

# RL78/G14

User's Manual: Hardware

16-Bit Single-Chip Microcontrollers

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

## 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 be responsible for determining what licenses are required from any third parties, and obtaining such licenses for the lawful import, export, manufacture, sales, utilization, distribution or other disposal of any products incorporating Renesas Electronics products, if required.
5. 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.
6. 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.

7. No semiconductor product is absolutely secure. Notwithstanding any security measures or features that may be implemented in Renesas Electronics hardware or software products, Renesas Electronics shall have absolutely no liability arising out of any vulnerability or security breach, including but not limited to any unauthorized access to or use of a Renesas Electronics product or a system that uses a Renesas Electronics product. RENESAS ELECTRONICS DOES NOT WARRANT OR GUARANTEE THAT RENESAS ELECTRONICS PRODUCTS, OR ANY SYSTEMS CREATED USING RENESAS ELECTRONICS PRODUCTS WILL BE INVULNERABLE OR FREE FROM CORRUPTION, ATTACK, VIRUSES, INTERFERENCE, HACKING, DATA LOSS OR THEFT, OR OTHER SECURITY INTRUSION ("Vulnerability Issues"). RENESAS ELECTRONICS DISCLAIMS ANY AND ALL RESPONSIBILITY OR LIABILITY ARISING FROM OR RELATED TO ANY VULNERABILITY ISSUES. FURTHERMORE, TO THE EXTENT PERMITTED BY APPLICABLE LAW, RENESAS ELECTRONICS DISCLAIMS ANY AND ALL WARRANTIES, EXPRESS OR IMPLIED, WITH RESPECT TO THIS DOCUMENT AND ANY RELATED OR ACCOMPANYING SOFTWARE OR HARDWARE, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.
8. 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.
9. 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.
10. 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.
11. 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.

12. 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.
13. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
14. 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. 5.0-1 October 2020)

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

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

## 1. Precaution against Electrostatic Discharge (ESD)

A strong electrical field, when exposed to a CMOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop the generation of static electricity as much as possible, and quickly dissipate it when it occurs. Environmental control must be adequate. When it is dry, a humidifier should be used. This is recommended to avoid using insulators that can 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 must be grounded. The operator must also be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions must be taken for printed circuit boards with mounted semiconductor devices.

## 2. Processing at power-on

The state of the product is undefined at the time when power is supplied. The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the time when power is supplied. In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the time when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the time when power is supplied until the power reaches the level at which resetting is specified.

## 3. Input of signal during power-off state

Do not input signals or an I/O pull-up power supply while the device is powered off. 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. Follow the guideline for input signal during power-off state as described in your product documentation.

## 4. Handling of unused pins

Handle unused pins in accordance with the directions given under handling of unused pins in the manual. The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of the LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible.

## 5. Clock signals

After applying a reset, only release the reset line after the operating clock signal becomes stable. When switching the clock signal during program execution, wait until the target clock signal is stabilized. When the clock signal is generated with an external resonator or from an external oscillator during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Additionally, when switching to a clock signal produced with an external resonator or by an external oscillator while program execution is in progress, wait until the target clock signal is stable.

## 6. 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  $V_{IL}$  (Max.) and  $V_{IH}$  (Min.) due to noise, for example, 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  $V_{IL}$  (Max.) and  $V_{IH}$  (Min.).

## 7. Prohibition of access to reserved addresses

Access to reserved addresses is prohibited. The reserved addresses are provided for possible future expansion of functions. Do not access these addresses as the correct operation of the LSI is not guaranteed.

## 8. Differences between products

Before changing from one product to another, for example to a product with a different part number, confirm that the change will not lead to problems. The characteristics of a microprocessing unit or microcontroller unit products in the same group but having a different part number might differ in terms of internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

# How to Use This Manual

## Readers

This manual is intended for user engineers who wish to understand the functions of the RL78/G14 and design and develop application systems and programs for these devices. The target products are as follows.

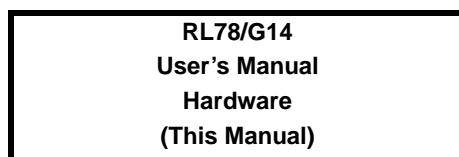
- 30-pin: R5F104Ax (x = A, C to G)
- 32-pin: R5F104Bx (x = A, C to G)
- 36-pin: R5F104Cx (x = A, C to G)
- 40-pin: R5F104Ex (x = A, C to H)
- 44-pin: R5F104Fx (x = A, C to H, J)
- 48-pin: R5F104Gx (x = A, C to H, J, K, L)
- 52-pin: R5F104Jx (x = C to H, J)
- 64-pin: R5F104Lx (x = C to H, J, K, L)
- 80-pin: R5F104Mx (x = F, G, H, J, K, L)
- 100-pin: R5F104Px (x = F, G, H, J, K, L)

## Purpose

This manual is intended to give users an understanding of the functions described in the **Organization** below.

## Organization

The RL78/G14 manual is separated into two parts: this manual and the software edition (common to the RL78 family).



- Pin functions
- Internal block functions
- Interrupts
- Other on-chip peripheral functions
- Electrical specifications
- CPU functions
- Instruction set
- Explanation of each instruction

## How to Read This Manual

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

- To gain a general understanding of functions:

→ Read this manual in the order of the **CONTENTS**. The mark "<R>" shows major revised points. The revised points can be easily searched by copying an "<R>" in the PDF file and specifying it in the "Find what:" field.

- How to interpret the register format:

→ For a bit number enclosed in angle brackets, the bit name is defined as a reserved word in the assembler, and is defined as an sfr variable using the #pragma sfr directive in the compiler.

- To know details of the RL78/G14 Microcontroller instructions:

→ Refer to the separate document **RL78 Family User's Manual Software (R01US0015E)**.

<b>Conventions</b>	Data significance:	Higher digits on the left and lower digits on the right
	Active low representations:	<u>xxx</u> (overscore over pin and signal name)
<b>Note:</b>		Footnote for item marked with Note in the text
<b>Caution:</b>		Information requiring particular attention
<b>Remark:</b>		Supplementary information
	Numerical representations:	Binary.....xxxx or xxxxB Decimal.....xxxx Hexadecimal .....xxxxH

**Related Documents** The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

#### Documents Related to Devices

Document Name	Document No.
RL78/G14 User's Manual Hardware	This manual
RL78 Family User's Manual Software	R01US0015E

#### Documents Related to Flash Memory Programming (User's Manual)

Document Name	Document No.
PG-FP5 Flash Memory Programmer User's Manual	—
RL78, 78K, V850, RX100, RX200, RX600 (Except RX64x), R8C, SH	R20UT2923E
Common	R20UT2922E
Setup Manual	R20UT0930E

**Caution** The related documents listed above are subject to change without notice. Be sure to use the latest version of each document when designing.

#### Other Documents

Document Name	Document No.
Renesas Microcontrollers RL78 Family	R01CP0003E
Semiconductor Package Mount Manual	R50ZZ0003E
Semiconductor Reliability Handbook	R51ZZ0001E

**Caution** The related documents listed above are subject to change without notice. Be sure to use the latest version of each document when designing.

All trademarks and registered trademarks are the property of their respective owners.

EEPROM is a trademark of Renesas Electronics Corporation.

SuperFlash is a registered trademark of Silicon Storage Technology, Inc. in several countries including the United States and Japan.

Caution: This product uses SuperFlash® technology licensed from Silicon Storage Technology, Inc.
--

# CONTENTS

1.	OUTLINE .....	1
1.1	Features .....	1
1.2	Ordering Information .....	5
1.3	Pin Configuration (Top View) .....	10
1.3.1	30-pin products .....	10
1.3.2	32-pin products .....	11
1.3.3	36-pin products .....	13
1.3.4	40-pin products .....	14
1.3.5	44-pin products .....	15
1.3.6	48-pin products .....	16
1.3.7	52-pin products .....	18
1.3.8	64-pin products .....	19
1.3.9	80-pin products .....	21
1.3.10	100-pin products .....	22
1.4	Pin Identification .....	24
1.5	Block Diagram .....	25
1.5.1	30-pin products .....	25
1.5.2	32-pin products .....	26
1.5.3	36-pin products .....	27
1.5.4	40-pin products .....	28
1.5.5	44-pin products .....	29
1.5.6	48-pin products .....	30
1.5.7	52-pin products .....	31
1.5.8	64-pin products .....	32
1.5.9	80-pin products .....	33
1.5.10	100-pin products .....	34
1.6	Outline of Functions .....	35
2.	PIN FUNCTIONS .....	54
2.1	Port Functions .....	54
2.1.1	30-pin (Code Flash Memory 16 KB to 64 KB) .....	55
2.1.2	32-pin (Code Flash Memory 16 KB to 64 KB) .....	57
2.1.3	36-pin (Code Flash Memory 16 KB to 64 KB) .....	59
2.1.4	40-pin (Code Flash Memory 16 KB to 64 KB) .....	61
2.1.5	44-pin (Code Flash Memory 16 KB to 64 KB) .....	63
2.1.6	48-pin (Code Flash Memory 16 KB to 64 KB) .....	65
2.1.7	52-pin (Code Flash Memory 16 KB to 64 KB) .....	67
2.1.8	64-pin (Code Flash Memory 16 KB to 64 KB) .....	69
2.1.9	30-pin (Code Flash Memory 96 KB to 256 KB) .....	71
2.1.10	32-pin (Code Flash Memory 96 KB to 256 KB) .....	73
2.1.11	36-pin (Code Flash Memory 96 KB to 256 KB) .....	75
2.1.12	40-pin (Code Flash Memory 96 KB to 256 KB) .....	77
2.1.13	44-pin (Code Flash Memory 96 KB to 256 KB) .....	79
2.1.14	48-pin (Code Flash Memory 96 KB to 512 KB) .....	81
2.1.15	52-pin (Code Flash Memory 96 KB to 256 KB) .....	83

2.1.16	64-pin (Code Flash Memory 96 KB to 512 KB) .....	85
2.1.17	80-pin .....	87
2.1.18	100-pin .....	90
2.2	Functions other than port pins .....	93
2.2.1	Functions for each product .....	93
2.2.2	Pins for each product (pins other than port pins) .....	98
2.3	Connection of Unused Pins .....	100
2.4	Pin Block Diagrams .....	101
<b>3.</b>	<b>CPU ARCHITECTURE .....</b>	<b>121</b>
3.1	Memory Space .....	121
3.1.1	Internal program memory space .....	137
3.1.2	Mirror area .....	140
3.1.3	Internal data memory space .....	142
3.1.4	Special function register (SFR) area .....	143
3.1.5	Extended special function register (2nd SFR: 2nd Special Function Register) area .....	143
3.1.6	Data memory addressing .....	144
3.2	Processor Registers .....	145
3.2.1	Control registers .....	145
3.2.2	General-purpose registers .....	148
3.2.3	ES and CS registers .....	149
3.2.4	Special function registers (SFRs) .....	150
3.2.5	Extended special function registers (2nd SFRs: 2nd Special Function Registers) .....	156
3.3	Instruction Address Addressing .....	166
3.3.1	Relative addressing .....	166
3.3.2	Immediate addressing .....	166
3.3.3	Table indirect addressing .....	167
3.3.4	Register direct addressing .....	168
3.4	Addressing for Processing Data Addresses .....	169
3.4.1	Implied addressing .....	169
3.4.2	Register addressing .....	169
3.4.3	Direct addressing .....	170
3.4.4	Short direct addressing .....	171
3.4.5	SFR addressing .....	172
3.4.6	Register indirect addressing .....	173
3.4.7	Based addressing .....	174
3.4.8	Based indexed addressing .....	177
3.4.9	Stack addressing .....	178
<b>4.</b>	<b>PORT FUNCTIONS .....</b>	<b>181</b>
4.1	Port Functions .....	181
4.2	Port Configuration .....	182
4.2.1	Port 0 .....	184
4.2.2	Port 1 .....	184
4.2.3	Port 2 .....	185
4.2.4	Port 3 .....	187
4.2.5	Port 4 .....	187
4.2.6	Port 5 .....	188
4.2.7	Port 6 .....	188

4.2.8	Port 7 .....	188
4.2.9	Port 8 .....	189
4.2.10	Port 10 .....	189
4.2.11	Port 11 .....	189
4.2.12	Port 12 .....	190
4.2.13	Port 13 .....	190
4.2.14	Port 14 .....	191
4.2.15	Port 15 .....	191
4.3	Registers Controlling Port Function .....	192
4.3.1	Port mode registers (PMxx) .....	199
4.3.2	Port registers (Pxx) .....	200
4.3.3	Pull-up resistor option registers (PUxx) .....	202
4.3.4	Port input mode registers (PMxx) .....	203
4.3.5	Port output mode registers (POMxx) .....	204
4.3.6	Port mode control registers (PMCxx) .....	205
4.3.7	A/D port configuration register (ADPC) .....	206
4.3.8	Peripheral I/O redirection register 0 (PIOR0) .....	207
4.3.9	Peripheral I/O redirection register 1 (PIOR1) .....	209
4.3.10	Global digital input disable register (GDIDIS) .....	210
4.4	Port Function Operations .....	211
4.4.1	Writing to I/O port .....	211
4.4.2	Reading from I/O port .....	211
4.4.3	Operations on I/O port .....	211
4.4.4	Handling different potential (1.8 V, 2.5 V, 3 V) by using EVDD $\leq$ VDD .....	212
4.4.5	Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers .....	212
4.5	Register Settings When Using Alternate Function .....	215
4.5.1	Basic concept when using alternate function .....	215
4.5.2	Register settings for alternate function whose output function is not used .....	216
4.5.3	Register setting examples for used port and alternate functions .....	218
4.6	Cautions When Using Port Function .....	232
4.6.1	Cautions on 1-Bit Manipulation Instruction for Port Register n (Pn) .....	232
4.6.2	Notes on specifying the pin settings .....	233
5.	CLOCK GENERATOR .....	234
5.1	Functions of Clock Generator .....	234
5.2	Configuration of Clock Generator .....	237
5.3	Registers Controlling Clock Generator .....	239
5.3.1	Clock operation mode control register (CMC) .....	239
5.3.2	System clock control register (CKC) .....	242
5.3.3	Clock operation status control register (CSC) .....	244
5.3.4	Oscillation stabilization time counter status register (OSTC) .....	245
5.3.5	Oscillation stabilization time select register (OSTS) .....	247
5.3.6	Peripheral enable registers 0, 1 (PER0, PER1) .....	249
5.3.7	Subsystem clock supply mode control register (OSMC) .....	254
5.3.8	High-speed on-chip oscillator frequency select register (HOCODIV) .....	255
5.3.9	High-speed on-chip oscillator trimming register (HIOTRM) .....	256
5.4	System Clock Oscillator .....	257
5.4.1	X1 oscillator .....	257
5.4.2	XT1 oscillator .....	257

5.4.3	High-speed on-chip oscillator .....	261
5.4.4	Low-speed on-chip oscillator .....	261
5.5	Clock Generator Operation .....	262
5.6	Controlling Clock .....	264
5.6.1	Example of setting high-speed on-chip oscillator .....	264
5.6.2	Example of setting X1 oscillation clock .....	266
5.6.3	Example of setting XT1 oscillation clock .....	268
5.6.4	CPU clock status transition diagram .....	269
5.6.5	Condition before changing CPU clock and processing after changing CPU clock .....	275
5.6.6	Time required for switchover of CPU clock and main system clock .....	277
5.6.7	Conditions before clock oscillation is stopped .....	278
5.7	Resonator and Oscillator Constants .....	279
6.	TIMER ARRAY UNIT .....	282
6.1	Functions of Timer Array Unit .....	284
6.1.1	Independent channel operation function .....	284
6.1.2	Simultaneous channel operation function .....	285
6.1.3	8-bit timer operation function (channels 1 and 3 only) .....	286
6.1.4	LIN-bus supporting function (channel 3 of unit 0 only) .....	287
6.2	Configuration of Timer Array Unit .....	288
6.2.1	Timer count register mn (TCRmn) .....	293
6.2.2	Timer data register mn (TDRmn) .....	295
6.3	Registers Controlling Timer Array Unit .....	296
6.3.1	Peripheral enable register 0 (PER0) .....	297
6.3.2	Timer clock select register m (TPSm) .....	298
6.3.3	Timer mode register mn (TMRmn) .....	301
6.3.4	Timer status register mn (TSRmn) .....	306
6.3.5	Timer channel enable status register m (TEm) .....	307
6.3.6	Timer channel start register m (TSm) .....	308
6.3.7	Timer channel stop register m (TTm) .....	310
6.3.8	Timer input select register 0 (TIS0) .....	311
6.3.9	Timer output enable register m (TOEm) .....	312
6.3.10	Timer output register m (TOM) .....	313
6.3.11	Timer output level register m (TOLm) .....	314
6.3.12	Timer output mode register m (TOMm) .....	315
6.3.13	Input switch control register (ISC) .....	316
6.3.14	Noise filter enable registers 1, 2 (NFEN1, NFEN2) .....	317
6.3.15	Registers controlling port functions of pins to be used for timer I/O .....	320
6.4	Basic Rules of Timer Array Unit .....	321
6.4.1	Basic rules of simultaneous channel operation function .....	321
6.4.2	Basic rules of 8-bit timer operation function (channels 1 and 3 only) .....	323
6.5	Operation of Counter .....	324
6.5.1	Count clock (fTCLK) .....	324
6.5.2	Start timing of counter .....	326
6.5.3	Operation of counter .....	327
6.6	Channel Output (TOmn pin) Control .....	332
6.6.1	TOmn pin output circuit configuration .....	332
6.6.2	TOmn Pin Output Setting .....	333
6.6.3	Cautions on Channel Output Operation .....	334

6.6.4	Collective manipulation of TOmn bit .....	339
6.6.5	Timer Interrupt and TOmn Pin Output at Operation Start .....	340
6.7	Timer Input (TImn) Control .....	341
6.7.1	TImn input circuit configuration .....	341
6.7.2	Noise filter .....	341
6.7.3	Cautions on channel input operation .....	342
6.8	Independent Channel Operation Function of Timer Array Unit .....	343
6.8.1	Operation as interval timer/square wave output .....	343
6.8.2	Operation as external event counter .....	348
6.8.3	Operation as frequency divider (channel 0 of unit 0 only) .....	352
6.8.4	Operation as input pulse interval measurement .....	356
6.8.5	Operation as input signal high-/low-level width measurement .....	360
6.8.6	Operation as delay counter .....	364
6.9	Simultaneous Channel Operation Function of Timer Array Unit .....	368
6.9.1	Operation as one-shot pulse output function .....	368
6.9.2	Operation as PWM function .....	375
6.9.3	Operation as multiple PWM output function .....	382
6.10	Cautions When Using Timer Array Unit .....	390
6.10.1	Cautions When Using Timer output .....	390
7.	TIMER RJ .....	391
7.1	Functions of Timer RJ .....	391
7.2	Configuration of Timer RJ .....	392
7.3	Registers Controlling Timer RJ .....	393
7.3.1	Peripheral enable register 1 (PER1) .....	394
7.3.2	Subsystem clock supply mode control register (OSMC) .....	395
7.3.3	Timer RJ counter register 0 (TRJ0) .....	396
7.3.4	Timer RJ control register 0 (TRJCR0) .....	397
7.3.5	Timer RJ I/O control register 0 (TRJIOC0) .....	399
7.3.6	Timer RJ mode register 0 (TRJMR0) .....	401
7.3.7	Timer RJ event pin select register 0 (TRJISR0) .....	402
7.3.8	Port mode registers 0, 3, 4, 5 (PM0, PM3, PM4, PM5) .....	403
7.4	Timer RJ Operation .....	405
7.4.1	Reload Register and Counter Rewrite Operation .....	405
7.4.2	Timer Mode .....	406
7.4.3	Pulse Output Mode .....	407
7.4.4	Event Counter Mode .....	408
7.4.5	Pulse Width Measurement Mode .....	410
7.4.6	Pulse Period Measurement Mode .....	411
7.4.7	Coordination with Event Link Controller (ELC) .....	412
7.4.8	Output Settings for Each Mode .....	412
7.5	Cautions for Timer RJ .....	413
7.5.1	Count Operation Start and Stop Control .....	413
7.5.2	Access to Flags (Bits TEDGF and TUNDF in TRJCR0 Register) .....	413
7.5.3	Access to Counter Register .....	413
7.5.4	When Changing Mode .....	413
7.5.5	Procedure for Setting Pins TRJO0 and TRJIO0 .....	414
7.5.6	When Timer RJ is not Used .....	414
7.5.7	When Timer RJ Operating Clock is Stopped .....	414

7.5.8	Procedure for Setting STOP Mode (Event Counter Mode) .....	415
7.5.9	Functional Restriction in STOP Mode (Event Counter Mode Only) .....	415
7.5.10	When Count is Forcibly Stopped by TSTOP Bit .....	415
7.5.11	Digital Filter .....	415
7.5.12	When Selecting f <sub>IL</sub> as Count Source .....	415
8.	TIMER RD .....	416
8.1	Functions of Timer RD .....	416
8.2	Configuration of Timer RD .....	417
8.3	Registers Controlling Timer RD .....	418
8.3.1	Peripheral enable register 1 (PER1) .....	419
8.3.2	Timer RD ELC register (TRDELC) .....	420
8.3.3	Timer RD start register (TRDSTR) .....	421
8.3.4	Timer RD mode register (TRDMR) .....	422
8.3.5	Timer RD PWM function select register (TRDPMR) .....	423
8.3.6	Timer RD function control register (TRDFCR) .....	424
8.3.7	Timer RD output master enable register 1 (TRDOER1) .....	426
8.3.8	Timer RD output master enable register 2 (TRDOER2) .....	427
8.3.9	Timer RD output control register (TRDOCR) .....	428
8.3.10	Timer RD digital filter function select register i (TRDDFi) (i = 0 or 1) .....	432
8.3.11	Timer RD control register i (TRDCRi) (i = 0 or 1) .....	435
8.3.12	Timer RD I/O control register Ai (TRDIORAi) (i = 0 or 1) .....	440
8.3.13	Timer RD I/O control register Ci (TRDIORCi) (i = 0 or 1) .....	442
8.3.14	Timer RD status register 0 (TRDSR0) .....	444
8.3.15	Timer RD status register 1 (TRDSR1) .....	448
8.3.16	Timer RD interrupt enable register i (TRDIERi) (i = 0 or 1) .....	452
8.3.17	Timer RD PWM function output level control register i (TRDPOCRI) (i = 0 or 1) .....	453
8.3.18	Timer RD counter i (TRDi) (i = 0 or 1) .....	454
8.3.19	Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi) (i = 0 or 1) [Input Capture Function] .....	456
8.3.20	Port mode register 1 (PM1) .....	467
8.4	Items Common to Multiple Modes .....	468
8.4.1	Count Sources .....	468
8.4.2	Buffer Operation .....	469
8.4.3	Synchronous Operation .....	472
8.4.4	Pulse Output Forced Cutoff .....	473
8.4.5	Event Input from Event Link Controller (ELC) .....	475
8.4.6	Event Output to Event Link Controller (ELC)/Data Transfer Controller (DTC) .....	475
8.5	Timer RD Operation .....	476
8.5.1	Input Capture Function .....	476
8.5.2	Output Compare Function .....	481
8.5.3	PWM Function .....	487
8.5.4	Reset Synchronous PWM Mode .....	492
8.5.5	Complementary PWM Mode .....	496
8.5.6	PWM3 Mode .....	501
8.6	Timer RD Interrupt .....	505
8.7	Cautions for Timer RD .....	507
8.7.1	SFR Read/Write Access .....	507
8.7.2	Mode Switching .....	508

8.7.3	Count Source .....	508
8.7.4	Input Capture Function .....	508
8.7.5	Procedure for Setting Pins TRDIOAi, TRDIOBi, TRDIOC <sub>i</sub> , and TRDIOD <sub>i</sub> (i = 0 or 1) .....	509
8.7.6	External clock TRDCLK .....	509
8.7.7	Reset Synchronous PWM Mode .....	509
8.7.8	Complementary PWM Mode .....	510
9.	TIMER RG .....	514
9.1	Functions of Timer RG .....	514
9.2	Configuration of Timer RG .....	515
9.3	Registers Controlling Timer RG .....	516
9.3.1	Peripheral enable register 1 (PER1) .....	517
9.3.2	Timer RG mode register (TRGMR) .....	518
9.3.3	Timer RG count control register (TRGCNTC) .....	519
9.3.4	Timer RG control register (TRGCR) .....	520
9.3.5	Timer RG interrupt enable register (TRGIER) .....	521
9.3.6	Timer RG status register (TRGSR) .....	522
9.3.7	Timer RG I/O control register (TRGIOR) .....	524
9.3.8	Timer RG counter (TRG) .....	526
9.3.9	Timer RG general registers A, B, C, and D (TRGGRA, TRGGRB, TRGGRC, TRGGRD) .....	527
9.3.10	Port mode registers 0, 5 (PM0, PM5) .....	529
9.4	Timer RG Operation .....	530
9.4.1	Items Common to Multiple Modes and Functions .....	530
9.4.2	Timer Mode (Input Capture Function) .....	535
9.4.3	Timer Mode (Output Compare Function) .....	538
9.4.4	PWM Mode .....	542
9.4.5	Phase Counting Mode .....	546
9.5	Timer RG Interrupt .....	548
9.6	Cautions for Timer RG .....	550
9.6.1	Phase Difference, Overlap, and Pulse Width in Phase Counting Mode .....	550
9.6.2	Mode Switching .....	550
9.6.3	Count Source Switching .....	550
9.6.4	Procedure for Setting Pins TRGIOA and TRGIOB .....	551
9.6.5	External Clock TRGCLKA, TRGCLKB .....	551
9.6.6	SFR Read/Write Access .....	552
9.6.7	Input Capture Operation when Count is Stopped .....	552
10.	REAL-TIME CLOCK .....	553
10.1	Functions of Real-time Clock .....	553
10.2	Configuration of Real-time Clock .....	553
10.3	Registers Controlling Real-time Clock .....	555
10.3.1	Peripheral enable register 0 (PER0) .....	556
10.3.2	Subsystem clock supply mode control register (OSMC) .....	557
10.3.3	Real-time clock control register 0 (RTCC0) .....	558
10.3.4	Real-time clock control register 1 (RTCC1) .....	559
10.3.5	Second count register (SEC) .....	561
10.3.6	Minute count register (MIN) .....	561
10.3.7	Hour count register (HOUR) .....	562

