1689	Table 23.42 PIC2BADCCnEDGSEL Register Contents: Functional description corrected (channel group → scan group)
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1696	Table 24.1 Units: Number of units corrected (2 → 1 (2 channels))
		1696	Table 24.2 Index: Meaning of n corrected (individual EMU2 units → individual EMU2 channels)
		1699	Figure 24.1 Overall Configuration of EMU2: Corrected (EMU2 → EMU2n, ADCC → ADCCn, RDC2→RDC2n, TSG3 → TSG3n)
		1845	24.4.5 Input IP, (4) dq-axis Current Transformation: Description corrected
		Section 25 R/D Converter (RDC2)	
		All	Erroneous bit name corrected (INTFj → INTFLGj, INTCj → INTCLRj)
		1902	Table 25.41 RDC2nCMINT Register Contents: Erroneous bit name corrected (INTF2 → INTFLG2, INTF1 → INTFLG1, INTF0→ INTFLG0)
		Section 26 A/D Converter (ADCC)	
		1995	26.5.4 Procedure for Setting Pin-Level Self-Diagnosis: Description corrected
		1998	26.7 Notes: 6. added
		Section 27 Functional Safety	
		2050	Table 27.49 GR1STERSTR Register Contents: Term used in the functional description for bit 5+8×n (EXSEDFn), corrected (GRAM→ global RAM)
		2064	(7) ITECCCTL_PE1/PE2 — Instruction Cache Tag RAM ECC Control Register: R/W attribute of bit 1 in the bit chart, corrected
		2070	27.2.8 ECC for Peripheral RAM (32 Bits), 27.2.8.1 Overview: Description of “Address Capture” corrected
		2074	Table 27.68 E710CTL Register Contents: Functional description on [Clearing conditions] for bit 2 (ECER2F), bit 1 (ECER1F), and bit 0 (ECEMF), corrected
		2075	Table 27.69 E710TMC Register Contents: Functional description on bit 4 (ECTRRS) corrected

Rev.	Date	Description	
		Page	Summary
1.10	Jan 20, 2016	2083	Table 27.70 Transfer Paths to which Data Parity is Applied: Access destination of the access Source (CPU1, CPU2, DMAC, DTS), corrected (DMA → DMAC)
		2097	(4) MGDGRSCTL_VCI/PE1/PE2 — GRG Control Register: Index n deleted
		Section 28 Error Control Module (ECM)	
		All	Terms corrected (Non-maskable interrupt→ FE level interrupt)
		2182	28.5 Usage Notes: Added
		Section 30 On-Chip Debugging Unit (OCD)	
		2191	30.1 Debug Function, (1) Debug Interface: Description corrected (on-chip RAM data, peripheral registers→ the on-chip RAM, data peripheral registers)
		Section 31 Flash Memory	
		2213	Figure 31.3 Data Flash Memory Map (64 bytes × 512 configuration): Corrected (FF20 7E00$_H$ → FF20 7FC0$_H$)
		2214	Table 31.1 Differences between Modes: Note 1 added
		2220	Figure 31.6 Concept of Self-Programming: Terms corrected (FACI Command)
		2227	Table 31.16 Setting Area of Option Bytes: Added
		Section 35 Electrical Characteristics	
		2258	Table 35.12 DC Characteristics (Supply Current: C1M): Conditions corrected (SYSVCC = PLLVCC = VCC = 3.0 to 3.6 V → SYSVCC = VCC = 3.0 to 3.6 V)
		2260	Figure 35.2 Power On/Off Timings: Position of arrow for tPLL1L0, corrected
		2261	Figure 35.3 Power On/Off Timings when EPT is Used: Position of arrow for tPLL1L0, corrected
		2264	Table 35.20 Control Signals: Note 1 corrected
		2265	Table 35.21 CSIH Timing in Master Mode: CSIHnSO output hold time (vs. CSIHnSC) added
		2266	Figure 35.7 CSIH Timing (Master Mode) (1/4): tHSOM added
		2270	Table 35.22 CSIH Timing in Slave Mode: CSIHnSO output hold time (vs. CSIHnSC), and slave output release time added
		2271	Figure 35.8 CSIH Timing (Slave Mode) (1/3): tHSOS added
		2273	Figure 35.8 CSIH Timing (Slave Mode) (3/3): tREL added
		2275	Table 35.24 SCI3 Timing (Slave Mode): Transmit data delay time (tTXD) corrected, Note 1 corrected, Note 2 deleted
		2275	Figure 35.11 SCI Input/Output Timing in Clock Synchronous Mode (as a Slave): Added
		2278	Table 35.29 JTAG/NEXUS Timing: Max. value of tTDOD and tRDYD corrected (tTCKcyc – 20 → tTCKW – 20)
		2282	Table 35.33 RDC Conversion Performance: Min. value of 12-bit resolution for the maximum angular velocity, corrected (240000(12000) → 240000(120000))
		2285	Table 35.35 Error Detect Characteristics: Error in the set register for the breaking of resolver signal, corrected (RD2nCON.BDVTH = 1b → RDC2nCON.BDVTH = 1b)
1.20	Jan 20, 2016	Section 1 Overview	
		52	Figure 1.1 Internal Block Diagram of RH850/C1H: Note added
		53	Figure 1.2 Internal Block Diagram of RH850/C1M: Note added
		Section 3 CPU System	
		137	P-Bus: Reference added to the description
		138, 139	3.1.2 Periphral Group Configuration: Added
		161	Table 3.32 FPIPR Register Contents: Functional description on bits 4 to 0 (FPIPR), corrected
		163	Table 3.34 PMR Register Contents: Functional description (Priority of bit 15) on bits 15 to 0 (PM15-0), corrected
		206	3.3.3.2 Operations of the LDL.W and STC.W Instructions: Description corrected, Note 1 added
		207	3.4.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction: Reference added to the description of “Example 2”
		Section 6 Interrupt	
		221	6.2.1 Configuration of Registers: Description corrected (Peripheral IP group 0 → Peripheral group 0)
		239	6.4 Description of Operations in Terms of Interrupt Exception Handlers and Order of Priority: CAUTION added