10.3.8	Day count register (DAY) .....	564
10.3.9	Week count register (WEEK) .....	565
10.3.10	Month count register (MONTH) .....	566
10.3.11	Year count register (YEAR) .....	566
10.3.12	Watch error correction register (SUBCUD) .....	567
10.3.13	Alarm minute register (ALARMWM) .....	568
10.3.14	Alarm hour register (ALARMWH) .....	568
10.3.15	Alarm week register (ALARMWW) .....	568
10.3.16	Port mode register 3 (PM3) .....	569
10.3.17	Port register 3 (P3) .....	569
10.4	Real-time Clock Operation .....	570
10.4.1	Starting operation of real-time clock .....	570
10.4.2	Shifting to HALT/STOP mode after starting operation .....	571
10.4.3	Reading/writing real-time clock .....	572
10.4.4	Setting alarm of real-time clock .....	574
10.4.5	1 Hz output of real-time clock .....	575
10.4.6	Example of watch error correction of real-time clock .....	576
11.	12-BIT INTERVAL TIMER .....	581
11.1	Functions of 12-bit Interval Timer .....	581
11.2	Configuration of 12-bit Interval Timer .....	581
11.3	Registers Controlling 12-bit Interval Timer .....	582
11.3.1	Peripheral enable register 0 (PER0) .....	582
11.3.2	Subsystem clock supply mode control register (OSMC) .....	583
11.3.3	12-bit interval timer control register (ITMC) .....	584
11.4	12-bit Interval Timer Operation .....	585
11.4.1	12-bit interval timer operation timing .....	585
11.4.2	Start of count operation and re-enter to HALT/STOP mode after returned from HALT/STOP mode .....	586
12.	CLOCK OUTPUT/BUZZER OUTPUT CONTROLLER .....	587
12.1	Functions of Clock Output/Buzzer Output Controller .....	587
12.2	Configuration of Clock Output/Buzzer Output Controller .....	589
12.3	Registers Controlling Clock Output/Buzzer Output Controller .....	589
12.3.1	Clock output select registers n (CKSn) .....	589
12.3.2	Registers controlling port functions of pins to be used for clock or buzzer output .....	591
12.4	Operations of Clock Output/Buzzer Output Controller .....	592
12.4.1	Operation as output pin .....	592
12.5	Cautions of clock output/buzzer output controller .....	592
13.	WATCHDOG TIMER .....	593
13.1	Functions of Watchdog Timer .....	593
13.2	Configuration of Watchdog Timer .....	594
13.3	Register Controlling Watchdog Timer .....	595
13.3.1	Watchdog timer enable register (WDTE) .....	595
13.4	Operation of Watchdog Timer .....	596
13.4.1	Controlling operation of watchdog timer .....	596
13.4.2	Setting overflow time of watchdog timer .....	597
13.4.3	Setting window open period of watchdog timer .....	598

13.4.4	Setting watchdog timer interval interrupt .....	599
13.5	Restrictions on the Watchdog Timer .....	599
14.	A/D CONVERTER .....	600
14.1	Function of A/D Converter .....	600
14.2	Configuration of A/D Converter .....	603
14.3	Registers Controlling A/D Converter .....	605
14.3.1	Peripheral enable register 0 (PER0) .....	606
14.3.2	A/D converter mode register 0 (ADM0) .....	607
14.3.3	A/D converter mode register 1 (ADM1) .....	615
14.3.4	A/D converter mode register 2 (ADM2) .....	616
14.3.5	10-bit A/D conversion result register (ADCR) .....	618
14.3.6	8-bit A/D conversion result register (ADCRH) .....	618
14.3.7	Analog input channel specification register (ADS) .....	619
14.3.8	Conversion result comparison upper limit setting register (ADUL) .....	621
14.3.9	Conversion result comparison lower limit setting register (ADLL) .....	621
14.3.10	A/D test register (ADTES) .....	622
14.3.11	Registers controlling port function of analog input pins .....	623
14.4	A/D Converter Conversion Operations .....	624
14.5	Input Voltage and Conversion Results .....	626
14.6	A/D Converter Operation Modes .....	627
14.6.1	Software trigger mode (select mode, sequential conversion mode) .....	627
14.6.2	Software trigger mode (select mode, one-shot conversion mode) .....	628
14.6.3	Software trigger mode (scan mode, sequential conversion mode) .....	629
14.6.4	Software trigger mode (scan mode, one-shot conversion mode) .....	630
14.6.5	Hardware trigger no-wait mode (select mode, sequential conversion mode) .....	631
14.6.6	Hardware trigger no-wait mode (select mode, one-shot conversion mode) .....	632
14.6.7	Hardware trigger no-wait mode (scan mode, sequential conversion mode) .....	633
14.6.8	Hardware trigger no-wait mode (scan mode, one-shot conversion mode) .....	634
14.6.9	Hardware trigger wait mode (select mode, sequential conversion mode) .....	635
14.6.10	Hardware trigger wait mode (select mode, one-shot conversion mode) .....	636
14.6.11	Hardware trigger wait mode (scan mode, sequential conversion mode) .....	637
14.6.12	Hardware trigger wait mode (scan mode, one-shot conversion mode) .....	638
14.7	A/D Converter Setup Flowchart .....	639
14.7.1	Setting up software trigger mode .....	640
14.7.2	Setting up hardware trigger no-wait mode .....	641
14.7.3	Setting up hardware trigger wait mode .....	642
14.7.4	Setup when temperature sensor output voltage/internal reference voltage is selected (example for software trigger mode and one-shot conversion mode) .....	643
14.7.5	Setting up test mode .....	644
14.8	SNOOZE Mode Function .....	645
14.9	How to Read A/D Converter Characteristics Table .....	649
14.10	Cautions for A/D Converter .....	652
15.	D/A CONVERTER .....	656
15.1	Functions of D/A Converter .....	656
15.2	Configuration of D/A Converter .....	657
15.3	Registers Controlling D/A Converter .....	658
15.3.1	Peripheral enable register 1 (PER1) .....	658

15.3.2	D/A converter mode register (DAM) .....	659
15.3.3	D/A conversion value setting register i (DACS <sub>i</sub> ) ( $i = 0, 1$ ) .....	659
15.3.4	Event output destination select register n (ELSELR <sub>n</sub> ), $n = 00$ to 25 .....	660
15.3.5	Registers controlling port functions of analog input pins .....	660
15.4	Operations of D/A Converter .....	661
15.4.1	Operation in Normal Mode .....	661
15.4.2	Operation in Real-Time Output Mode .....	662
15.4.3	Timing for Outputting D/A Conversion Value .....	663
15.5	Cautions for D/A Converter .....	664
16.	COMPARATOR .....	665
16.1	Functions of Comparator .....	665
16.2	Configuration of Comparator .....	666
16.3	Registers Controlling Comparator .....	667
16.3.1	Peripheral enable register 1 (PER1) .....	667
16.3.2	Comparator mode setting register (COMPMDR) .....	668
16.3.3	Comparator filter control register (COMPFIR) .....	670
16.3.4	Comparator output control register (COMPOCR) .....	672
16.3.5	Registers controlling port functions of analog input pins .....	673
16.4	Operation .....	674
16.4.1	Comparator i Digital Filter ( $i = 0$ or 1) .....	677
16.4.2	Comparator i ( $i = 0$ or 1) Interrupts .....	677
16.4.3	Event signal output to event link controller (ELC) .....	678
16.4.4	Comparator i Output ( $i = 0$ or 1) .....	679
16.4.5	Stopping or Supplying Comparator Clock .....	679
17.	SERIAL ARRAY UNIT .....	680
17.1	Functions of Serial Array Unit .....	682
17.1.1	3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) .....	682
17.1.2	UART (UART0 to UART3) .....	683
17.1.3	Simplified I <sup>2</sup> C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) .....	684
17.2	Configuration of Serial Array Unit .....	685
17.2.1	Shift register .....	689
17.2.2	Lower 8/9 bits of the serial data register mn (SDRmn) .....	689
17.3	Registers Controlling Serial Array Unit .....	691
17.3.1	Peripheral enable register 0 (PER0) .....	692
17.3.2	Serial clock select register m (SPSm) .....	693
17.3.3	Serial mode register mn (SMRmn) .....	695
17.3.4	Serial communication operation setting register mn (SCRmn) .....	696
17.3.5	Serial data register mn (SDRmn) .....	699
17.3.6	Serial flag clear trigger register mn (SIRmn) .....	701
17.3.7	Serial status register mn (SSRmn) .....	702
17.3.8	Serial channel start register m (SSm) .....	704
17.3.9	Serial channel stop register m (STm) .....	705
17.3.10	Serial channel enable status register m (SEm) .....	706
17.3.11	Serial output enable register m (SOEm) .....	707
17.3.12	Serial output register m (SOm) .....	708
17.3.13	Serial output level register m (SOLm) .....	709
17.3.14	Serial standby control register m (SSCm) .....	711

17.3.15	Input switch control register (ISC) .....	712
17.3.16	Noise filter enable register 0 (NFEN0) .....	713
17.3.17	Registers controlling port functions of serial input/output pins .....	714
17.4	Operation Stop Mode .....	715
17.4.1	Stopping the operation by units .....	715
17.4.2	Stopping the operation by channels .....	716
17.5	Operation of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) Communication .....	717
17.5.1	Master transmission .....	720
17.5.2	Master reception .....	728
17.5.3	Master transmission/reception .....	736
17.5.4	Slave transmission .....	744
17.5.5	Slave reception .....	752
17.5.6	Slave transmission/reception .....	758
17.5.7	SNOOZE mode function .....	766
17.5.8	Calculating transfer clock frequency .....	770
17.5.9	Procedure for processing errors that occurred during 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) communication .....	772
17.6	Clock Synchronous Serial Communication with Slave Select Input Function .....	773
17.6.1	Slave transmission .....	777
17.6.2	Slave reception .....	787
17.6.3	Slave transmission/reception .....	794
17.6.4	Calculating transfer clock frequency .....	804
17.6.5	Procedure for processing errors that occurred during slave select input function communication .....	806
17.7	Operation of UART (UART0 to UART3) Communication .....	807
17.7.1	UART transmission .....	810
17.7.2	UART reception .....	819
17.7.3	SNOOZE mode function .....	826
17.7.4	Calculating baud rate .....	834
17.7.5	Procedure for processing errors that occurred during UART (UART0 to UART3) communication .....	838
17.8	LIN Communication Operation .....	839
17.8.1	LIN transmission .....	839
17.8.2	LIN reception .....	842
17.9	Operation of Simplified I <sup>2</sup> C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) Communication .....	847
17.9.1	Address field transmission .....	850
17.9.2	Data transmission .....	855
17.9.3	Data reception .....	858
17.9.4	Stop condition generation .....	862
17.9.5	Calculating transfer rate .....	863
17.9.6	Procedure for processing errors that occurred during simplified I <sup>2</sup> C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) communication .....	866
18.	SERIAL INTERFACE IICA .....	867
18.1	Functions of Serial Interface IICA .....	867
18.2	Configuration of Serial Interface IICA .....	870
18.3	Registers Controlling Serial Interface IICA .....	873
18.3.1	Peripheral enable register 0 (PER0) .....	874

18.3.2	IICA control register n0 (IICCTLn0) .....	874
18.3.3	IICA status register n (IICSn) .....	879
18.3.4	IICA flag register n (IICFn) .....	881
18.3.5	IICA control register n1 (IICCTLn1) .....	883
18.3.6	IICA low-level width setting register n (IICWL <sub>n</sub> ) .....	885
18.3.7	IICA high-level width setting register n (IICWH <sub>n</sub> ) .....	885
18.3.8	Port mode register 6 (PM6) .....	886
18.4	I <sup>2</sup> C Bus Mode Functions .....	887
18.4.1	Pin configuration .....	887
18.4.2	Setting transfer clock by using IICWL <sub>n</sub> and IICWH <sub>n</sub> registers .....	888
18.5	I <sup>2</sup> C Bus Definitions and Control Methods .....	890
18.5.1	Start conditions .....	890
18.5.2	Addresses .....	891
18.5.3	Transfer direction specification .....	891
18.5.4	Acknowledge (ACK) .....	892
18.5.5	Stop condition .....	893
18.5.6	Wait .....	894
18.5.7	Canceling wait .....	896
18.5.8	Interrupt request (INTIICAn) generation timing and wait control .....	897
18.5.9	Address match detection method .....	898
18.5.10	Error detection .....	898
18.5.11	Extension code .....	899
18.5.12	Arbitration .....	900
18.5.13	Wakeup function .....	902
18.5.14	Communication reservation .....	905
18.5.15	Cautions .....	909
18.5.16	Communication operations .....	910
18.5.17	Timing of I <sup>2</sup> C interrupt request (INTIICAn) occurrence .....	918
18.6	Timing Charts .....	939
19.	DATA TRANSFER CONTROLLER (DTC) .....	954
19.1	Functions of DTC .....	954
19.2	Configuration of DTC .....	956
19.3	Registers Controlling DTC .....	957
19.3.1	Allocation of DTC Control Data Area and DTC Vector Table Area .....	958
19.3.2	Control Data Allocation .....	959
19.3.3	Vector Table .....	961
19.3.4	Peripheral enable register 1 (PER1) .....	963
19.3.5	DTC control register j (DTCCRj) (j = 0 to 23) .....	964
19.3.6	DTC block size register j (DTBLSj) (j = 0 to 23) .....	965
19.3.7	DTC transfer count register j (DTCCTj) (j = 0 to 23) .....	965
19.3.8	DTC transfer count reload register j (DTRLDj) (j = 0 to 23) .....	966
19.3.9	DTC source address register j (DTSARj) (j = 0 to 23) .....	966
19.3.10	DTC destination address register j (DTDARj) (j = 0 to 23) .....	966
19.3.11	DTC activation enable register i (DTCENi) (i = 0 to 4) .....	967
19.3.12	DTC base address register (DTCBAR) .....	970
19.4	DTC Operation .....	970
19.4.1	Activation Sources .....	971
19.4.2	Normal Mode .....	972

19.4.3	Repeat Mode .....	975
19.4.4	Chain Transfers .....	979
19.5	Cautions for DTC .....	981
19.5.1	Setting DTC Control Data and Vector Table .....	981
19.5.2	Allocation of DTC Control Data Area and DTC Vector Table Area .....	981
19.5.3	DTC Pending Instruction .....	982
19.5.4	Operation when Accessing Data Flash Memory Space .....	982
19.5.5	Number of DTC Execution Clock Cycles .....	983
19.5.6	DTC Response Time .....	984
19.5.7	DTC Activation Sources .....	984
19.5.8	Operation in Standby Mode Status .....	985
20.	EVENT LINK CONTROLLER (ELC) .....	986
20.1	Functions of ELC .....	986
20.2	Configuration of ELC .....	986
20.3	Registers Controlling ELC .....	987
20.3.1	Event output destination select register n (ELSELRn) (n = 00 to 25) .....	988
20.4	ELC Operation .....	991
21.	INTERRUPT FUNCTIONS .....	993
21.1	Interrupt Function Types .....	993
21.2	Interrupt Sources and Configuration .....	993
21.3	Registers Controlling Interrupt Functions .....	1000
21.3.1	Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H) .....	1004
21.3.2	Interrupt mask flag registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H) .....	1006
21.3.3	Priority specification flag registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H) .....	1008
21.3.4	External interrupt rising edge enable registers (EGP0, EGP1), external interrupt falling edge enable registers (EGN0, EGN1) .....	1011
21.3.5	Program status word (PSW) .....	1013
21.4	Interrupt Servicing Operations .....	1014
21.4.1	Maskable interrupt request acknowledgment .....	1014
21.4.2	Software interrupt request acknowledgment .....	1017
21.4.3	Multiple interrupt servicing .....	1017
21.4.4	Interrupt servicing during division instruction .....	1021
21.4.5	Interrupt request hold .....	1023
22.	KEY INTERRUPT FUNCTION .....	1024
22.1	Functions of Key Interrupt .....	1024
22.2	Configuration of Key Interrupt .....	1024
22.3	Register Controlling Key Interrupt .....	1026
22.3.1	Key return mode register (KRM) .....	1026
22.3.2	Port mode register 7 (PM7) .....	1027
23.	STANDBY FUNCTION .....	1028
23.1	Standby Function .....	1028
23.2	Registers controlling standby function .....	1029
23.3	Standby Function Operation .....	1030
23.3.1	HALT mode .....	1030

23.3.2 STOP mode .....	1035
23.3.3 SNOOZE mode .....	1040
<b>24. RESET FUNCTION .....</b>	<b>1044</b>
24.1 Timing of Reset Operation .....	1046
24.2 Register for Confirming Reset Source .....	1050
24.2.1 Reset control flag register (RESF) .....	1050
<b>25. POWER-ON-RESET CIRCUIT .....</b>	<b>1052</b>
25.1 Functions of Power-on-reset Circuit .....	1052
25.2 Configuration of Power-on-reset Circuit .....	1053
25.3 Operation of Power-on-reset Circuit .....	1053
<b>26. VOLTAGE DETECTOR .....</b>	<b>1057</b>
26.1 Functions of Voltage Detector .....	1057
26.2 Configuration of Voltage Detector .....	1058
26.3 Registers Controlling Voltage Detector .....	1058
26.3.1 Voltage detection register (LVIM) .....	1059
26.3.2 Voltage detection level register (LVIS) .....	1060
26.4 Operation of Voltage Detector .....	1063
26.4.1 When used as reset mode .....	1063
26.4.2 When used as interrupt mode .....	1065
26.4.3 When used as interrupt and reset mode .....	1067
26.5 Cautions for Voltage Detector .....	1073
<b>27. SAFETY FUNCTIONS .....</b>	<b>1075</b>
27.1 Overview of Safety Functions .....	1075
27.2 Registers Used by Safety Functions .....	1076
27.3 Operation of Safety Functions .....	1076
27.3.1 Flash memory CRC operation function (high-speed CRC) .....	1076
27.3.1.1 Flash memory CRC control register (CRC0CTL) .....	1077
27.3.1.2 Flash memory CRC operation result register (PGCRCL) .....	1078
27.3.2 CRC operation function (general-purpose CRC) .....	1080
27.3.2.1 CRC input register (CRCIN) .....	1081
27.3.2.2 CRC data register (CRCD) .....	1082
27.3.3 RAM parity error detection function .....	1083
27.3.3.1 RAM parity error control register (RPECTL) .....	1083
27.3.4 RAM guard function .....	1085
27.3.4.1 Invalid memory access detection control register (IAWCTL) .....	1085
27.3.5 SFR guard function .....	1086
27.3.5.1 Invalid memory access detection control register (IAWCTL) .....	1086
27.3.6 Invalid memory access detection function .....	1087
27.3.6.1 Invalid memory access detection control register (IAWCTL) .....	1088
27.3.7 Frequency detection function .....	1089
27.3.7.1 Timer input select register 0 (TIS0) .....	1090
27.3.8 A/D test function .....	1091
27.3.8.1 A/D test register (ADTES) .....	1093
27.3.8.2 Analog input channel specification register (ADS) .....	1094
27.3.9 Digital output signal level detection function for I/O pins .....	1096

27.3.9.1 Port mode select register (PMS) .....	1096
28. REGULATOR .....	1097
28.1 Regulator Overview .....	1097
29. OPTION BYTE .....	1098
29.1 Functions of Option Bytes .....	1098
29.1.1 User option byte (000C0H to 000C2H/010C0H to 010C2H) .....	1098
29.1.2 On-chip debug option byte (000C3H/ 010C3H) .....	1099
29.2 Format of User Option Byte .....	1100
29.3 Format of On-chip Debug Option Byte .....	1106
29.4 Setting of Option Byte .....	1107
30. FLASH MEMORY .....	1108
30.1 Serial Programming Using Flash Memory Programmer .....	1109
30.1.1 Programming Environment .....	1111
30.1.2 Communication Mode .....	1111
30.2 Serial Programming Using External Device (that Incorporates UART) .....	1112
30.2.1 Programming Environment .....	1112
30.2.2 Communication Mode .....	1113
30.3 Connection of Pins on Board .....	1114
30.3.1 P40/TOOL0 pin .....	1114
30.3.2 RESET pin .....	1114
30.3.3 Port pins .....	1115
30.3.4 REGC pin .....	1115
30.3.5 X1 and X2 pins .....	1115
30.3.6 Power supply .....	1115
30.4 Programming Method .....	1116
30.4.1 Serial programming procedure .....	1116
30.4.2 Flash memory programming mode .....	1117
30.4.3 Selecting communication mode .....	1119
30.4.4 Communication commands .....	1120
30.5 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value) .....	1122
30.6 Self-Programming .....	1123
30.6.1 Self-programming procedure .....	1124
30.6.2 Boot swap function .....	1125
30.6.3 Flash shield window function .....	1127
30.7 Security Settings .....	1128
30.8 Data Flash .....	1130
30.8.1 Data flash overview .....	1130
30.8.2 Register controlling data flash memory .....	1131
30.8.2.1 Data flash control register (DFLCTL) .....	1131
30.8.3 Procedure for accessing data flash memory .....	1132
31. ON-CHIP DEBUG FUNCTION .....	1133
31.1 Connecting E1 On-chip Debugging Emulator .....	1133
31.2 On-Chip Debug Security ID .....	1134
31.3 Securing of User Resources .....	1134

32. BCD CORRECTION CIRCUIT .....	1136
32.1 BCD Correction Circuit Function .....	1136
32.2 Registers Used by BCD Correction Circuit .....	1136
32.2.1 BCD correction result register (BCDADJ) .....	1136
32.3 BCD Correction Circuit Operation .....	1137
33. INSTRUCTION SET .....	1139
33.1 Conventions Used in Operation List .....	1140
33.1.1 Operand identifiers and specification methods .....	1140
33.1.2 Description of operation column .....	1141
33.1.3 Description of flag operation column .....	1142
33.1.4 PREFIX instruction .....	1142
33.2 Operation List .....	1143
34. ELECTRICAL SPECIFICATIONS (TA = -40 to +85°C) .....	1161
34.1 Absolute Maximum Ratings .....	1162
34.2 Oscillator Characteristics .....	1164
34.2.1 X1, XT1 characteristics .....	1164
34.2.2 On-chip oscillator characteristics .....	1164
34.3 DC Characteristics .....	1165
34.3.1 Pin characteristics .....	1165
34.3.2 Supply current characteristics .....	1170
34.4 AC Characteristics .....	1184
34.5 Peripheral Functions Characteristics .....	1191
34.5.1 Serial array unit .....	1191
34.5.2 Serial interface IICA .....	1219
34.6 Analog Characteristics .....	1223
34.6.1 A/D converter characteristics .....	1223
34.6.2 Temperature sensor characteristics/internal reference voltage characteristic .....	1227
34.6.3 D/A converter characteristics .....	1227
34.6.4 Comparator .....	1228
34.6.5 POR circuit characteristics .....	1228
34.6.6 LVD circuit characteristics .....	1229
34.6.7 Power supply voltage rising slope characteristics .....	1230
34.7 RAM Data Retention Characteristics .....	1231
34.8 Flash Memory Programming Characteristics .....	1231
34.9 Dedicated Flash Memory Programmer Communication (UART) .....	1231
34.10 Timing of Entry to Flash Memory Programming Modes .....	1232
35. ELECTRICAL SPECIFICATIONS (G: INDUSTRIAL APPLICATIONS TA = -40 to +105°C) .....	1233
35.1 Absolute Maximum Ratings .....	1235
35.2 Oscillator Characteristics .....	1237
35.2.1 X1, XT1 characteristics .....	1237
35.2.2 On-chip oscillator characteristics .....	1237
35.3 DC Characteristics .....	1238
35.3.1 Pin characteristics .....	1238
35.3.2 Supply current characteristics .....	1243
35.4 AC Characteristics .....	1257
35.5 Peripheral Functions Characteristics .....	1263

35.5.1	Serial array unit .....	1263
35.5.2	Serial interface IICA .....	1285
35.6	Analog Characteristics .....	1286
35.6.1	A/D converter characteristics .....	1286
35.6.2	Temperature sensor characteristics/internal reference voltage characteristic .....	1290
35.6.3	D/A converter characteristics .....	1290
35.6.4	Comparator .....	1291
35.6.5	POR circuit characteristics .....	1291
35.6.6	LVD circuit characteristics .....	1292
35.6.7	Power supply voltage rising slope characteristics .....	1293
35.7	RAM Data Retention Characteristics .....	1294
35.8	Flash Memory Programming Characteristics .....	1294
35.9	Dedicated Flash Memory Programmer Communication (UART) .....	1294
35.10	Timing of Entry to Flash Memory Programming Modes .....	1295
36.	PACKAGE DRAWINGS .....	1296
36.1	30-pin Package .....	1296
36.2	32-pin Package .....	1297
36.3	36-pin Package .....	1300
36.4	40-pin Package .....	1301
36.5	44-pin Package .....	1303
36.6	48-pin Package .....	1305
36.7	52-pin Package .....	1309
36.8	64-pin Package .....	1310
36.9	80-pin Package .....	1315
36.10	100-pin Package .....	1318
APPENDIX A	REVISION HISTORY .....	1321
A.1	Major Revisions in This Edition .....	1321
A.2	Revision History of Preceding Editions .....	1322

## CHAPTER 1 OUTLINE

### 1.1 Features

Ultra-low power consumption technology

- VDD = single power supply voltage of 1.6 to 5.5 V which can operate a 1.8 V device at a low voltage
- HALT mode
- STOP mode
- SNOOZE mode

RL78 CPU core

- CISC architecture with 3-stage pipeline
- Minimum instruction execution time: Can be changed from high speed (0.03125  $\mu$ s: @ 32 MHz operation with high-speed on-chip oscillator) to ultra-low speed (30.5  $\mu$ s: @ 32.768 kHz operation with subsystem clock)
- Multiply/divide/multiply & accumulate instructions are supported.
- Address space: 1 MB
- General-purpose registers: (8-bit register  $\times$  8)  $\times$  4 banks
- On-chip RAM: 2.5 to 48 KB

Code flash memory

- Code flash memory: 16 to 512 KB
- Block size: 1 KB
- Prohibition of block erase and rewriting (security function)
- On-chip debug function
- Self-programming (with boot swap function/flash shield window function)

Data flash memory

- Data flash memory: 4 KB and 8 KB
- Back ground operation (BGO): Instructions can be executed from the program memory while rewriting the data flash memory.
- Number of rewrites: 1,000,000 times (TYP.)
- Voltage of rewrites: VDD = 1.8 to 5.5 V

High-speed on-chip oscillator

- Select from 64 MHz, 48 MHz, 32 MHz, 24 MHz, 16 MHz, 12 MHz, 8 MHz, 6 MHz, 4 MHz, 3 MHz, 2 MHz, and 1 MHz
- High accuracy:  $\pm 1.0\%$  (VDD = 1.8 to 5.5 V, TA = -20 to +85°C)

Operating ambient temperature

- TA = -40 to +85°C (A: Consumer applications, D: Industrial applications)
- TA = -40 to +105°C (G: Industrial applications)

Power management and reset function

- On-chip power-on-reset (POR) circuit
- On-chip voltage detector (LVD) (Select interrupt and reset from 14 levels)

Data transfer controller (DTC)

- Transfer modes: Normal transfer mode, repeat transfer mode, block transfer mode
- Activation sources: Activated by interrupt sources.
- Chain transfer function

Event link controller (ELC)

- Event signals of 19 to 26 types can be linked to the specified peripheral function.