Rev.	Date	Description	
		Page	Summary
1.20	Jan 20, 2016	Section 7 DMA Controller	
		263	Figure 7.4 DMAC Channel Arbitration: Corrected (DMA transfer request and DMA transfer request acceptance for channel 1)
		264	7.2.2.3 Interface Arbitration: Description added
		274	7.3.1 DMA Suspension and Resume by Software Control: Bit name corrected, CAUTION added
		274	7.3.2 Suspension, Resume, and Transfer Abort of a DMAC Channel: Description corrected
		275, 276	7.3.3 Suspension, Resume, and Transfer Abort of a DTS: Description corrected, Note 1 added
		277	Table 7.5 List of Suspend, Resume, and Transfer Abort Functions: “DMA suspension and resume by software control” and “Suspension and resume of a DTS”, corrected (Bit name), Note 2 added
		298	Table 7.11 DMACTL Register Contents: CAUTION added
		299	Table 7.12 DTSCTL1 Register Contents: CAUTION added
		322	Recommended setup procedure of the DTS channel master setting register: Description (bit position) corrected
		327	Table 7.42 DTCTn Register Contents: Functional description on bit 27 (ESE), corrected (Bit name)
		Section 11 Clocked Serial Interface H (CSIH)	
		423	Table 11.24 CSIHnCFGx Register Contents: Functional description on bits 14 to 12, corrected (“Idle time with CSIHnCTL1.CSIHnSIT = 1” deleted)
		498, 499	Table 11.43 Notes on Setting Registers: Description of register for CSIHnRX0W and CSIHnRX0H, corrected
		Section 12 Serial Communication Interface 3 (SCI3)	
		524	Figure 12.8 Example Flowchart for Stopping the SCI3 after Serial Transmission: Note corrected
		527	Figure 12.10 Example of Serial Reception Flowchart (1): Description of [1] corrected
		532	Figure 12.15 Example of Multi-Processor Serial Reception Flowchart (1): Description of [1] corrected
		533	Figure 12.16 Example of Multi-Processor Serial Reception Flowchart (2): Note corrected
		539	Figure 12.22 Example Flowchart for Stopping the SCI3 after Serial Transmission: Note corrected
		541	Figure 12.24 Example of Serial Reception Flowchart: Note corrected
		543	Figure 12.25 Example of Simultaneous Serial Transmission and Reception Flowchart: Note corrected
		547	12.5.6 External Clock Input in Clock Synchronous Mode: Description corrected
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1714	Table 24.27 List of EMU2 Register Functions (Angle Generation IP): “Sign/Data Size” for “EMU2n electrical angle generation coefficient register”, corrected
		1726	24.3.7 EMU2nADTRG — EMU2n A/D Conversion Start Trigger Select Register (n = 0, 1): Description of register corrected (Register name)
		Section 26 A/D Converter (ADCC)	
		1949	Table 26.11 ADCCnVCRj Register Contents: Functional description on bits 5 to 0 (GCTRL[5:0]), corrected
		1992	26.4.7.2 A/D conversion circuit self-diagnosis function: Description corrected
		2001	26.7.1 Notes for the Register Setting: Subsection title added
		2002, 2003	26.7.2 Notes during Injection Current Application: Added
		Section 27 Functional Safety	
		2007	Table 27.3 Overview of the Code Flash ECC: Description of “Self-diagnosis” corrected
		2028	Table 27.25 Overview of the Local RAM ECC of CPU1 and CPU2: “Address capture” deleted
		2032	Table 27.30 LRTSTCTL Register Contents: Functional description on bits 5 to 2 (APTEST3 to APTEST0), corrected (WORD3 to WORD0 → (Bank3 to Bank0)
		2038	Table 27.36 LR1STERSTR Register Contents: Functional description on bits 1+8n and 0+8n, corrected (Error flag for 1 function)
		2045, 2046	Table 27.41 GRTSTCTL Register Contents: Functional description on bits 7 to 4 (APTEST3 to APTEST0), bit 2 (DECINEN) and bits 1, 0 (DATSEL), corrected
		2047	Table 27.42 GRTDATBFn Register Contents: Note added

Rev.	Date	Description	
		Page	Summary
1.20	Jan 20, 2016	2066	(5) ID1STERSTR_PE1/PE2 — Instruction Cache Data RAM 1st Error Status Register: Description of register corrected
		2072	(11) IT1STERSTR_PE1/PE2 — Instruction Cache Tag RAM 1st Error Status Register: Description of register corrected
		2073	(12) IT1STEADR0_PE1/PE2 — Instruction Cache Tag RAM 1st Error Address Register: Description of register corrected
		2092	27.3 Lockstep: Description corrected (“Interconnect” added)
		2106	Table 27.94 FSGDxxDPROTn Register Contents: Functional description on bit 4 (PROTDEB), corrected
		2109	Table 27.98 ERRSLVxxTYPE Register Contents: Functional description on bits 15 to 13 (PEID[2:0]), 9, 8 (SPID[1:0]), 6 (UM), 4 to 1 (STRB[3:0]), 0 (WRITE), corrected
		2116	27.5.3.3 Data Counter: CAUTION corrected
		Section 28 Error Control Module (ECM)	
		2184	28.4.1 Operations for ERROROUT Output: Description corrected
		Section 31 Flash Memory	
		2235	31.11 Notes: Description of (4) and (5), corrected
		Section 35 Electrical Characteristics	
		2261	Table 35.11 DC Characteristics (Supply Current: C1H): “Analog power supply current (A0VCC, A1VCC power supply)” corrected
		2262	Table 35.12 DC Characteristics (Supply Current: C1M): “Analog power supply current (A0VCC, A1VCC power supply)” corrected
		2264	Table 35.13 Power On/Off Timings: “PLL1 lock-in time” corrected
		2265	Table 35.14 Power On/Off Timings: “PLL1 lock-in time” corrected
1.30	Jul 29, 2016	Section 2 Pins	
		77	(4) PMn — Port Mode Register: Bit names of bits 15 and 0 in the bit chart, corrected
		77	Table 2.10 PMn Register Contents: Bit name, corrected (PMn[15:0] → PMn_[15:0])
		124	2.3.2.4 Setting Procedures of Peripheral Function DNF: Description of (4), corrected
		134	Table 2.62 Example Handling of Unused Pins: Description of Example handling of unused pins for AUDCK, $\overline{\text{AUDSYNC}}$ pins, corrected
		134	Table 2.62 Example Handling of Unused Pins: Description of Internal pull-up/pull-down resistor for AUDCK, $\overline{\text{AUDSYNC}}$ pins and AUDATAn pin, corrected
		135	Table 2.62 Example Handling of Unused Pins: Description of Example handling of unused pins for DCUTDI pin, DCUTCK pin, and DCUTMS pin, corrected
		135	Table 2.62 Example Handling of Unused Pins: Description of Internal pull-up/pull-down resistor for DCUTDI pin, DCUTCK pin, and DCUTMS pin, corrected
		Section 3 CPU System	
		148	(e) PSW — Program status word: Description corrected
		187	(b) PEGGnMK — PE guard area n mask setting register: Description of Example, corrected
		196	Table 3.74 IPGPMTUM4 Register Contents: Functional description on bits 1 and 0 (W0 and R0), corrected
		198	Table 3.76 SEGCONT Register Contents: Functional description on bit 8 (VCRE), corrected (*2 added)
		199	Table 3.76 SEGCONT Register Contents: Functional description on bit 4 (VCIE), corrected. Note 2 added.
		202	(c) SEGADDR — Error factor retention register (address): Description corrected
		207	3.4.1 Synchronization of Store Instruction Completion and Subsequent Instruction Generation: Description corrected
		207, 208	3.4.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction: Description corrected
		208	3.4.1.3 When switching the code flash memory area: Added
		Section 6 Interrupt	
		235, 236	6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers: Bit names for registers PINTCLR4 to PINTCLR7, corrected
		255	Table 6.13 Interrupt Response Times, corrected