Serial interfaces

- CSI: 3 to 8 channels
- UART/UART (LIN-bus supported): 3 or 4 channels
- I<sup>2</sup>C/simplified I<sup>2</sup>C: 3 to 8 channels

Timer

- 16-bit timer: 8 to 12 channels  
(Timer Array Unit (TAU): 4 to 8 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)
- 12-bit interval timer: 1 channel
- Real-time clock: 1 channel (calendar for 99 years, alarm function, and clock correction function)
- Watchdog timer: 1 channel (operable with the dedicated low-speed on-chip oscillator)

A/D converter

- 8/10-bit resolution A/D converter (V<sub>DD</sub> = 1.6 to 5.5 V)
- Analog input: 8 to 20 channels
- Internal reference voltage (1.45 V) and temperature sensor

D/A converter

- 8-bit resolution D/A converter (V<sub>DD</sub> = 1.6 to 5.5 V)
- Analog output: None or up to two channels
- Output voltage: 0 V to V<sub>DD</sub>
- Real-time output function

Comparator

- None or up to two channels
- Operating modes: Comparator high-speed mode, comparator low-speed mode, window mode
- The external reference voltage or internal reference voltage can be selected as the reference voltage.

### I/O port

- I/O port: 26 to 92 (N-ch open drain I/O [withstand voltage of 6 V]: 2 to 4, N-ch open drain I/O [V<sub>DD</sub> withstand voltage/EV<sub>DD</sub> withstand voltage]: 10 to 28)
- Can be set to N-ch open drain, TTL input buffer, and on-chip pull-up resistor
- Different potential interface: Can connect to a 1.8/2.5/3 V device
- On-chip key interrupt function
- On-chip clock output/buzzer output controller

### Others

- On-chip BCD (binary-coded decimal) correction circuit

**Remark** The functions mounted depend on the product. See **1.6 Outline of Functions**.

○ ROM, RAM capacities

Flash ROM	Data flash	RAM	RL78/G14			
			30 pins	32 pins	36 pins	40 pins
192 KB	8 KB	20 KB	—	—	—	R5F104EH
128 KB	8 KB	16 KB	R5F104AG	R5F104BG	R5F104CG	R5F104EG
96 KB	8 KB	12 KB	R5F104AF	R5F104BF	R5F104CF	R5F104EF
64 KB	4 KB	5.5 KB Note	R5F104AE	R5F104BE	R5F104CE	R5F104EE
48 KB	4 KB	5.5 KB Note	R5F104AD	R5F104BD	R5F104CD	R5F104ED
32 KB	4 KB	4 KB	R5F104AC	R5F104BC	R5F104CC	R5F104EC
16 KB	4 KB	2.5 KB	R5F104AA	R5F104BA	R5F104CA	R5F104EA

Flash ROM	Data flash	RAM	RL78/G14			
			44 pins	48 pins	52 pins	64 pins
512 KB	8 KB	48 KB Note	—	R5F104GL	—	R5F104LL
384 KB	8 KB	32 KB	—	R5F104GK	—	R5F104LK
256 KB	8 KB	24 KB Note	R5F104FJ	R5F104GJ	R5F104JJ	R5F104LJ
192 KB	8 KB	20 KB	R5F104FH	R5F104GH	R5F104JH	R5F104LH
128 KB	8 KB	16 KB	R5F104FG	R5F104GG	R5F104JG	R5F104LG
96 KB	8 KB	12 KB	R5F104FF	R5F104GF	R5F104JF	R5F104LF
64 KB	4 KB	5.5 KB Note	R5F104FE	R5F104GE	R5F104JE	R5F104LE
48 KB	4 KB	5.5 KB Note	R5F104FD	R5F104GD	R5F104JD	R5F104LD
32 KB	4 KB	4 KB	R5F104FC	R5F104GC	R5F104JC	R5F104LC
16 KB	4 KB	2.5 KB	R5F104FA	R5F104GA	—	—

Flash ROM	Data flash	RAM	RL78/G14	
			80 pins	100 pins
512 KB	8 KB	48 KB Note	R5F104ML	R5F104PL
384 KB	8 KB	32 KB	R5F104MK	R5F104PK
256 KB	8 KB	24 KB Note	R5F104MJ	R5F104PJ
192 KB	8 KB	20 KB	R5F104MH	R5F104PH
128 KB	8 KB	16 KB	R5F104MG	R5F104PG
96 KB	8 KB	12 KB	R5F104MF	R5F104PF

**Note** The flash library uses RAM in self-programming and rewriting of the data flash memory.

The target products and start address of the RAM areas used by the flash library are shown below.

R5F104xD (x = A to C, E to G, J, L): Start address FE900H

R5F104xE (x = A to C, E to G, J, L): Start address FE900H

R5F104xJ (x = F, G, J, L, M, P): Start address F9F00H

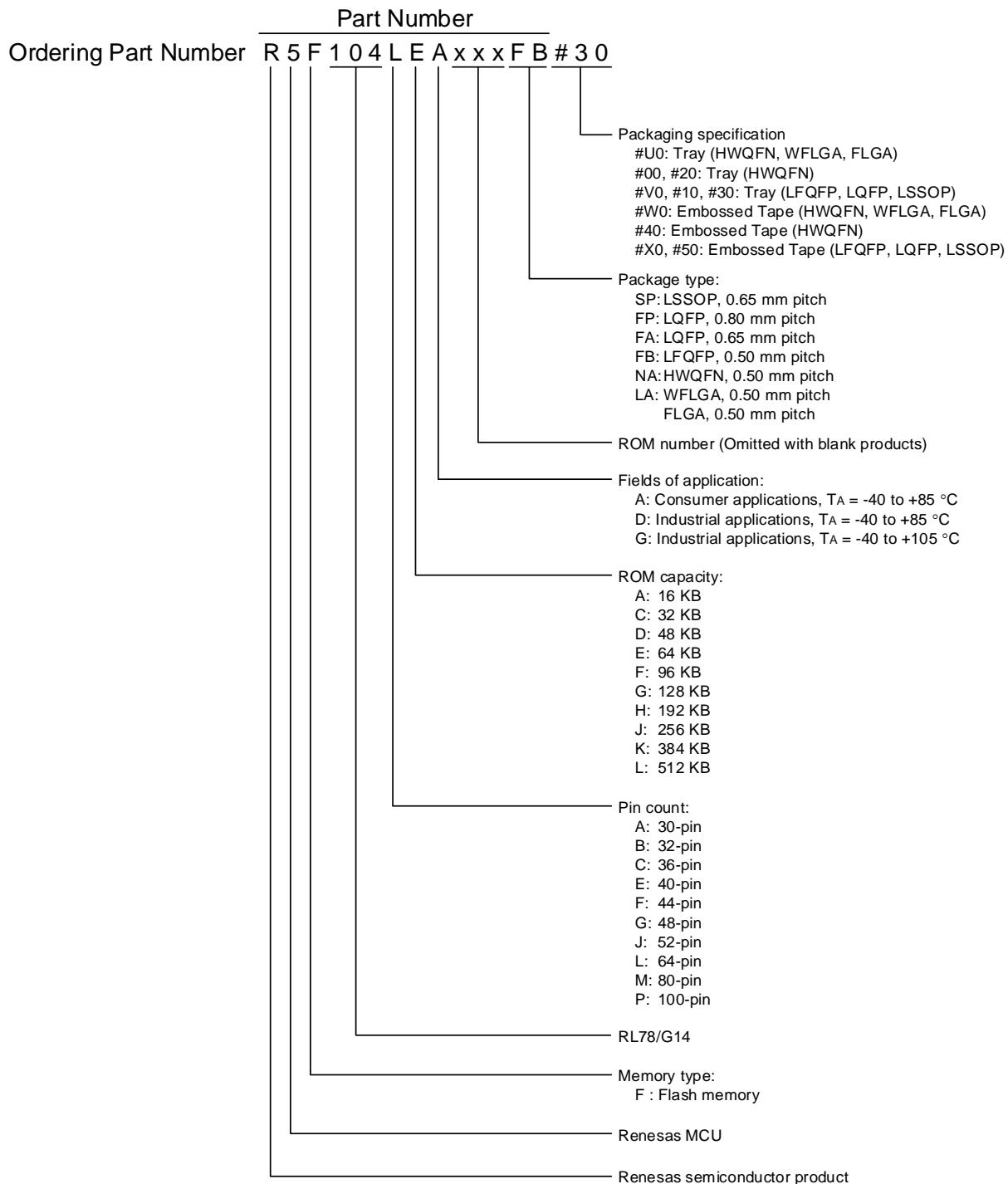
R5F104xL (x = G, L, M, P): Start address F3F00H

For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

## 1.2 Ordering Information

&lt;R&gt;

**Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14**



&lt;R&gt;

**Table 1 - 1 List of Ordering Part Numbers (1/4)**

Pin count	Package	Fields of Application Note	Ordering Part Number		RENESAS Code
			Part Number	Packaging specification	
30 pins	30-pin plastic LSSOP (7.62 mm (300), 0.65 mm pitch)	A	R5F104AAASP, R5F104ACASP, R5F104ADASP, R5F104AEASP, R5F104AFASP, R5F104AGASP	#V0, #10, #30, #X0, #50	PLSP0030JB-B
		D	R5F104AADSP, R5F104ACDSP, R5F104ADDSP, R5F104AEDSP, R5F104AFDSP, R5F104AGDSP		
		G	R5F104AAGSP, R5F104ACGSP, R5F104ADGSP, R5F104AEGSP, R5F104AFGSP, R5F104AGGSP		
32 pins	32-pin plastic HWQFN (5 × 5 mm, 0.5 mm pitch)	A	R5F104BAANA, R5F104BCANA, R5F104BDANA, R5F104BEANA, R5F104BFANA, R5F104BGANA	#U0, #W0	PWQN0032KB-A
		D	R5F104BADNA, R5F104BCDNA, R5F104BDDNA, R5F104BEDNA, R5F104BFDNA, R5F104BGDNA	#U0, #W0	PWQN0032KB-A
		G	R5F104BAGNA, R5F104BCGNA, R5F104BDGNA, R5F104BEGNA, R5F104BFGNA, R5F104BGGNA	#U0, #W0 #00, #20, #40	PWQN0032KB-A PWQN0032KE-A
	32-pin plastic LQFP (7 × 7, 0.8 mm pitch)	A	R5F104BAAFP, R5F104BCAfp, R5F104BDAFP, R5F104BEAfp, R5F104BFAfp, R5F104BGAfp	#V0, #10, #30, #X0, #50	PLQP0032GB-A
		D	R5F104BADFP, R5F104BCDFP, R5F104BDDFP, R5F104BEDFP, R5F104BFDfp, R5F104BGDFP		
		G	R5F104BAGFP, R5F104BCGFP, R5F104BDGFP, R5F104BEGFP, R5F104BFGFP, R5F104BGGFP		
36 pins	36-pin plastic WFLGA (4 × 4 mm, 0.5 mm pitch)	A	R5F104CAALA, R5F104CCALA, R5F104CDALA, R5F104CEALA, R5F104CFALA, R5F104CGALA	#U0, #W0	PWLG0036KA-A
		G	R5F104CAGLA, R5F104CCGLA, R5F104CDGLA, R5F104CEGLA, R5F104CFGGLA, R5F104CGGLA		
40 pins	40-pin plastic HWQFN (6 × 6 mm, 0.5 mm pitch)	A	R5F104EAANA, R5F104ECANA, R5F104EDANA, R5F104EEANA, R5F104EFANA, R5F104EGANA, R5F104EHANA	#U0, #W0 #00, #20, #40	PWQN0040KC-A PWQN0040KD-A
		D	R5F104EADNA, R5F104ECDNA, R5F104EDDNA, R5F104EEDNA, R5F104EFDNA, R5F104EGDNA, R5F104EHDNA	#U0, #W0	PWQN0040KC-A
		G	R5F104EAGNA, R5F104ECGNA, R5F104EDGNA, R5F104EEGNA, R5F104EFGNA, R5F104EGGNA, R5F104EHGNA	#U0, #W0 #00, #20, #40	PWQN0040KC-A PWQN0040KD-A
	44-pin plastic LQFP (10 × 10, 0.8 mm pitch)	A	R5F104FAAfp, R5F104FC AFP, R5F104FDAFP, R5F104FEAfp, R5F104FF AFP, R5F104FG AFP, R5F104FH AFP, R5F104FJAfp	#V0, #X0 #10, #30, #50	PLQP0044GC-A PLQP0044GC-A/ PLQP0044GC-D
		D	R5F104FADFP, R5F104FCDFP, R5F104FDDFP, R5F104FEDFP, R5F104FFDFP, R5F104FGDFP, R5F104FHDFP, R5F104FJDFP	#V0, #X0 #10, #30, #50	PLQP0044GC-A PLQP0044GC-A/ PLQP0044GC-D
		G	R5F104FAGFP, R5F104FCGFP, R5F104FDGFP, R5F104FEGFP, R5F104FFGFP, R5F104FGGFP, R5F104FHGFP, R5F104FJGFP	#V0, #X0 #10, #30, #50	PLQP0044GC-A PLQP0044GC-A/ PLQP0044GC-D

**Note** For the fields of application, refer to **Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14**.

**Caution** The ordering part numbers represent the numbers at the time of publication. For the latest ordering part numbers, refer to the target product page of the Renesas Electronics website.

&lt;R&gt;

**Table 1 - 1 List of Ordering Part Numbers (2/4)**

Pin count	Package	Fields of Application Note	Ordering Part Number		RENESAS Code
			Part Number	Packaging specification	
48 pins	48-pin plastic LFQFP (7 × 7 mm, 0.5 mm pitch)	A	R5F104GKAFB, R5F104GLAFB	#10, #30, #50	PLQP0048KB-B
			R5F104GAAB, R5F104GCAF, R5F104GDAFB, R5F104GEAFB, R5F104GFAB, R5F104GGAFB, R5F104GHAFB, R5F104GJAFB	#V0, #X0	PLQP0048KF-A
				#10, #30, #50	PLQP0048KB-B
		D	R5F104GADFB, R5F104GCDFB, R5F104GDDFB, R5F104GEDFB, R5F104GFDFB, R5F104GGDFB, R5F104GHDFB, R5F104GJDFB	#V0, #X0	PLQP0048KF-A
				#10, #30, #50	PLQP0048KB-B
		G	R5F104GKGFB, R5F104GLGFB	#10, #30, #50	PLQP0048KB-B
			R5F104GAGFB, R5F104GCGFB, R5F104GDGFB, R5F104GEGFB, R5F104GFGFB, R5F104GGGFB, R5F104GHGFB, R5F104GJGFB	#V0, #X0	PLQP0048KF-A
				#10, #30, #50	PLQP0048KB-B
		A	R5F104GAANA, R5F104GCANA, R5F104GDANA, R5F104GEANA, R5F104GFANA, R5F104GGANA, R5F104GHANA, R5F104GJANA, R5F104GKANA, R5F104GLANA	#U0, #W0	PWQN0048KB-A
				#00, #20, #40	PWQN0048KE-A
		D	R5F104GADNA, R5F104GCDNA, R5F104GDDNA, R5F104GEDNA, R5F104GFDNA, R5F104GGDNA, R5F104GHDNA, R5F104GJDNA	#U0, #W0	PWQN0048KB-A
		G	R5F104GAGNA, R5F104GCGNA, R5F104GDGNA, R5F104GEGNA, R5F104GFGNA, R5F104GGGNA, R5F104GHGNA, R5F104GJGNA, R5F104GKGNNA, R5F104GLGNA	#U0, #W0	PWQN0048KB-A
				#00, #20, #40	PWQN0048KE-A
52 pins	52-pin plastic LQFP (10 × 10 mm, 0.65 mm pitch)	A	R5F104JCAFA, R5F104JDAFA, R5F104JEFAA, R5F104JFAFA, R5F104JGFAA, R5F104JHFAA, R5F104JJFAA	#V0, #10, #30, #X0, #50	PLQP0052JA-A
		D	R5F104JCDFA, R5F104JDDFA, R5F104JEDFA, R5F104JFDFA, R5F104JGDFA, R5F104JHDFA, R5F104JJDDFA		
		G	R5F104JCGFA, R5F104JDGFA, R5F104JEGFA, R5F104JFGFA, R5F104JGGFA, R5F104JHGFA, R5F104JJGFA		
64 pins	64-pin plastic LQFP (12 × 12 mm, 0.65 mm pitch)	A	R5F104LKAFA, R5F104LLAFA	#10, #30, #50	PLQP0064JA-A
			R5F104LCAFA, R5F104LDAFA, R5F104LEAFA, R5F104LFAFA, R5F104LGAFA, R5F104LHFAA, R5F104LJFAA	#V0, #10, #30, #X0, #50	
		D	R5F104LCDFA, R5F104LDDFA, R5F104LEDFA, R5F104LFdfa, R5F104LGdfa, R5F104LHDfa, R5F104LJDfa		
		G	R5F104LKGFA, R5F104LLGFA	#10, #30, #50	
			R5F104LCGFA, R5F104LDGFA, R5F104LEGFA, R5F104LFGFA, R5F104LGGFA, R5F104LHGFA, R5F104LJGFA	#V0, #10, #30, #X0, #50	

**Note** For the fields of application, refer to **Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14**.

**Caution** The ordering part numbers represent the numbers at the time of publication. For the latest ordering part numbers, refer to the target product page of the Renesas Electronics website.

&lt;R&gt;

**Table 1 - 1 List of Ordering Part Numbers (3/4)**

Pin count	Package	Fields of Application Note	Ordering Part Number		RENESAS Code
			Part Number	Packaging specification	
64 pins	64-pin plastic LFQFP (10 × 10 mm, 0.5 mm pitch)	A	R5F104LKAFB, R5F104LLAFB	#10, #30, #50	PLQP0064KB-C
			R5F104LCABF, R5F104LDAFB, R5F104LEAFB, R5F104LFAFB, R5F104LGAFB, R5F104LHAFB, R5F104LJAFB	#V0, #X0	PLQP0064KF-A
				#10, #30, #50	PLQP0064KB-C
		D	R5F104LCDFB, R5F104LDDFB, R5F104LEDFB, R5F104LFDFB, R5F104LGDFB, R5F104LHDFB, R5F104LJDFB	#V0, #X0	PLQP0064KF-A
				#10, #30, #50	PLQP0064KB-C
		G	R5F104LKGFB, R5F104LLGFB	#10, #30, #50	PLQP0064KB-C
			R5F104LCGFB, R5F104LDGFB, R5F104LEGFB, R5F104LFGFB, R5F104LGGFB, R5F104LHGFB, R5F104LJGFB	#V0, #X0	PLQP0064KF-A
				#10, #30, #50	PLQP0064KB-C
	64-pin plastic FLGA (5 × 5 mm, 0.5 mm pitch)	A	R5F104LCALA, R5F104LDALA, R5F104LEALA, R5F104LFALA, R5F104LGALA, R5F104LHALA, R5F104LJALA, R5F104LKALA, R5F104LLALA	#U0, #W0	PWLG0064KA-A
		G	R5F104LCGLA, R5F104LDGLA, R5F104LEGLA, R5F104LFGLA, R5F104LGGLA, R5F104LHGLA, R5F104LJGLA, R5F104LKGLA, R5F104LLGLA		
	64-pin plastic LQFP (14 × 14 mm, 0.8 mm pitch)	A	R5F104LCAP, R5F104LDAFP, R5F104LEAFP, R5F104LFAFP, R5F104LG AFP, R5F104LHAFP, R5F104LJA FP	#V0, #10, #30, #X0, #50	PLQP0064GA-A
		D	R5F104LCDFP, R5F104LDDFP, R5F104LEDFP, R5F104LFDFP, R5F104LGDFP, R5F104LHDFP, R5F104LJDFP		
		G	R5F104LCGFP, R5F104LDGFP, R5F104LEGFP, R5F104LFGFP, R5F104LGGFP, R5F104LHGFP, R5F104LJGFP		
80 pins	80-pin plastic LFQFP (12 × 12 mm, 0.5 mm pitch)	A	R5F104MKAFB, R5F104MLAFB	#10, #30, #50	PLQP0080KB-B
			R5F104MFAFB, R5F104MGAFB, R5F104MHAFB, R5F104MJAFB	#V0, #X0	PLQP0080KE-A
				#10, #30, #50	PLQP0080KB-B
		D	R5F104MFDFB, R5F104MGDFB, R5F104MHDFB, R5F104MJDFB	#V0, #X0	PLQP0080KE-A
				#10, #30, #50	PLQP0080KB-B
		G	R5F104MKGFB, R5F104MLGFB	#10, #30, #50	PLQP0080KB-B
			R5F104MFGFB, R5F104MGGFB, R5F104MHGFB, R5F104MJGFB	#V0, #X0	PLQP0080KE-A
				#10, #30, #50	PLQP0080KB-B
	80-pin plastic LQFP (14 × 14 mm, 0.65 mm pitch)	A	R5F104MKAFA, R5F104MLAFA	#10, #30, #50	PLQP0080JB-E
			R5F104MFAFA, R5F104MGAFA, R5F104MHAFA, R5F104MJAFA	#V0, #10, #30, #X0, #50	
		D	R5F104MF DFA, R5F104MG DFA, R5F104MH DFA, R5F104MJ DFA	#V0, #10, #30, #X0, #50	
		G	R5F104MKGFA, R5F104MLGFA	#10, #30, #50	
			R5F104MFGFA, R5F104MGGFA, R5F104MHGFA, R5F104MJGFA	#V0, #10, #30, #X0, #50	

**Note** For the fields of application, refer to **Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14**.

**Caution** The ordering part numbers represent the numbers at the time of publication. For the latest ordering part numbers, refer to the target product page of the Renesas Electronics website.

&lt;R&gt;

**Table 1 - 1 List of Ordering Part Numbers (4/4)**

Pin count	Package	Fields of Application Note	Ordering Part Number		RENESAS Code
			Part Number	Packaging specification	
100 pins	100-pin plastic LFQFP (14 × 14 mm, 0.5 mm pitch)	A	R5F104PKAFB, R5F104PLAFB	#10, #30, #50	PLQP0100KB-B
			R5F104PFAFB, R5F104PGAFB, R5F104PHAFB, R5F104PJAFB	#V0, #X0 #10, #30, #50	PLQP0100KE-A PLQP0100KB-B
		D	R5F104PFDFB, R5F104PGDFB, R5F104PHDFB, R5F104PJDFB	#V0, #X0 #10, #30, #50	PLQP0100KE-A PLQP0100KB-B
			R5F104PKGFB, R5F104PLGFB	#10, #30, #50	PLQP0100KB-B
		G	R5F104PFGFB, R5F104PGGFB, R5F104PHGFB, R5F104PJGFB	#V0, #X0 #10, #30, #50	PLQP0100KE-A PLQP0100KB-B
	100-pin plastic LQFP (14 × 20 mm, 0.65 mm pitch)	A	R5F104PKAFA, R5F104PLAFA	#10, #30, #50	PLQP0100JC-A
			R5F104PFAFA, R5F104PGAFA, R5F104PHFAFA, R5F104PJFAFA	#V0, #10, #30, #X0, #50	
		D	R5F104PF DFA, R5F104PG DFA, R5F104PH DFA, R5F104PJ DFA	#V0, #10, #30, #X0, #50	
			R5F104PKGFA, R5F104PLGFA	#10, #30, #50	
		G	R5F104PFGFA, R5F104PGGFA, R5F104PHGFA, R5F104PJGFA	#V0, #10, #30, #X0, #50	

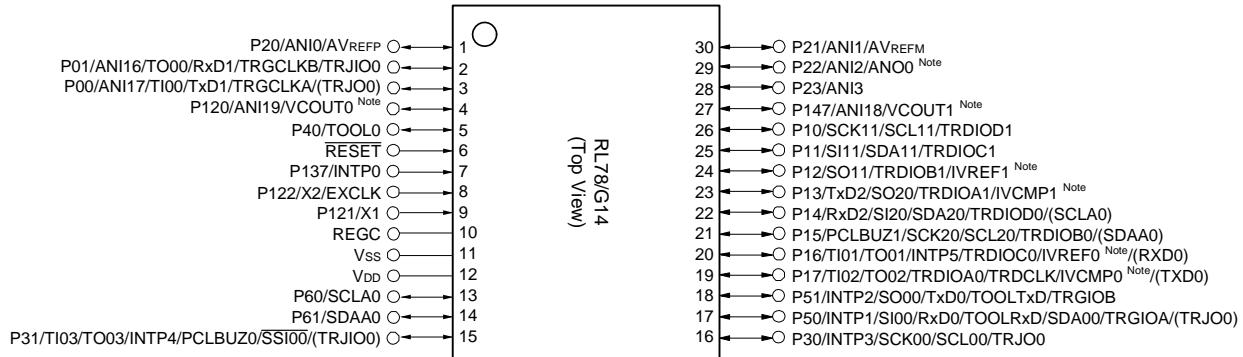
**Note** For the fields of application, refer to **Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14**.

**Caution** The ordering part numbers represent the numbers at the time of publication. For the latest ordering part numbers, refer to the target product page of the Renesas Electronics website.

## 1.3 Pin Configuration (Top View)

### 1.3.1 30-pin products

- 30-pin plastic LSSOP (7.62 mm (300), 0.65 mm pitch)



**Note** Mounted on the 96 KB or more code flash memory products.

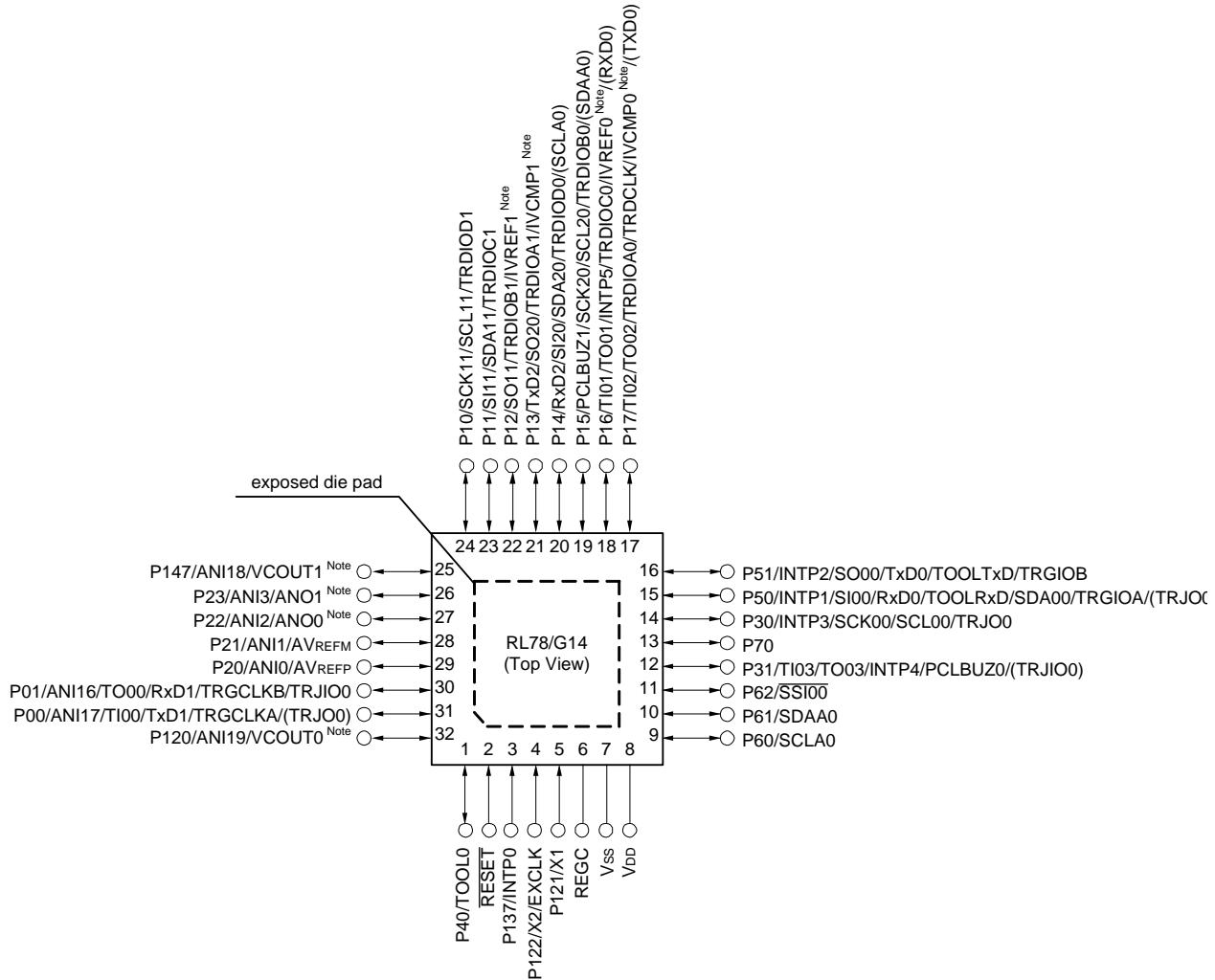
**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.2 32-pin products

- 32-pin plastic HWQFN (5 × 5 mm, 0.5 mm pitch)



**Note** Mounted on the 96 KB or more code flash memory products.

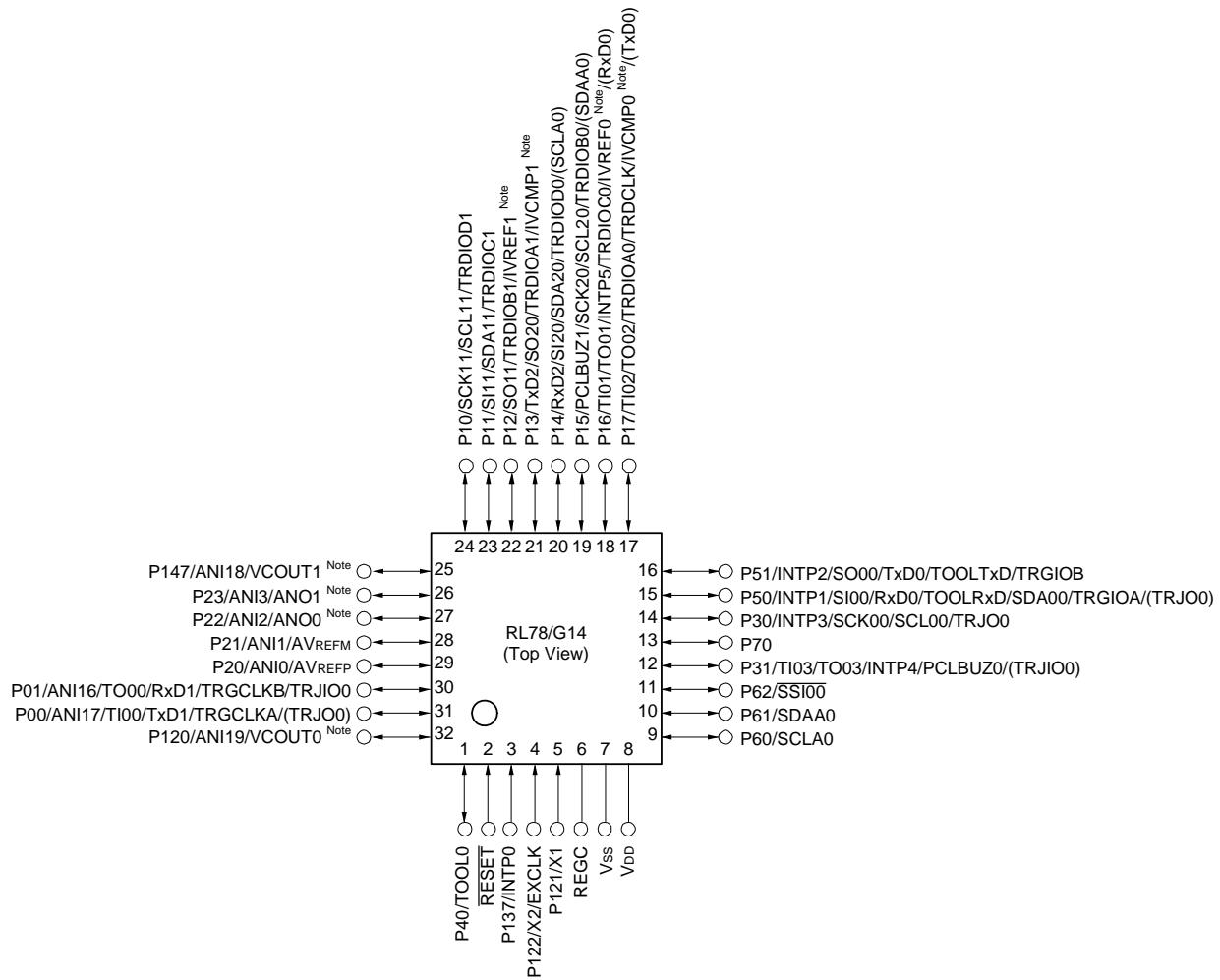
**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

**Remark 3.** It is recommended to connect an exposed die pad to Vss.

- 32-pin plastic LQFP ( $7 \times 7$  mm, 0.8 mm pitch)



**Note** Mounted on the 96 KB or more code flash memory products.

**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

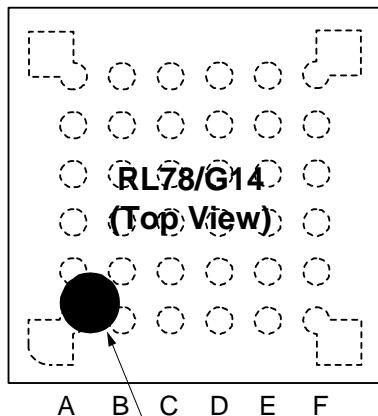
**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

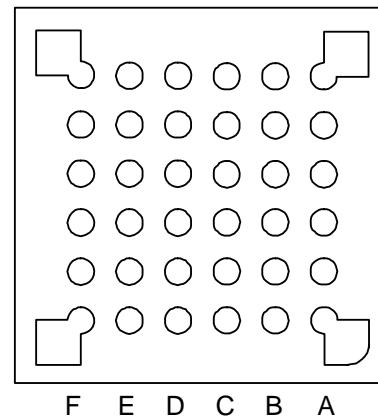
### 1.3.3 36-pin products

- 36-pin plastic WFLGA (4 × 4 mm, 0.5 mm pitch)

Top View



Bottom View



INDEX MARK

	A	B	C	D	E	F	
6	P60/SCLA0	VDD	P121/X1	P122/X2/EXCLK	P137/INTP0	P40/TOOL0	6
5	P62/SSI00	P61/SDAA0	Vss	REGC	RESET	P120/ANI19/ VCOUT0 Note	5
4	P72/SO21	P71/SI21/ SDA21	P14/RxD2/SI20/ SDA20/TRDIO0/ (SCLA0)	P31/TI03/TO03/ INTP4/PCLBUZ0/ (TRJ00)	P00/TI00/TxD1/ TRGCLKA/ (TRJ00)	P01/TO00/ RxD1/TRGCLKB/ (TRJ00)	4
3	P50/INTP1/ SI00/RxD0/ TOOLRxD/ SDA00/TRGIOA/ (TRJ00)	P70/SCK21/ SCL21	P15/PCLBUZ1/ SCK20/SCL20/ TRDIOB0/ (SDAA0)	P22/ANI2/ ANO0 Note	P20/ANI0/ AVREFP	P21/ANI1/ AVREFM	3
2	P30/INTP3/ SCK00/SCL00/ TRJ00	P16/TI01/TO01/ INTP5/TRDIOC0/ IVREF0 Note/ (RXD0)	P12/SO11/ TRDIOB1/ IVREF1 Note	P11/SI11/ SDA11/ TRDIOC1	P24/ANI4	P23/ANI3/ ANO1 Note	2
1	P51/INTP2/ SO00/TxD0/ TOOLTxD/ TRGIOB	P17/TI02/TO02/ TRDIOA0/ TRDCLK/ IVCMP0 Note/ (TXD0)	P13/TxD2/ SO20/TRDIOA1/ IVCMP1 Note	P10/SCK11/ SCL11/ TRDIOD1	P147/ANI18/ VCOUT1 Note	P25/ANI5	1

**Note** Mounted on the 96 KB or more code flash memory products.

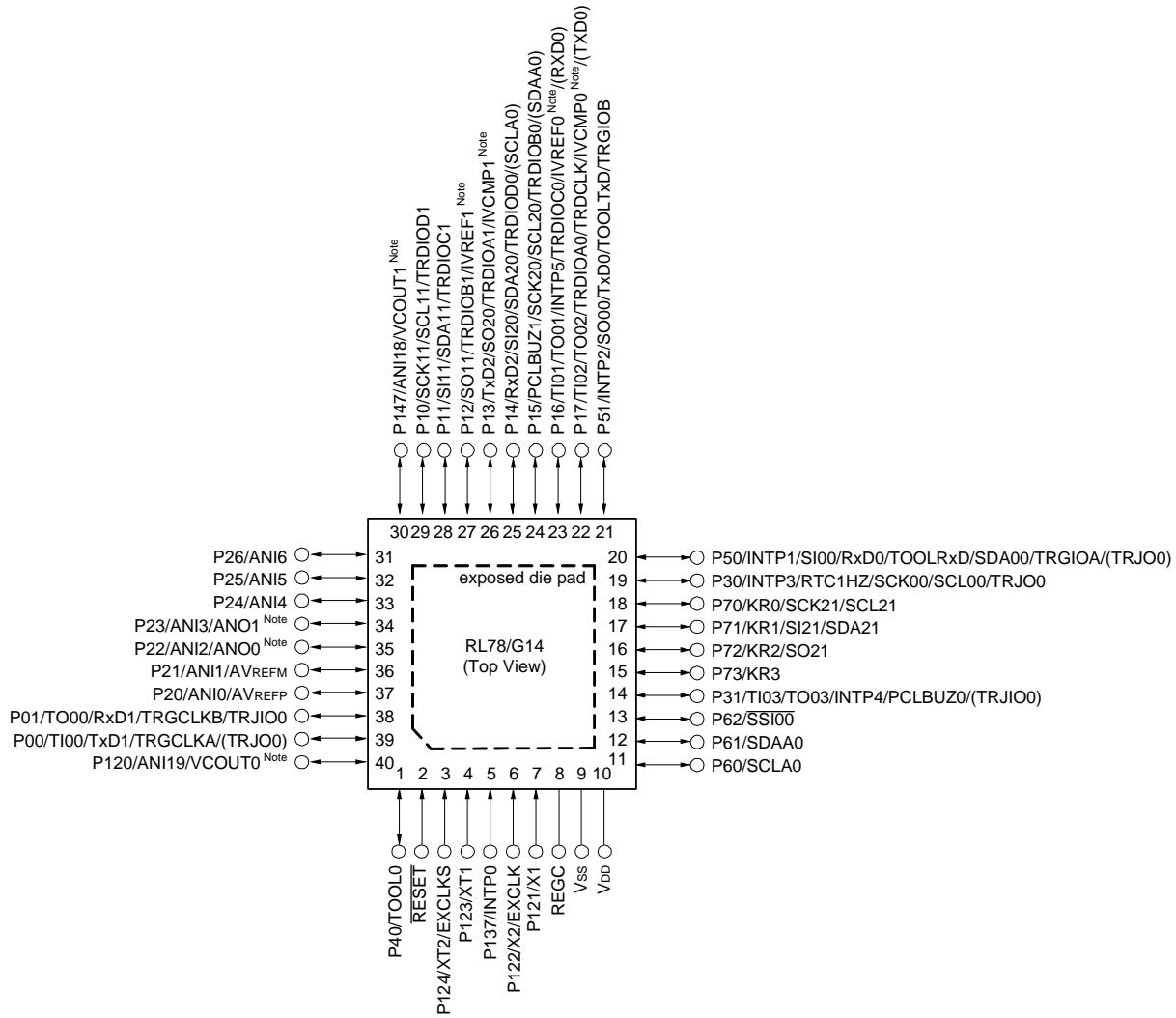
**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.4 40-pin products

- 40-pin plastic HWQFN (6 × 6 mm, 0.5 mm pitch)



**Note** Mounted on the 96 KB or more code flash memory products.

**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

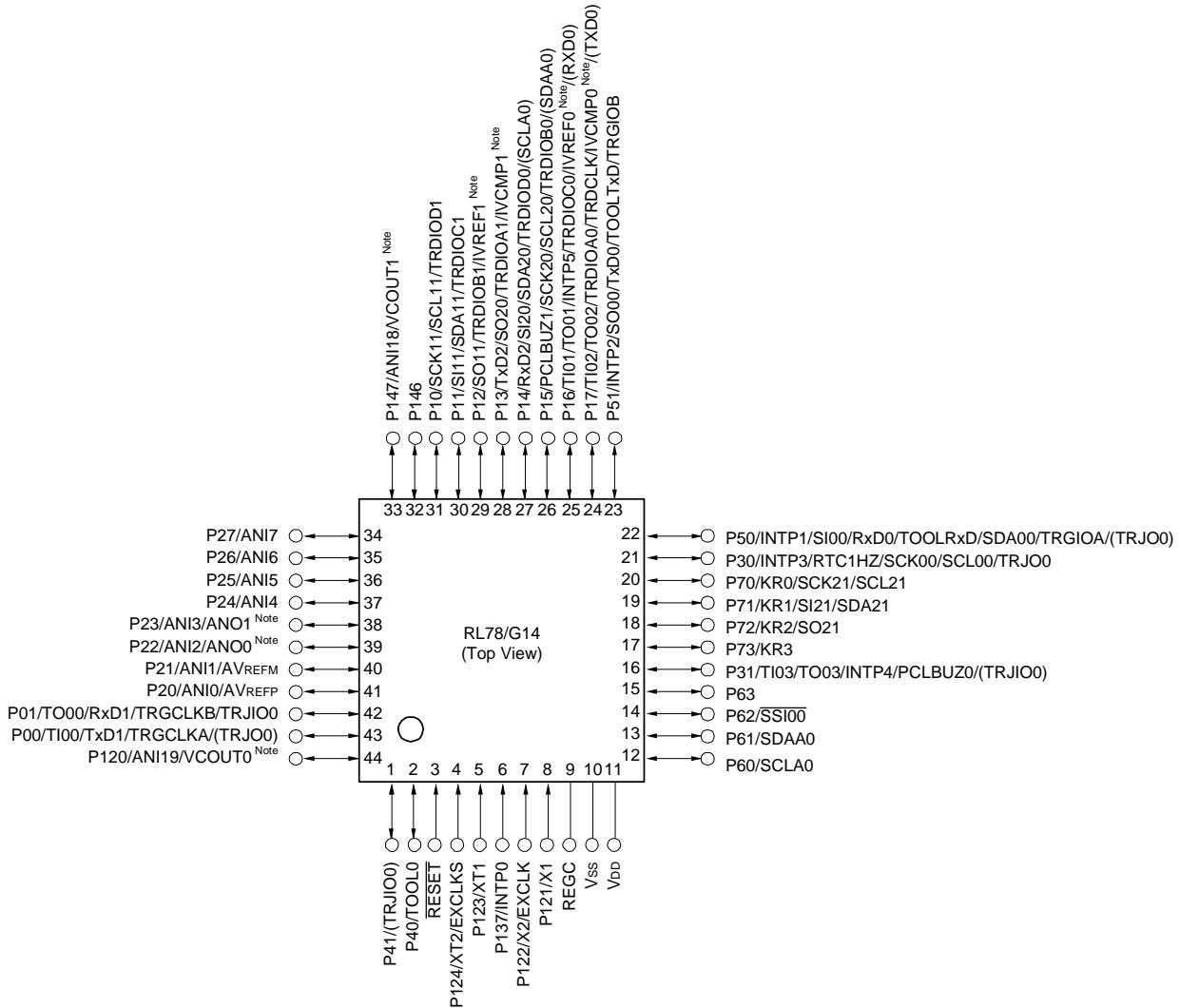
**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

**Remark 3.** It is recommended to connect an exposed die pad to Vss.

### 1.3.5 44-pin products

- 44-pin plastic LQFP (10 × 10 mm, 0.8 mm pitch) v



**Note** Mounted on the 96 KB or more code flash memory products.

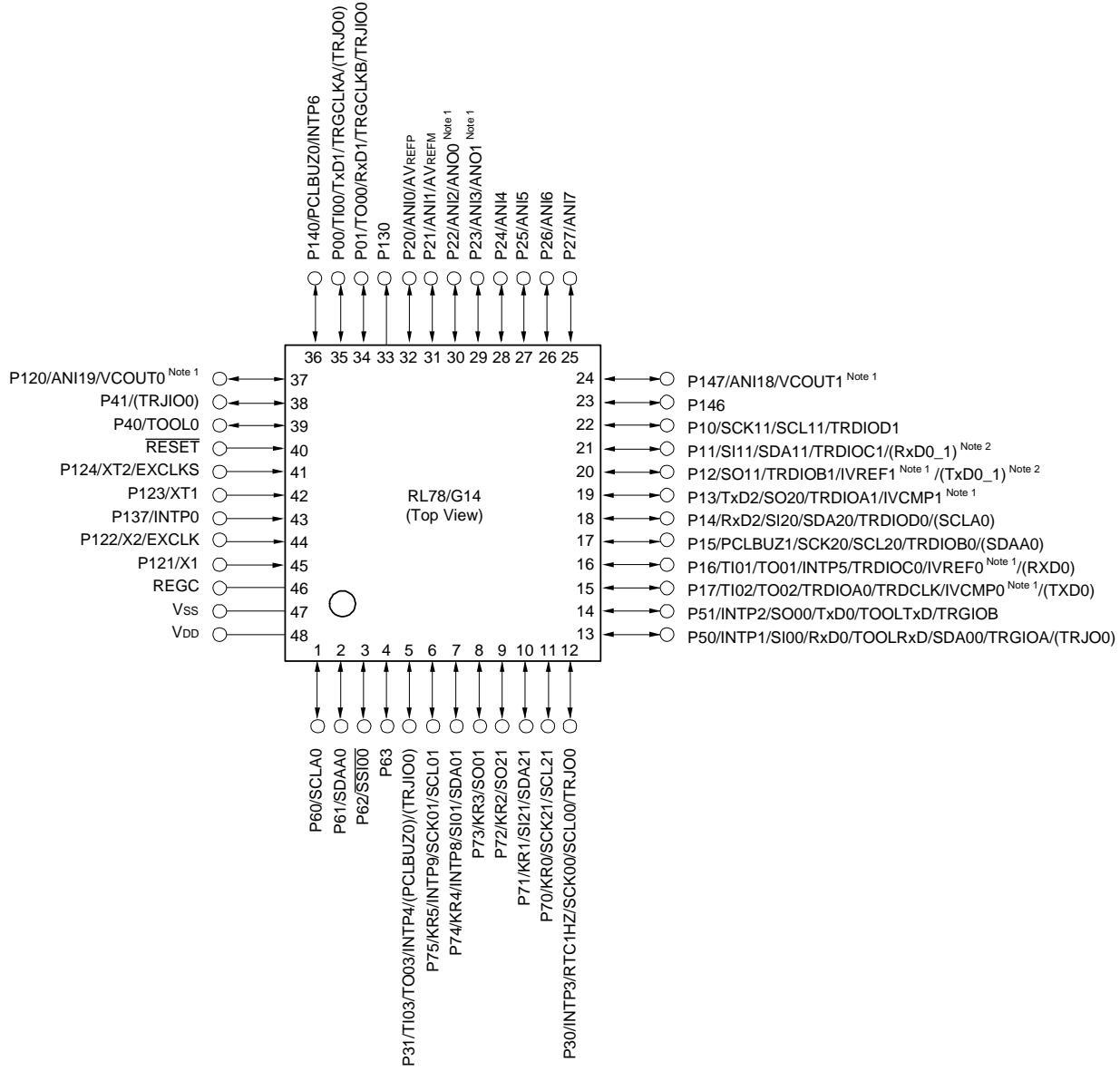
**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.6 48-pin products