Rev.	Date	Description	
		Page	Summary
1.30	Jul 29, 2016	Section 7 DMA Controller	
		258	7.2.1.2 Executing a DMA Cycle: Description corrected
		261	7.2.1.6 Continuous Transfer: Description corrected
		274	7.3.2 Suspension, Resume, and Transfer Abort of a DMAC Channel: Description corrected
		277	Table 7.5 List of Suspend, Resume, and Transfer Abort Functions: Description of How to execute the function for Suspension and resume of a DMAC channel function, corrected (*3 added). Note 3 added.
		278	7.4.2 DMA Transfer Error: Description added
		282	7.5.3 Master Information Inherit Function: Description corrected
		320	Table 7.35 DTRTRDAT Register Contents: Functional description on bits 6 to 0 (TRDAT[6:0]), corrected
		330	7.10.2.4 DTCTn — DMAC Transfer Control Register: CAUTIONS corrected
		343	Figure 7.12 Structure of the TI: TI-H corrected
		345	7.11.1.4 Caution about accessing the TI: Description corrected
		Section 8 Resets	
		362	8.3 Reset Sources: Source and Description, corrected
		367	8.4.4 SWRESA — Software Reset Request Register: Note 1 corrected
		Section 9 Power Supply Circuit	
		373	9.3.3 PROT0PHCMD — Protect 0 Command Register: Description added
		Section 10 Clock Controller	
		378	Table 10.2 Clocks and Functional Modules: Functional Module Name for High speed peripheral clock and Low speed peripheral clock (peripheral clock), corrected
		391	10.2.10 PROT1PHCMD Protect 1 Command Register: Note 1 corrected
		Section 11 Clocked Serial Interface H (CSIH)	
		448	Figure 11.14 Chip Select and RCB Example, corrected
		484	Figure 11.51 Master in Transmit-only Buffer Mode, CSIHnCTL1.CSIHnJE = 0: CSIHnMCTL2.CSIHnSOP[6:0]) corrected
		488	Figure 11.53 Master in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: CSIHnMCTL2.CSIHnSOP[6:0]) and CSIHnMRWP0.CSIHnRRA[6:0], corrected
		489	11.5.3.1 Transmission/Reception in Master Mode when Job Mode is Disabled: Description corrected
		492	Figure 11.55 Slave in Dual Buffer Mode, CSIHnCTL1.CSIHnJE = 0: CSIHnMCTL2.CSIHnSOP[6:0]) and CSIHnMRWP0.CSIHnRRA[6:0], corrected
		493	11.5.3.3 Transmission/Reception in Slave Mode when Job Mode is Disabled: Description corrected
		499	Table 11.43 Notes on Setting Registers: Contents for CSIHnRX0W, corrected
		500	Table 11.43 Notes on Setting Registers: Contents for CSIHnRX0H, corrected
		Section 13 LIN Master Interface (RLIN2)	
		567	Table 13.23 RLN21nmLiMST Register Contents: Functional description on bits 7 to 2 (Reserved), corrected
		575	13.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register: Description corrected
		592	Table 13.35 Types of Status: Status Set Condition for Header reception end, corrected
		Section 14 CAN Interface (RS-CAN)	
		606	Figure 14.1 RS-CAN Module Block Diagram, corrected
		641	14.3.5 RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 3): Description on BLF Flag, corrected
		682	14.3.31 RSCAN0CFCCk — Transmit/Receive FIFO Buffer Configuration and Control Register k (k = 0 to 11): Description on CFITR Bit and CFITSS Bit, corrected
		770	14.5.1.2 Bit Timing Setting: Description corrected
		772	Figure 14.19 Receive Rule Setting Procedure, corrected (y = 0, deleted)
		Section 17 Timer Array Unit D (TAUD)	
		855	Table 17.33 TAUDnTOL Register Contents: Functional description on bits 15 to 0, corrected
		989	Table 17.121 Control Bit Settings in Synchronous Channel Output Mode 1: Table title, corrected
		1000	Table 17.129 Control Bit Settings in Independent Channel Output Mode 2: Table title, corrected
		1013	Table 17.137 Control Bit Settings in Synchronous Channel Output Mode 1: Table title, corrected