- 48-pin plastic LFQFP (7 × 7 mm, 0.5 mm pitch)



**Note 1.** Mounted on the 96 KB or more code flash memory products.

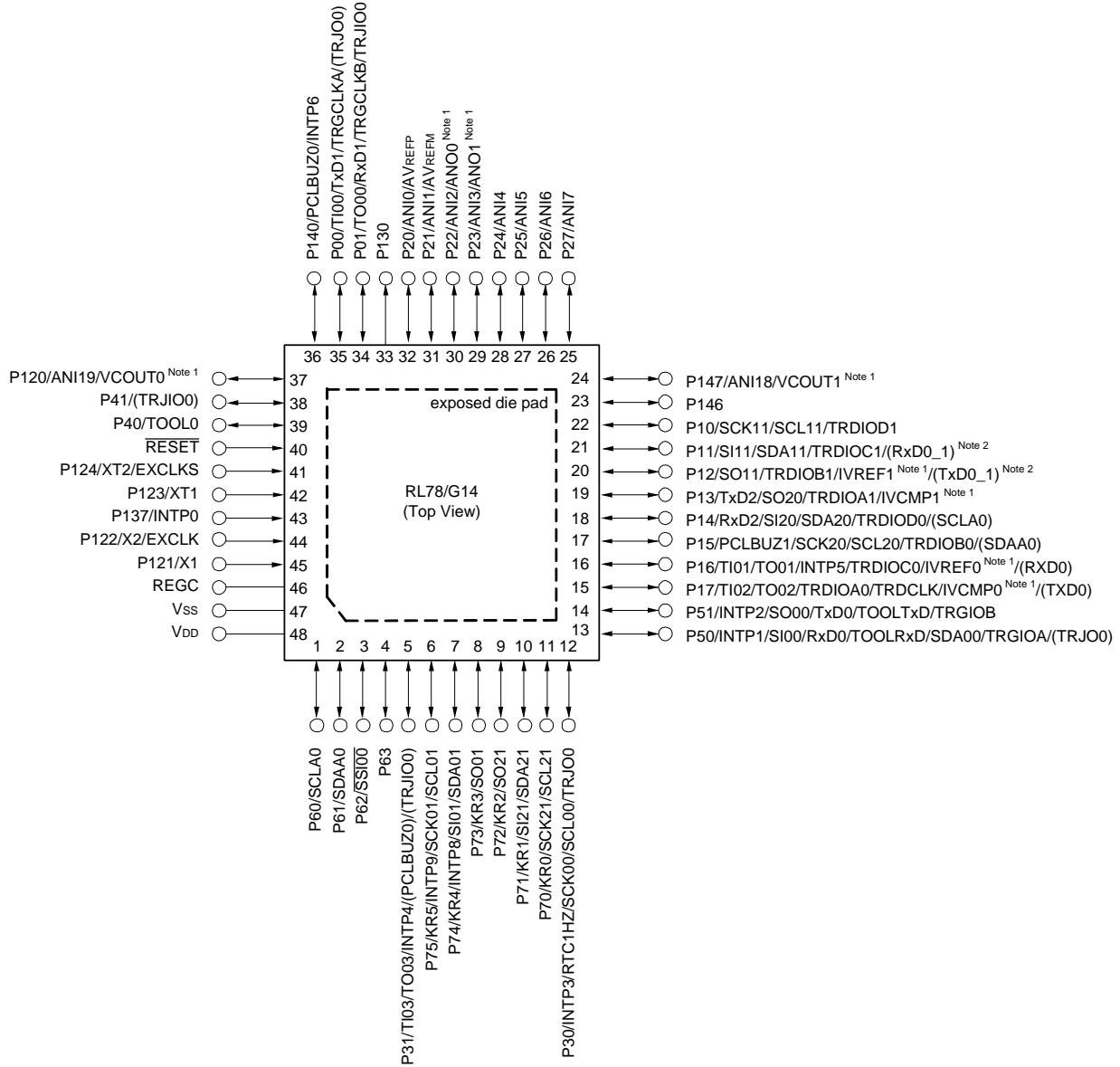
**Note 2.** Mounted on the 384 KB or more code flash memory products.

**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

- 48-pin plastic HWQFN ( $7 \times 7$  mm, 0.5 mm pitch)



**Note 1.** Mounted on the 96 KB or more code flash memory products.

**Note 2.** Mounted on the 384 KB or more code flash memory products.

**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

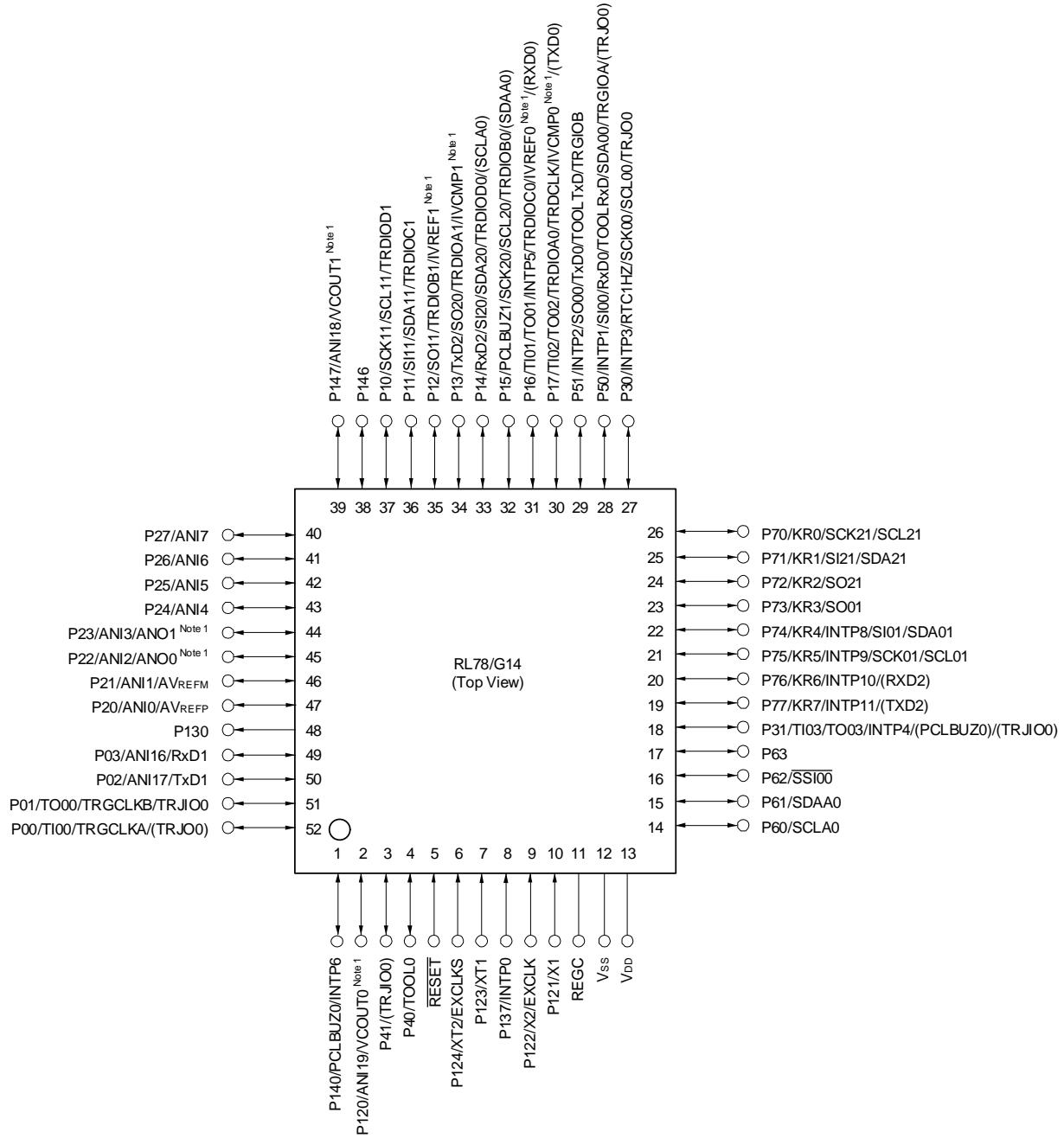
**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

**Remark 3.** It is recommended to connect an exposed die pad to Vss.

### 1.3.7 52-pin products

- 52-pin plastic LQFP (10 × 10 mm, 0.65 mm pitch)



**Note 1.** Mounted on the 96 KB or more code flash memory products.

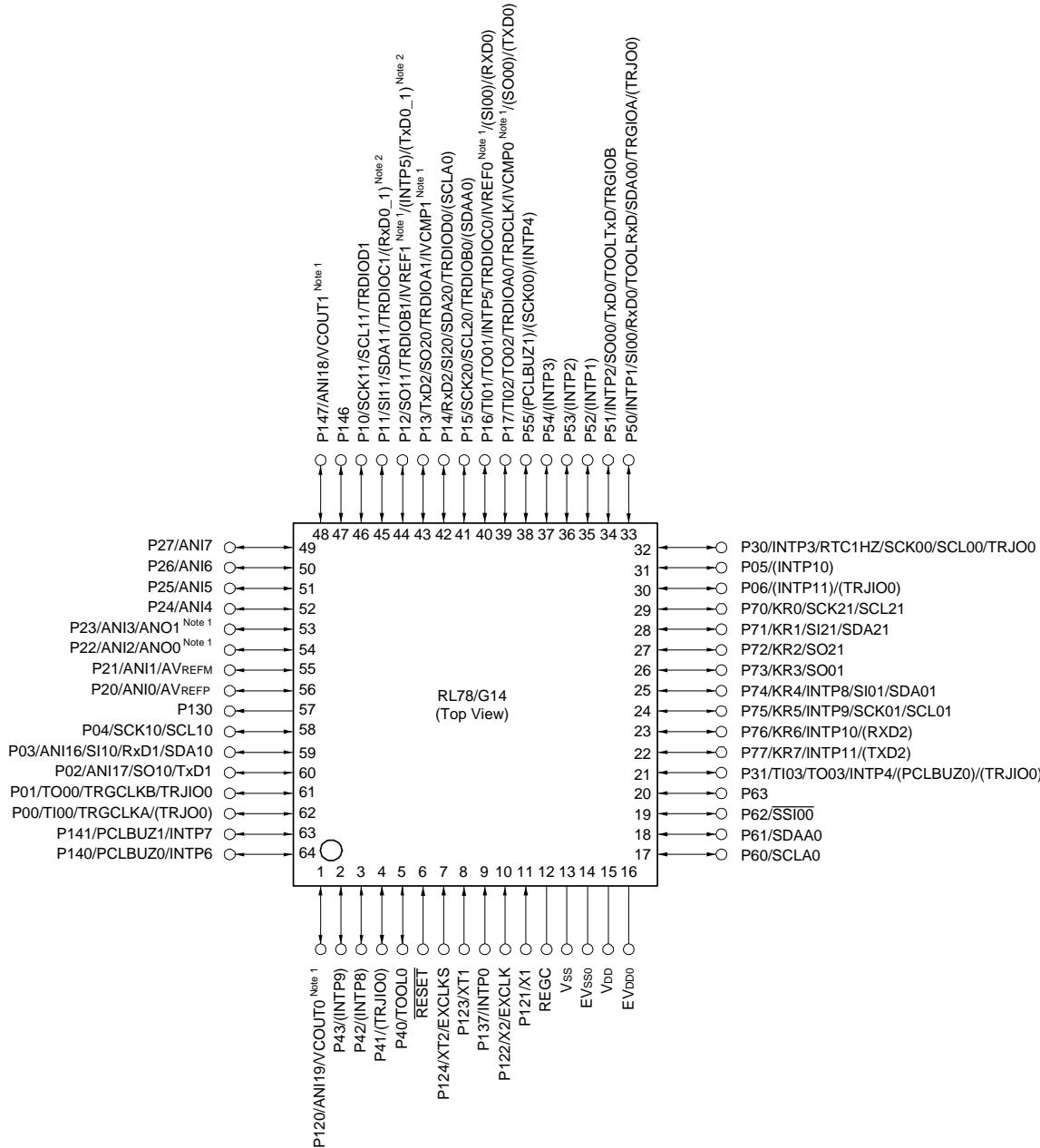
**Caution** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.8 64-pin products

- 64-pin plastic LQFP (14 × 14 mm, 0.8 mm pitch)
- 64-pin plastic LQFP (12 × 12 mm, 0.65 mm pitch)
- 64-pin plastic LFQFP (10 × 10 mm, 0.5 mm pitch)



**Note 1.** Mounted on the 96 KB or more code flash memory products.

**Note 2.** Mounted on the 384 KB or more code flash memory products.

**Caution 1. Make EVSS0 pin the same potential as Vss pin.**

**Caution 2. Make VDD pin the potential that is higher than EVDD0 pin.**

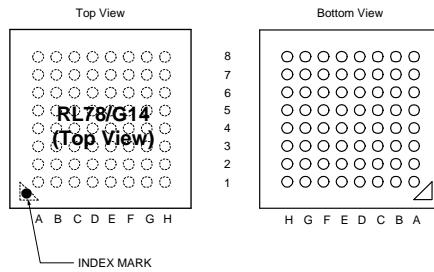
**Caution 3. Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).**

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** When using the microcontroller for an application where the noise generated inside the microcontroller must be reduced, it is recommended to supply separate powers to the VDD and EVDD0 pins and connect the Vss and EVSS0 pins to separate ground lines.

**Remark 3.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

- 64-pin plastic FLGA ( $5 \times 5$  mm, 0.5 mm pitch)



	A	B	C	D	E	F	G	H
8	EV <sub>DD0</sub>	EV <sub>SS0</sub>	P121/X1	P122/X2/ EXCLK	P137/INTP0	P123/XT1	P124/XT2/ EXCLKS	P120/ANI19/ VCOUT0 Note 1
7	P60/SCLA0	V <sub>DD</sub>	V <sub>ss</sub>	REGC	RESET	P01/T000/ TRGCLKB/ TRJIO0	P00/TI00/ TRGCLKA/ (TRJIO0)	P140/ PCLBUZ0/ INTP6
6	P61/SDAA0	P62/SSI00	P63	P40/TOOL0	P41/(TRJIO0)	P43/(INTP9)	P02/ANI17/ SO10/TxD1	P141/ PCLBUZ1/ INTP7
5	P77/KR7/ INTP11/(TXD2)	P31/TI03/ TO03/INTP4/ (PCLBUZ0)/ (TRJIO0)	P53/(INTP2)	P42/(INTP8)	P03/ANI16/ SI10/RxD1/ SDA10	P04/SCK10/ SCL10	P130	P20/ANI0/ AVREFP
4	P75/KR5/ INTP9/ SCK01/ SCL01	P76/KR6/ INTP10/ (RXD2)	P52/(INTP1)	P54/(INTP3)	P16/TI01/ TO01/INTP5/ TRDIOC0/ IVREF0 Note 1/ (SI00)/(RXD0)	P21/ANI1/ AVREFM	P22/ANI2/ ANO0 Note 1	P23/ANI3/ ANO1 Note 1
3	P70/KR0/ SCK21/ SCL21	P73/KR3/ SO01	P74/KR4/ INTP8/SI01/ SDA01	P17/TI02/TO02/ TRDIOAO/ TRDCLK/ IVCMP0 Note 1/ (SO00)/(TXD0)	P15/SCK20/ SCL20/ TRDIOB0/ (SDAA0)	P12/SO11/ TRDIOB1/ IVREF1 Note 1/ (INTP5)/ (TxDO_1) Note 2	P24/ANI4	P26/ANI6
2	P30/INTP3/ RTC1HZ/ SCK00/ SCL00/TRJIO0	P72/KR2/ SO21	P71/KR1/ SI21/SDA21	P06/(INTP11)/ (TRJIO0)	P14/RxD2/ SI20/SDA20/ TRDIOD0/ (SCLA0)	P11/SI11/ SDA11/ TRDIOC1/ (RxDO_1) Note 2	P25/ANI5	P27/ANI7
1	P05/(INTP10)	P50/INTP1/ SI00/RxD0/ TOOLRXD/ SDA00/ TRGIOA/ (TRJIO0)	P51/INTP2/ SO00/TxD0/ TOOLTxD/ TRGIOB	P55/(PCLBUZ1)/ (SCK00)/ (INTP4)	P13/TxD2/ SO20/ TRDIOA1/ IVCMP1 Note 1	P10/SCK11/ SCL11/ TRDIOD1	P146	P147/ANI18/ VCOUT1 Note 1

**Note 1.** Mounted on the 96 KB or more code flash memory products.

**Note 2.** Mounted on the 384 KB or more code flash memory products.

**Caution 1. Make EV<sub>SS0</sub> pin the same potential as V<sub>SS</sub> pin.**

**Caution 2. Make V<sub>DD</sub> pin the potential that is higher than EV<sub>DD0</sub> pin.**

**Caution 3. Connect the REGC pin to V<sub>ss</sub> pin via a capacitor (0.47 to 1  $\mu$ F).**

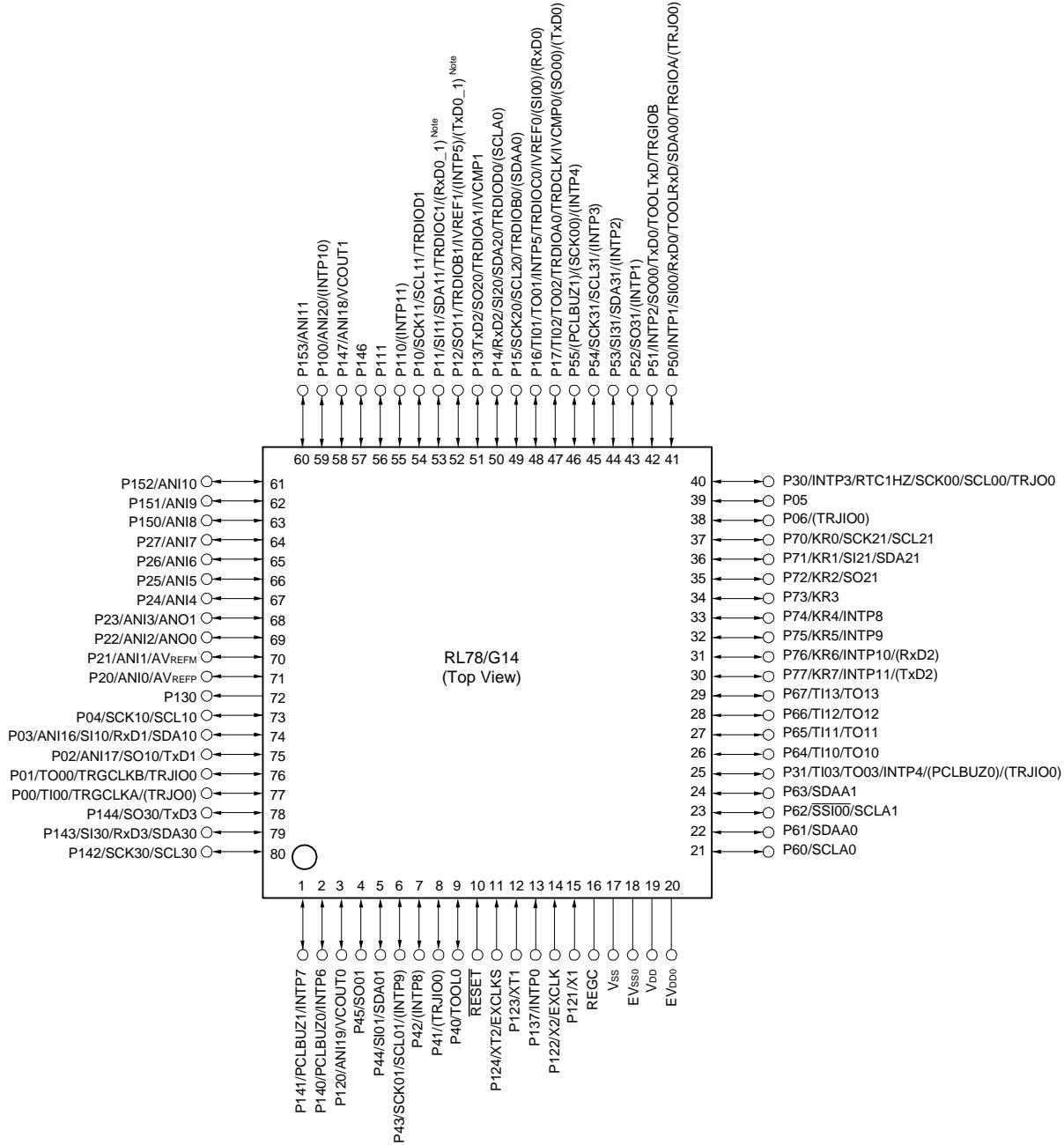
**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** When using the microcontroller for an application where the noise generated inside the microcontroller must be reduced, it is recommended to supply separate powers to the V<sub>DD</sub> and EV<sub>DD0</sub> pins and connect the V<sub>ss</sub> and EV<sub>SS0</sub> pins to separate ground lines.

**Remark 3.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.9 80-pin products

- 80-pin plastic LQFP (14 × 14 mm, 0.65 mm pitch)
- 80-pin plastic LFQFP (12 × 12 mm, 0.5 mm pitch)



**Note** Mounted on the 384 KB or more code flash memory products.

**Caution 1. Make EV<sub>ss0</sub> pin the same potential as V<sub>ss</sub> pin.**

**Caution 2. Make V<sub>dd</sub> pin the potential that is higher than EV<sub>dd0</sub> pin.**

**Caution 3. Connect the REGC pin to V<sub>ss</sub> pin via a capacitor (0.47 to 1  $\mu$ F).**

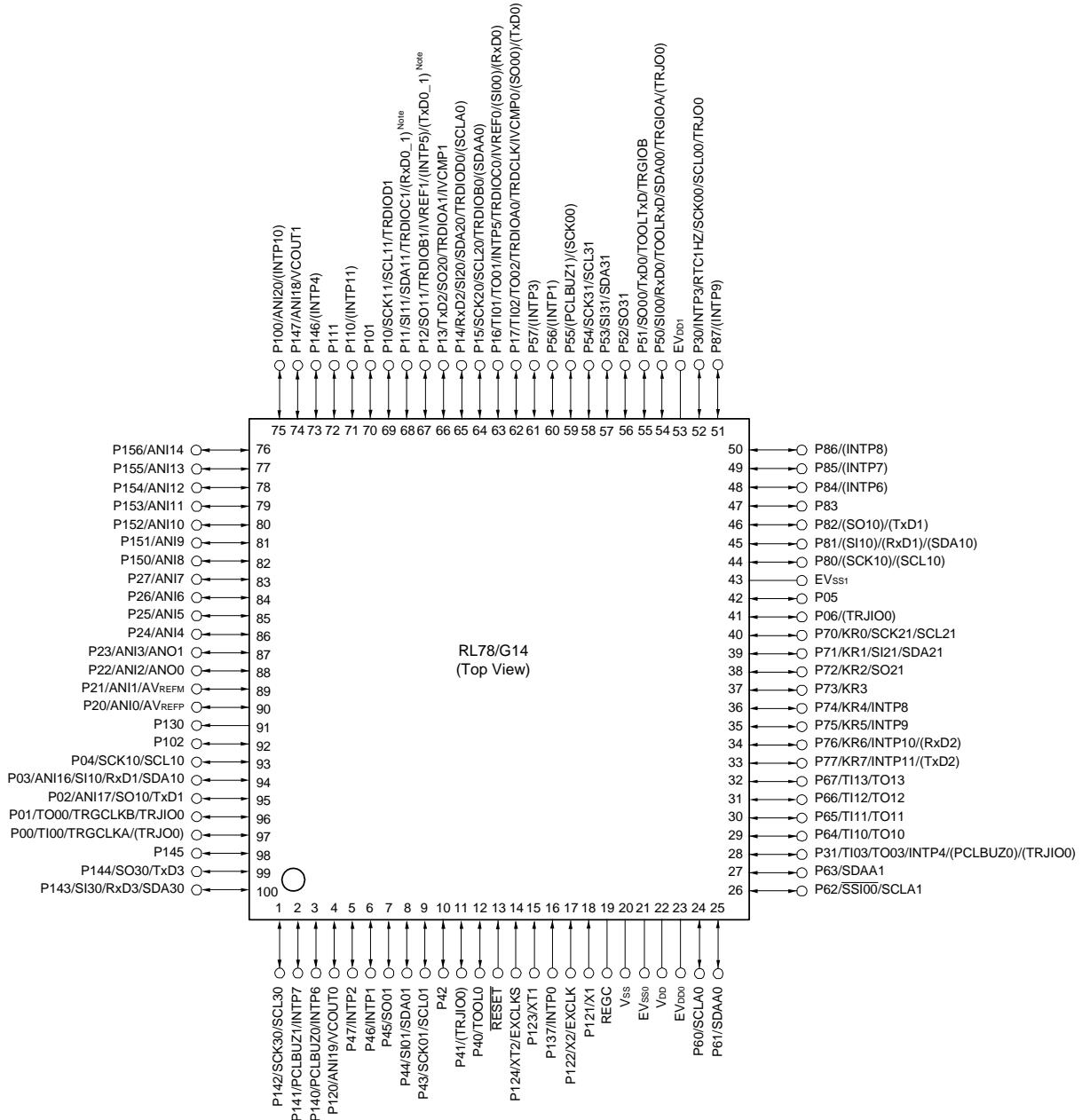
**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** When using the microcontroller for an application where the noise generated inside the microcontroller must be reduced, it is recommended to supply separate powers to the V<sub>dd</sub> and EV<sub>dd0</sub> pins and connect the V<sub>ss</sub> and EV<sub>ss0</sub> pins to separate ground lines.