Rev.	Date	Description	
		Page	Summary
1.30	Jul 29, 2016	1040	Table 17.160 Control Bit Settings for Slave Channel 1 in Synchronous Channel Output Mode 1: Table title corrected. Setting for TAUDnTOM.TAUDnTOMm and TAUDnTOC.TAUDnTOCm bits, corrected.
		Section 18 Timer Array Unit J (TAUJ)	
		1158	Table 18.26 TAUJnTOL Register Contents: Functional description on bits 3 to 0, corrected
		1221	Table 18.66 Control Bit Settings for Synchronous Channel Output Mode 1: Table title, corrected
		Section 18 Timer Array Unit J (TAUJ)	
		1466	20.4.1.5 Operating Procedure: Operation for During operation, corrected
		1466	20.4.1.5 Operating Procedure: State of TAPA for Stopping operation, corrected
		Section 20 Timer Option Module (TAPA)	
		1894	25.3.12 RDC2nINIT — RDC2n Initializing Register 2 (n = 0, 1): Section title, corrected
		1896	Table 25.30 RDC2nCON Register Contents: Note 3, corrected
		1903	Table 25.39 RDC2nERDET Register Contents: Functional description on bits 9 and 8, corrected (*3 added). Note 3 added.
		1922	Figure 25.6 Flowchart of Built-In Self-Test (BIST) Process: Process of "Wait for BIST recovery time" added to the flow. Note 1 and Note 2 added.
		1927	Figure 25.11 RDC2 Initial Operation Flow: Precess of "Wait for BIST recovery time" added to the flow. Note 1 to Note 4 added.
		Section 26 A/D Converter (ADCC)	
		2004	Table 26.49 Notes on Setting Registers, corrected
		Section 27 Functional Safety	
		2142	Table 27.125 CLMATEST Register Contents: Functional description on bit 1 (MONCLKMSK), corrected
		2144	27.6.4 Detection of Abnormal Clock Frequency: Description corrected
		2147	27.6.5 Self-Diagnosis: Description corrected
		Section 28 Error Control Module (ECM)	
		2154	Table 28.8 List of Error Sources and Safety Processing: Error Source for PE guard (PEG) module, corrected
		2156	Table 28.9 Merging of Error Sources: Error Source for PE guard (PEG) module, corrected
		2161	28.3.4 ECMmESSTR0 (m = M/C) — ECM Master/Checker Error Source Status Register 0: Description corrected
		2162	28.3.5 ECMmESSTR1 (m = M/C) — ECM Master/Checker Error Source Status Register 1: Description corrected
		Section 31 Flash Memory	
		2223	Table 31.4 Summary of Security Functions: Description of OTP function, corrected
		Section 35 Electrical Characteristics	
		2288	35.4 A/D Converter Characteristics: Description of “Errors in the External Circuit of the A/D Converter” added
		2290	Table 35.35 RDC Analog Pin Characteristics: Note 2 corrected ((±10%) → (±0%)). Note 4 corrected ((velocity voltage provided) → (high-speed rotation)). Note 6 corrected ((±10%) → (±0%)).
		2294	Table 35.39 Data Flash Basic Characteristics: Note 1 corrected (64KB → 64B)
1.40	Jan 30, 2017	Section 8 Resets	
		361	8.2.2 Internal Reset State: Description on tINTRES, deleted
		Section 11 Clocked Serial Interface H (CSIH)	
		422	Table 11.24 CSIHnCFGx Register Contents (2/5): CAUTION for the functional description on the CSIHnDLSx[3:0] bits, modified
		Section 12 Serial Communication Interface 3 (SCI3)	
		520	12.4.1.2 Receive Data Sampling Timing and Reception Margin: Bit name corrected (SABCS → ABCS)
		523	12.4.1.6 Serial Data Transmission (Asynchronous Mode): “Supplementary note on operation when data transmission in asynchronous mode is enabled” added
		524	Figure 12.7 Example of Serial Transmission Flowchart: Description in [1] modified (high level → high level (preamble))
		542	Figure 12.24 Example of Serial Reception Flowchart: Note modified
		546	12.4.5 Interrupt Sources: Description of Caution changed