**Remark 3.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

### 1.3.10 100-pin products

- 100-pin plastic LFQFP (14 × 14 mm, 0.5 mm pitch)



**Note** Mounted on the 384 KB or more code flash memory products.

**Caution 1.** Make EV<sub>SS0</sub>, EV<sub>SS1</sub> pins the same potential as V<sub>SS</sub> pin.

**Caution 2.** Make V<sub>DD</sub> pin the potential that is higher than EV<sub>DD0</sub>, EV<sub>DD1</sub> pins (EV<sub>DD0</sub> = EV<sub>DD1</sub>).

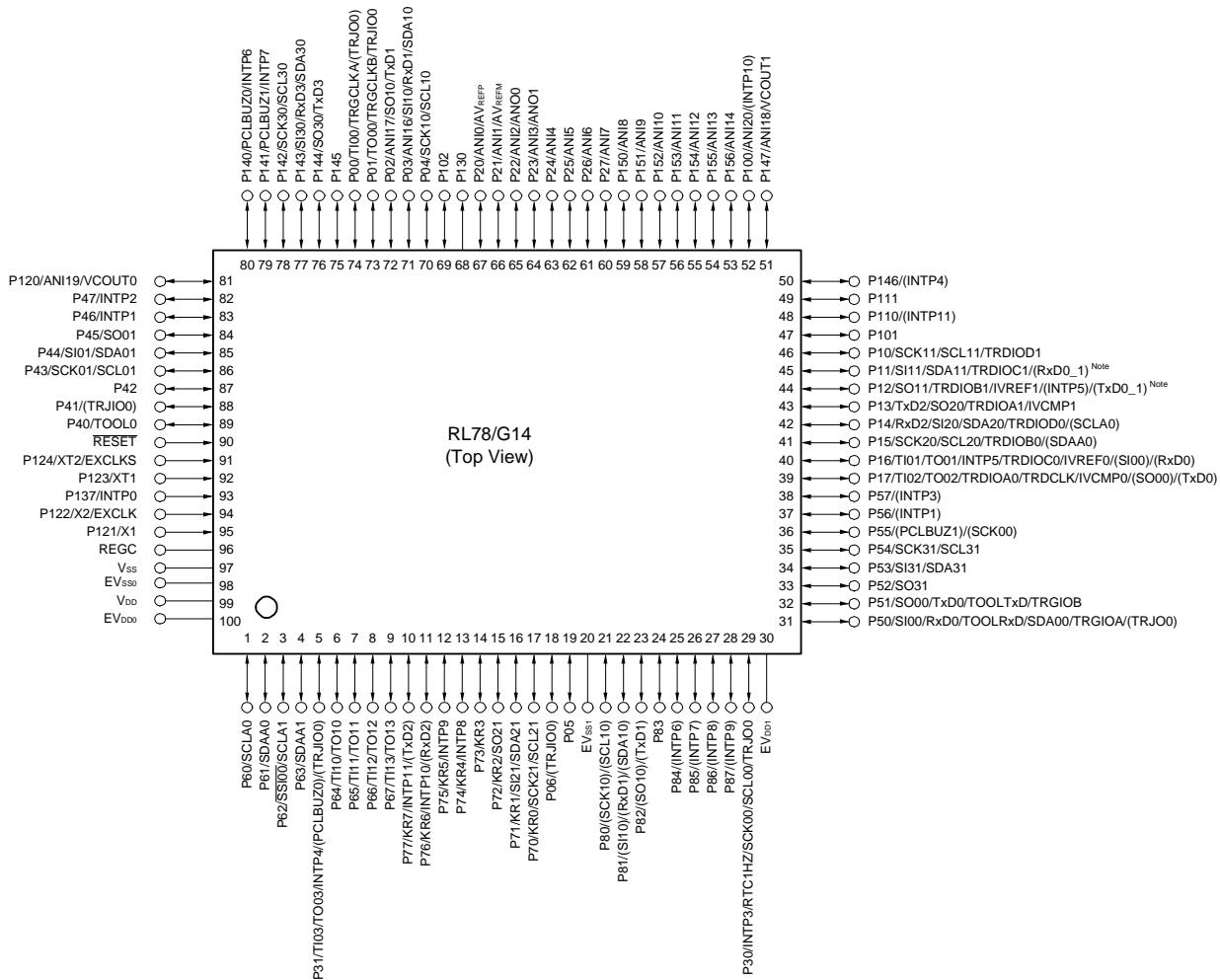
**Caution 3.** Connect the REGC pin to V<sub>SS</sub> pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see **1.4 Pin Identification**.

**Remark 2.** When using the microcontroller for an application where the noise generated inside the microcontroller must be reduced, it is recommended to supply separate powers to the V<sub>DD</sub>, EV<sub>DD0</sub> and EV<sub>DD1</sub> pins and connect the V<sub>SS</sub>, EV<sub>SS0</sub> and EV<sub>SS1</sub> pins to separate ground lines.

**Remark 3.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

- 100-pin plastic LQFP (14 × 20 mm, 0.65 mm pitch)



**Note** Mounted on the 384 KB or more code flash memory products.

**Caution 1.** Make EVss0, EVss1 pins the same potential as Vss pin.

**Caution 2. Make VDD pin the potential that is higher than EVDD0, EVDD1 pins (EVDD0 = EVDD1).**

**Caution 3.** Connect the REGC pin to Vss pin via a capacitor (0.47 to 1  $\mu$ F).

**Remark 1.** For pin identification, see 1.4 Pin Identification.

**Remark 2.** When using the microcontroller for an application where the noise generated inside the microcontroller must be reduced, it is recommended to supply separate powers to the V<sub>DD</sub>, EV<sub>DD0</sub> and EV<sub>DD1</sub> pins and connect the V<sub>SS</sub>, EV<sub>VSS0</sub> and EV<sub>VSS1</sub> pins to separate ground lines.

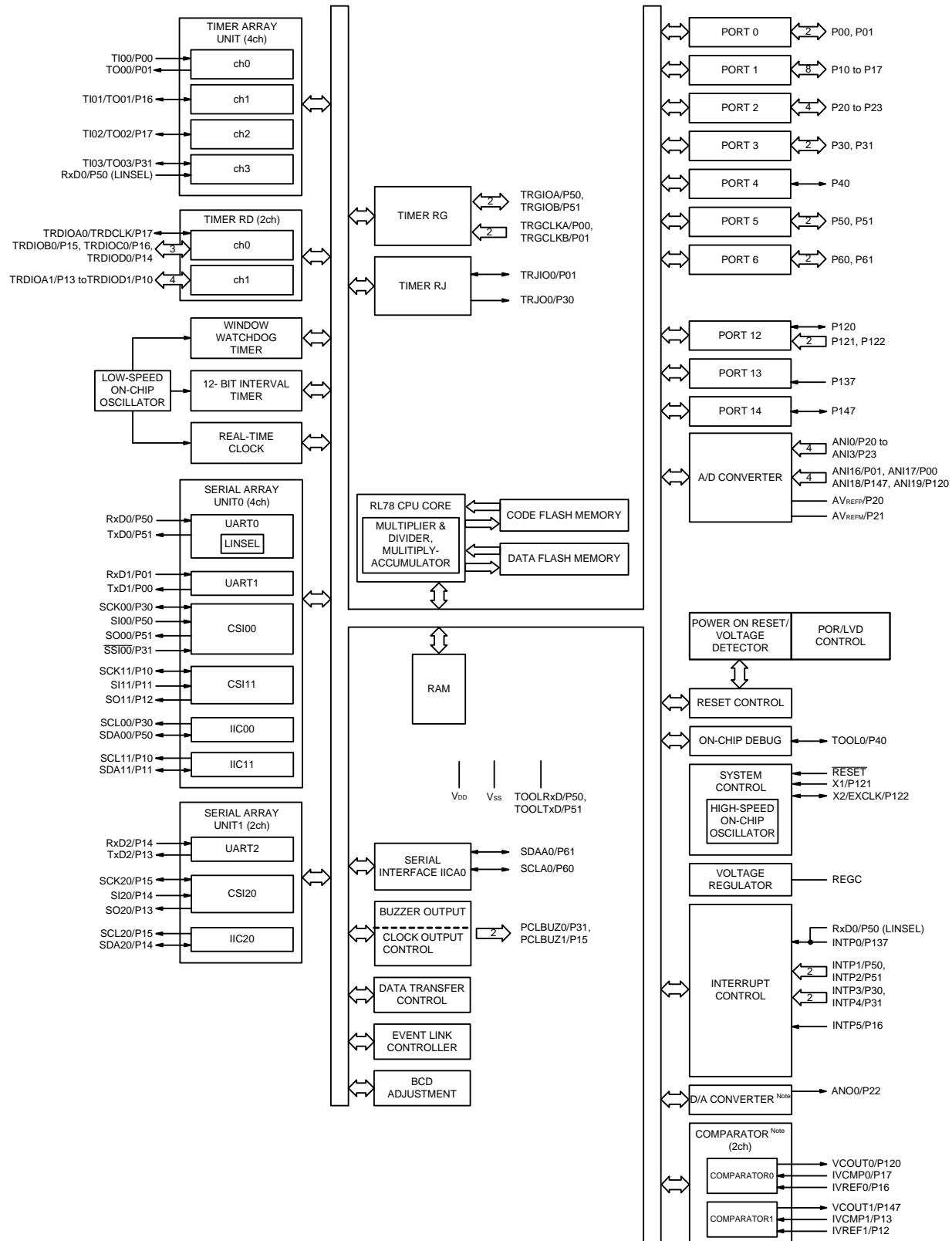
**Remark 3.** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

## 1.4 Pin Identification

ANIO to ANI14,: ANI16 to ANI20 ANO0, ANO1: AVREFM: AVREFP: EVDD0, EVDD1: EVSS0, EVSS1: EXCLK: EXCLKS: INTP0 to INTP11: IVCMP0, IVCMP1: IVREF0, IVREF1: KR0 to KR7: P00 to P06: P10 to P17: P20 to P27: P30, P31: P40 to P47: P50 to P57: P60 to P67: P70 to P77: P80 to P87: P100 to P102: P110, P111: P120 to P124: P130, P137: P140 to P147: P150 to P156: PCLBUZ0, PCLBUZ1: REGC: RESET: RTC1HZ:	Analog input  Analog output  A/D converter reference potential (– side) input  A/D converter reference potential (+ side) input  Power supply for port  Ground for port  External clock input (main system clock)  External clock input (subsystem clock)  External interrupt input  Comparator input  Comparator reference input  Key return  Port 0  Port 1  Port 2  Port 3  Port 4  Port 5  Port 6  Port 7  Port 8  Port 10  Port 11  Port 12  Port 13  Port 14  Port 15  Programmable clock output/buzzer output  Regulator capacitance  Reset  Real-time clock correction clock  (1 Hz) output	RxD0 to RxD3:  SCK00, SCK01, SCK10,:  SCK11, SCK20, SCK21,  SCK30, SCK31  SCLA0, SCLA1,:  SCL00, SCL01, SCL10, SCL11,:  SCL20, SCL21, SCL30,  SCL31  SDAA0, SDAA1, SDA00,:  SDA01, SDA10, SDA11,  SDA20, SDA21, SDA30,  SDA31  <u>SSI00</u> :  TI00 to TI03,:  TI10 to TI13  TO00 to TO03,:  TO10 to TO13, TRJ00  TOOL0:  TOOLRxD, TOOLTxD:  TRDCLK, TRGCLKA,:  TRGCLKB  TRDIOA0, TRDIOB0,:  TRDIOD0,  TRDIOA1, TRDIOB1,  TRDIOD1,  TRDIOC1, TRDIOD1,  TRGIOA, TRGIOB, TRJ00  TxD0 to TxD3:  VCOUT0, VCOUT1:  VDD:  Vss:  X1, X2:  XT1, XT2:	Receive data  Serial clock input/output  Serial clock input/output  Serial clock output  Serial data input/output  Serial data output  Serial data input  Serial interface chip select input  Timer input  Timer output  Data input/output for tool  Data input/output for external device  Timer external input clock  Transmit data  Comparator output  Power supply  Ground  Crystal oscillator (main system clock)  Crystal oscillator (subsystem clock)
--	---	--	--

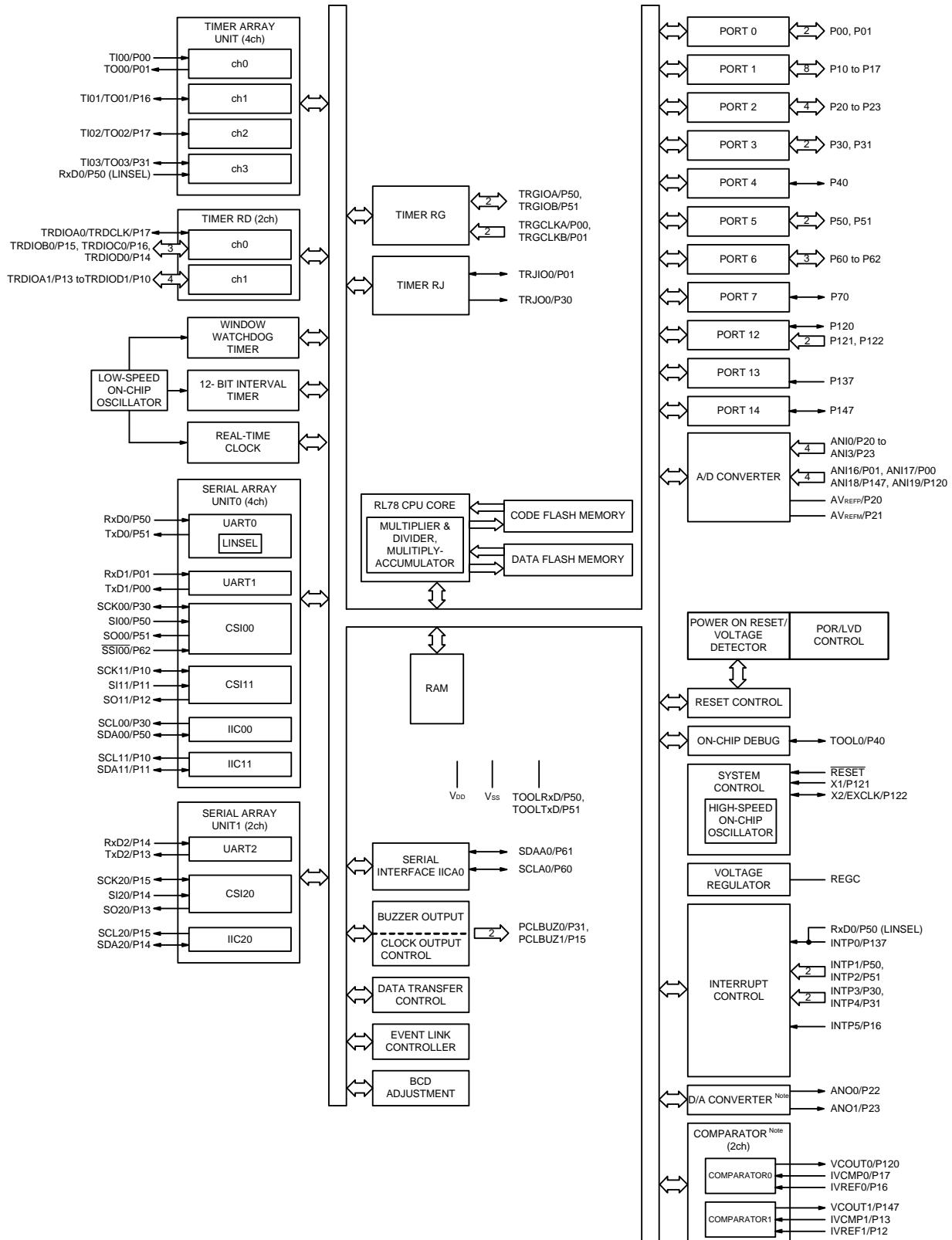
## 1.5 Block Diagram

### 1.5.1 30-pin products



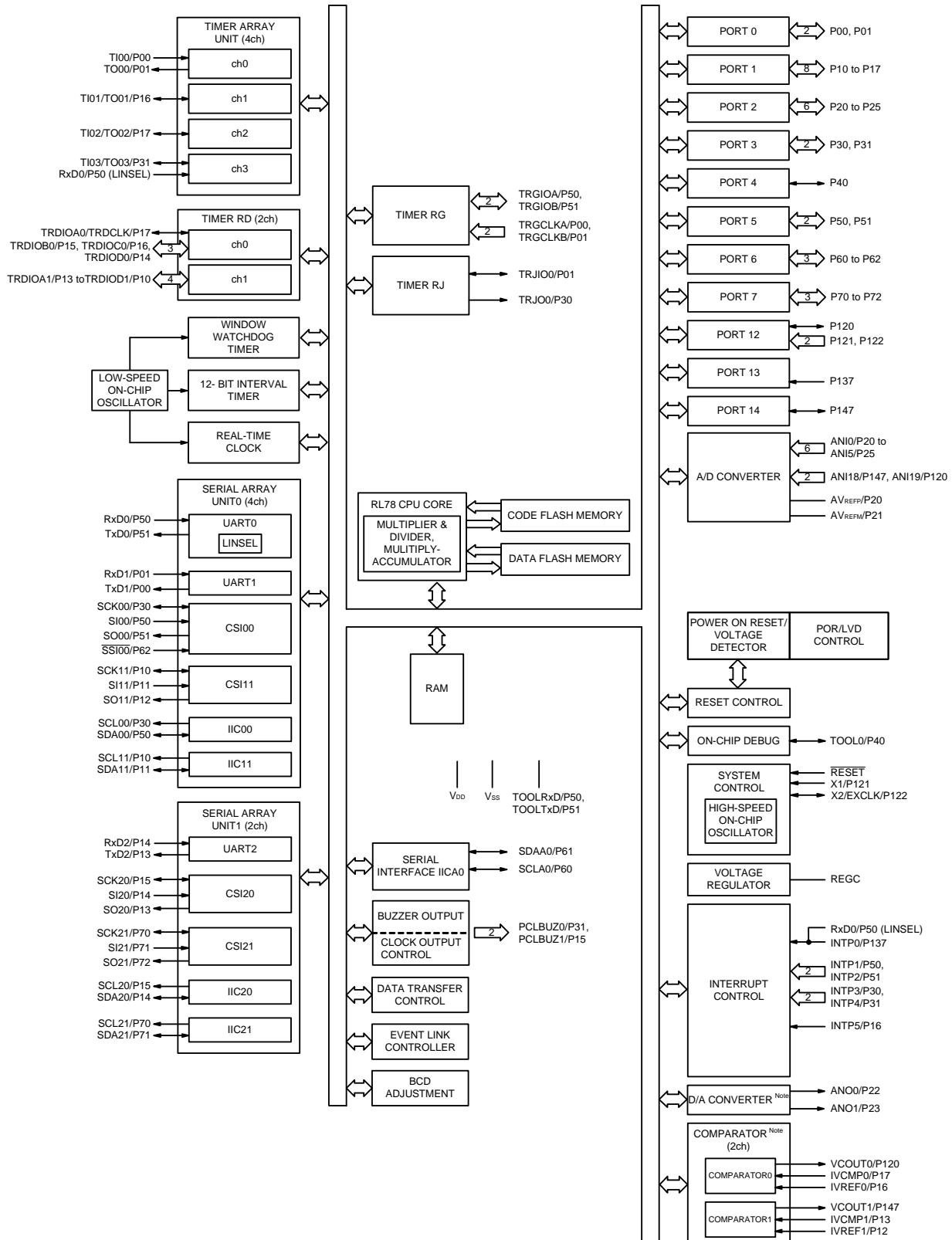
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.2 32-pin products



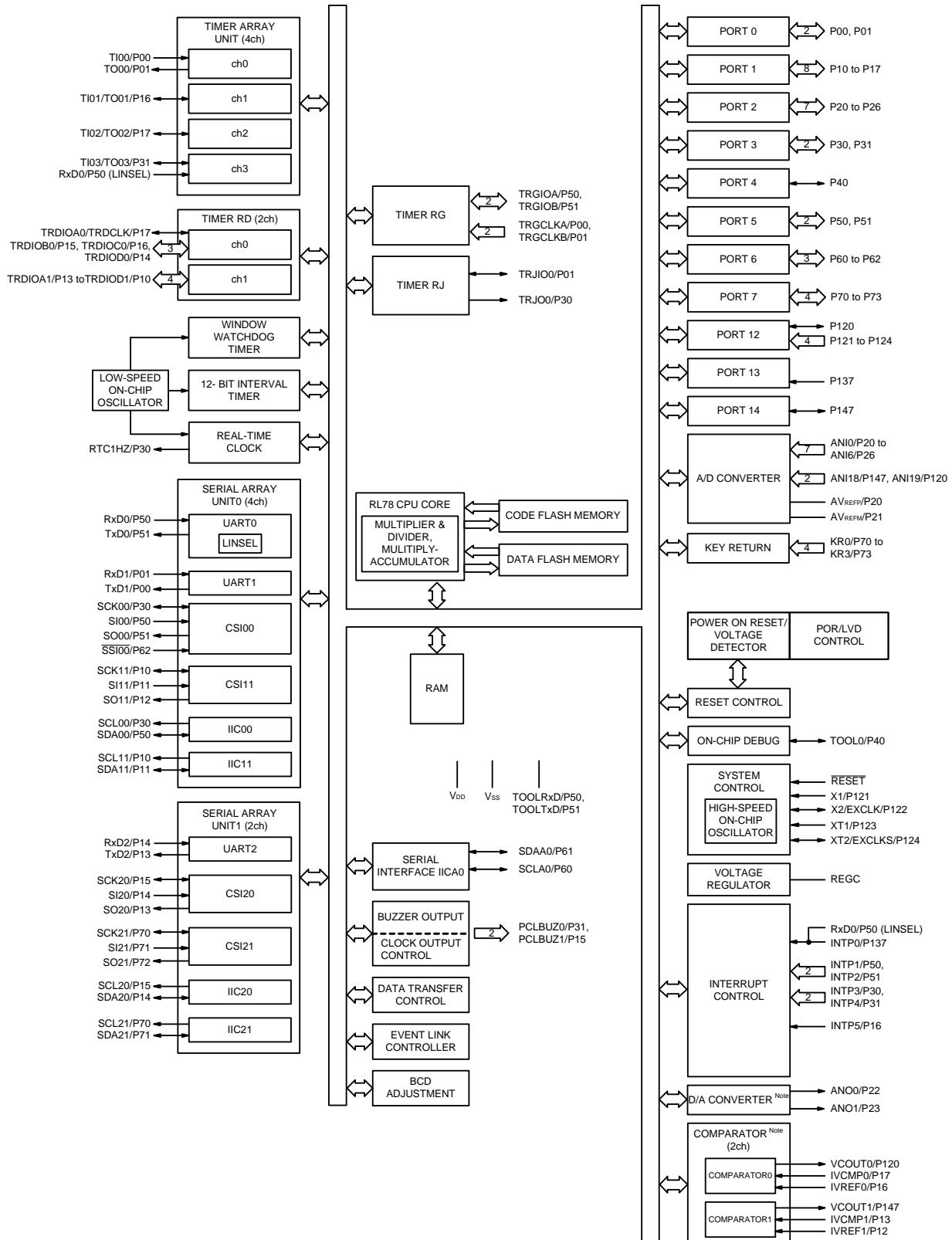
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.3 36-pin products