Rev.	Date	Description	
		Page	Summary
1.40	Jan 30, 2017	Section 17 Timer Array Unit D (TAUD)	
		940	17.4.9.8, (6) Specific Timing Diagrams: Overflow Operation, (c) TAUDnCMORm.TAUDnCOS[1:0] = 10B: Description modified
		941	17.4.9.8, (6) Specific Timing Diagrams: Overflow Operation, (d) TAUDnCMORm.TAUDnCOS[1:0] = 11B: Description modified
		974	17.4.10.2, (6) Specific Timing Diagrams, (a) Operation start and stop: Description error on the bit name, modified
		1001	Table 17.131 Operating Procedure for One-Shot Pulse Output Function: TAUDn Status of During Operation, modified
		1034	17.4.12.5, (1) Overview: Prerequisites modified
		1059	Table 17.172 Operation of TAUDTTOUTm upon Occurrence of an Interrupt on Slave Channel 2: TAUDTTOUTm Set/Reset Timing for a count up when TAUDnTDL.TAUDnTDLm = 1, modified
		1100	Table 17.211 Operating Procedure for Non-Complementary Modulation Output Function Type 1 (2/2): TAUDn Status of During Operation, Modified
		1104	17.4.13.2, (2) Equations: Modified
		1119	17.4.13.3, (2) Equations: Modified
		1131	Table 17.242 Operating Procedure for Complementary Modulation Output Function (2/2): TAUDn Status of During Operation, modified
		Section 18 Timer Array Unit J (TAUJ)	
		1180	Table 18.33 Contents of TAUJnCMORm Register for Interval Timer Function: Function of the TAUJnSTS[2:0] bits, modified
		1190	Figure 18.22 Counter Triggered by Rising TAUJTTINm Input Edge (TAUJnCMURm.TAUJnTIS[1:0] = 01B), TAUJnCMORm.TAUJnMD0 = 1: Modified
		1191	18.4.9.3, (1) Overview: Description modified
		1204	18.4.9.4, (6) Specific timing diagrams: overflow behavior, (c) TAUJnCMORm.TAUJnCOS[1:0] = 10B: Description modified
		1204	18.4.9.4, (6) Specific timing diagrams: overflow behavior, (d) TAUJnCMORm.TAUJnCOS[1:0] = 11B: Description modified
		1208	Table 18.56 Operating Procedure for TAUJTTINm Input Position Detection Function: Operation of During operation, modified
		1213	Table 18.60 Operating Procedure for TAUJTTINm Input Period Count Detection Function: Operation of During operation, modified
		Section 19 Motor Control Timer (TSG3)	
		1312	Table 19.51 List of Flags: Flag Name of Number (9), modified
		Section 22 Encoder Timer (ENCA)	
		1540	Figure 22.17 Overflow Occurrence and Setting for Clearing of the Overflow Flag: Waveform of the ENCAnCSF flag, modified
		1541	Figure 22.18 Underflow Occurrence and Setting for Clearing of the Underflow Flag: Waveform of the ENCAnCSF flag, modified
		1542	Figure 22.19 Timing Chart of Counter Clearing and Capture Operation by Encoder Clear Input (ENCAnEC pin): Waveform of the ENCAnCSF flag, modified
		1543	Figure 22.20 Conflict between Overflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin): Waveform of the ENCAnCSF flag, modified
		1544	Figure 22.21 Conflict between Underflow Occurrence and Clear Operation by Encoder Clear Input (ENCAnEC pin): Waveform of the ENCAnCSF flag, modified
		1545	Figure 22.22 Overflow Operation Immediately after Startup: E Waveform of the NCAnCSF flag, modified
		1546	Figure 22.23 Underflow Operation Immediately after Startup: Waveform of the ENCAnCSF flag, modified
		1547	Figure 22.24 Using the ENCAnLDE Function Immediately after Startup: Waveform of the ENCAnCSF flag, modified
		1548	Figure 22.25 ENCAnLDE Function (when ENCAnLDE = 0): Waveform of the ENCAnCSF flag, modified
		1549	Figure 22.26 ENCAnLDE Function (when ENCAnLDE=1): Waveform of the ENCAnCSF flag, modified
		1550	Figure 22.27 Conflict between ENCAnLDE Function and Rewriting of ENCAnCCR0 Register: Waveform of the ENCAnCSF flag, modified

Rev.	Date	Description	
		Page	Summary
1.40	Jan 30, 2017	1551	Figure 22.28 Conflict between ENCAnLDE Function and Clear Operation by Encoder Clear Input: Waveform of the ENCAnCSF flag, modified
		1553	Figure 22.29 Up-counting after Conflict between ENCAnLDE Function and Clearing of the Encoder Counter: Waveform of the ENCAnCSF flag, modified
		1554	Figure 22.30 Capture Operation between Counter Clocks (ENCAnCCR1): Waveform of the ENCAnCSF flag, modified
		1554	Figure 22.31 Capture Operation between Counter Clocks (ENCAnCCR0): Waveform of the ENCAnCSF flag, modified
		1555	Figure 22.32 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS=0: Waveform of the ENCAnCSF flag, modified
		1556	Figure 22.33 Encoder Operation when Compare Match Clear Control is Enabled and ENCAnCTS=1: Waveform of the ENCAnCSF flag, modified
		1557	Figure 22.34 Encoder Operation when Compare Match Clear Control is Disabled: Waveform of the ENCAnCSF fla, modified
		1558	Figure 22.35 Capture Operation Performed upon Clearing by ENCAnEC, ENCAnE0, and ENCAnE1 when ENCAnSCE = 1: Waveform of the ENCAnCSF flag, modified
		1562	Figure 22.40 Capture Operation Performed upon Clearing by ENCAnEC when ENCAnSCE = 0: Waveform of the ENCAnCSF flag, modified
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1849	Figure 24.12 Process Flow of PI Control IP: Position of “—” under the target values of d- and q-axis changed
		Section 29 Data CRC (DCRA)	
		2195	Table 29.9 DCRAnCTL Register Contents: Function of the DCRAnPOL bit, modified
		2196	Table 29.10 Setting Example of Initial Start Values (when read at a reset): The initial start value, EXOR value, DCRAnCOUT read value of the 16-bit CCITT, modified
		Section 31 Flash Memory	
		2230	31.9.2.2 FHVE3 — FHVE3 Control Register: Description modified
		2230	Table 31.12 FHVE3 Register Contents: Function of the FHVE3CNT bit, modified
		Appendix Package Dimensions	
		2296	Appendix Package Dimensions: Added
		2297	Appendix Package Dimensions: Added
1.50	Jun 23, 2017	Section 13 LIN Master Interface (RLIN2)	
		576	13.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register: The description of “For response reception” modified
		Section 19 Motor Control Timer (TSG3)	
		1357	Figure 19.57 Basic Operation Flow in HT-PWM Mode, NOTE: The names of bits, modified (TSG3nPRE → TSG3nPIE, TSG3nVRE → TSG3nVIE)
		Section 26 A/D Converter (ADCC)	
		1951	Table 26.11 ADCCnVCRj Register Contents: The functional description of the GCTRL[5:0] bits, modified (The reference, added)
		2005 to 2007	26.7.3 Notes when Using Simultaneous Track-and-Hold: Added
		Section 28 Error Control Module (ECM)	
		2178	Table 28.29 ECMPCMD1 Register Contents: The functional description of reserved bits, modified
		2190	28.4.5.1 Sequence of Writing to the Write-Protected Registers: Note 1, added
1.60	Apr 20, 2018	Section 1 Overview	
		50	Table 1.2 List of Products: Part numbers changed, wafer process manufacturing sites added
		Section 2 Pins	
		92	Table 2.24 List of the Pins which Require PIPC Register Setting: Pin name corrected (CSIHnSCK → CSIHnSC)
		98	Table 2.28 List of Registers in C1H Port Group 3: Bit map 7 of PMCSR3 register changed
		100	Table 2.30 List of Registers in C1H Port Group 5: Bit map 0 (PFC) added to the bottom table.
		107	Table 2.38 List of Pin Alternative Functions in C1H Port Group 5: 4th alternative function ALT_OUT4 for P5_2 corrected
		118	Table 2.48 List of Pin Alternative Functions in C1M Port Group 0: 7th alternative function ALT_OUT7 for P0_9 corrected