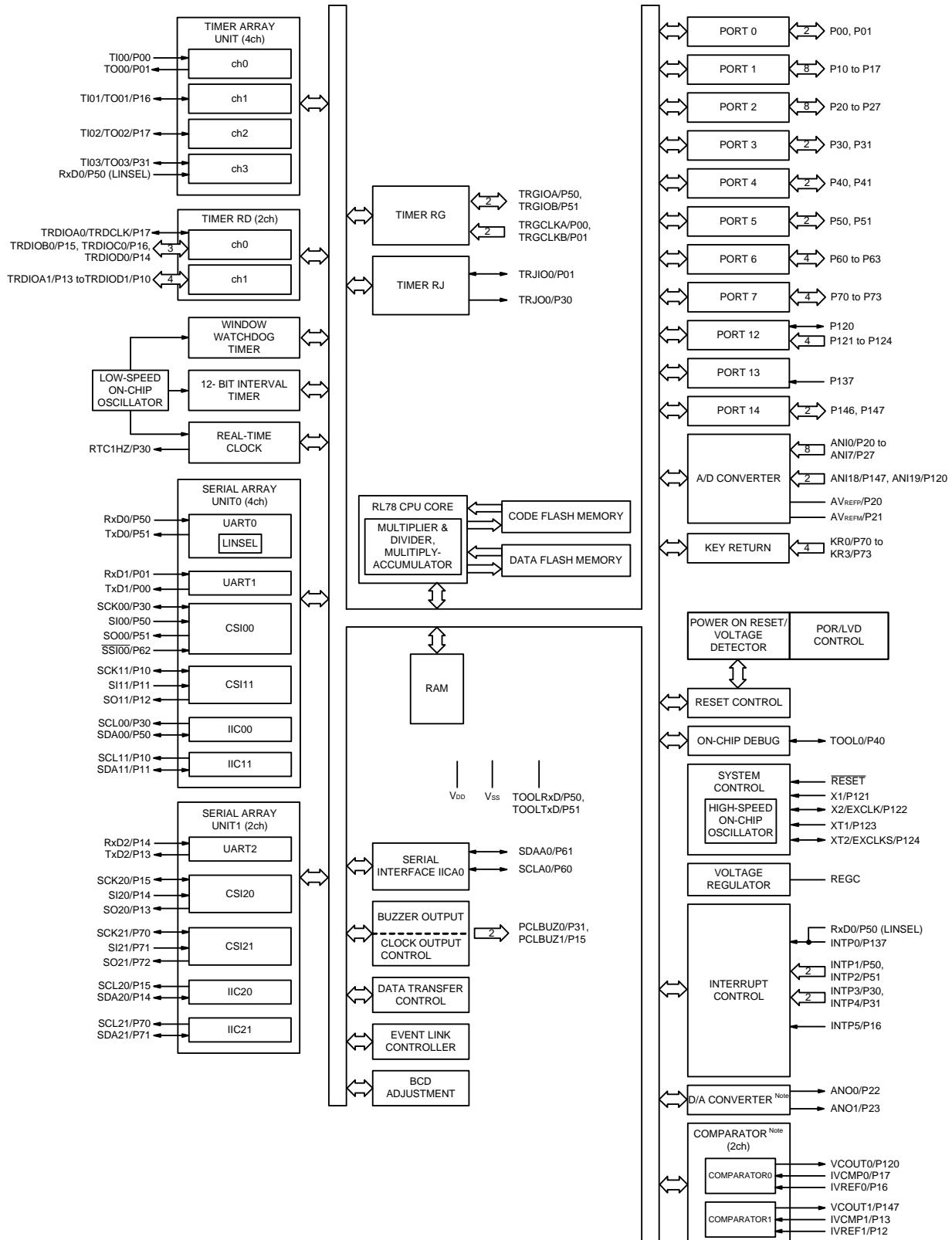
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.4 40-pin products



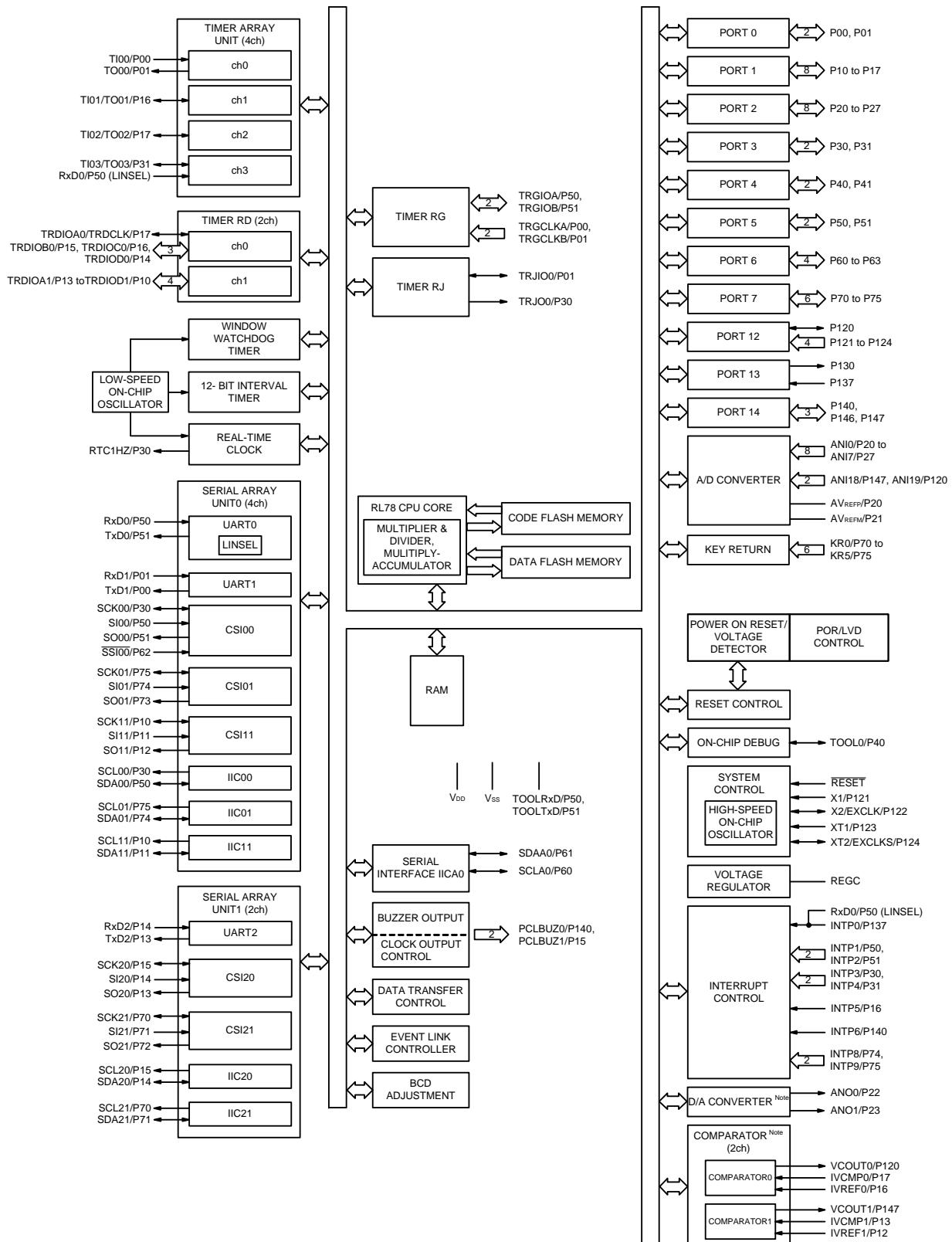
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.5 44-pin products



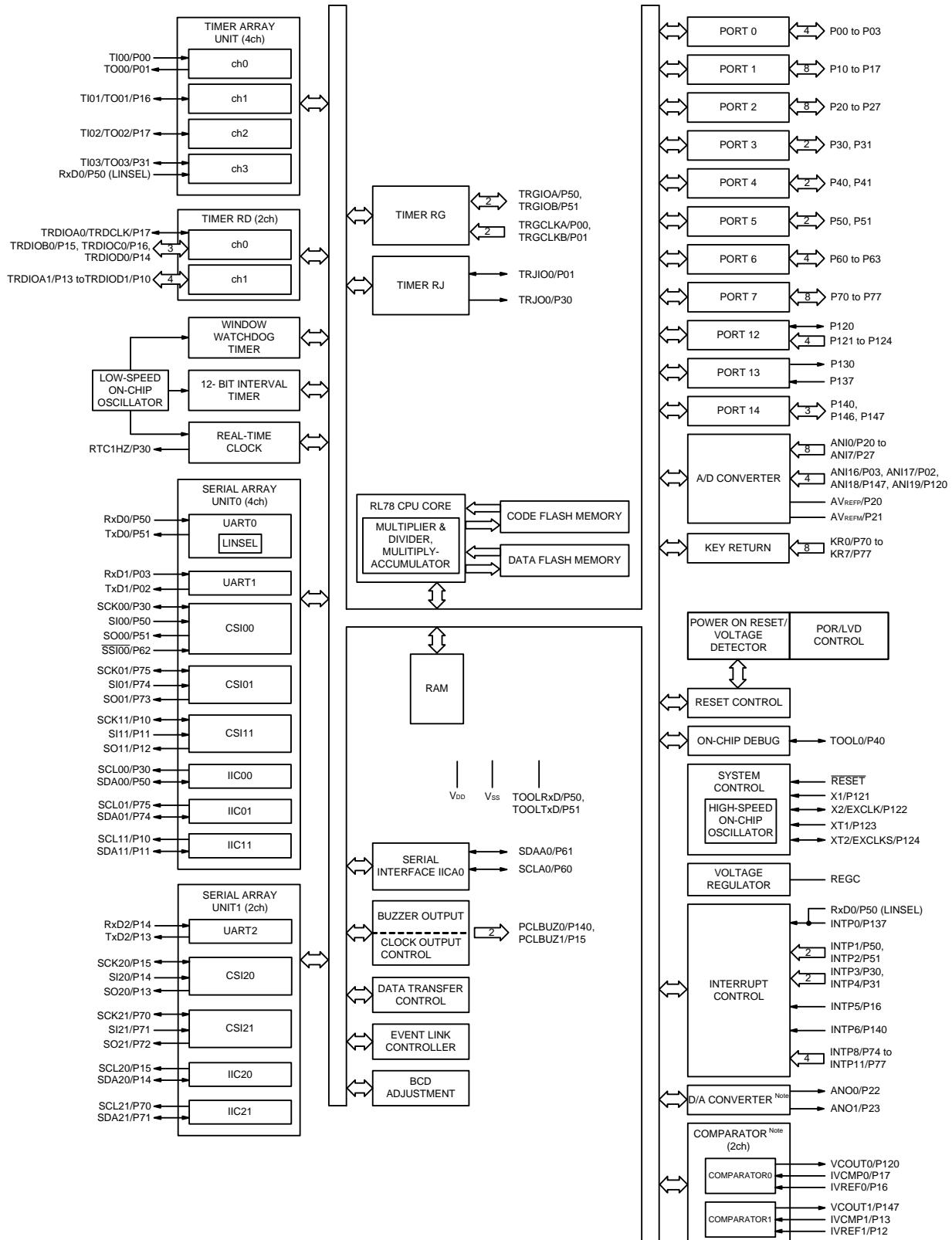
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.6 48-pin products



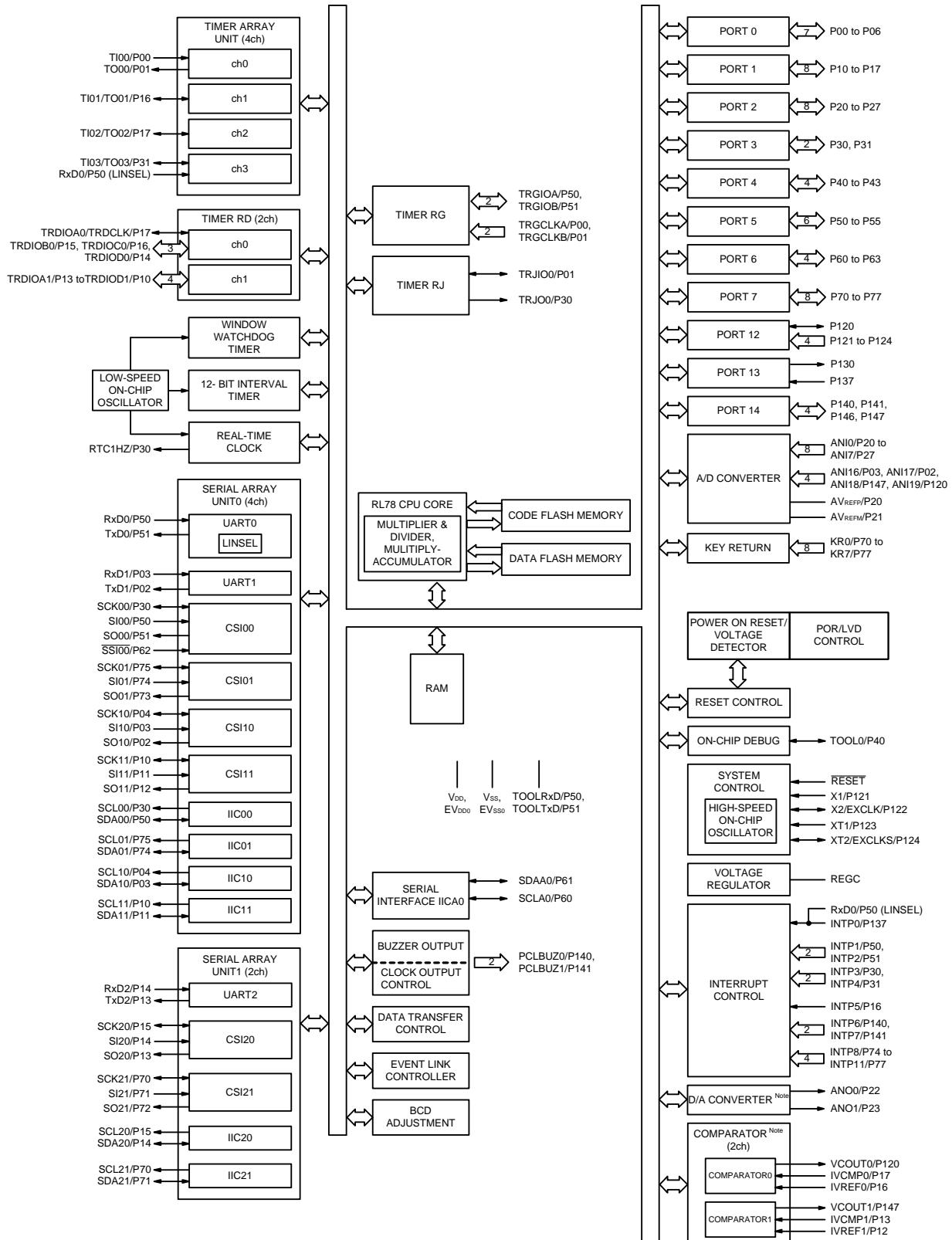
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.7 52-pin products



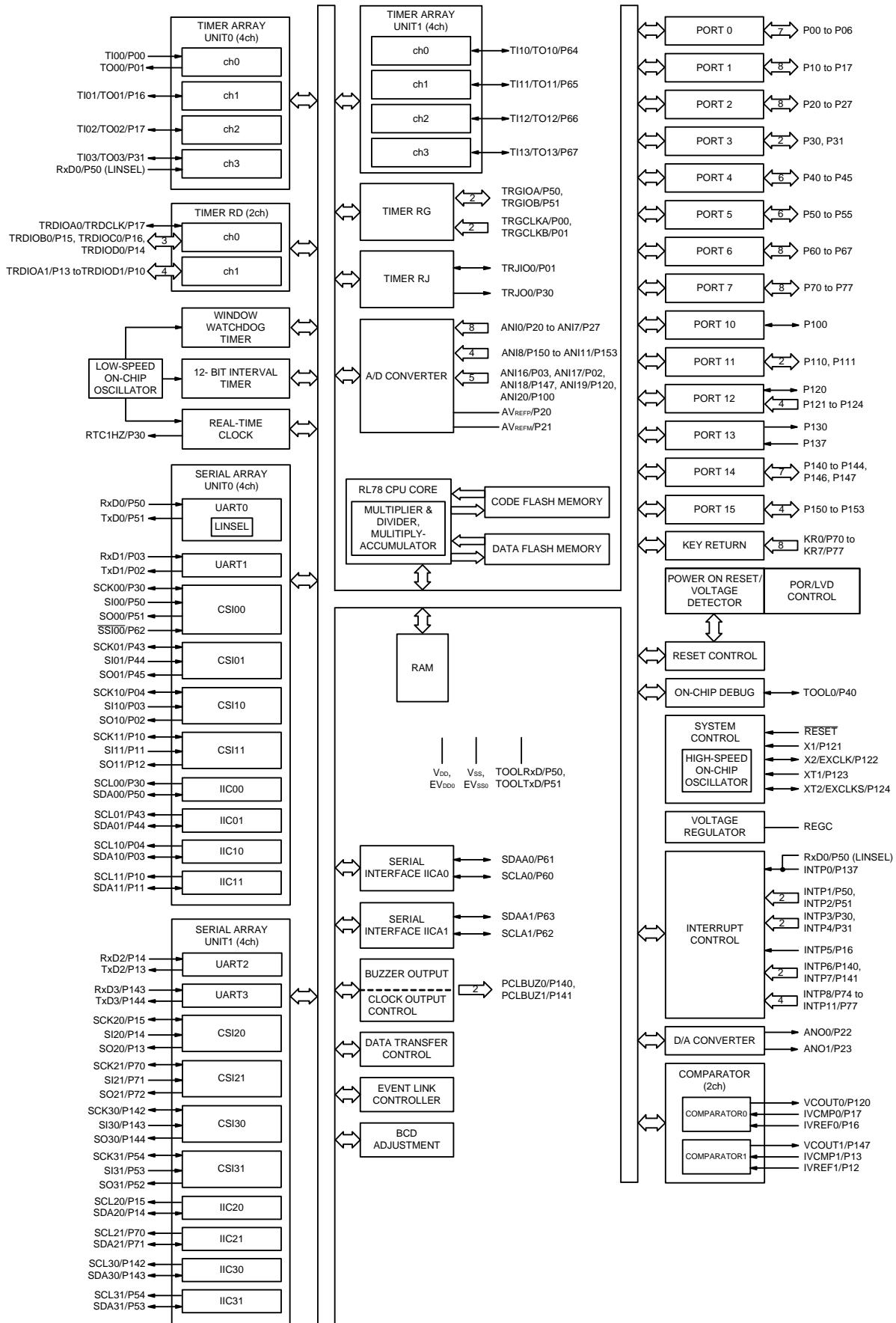
**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.8 64-pin products

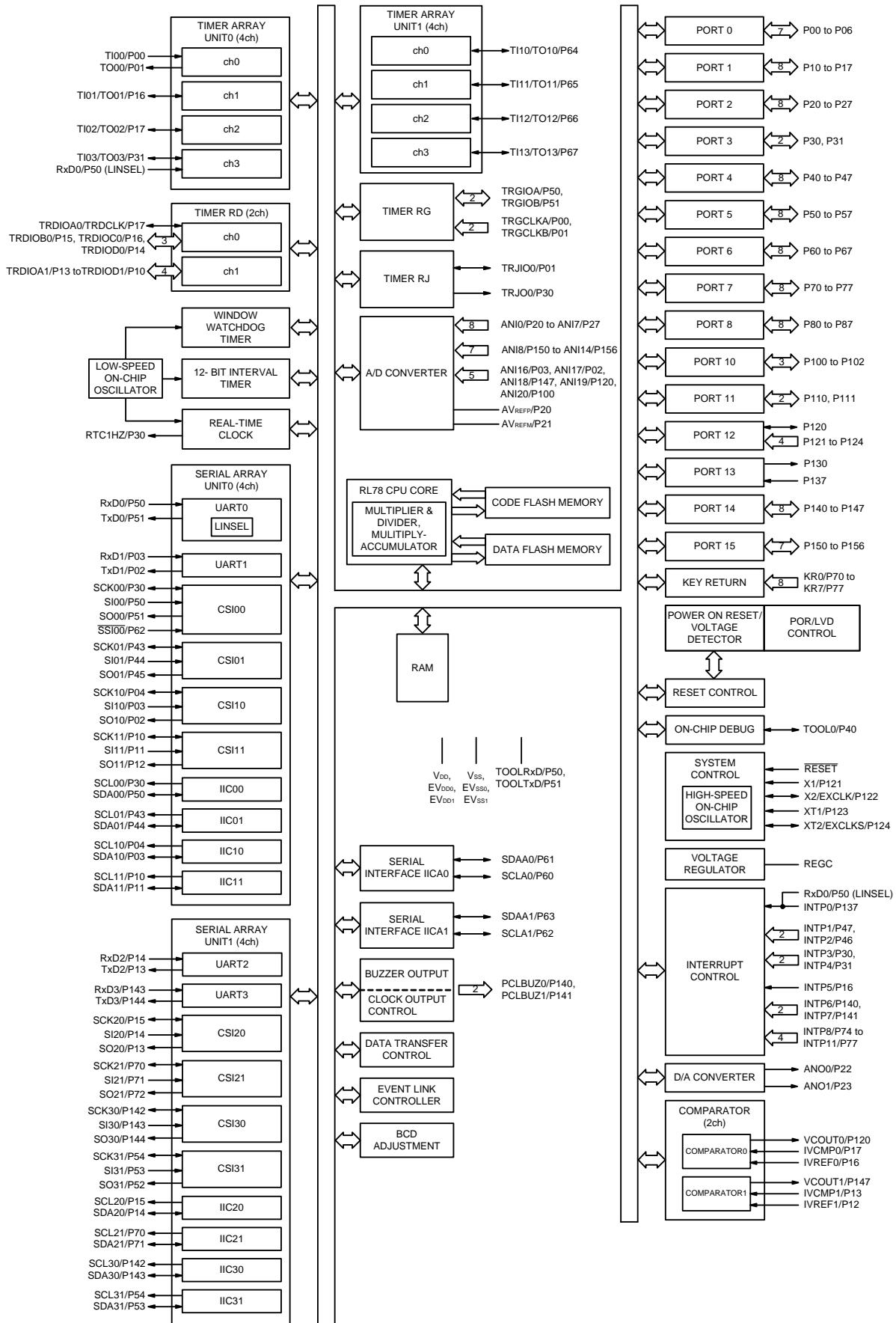


**Note** Mounted on the 96 KB or more code flash memory products.

### 1.5.9 80-pin products



### 1.5.10 100-pin products



## 1.6 Outline of Functions

[30-pin, 32-pin, 36-pin, 40-pin products (code flash memory 16 KB to 64 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item		30-pin	32-pin	36-pin	40-pin
		R5F104Ax (x = A, C to E)	R5F104Bx (x = A, C to E)	R5F104Cx (x = A, C to E)	R5F104Ex (x = A, C to E)
Code flash memory (KB)		16 to 64	16 to 64	16 to 64	16 to 64
Data flash memory (KB)		4	4	4	4
RAM (KB)		2.5 to 5.5 Note	2.5 to 5.5 Note	2.5 to 5.5 Note	2.5 to 5.5 Note
Address space		1 MB			
Main system clock	High-speed system clock	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD}$ = 2.7 to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD}$ = 2.4 to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD}$ = 1.8 to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD}$ = 1.6 to 5.5 V)			
	High-speed on-chip oscillator clock ( $f_{IH}$ )	HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD}$ = 2.7 to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD}$ = 2.4 to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD}$ = 1.8 to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD}$ = 1.6 to 5.5 V)			
Subsystem clock		—			XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz
Low-speed on-chip oscillator clock		15 kHz (TYP.): $V_{DD}$ = 1.6 to 5.5 V			
General-purpose register		8 bits $\times$ 32 registers (8 bits $\times$ 8 registers $\times$ 4 banks)			
Minimum instruction execution time		0.03125 $\mu$ s (High-speed on-chip oscillator clock: $f_{IH}$ = 32 MHz operation)			
		0.05 $\mu$ s (High-speed system clock: $f_{MX}$ = 20 MHz operation)			
		—			30.5 $\mu$ s (Subsystem clock: $f_{SUB}$ = 32.768 kHz operation)
Instruction set		<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits <math>\times</math> 8 bits, 16 bits <math>\times</math> 16 bits), Division (16 bits <math>\div</math> 16 bits, 32 bits <math>\div</math> 32 bits)</li> <li>• Multiplication and Accumulation (16 bits <math>\times</math> 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>			
I/O port	Total	26	28	32	36
	CMOS I/O	21	22	26	28
	CMOS input	3	3	3	5
	CMOS output	—	—	—	—
	N-ch open-drain I/O (6 V tolerance)	2	3	3	3
Timer	16-bit timer	8 channels (TAU: 4 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)			
	Watchdog timer	1 channel			
	Real-time clock (RTC)	1 channel			
	12-bit interval timer	1 channel			
	Timer output	Timer outputs: 13 channels PWM outputs: 9 channels			
	RTC output	—			1 • 1 Hz (subsystem clock: $f_{SUB}$ = 32.768 kHz)

(Note is listed on the next page.)