Rev.	Date	Description	
		Page	Summary
1.60	Apr 20, 2018	118	Table 2.49 List of Pin Alternative Functions in C1M Port Group 1: 6th alternative function ALT_IN6 for P1_5 corrected
		Section 3 CPU System	
		191	Table 3.67 IPGECRUM Register Contents: Functional description of b0 (VD bit) changed
		198	Table 3.76 SEGCONT Register Contents (1/2): Functional description of b8 (VCRE bit) partially added
		199	Table 3.76 SEGCONT Register Contents (2/2): Functional description of b4 (VCIE bit) partially added
		Section 6 Interrupt	
		221	6.2.1 Configuration of Registers: Description about writing to the registers, modified
		Section 7 DMA Controller	
		291	Table 7.8 List of DMA Trigger Sources (4/4): DMACTRG125 changed
		Section 10 Clock Controller	
		376	10.1 Features of the RH850/C1x Clock Controller: Description modified, description of Note 1 partially added
		377	Table 10.1 List of Clocks: Note 1 added
		392	10.3.1 Operation When the Divide Function is Used: Description modified
		Section 14 CAN Interface (RS-CAN)	
		747	14.4.2.2 Channel Modes, (1) Channel Stop Mode: Description modified, bit name corrected (GSLPR → CSLPR)
		Section 17 Timer Array Unit D (TAUD)	
		840	Table 17.15 TAUDnCNTm Read Values after Re-Enabling Counter: Gate count mode deleted
		844	Table 17.16 TAUDnCMORm Register Contents (3/3): Function of b4 to b0 (TAUDnMD[4:0] bits): Two operating modes specified as “setting prohibited” deleted, description “Settings other than the above are prohibited.” added under the table
		886	Table 17.44 Effect of TAUDnCMORm.TAUDnMD0 Bit on Generation of INTTAUDnIm when Counter is Triggered: Gate count mode deleted
		993	17.4.12.2 One-Shot Pulse Output Function, (1) Overview: Summary: Description changed
		993	17.4.12.2 One-Shot Pulse Output Function, (1) Overview: Prerequisites: Description partially added
		993	17.4.12.2 One-Shot Pulse Output Function, (1) Overview: Functional description of Master channel: Description partially added
		995	17.4.12.2 One-Shot Pulse Output Function, (3) Block Diagram and General Timing Diagram: Description modified
		995	Figure 17.81 General Timing Diagram of One-Shot Pulse Output Function (in the Case of an External Input Signal): Title modified
		996	Figure 17.82 General Timing Diagram of One-Shot Pulse Output Function (in the Case of a Software Trigger), added
		Section 18 Timer Array Unit J (TAUJ)	
		1151	Table 18.15 TAUJnCMORm Register Contents (3/3): Function of b4 to b0 (TAUJnMD[4:0] bits): Five operating modes specified as “setting prohibited” deleted, description “Settings other than the above are prohibited.” added under the table
		Section 19 Motor Control Timer (TSG3)	
		1247	19.3.8 TSG3nCTL7 — TSG3n Control Register 7: Description corrected (“TSG3nMD2-0 = 001” → “TSG3nMD2-0 = 010”)
		1357	Figure 19.57 Basic Operation Flow in HT-PWM Mode: Processing for “Initial setting”: Description of “Setting of HT-PWM mode” changed
		1371	Figure 19.66 Example of Operation During Reloading at 18-Bit Counter Peak Timing: Timing of INTTSG3nI2 interrupt changed
		1371	Figure 19.67 Example of Operation During Reloading at 18-Bit Counter Trough Timing: “INTTGS3nI1 interrupt” corrected to “INTTSG3nI1 interrupt”, timing of INTTSG3nI1 interrupt changed
		Section 24 Enhanced Motor Control Unit (EMU2)	
		1751	Table 24.59 EMU2nANGCTR Register Contents: Functional description of b0 (PHISEL bit) changed (EMUnRESTHSFT register → EMU2nRESTHSFT register)
		1760	Table 24.73 EMU2nCTRINMD Register Contents: Note 1 corrected

Rev.	Date	Description	
		Page	Summary
1.60	Apr 20, 2018	Section 25 R/D Converter (RDC2)	
		1915	25.4.1.7 Compare Match Interrupt: Description partially added
		—	Figure 25.3 Timing Chart for Compare Match Interrupt Request Signal, deleted
		1915	Figure 25.3 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is Off, added
		1916	Figure 25.4 Timing Chart for the Compare Match Interrupt Request Signal when Hysteresis is On, added
		1917	25.4.1.8 Encoder Pulse Output Function: Description partially added
		—	Figure 25.4 ABZ Output Example, deleted
		—	Figure 25.5 Hysteresis Output Example, deleted
		1917	Figure 25.5 Waveform of Encoder Phase Pulse Operation when Hysteresis is On, added
		1917	Figure 25.6 Waveform of Encoder Phase Pulse Operation when Hysteresis is Off, added
		Section 26 A/D Converter (ADCC)	
		1946	Figure 26.3 ADCC Functional Block Diagram, modified (scan group 4 end interrupt → scan group 4 scan end signal)
		1953	Table 26.12 ADCCnDRj Register Contents: Caution added
		Section 27 Functional Safety	
		2149	27.6.4 Detection of Abnormal Clock Frequency, (2) Definition of the input of initial values to the threshold registers: Description modified
		Section 28 Error Control Module (ECM)	
		2158	Table 28.8 List of Error Sources and Safety Processing (2/2): Note 13 added
		Section 31 Flash Memory	
		2232	Table 31.9 FRDCYCLD Register Contents: Note added
		Section 35 Electrical Characteristics	
		2292	Table 35.33 Parameters of C1M (R7F701271EAFP #**0) and C1H (R7F701270EABG #**0): Title changed
		2292	Table 35.34 Parameters of C1M (R7F701271EAFP #**4) and C1H (R7F701270EABG-C #**4), added

RH850/C1x User's Manual: Hardware

Publication Date: Rev.0.40 Jul 12, 2013
Rev.1.60 May 31, 2018

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-651-700, Fax: +44-1628-651-804

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709 Quantum Plaza, No.27 ZhichunLu, Haidian District, Beijing, 100191 P. R. China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, 200333 P. R. China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 5.0

RH850/C1x

Renesas Electronics Corporation

R01UH0414EJ0160