- Note** The flash library uses RAM in self-programming and rewriting of the data flash memory.  
The target products and start address of the RAM areas used by the flash library are shown below.  
R5F104xD (x = A to C, E to G, J, L): Start address FE900H  
R5F104xE (x = A to C, E to G, J, L): Start address FE900H  
For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

(2/2)

Item	30-pin	32-pin	36-pin	40-pin				
	R5F104Ax (x = A, C to E)	R5F104Bx (x = A, C to E)	R5F104Cx (x = A, C to E)	R5F104Ex (x = A, C to E)				
Clock output/buzzer output	2	2	2	2				
[30-pin, 32-pin, 36-pin products]								
• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: f <sub>MAIN</sub> = 20 MHz operation)								
[40-pin products]								
• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: f <sub>MAIN</sub> = 20 MHz operation)								
• 256 Hz, 512 Hz, 1,024 kHz, 2,048 kHz, 4,096 kHz, 8,192 kHz, 16,384 kHz, 32,768 kHz (Subsystem clock: f <sub>SUB</sub> = 32.768 kHz operation)								
8/10-bit resolution A/D converter	8 channels	8 channels	8 channels	9 channels				
Serial interface	[30-pin, 32-pin products]							
• CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I <sup>2</sup> C: 1 channel								
• CSI: 1 channel/UART: 1 channel/simplified I <sup>2</sup> C: 1 channel								
• CSI: 1 channel/UART: 1 channel/simplified I <sup>2</sup> C: 1 channel								
[36-pin, 40-pin products]								
• CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I <sup>2</sup> C: 1 channel								
• CSI: 1 channel/UART: 1 channel/simplified I <sup>2</sup> C: 1 channel								
• CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels								
I <sup>2</sup> C bus	1 channel	1 channel	1 channel	1 channel				
Data transfer controller (DTC)	28 sources							
Event link controller (ELC)	Event input: 19 Event trigger output: 7							
Event input: 20 Event trigger output: 7								
Vectored interrupt sources	Internal	24	24	24	24			
	External	6	6	6	7			
Key interrupt	—	—	—	—	4			
Reset	<ul style="list-style-type: none"> <li>• Reset by RESET pin</li> <li>• Internal reset by watchdog timer</li> <li>• Internal reset by power-on-reset</li> <li>• Internal reset by voltage detector</li> <li>• Internal reset by illegal instruction execution Note</li> <li>• Internal reset by RAM parity error</li> <li>• Internal reset by illegal-memory access</li> </ul>							
Power-on-reset circuit	<ul style="list-style-type: none"> <li>• Power-on-reset: 1.51 ±0.04 V (T<sub>A</sub> = -40 to +85°C) 1.51 ±0.06 V (T<sub>A</sub> = -40 to +105°C)</li> <li>• Power-down-reset: 1.50 ±0.04 V (T<sub>A</sub> = -40 to +85°C) 1.50 ±0.06 V (T<sub>A</sub> = -40 to +105°C)</li> </ul>							
Voltage detector	1.63 V to 4.06 V (14 stages)							
On-chip debug function	Provided							
Power supply voltage	V <sub>DD</sub> = 1.6 to 5.5 V (T <sub>A</sub> = -40 to +85°C) V <sub>DD</sub> = 2.4 to 5.5 V (T <sub>A</sub> = -40 to +105°C)							
Operating ambient temperature	T <sub>A</sub> = -40 to +85°C (A: Consumer applications, D: Industrial applications), T <sub>A</sub> = -40 to +105°C (G: Industrial applications)							

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution not is issued by emulation with the in-circuit emulator or on-chip debug emulator.

[30-pin, 32-pin, 36-pin, 40-pin products (code flash memory 96 KB to 256 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item	30-pin	32-pin	36-pin	40-pin
	R5F104Ax (x = F, G)	R5F104Bx (x = F, G)	R5F104Cx (x = F, G)	R5F104Ex (x = F to H)
Code flash memory (KB)	96 to 128	96 to 128	96 to 128	96 to 192
Data flash memory (KB)	8	8	8	8
RAM (KB)	12 to 16 Note	12 to 16 Note	12 to 16 Note	12 to 20 Note
Address space	1 MB			
Main system clock	High-speed system clock X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD}$ = 2.7 to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD}$ = 2.4 to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD}$ = 1.8 to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD}$ = 1.6 to 5.5 V)			
	High-speed on-chip oscillator clock ( $f_{IH}$ ) HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD}$ = 2.7 to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD}$ = 2.4 to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD}$ = 1.8 to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD}$ = 1.6 to 5.5 V)			
Subsystem clock		—		XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz
Low-speed on-chip oscillator clock	15 kHz (TYP.): $V_{DD}$ = 1.6 to 5.5 V			
General-purpose register	8 bits × 32 registers (8 bits × 8 registers × 4 banks)			
Minimum instruction execution time	0.03125 μs (High-speed on-chip oscillator clock: $f_{IH}$ = 32 MHz operation) 0.05 μs (High-speed system clock: $f_{MX}$ = 20 MHz operation) — 30.5 μs (Subsystem clock: $f_{SUB}$ = 32.768 kHz operation)			
Instruction set	<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits × 8 bits, 16 bits × 16 bits), Division (16 bits ÷ 16 bits, 32 bits ÷ 32 bits)</li> <li>• Multiplication and Accumulation (16 bits × 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>			
I/O port	Total	26	28	32
	CMOS I/O	21	22	26
	CMOS input	3	3	3
	CMOS output	—	—	—
	N-ch open-drain I/O (6 V tolerance)	2	3	3
Timer	16-bit timer	8 channels (TAU: 4 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)		
	Watchdog timer	1 channel		
	Real-time clock (RTC)	1 channel		
	12-bit interval timer	1 channel		
	Timer output	Timer outputs: 13 channels PWM outputs: 9 channels		
	RTC output	—		1 • 1 Hz (subsystem clock: $f_{SUB}$ = 32.768 kHz)

(Note is listed on the next page.)

- Note** The flash library uses RAM in self-programming and rewriting of the data flash memory.  
The target products and start address of the RAM areas used by the flash library are shown below.  
R5F104xJ (x = F, G, J, L, M, P): Start address F9F00H  
For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

(2/2)

Item	30-pin	32-pin	36-pin	40-pin
	R5F104Ax (x = F, G)	R5F104Bx (x = F, G)	R5F104Cx (x = F, G)	R5F104Ex (x = F to H)
Clock output/buzzer output	2	2	2	2
[30-pin, 32-pin, 36-pin products]				
	<ul style="list-style-type: none"> <li>• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: <math>f_{MAIN} = 20</math> MHz operation)</li> </ul>			
	[40-pin products]			
	<ul style="list-style-type: none"> <li>• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: <math>f_{MAIN} = 20</math> MHz operation)</li> <li>• 256 Hz, 512 Hz, 1,024 kHz, 2,048 kHz, 4,096 kHz, 8,192 kHz, 16,384 kHz, 32,768 kHz (Subsystem clock: <math>f_{SUB} = 32,768</math> kHz operation)</li> </ul>			
8/10-bit resolution A/D converter	8 channels	8 channels	8 channels	9 channels
D/A converter	1 channel	2 channels		
Comparator	2 channels			
Serial interface	[30-pin, 32-pin products]			
	<ul style="list-style-type: none"> <li>• CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>• CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>• CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> </ul>			
	[36-pin, 40-pin products]			
	<ul style="list-style-type: none"> <li>• CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>• CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>• CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul>			
I <sup>2</sup> C bus	1 channel	1 channel	1 channel	1 channel
Data transfer controller (DTC)	30 sources			31 sources
Event link controller (ELC)	Event input: 21 Event trigger output: 8	Event input: 21, Event trigger output: 9		Event input: 22 Event trigger output: 9
Vectored interrupt sources	Internal	24	24	24
	External	6	6	7
Key interrupt	—	—	—	4
Reset	<ul style="list-style-type: none"> <li>• Reset by <math>\overline{RESET}</math> pin</li> <li>• Internal reset by watchdog timer</li> <li>• Internal reset by power-on-reset</li> <li>• Internal reset by voltage detector</li> <li>• Internal reset by illegal instruction execution Note</li> <li>• Internal reset by RAM parity error</li> <li>• Internal reset by illegal-memory access</li> </ul>			
Power-on-reset circuit	<ul style="list-style-type: none"> <li>• Power-on-reset: <math>1.51 \pm 0.04</math> V (<math>T_A = -40</math> to <math>+85^\circ\text{C}</math>) <math>1.51 \pm 0.06</math> V (<math>T_A = -40</math> to <math>+105^\circ\text{C}</math>)</li> <li>• Power-down-reset: <math>1.50 \pm 0.04</math> V (<math>T_A = -40</math> to <math>+85^\circ\text{C}</math>) <math>1.50 \pm 0.06</math> V (<math>T_A = -40</math> to <math>+105^\circ\text{C}</math>)</li> </ul>			
Voltage detector	1.63 V to 4.06 V (14 stages)			
On-chip debug function	Provided			
Power supply voltage	$V_{DD} = 1.6$ to $5.5$ V ( $T_A = -40$ to $+85^\circ\text{C}$ ) $V_{DD} = 2.4$ to $5.5$ V ( $T_A = -40$ to $+105^\circ\text{C}$ )			
Operating ambient temperature	$T_A = -40$ to $+85^\circ\text{C}$ (A: Consumer applications, D: Industrial applications), $T_A = -40$ to $+105^\circ\text{C}$ (G: Industrial applications)			

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution not is issued by emulation with the in-circuit emulator or on-chip debug emulator.

[44-pin, 48-pin, 52-pin, 64-pin products (code flash memory 16 KB to 64 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item	44-pin	48-pin	52-pin	64-pin	
	R5F104Fx (x = A, C to E)	R5F104Gx (x = A, C to E)	R5F104Jx (x = C to E)	R5F104Lx (x = C to E)	
Code flash memory (KB)	16 to 64	16 to 64	32 to 64	32 to 64	
Data flash memory (KB)	4	4	4	4	
RAM (KB)	2.5 to 5.5 Note	2.5 to 5.5 Note	4 to 5.5 Note	4 to 5.5 Note	
Address space	1 MB				
Main system clock	High-speed system clock	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
	High-speed on-chip oscillator clock ( $f_{IH}$ )	HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
Subsystem clock		XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz			
Low-speed on-chip oscillator clock		15 kHz (TYP.): $V_{DD} = 1.6$ to 5.5 V			
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)			
Minimum instruction execution time		0.03125 µs (High-speed on-chip oscillator clock: $f_{IH} = 32$ MHz operation) 0.05 µs (High-speed system clock: $f_{MX} = 20$ MHz operation) 30.5 µs (Subsystem clock: $f_{SUB} = 32.768$ kHz operation)			
Instruction set		• Data transfer (8/16 bits) • Adder and subtractor/logical operation (8/16 bits) • Multiplication (8 bits × 8 bits, 16 bits × 16 bits), Division (16 bits ÷ 16 bits, 32 bits ÷ 32 bits) • Multiplication and Accumulation (16 bits × 16 bits + 32 bits) • Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.			
I/O port	Total	40	44	48	58
	CMOS I/O	31	34	38	48
	CMOS input	5	5	5	5
	CMOS output	—	1	1	1
	N-ch open-drain I/O (6 V tolerance)	4	4	4	4
Timer	16-bit timer	8 channels (TAU: 4 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)			
	Watchdog timer	1 channel			
	Real-time clock (RTC)	1 channel			
	12-bit interval timer	1 channel			
	Timer output	Timer outputs: 13 channels PWM outputs: 9 channels			
	RTC output	1 • 1 Hz (subsystem clock: $f_{SUB} = 32.768$ kHz)			

(Note is listed on the next page.)

- Note** The flash library uses RAM in self-programming and rewriting of the data flash memory.  
The target products and start address of the RAM areas used by the flash library are shown below.  
R5F104xD (x = A to C, E to G, J, L): Start address FE900H  
R5F104xE (x = A to C, E to G, J, L): Start address FE900H  
For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

(2/2)

Item	44-pin	48-pin	52-pin	64-pin	
	R5F104Fx (x = A, C to E)	R5F104Gx (x = A, C to E)	R5F104Jx (x = C to E)	R5F104Lx (x = C to E)	
Clock output/buzzer output	2	2	2	2	
<ul style="list-style-type: none"> <li>2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: fMAIN = 20 MHz operation)</li> <li>256 Hz, 512 Hz, 1,024 kHz, 2,048 kHz, 4,096 kHz, 8,192 kHz, 16,384 kHz, 32,768 kHz (Subsystem clock: fSUB = 32.768 kHz operation)</li> </ul>					
8/10-bit resolution A/D converter	10 channels	10 channels	12 channels	12 channels	
Serial interface	<p>[44-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul> <p>[48-pin, 52-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul> <p>[64-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul>				
	I <sup>2</sup> C bus	1 channel	1 channel	1 channel	1 channel
Data transfer controller (DTC)	29 sources	30 sources		31 sources	
Event link controller (ELC)	Event input: 20 Event trigger output: 7				
Vectored interrupt sources	Internal	24	24	24	24
	External	7	10	12	13
Key interrupt		4	6	8	8
Reset	<ul style="list-style-type: none"> <li>Reset by <u>RESET</u> pin</li> <li>Internal reset by watchdog timer</li> <li>Internal reset by power-on-reset</li> <li>Internal reset by voltage detector</li> <li>Internal reset by illegal instruction execution Note</li> <li>Internal reset by RAM parity error</li> <li>Internal reset by illegal-memory access</li> </ul>				
Power-on-reset circuit	<ul style="list-style-type: none"> <li>Power-on-reset: 1.51 ±0.04 V (TA = -40 to +85°C) 1.51 ±0.06 V (TA = -40 to +105°C)</li> <li>Power-down-reset: 1.50 ±0.04 V (TA = -40 to +85°C) 1.50 ±0.06 V (TA = -40 to +105°C)</li> </ul>				
Voltage detector	1.63 V to 4.06 V (14 stages)				
On-chip debug function	Provided				
Power supply voltage	VDD = 1.6 to 5.5 V (TA = -40 to +85°C) VDD = 2.4 to 5.5 V (TA = -40 to +105°C)				
Operating ambient temperature	TA = -40 to +85°C (A: Consumer applications, D: Industrial applications), TA = -40 to +105°C (G: Industrial applications)				

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution is not issued by emulation with the in-circuit emulator or on-chip debug emulator.

[44-pin, 48-pin, 52-pin, 64-pin products (code flash memory 96 KB to 256 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item	44-pin	48-pin	52-pin	64-pin	
	R5F104Fx (x = F to H, J)	R5F104Gx (x = F to H, J)	R5F104Jx (x = F to H, J)	R5F104Lx (x = F to H, J)	
Code flash memory (KB)	96 to 256	96 to 256	96 to 256	96 to 256	
Data flash memory (KB)	8	8	8	8	
RAM (KB)	12 to 24 Note	12 to 24 Note	12 to 24 Note	12 to 24 Note	
Address space	1 MB				
Main system clock	High-speed system clock	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
	High-speed on-chip oscillator clock ( $f_{IH}$ )	HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
Subsystem clock		XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz			
Low-speed on-chip oscillator clock		15 kHz (TYP.): $V_{DD} = 1.6$ to 5.5 V			
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)			
Minimum instruction execution time		0.03125 µs (High-speed on-chip oscillator clock: $f_{IH} = 32$ MHz operation) 0.05 µs (High-speed system clock: $f_{MX} = 20$ MHz operation) 30.5 µs (Subsystem clock: $f_{SUB} = 32.768$ kHz operation)			
Instruction set		<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits × 8 bits, 16 bits × 16 bits), Division (16 bits ÷ 16 bits, 32 bits ÷ 32 bits)</li> <li>• Multiplication and Accumulation (16 bits × 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>			
I/O port	Total	40	44	48	58
	CMOS I/O	31	34	38	48
	CMOS input	5	5	5	5
	CMOS output	—	1	1	1
	N-ch open-drain I/O (6 V tolerance)	4	4	4	4
Timer	16-bit timer	8 channels (TAU: 4 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)			
	Watchdog timer	1 channel			
	Real-time clock (RTC)	1 channel			
	12-bit interval timer	1 channel			
	Timer output	Timer outputs: 14 channels PWM outputs: 9 channels			
	RTC output	1 • 1 Hz (subsystem clock: $f_{SUB} = 32.768$ kHz)			

(Note is listed on the next page.)

- Note** The flash library uses RAM in self-programming and rewriting of the data flash memory.  
The target products and start address of the RAM areas used by the flash library are shown below.  
R5F104xJ (x = F, G, J, L, M, P): Start address F9F00H  
For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

(2/2)

Item	44-pin	48-pin	52-pin	64-pin	
	R5F104Fx (x = F to H, J)	R5F104Gx (x = F to H, J)	R5F104Jx (x = F to H, J)	R5F104Lx (x = F to H, J)	
Clock output/buzzer output	2	2	2	2	
<ul style="list-style-type: none"> <li>2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: fMAIN = 20 MHz operation)</li> <li>256 Hz, 512 Hz, 1.024 kHz, 2.048 kHz, 4.096 kHz, 8.192 kHz, 16.384 kHz, 32.768 kHz (Subsystem clock: fSUB = 32.768 kHz operation)</li> </ul>					
8/10-bit resolution A/D converter	10 channels	10 channels	12 channels	12 channels	
D/A converter	2 channels				
Comparator	2 channels				
Serial interface	<p>[44-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 1 channel/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul> <p>[48-pin, 52-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 1 channel/UART: 1 channel/simplified I<sup>2</sup>C: 1 channel</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul> <p>[64-pin products]</p> <ul style="list-style-type: none"> <li>CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul>				
	I <sup>2</sup> C bus	1 channel	1 channel	1 channel	1 channel
Data transfer controller (DTC)	31 sources	32 sources		33 sources	
Event link controller (ELC)	Event input: 22 Event trigger output: 9				
Vectored interrupt sources	Internal	24	24	24	24
	External	7	10	12	13
Key interrupt		4	6	8	8
Reset	<ul style="list-style-type: none"> <li>Reset by RESET pin</li> <li>Internal reset by watchdog timer</li> <li>Internal reset by power-on-reset</li> <li>Internal reset by voltage detector</li> <li>Internal reset by illegal instruction execution Note</li> <li>Internal reset by RAM parity error</li> <li>Internal reset by illegal-memory access</li> </ul>				
Power-on-reset circuit	<ul style="list-style-type: none"> <li>Power-on-reset: 1.51 ±0.04 V (TA = -40 to +85°C)</li> <li>1.51 ±0.06 V (TA = -40 to +105°C)</li> <li>Power-down-reset: 1.50 ±0.04 V (TA = -40 to +85°C)</li> <li>1.50 ±0.06 V (TA = -40 to +105°C)</li> </ul>				
Voltage detector	1.63 V to 4.06 V (14 stages)				
On-chip debug function	Provided				
Power supply voltage	V <sub>DD</sub> = 1.6 to 5.5 V (TA = -40 to +85°C) V <sub>DD</sub> = 2.4 to 5.5 V (TA = -40 to +105°C)				
Operating ambient temperature	TA = -40 to +85°C (A: Consumer applications, D: Industrial applications), TA = -40 to +105°C (G: Industrial applications)				

**Note** The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution is not issued by emulation with the in-circuit emulator or on-chip debug emulator.

[48-pin, 64-pin products (code flash memory 384 KB to 512 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item	48-pin	64-pin	
	R5F104Gx (x = K, L)	R5F104Lx (x = K, L)	
Code flash memory (KB)	384 to 512	384 to 512	
Data flash memory (KB)	8	8	
RAM (KB)	32 to 48 Note	32 to 48 Note	
Address space	1 MB		
Main system clock	High-speed system clock High-speed on-chip oscillator clock ( $f_{IH}$ )	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V) HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)	
Subsystem clock	XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz		
Low-speed on-chip oscillator clock	$15\text{ kHz (TYP.)}$ : $V_{DD} = 1.6$ to 5.5 V		
General-purpose register	8 bits $\times$ 32 registers (8 bits $\times$ 8 registers $\times$ 4 banks)		
Minimum instruction execution time	0.03125 $\mu\text{s}$ (High-speed on-chip oscillator clock: $f_{IH} = 32\text{ MHz}$ operation) 0.05 $\mu\text{s}$ (High-speed system clock: $f_{MX} = 20\text{ MHz}$ operation) 30.5 $\mu\text{s}$ (Subsystem clock: $f_{SUB} = 32.768\text{ kHz}$ operation)		
Instruction set	<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits <math>\times</math> 8 bits, 16 bits <math>\times</math> 16 bits), Division (16 bits <math>\div</math> 16 bits, 32 bits <math>\div</math> 32 bits)</li> <li>• Multiplication and Accumulation (16 bits <math>\times</math> 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>		
I/O port	Total	44	58
	CMOS I/O	34	48
	CMOS input	5	5
	CMOS output	1	1
	N-ch open-drain I/O (6 V tolerance)	4	4
Timer	16-bit timer	8 channels (TAU: 4 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)	
	Watchdog timer	1 channel	
	Real-time clock (RTC)	1 channel	
	12-bit interval timer	1 channel	
	Timer output	Timer outputs: 14 channels PWM outputs: 9 channels	
	RTC output	1 • 1 Hz (subsystem clock: $f_{SUB} = 32.768\text{ kHz}$ )	

(Note is listed on the next page.)

**Note**

The flash library uses RAM in self-programming and rewriting of the data flash memory.  
The target products and start address of the RAM areas used by the flash library are shown below.

R5F104xL (x = G, L, M, P): Start address F3F00H

For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

(2/2)

Item	48-pin	64-pin
	R5F104Gx (x = K, L)	R5F104Lx (x = K, L)
Clock output/buzzer output	2	2
	• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: fMAIN = 20 MHz operation) • 256 Hz, 512 Hz, 1.024 kHz, 2.048 kHz, 4.096 kHz, 8.192 kHz, 16.384 kHz, 32.768 kHz (Subsystem clock: fSUB = 32.768 kHz operation)	
8/10-bit resolution A/D converter	10 channels	12 channels
D/A converter	2 channels	
Comparator	2 channels	
Serial interface	[48-pin products] • CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 1 channel/UART: 1 channel/simplified I <sup>2</sup> C: 1 channel • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels [64-pin products] • CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels	
	I <sup>2</sup> C bus	1 channel
Data transfer controller (DTC)	32 sources	33 sources
Event link controller (ELC)	Event input: 22 Event trigger output: 9	
Vectorized interrupt sources	Internal	24
	External	10
Key interrupt		13
		6
Reset		8
	• Reset by <u>RESET</u> pin • Internal reset by watchdog timer • Internal reset by power-on-reset • Internal reset by voltage detector • Internal reset by illegal instruction execution Note • Internal reset by RAM parity error • Internal reset by illegal-memory access	
Power-on-reset circuit		• Power-on-reset: 1.51 ±0.04 V (TA = -40 to +85°C) 1.51 ±0.06 V (TA = -40 to +105°C) • Power-down-reset: 1.50 ±0.04 V (TA = -40 to +85°C) 1.50 ±0.06 V (TA = -40 to +105°C)
Voltage detector	1.63 V to 4.06 V (14 stages)	
On-chip debug function	Provided	
Power supply voltage	VDD = 1.6 to 5.5 V (TA = -40 to +85°C) VDD = 2.4 to 5.5 V (TA = -40 to +105°C)	
Operating ambient temperature	TA = -40 to +85°C (A: Consumer applications, D: Industrial applications), TA = -40 to +105°C (G: Industrial applications)	

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution is not issued by emulation with the in-circuit emulator or on-chip debug emulator.

[80-pin, 100-pin products (code flash memory 96 KB to 256 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item		80-pin	100-pin
		R5F104Mx (x = F to H, J)	R5F104Px (x = F to H, J)
Code flash memory (KB)		96 to 256	96 to 256
Data flash memory (KB)		8	8
RAM (KB)		12 to 24 Note	12 to 24 Note
Address space		1 MB	
Main system clock	High-speed system clock	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)	
	High-speed on-chip oscillator clock ( $f_{IH}$ )	HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)	
Subsystem clock		XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz	
Low-speed on-chip oscillator clock		15 kHz (TYP.): $V_{DD} = 1.6$ to 5.5 V	
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)	
Minimum instruction execution time	0.03125 µs (High-speed on-chip oscillator clock: $f_{IH} = 32$ MHz operation)		
	0.05 µs (High-speed system clock: $f_{MX} = 20$ MHz operation)		
	30.5 µs (Subsystem clock: $f_{SUB} = 32.768$ kHz operation)		
Instruction set		<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits × 8 bits, 16 bits × 16 bits), Division (16 bits ÷ 16 bits, 32 bits ÷ 32 bits)</li> <li>• Multiplication and Accumulation (16 bits × 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>	
I/O port	Total	74	92
	CMOS I/O	64	82
	CMOS input	5	5
	CMOS output	1	1
	N-ch open-drain I/O (6 V tolerance)	4	4
Timer	16-bit timer	12 channels (TAU: 8 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)	
	Watchdog timer	1 channel	
	Real-time clock (RTC)	1 channel	
	12-bit interval timer	1 channel	
	Timer output	Timer outputs: 18 channels PWM outputs: 12 channels	
	RTC output	1 • 1 Hz (subsystem clock: $f_{SUB} = 32.768$ kHz)	

**Note** In the case of the 24 KB, this is about 23 KB when the self-programming function and data flash function are used (For details, see **CHAPTER 3**).

(2/2)

Item	80-pin	100-pin
	R5F104Mx (x = F to H, J)	R5F104Px (x = F to H, J)
Clock output/buzzer output	2	2
	<ul style="list-style-type: none"> <li>2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: fMAIN = 20 MHz operation)</li> <li>256 Hz, 512 Hz, 1,024 kHz, 2,048 kHz, 4,096 kHz, 8,192 kHz, 16,384 kHz, 32,768 kHz (Subsystem clock: fSUB = 32.768 kHz operation)</li> </ul>	
8/10-bit resolution A/D converter	17 channels	20 channels
D/A converter	2 channels	2 channels
Comparator	2 channels	2 channels
Serial interface	[80-pin, 100-pin products] <ul style="list-style-type: none"> <li>CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> <li>CSI: 2 channels/UART: 1 channel/simplified I<sup>2</sup>C: 2 channels</li> </ul>	
I <sup>2</sup> C bus	2 channels	2 channels
Data transfer controller (DTC)	39 sources	39 sources
Event link controller (ELC)	Event input: 26 Event trigger output: 9	
Vectored interrupt sources	Internal	32
	External	13
Key interrupt	8	8
Reset	<ul style="list-style-type: none"> <li>Reset by <u>RESET</u> pin</li> <li>Internal reset by watchdog timer</li> <li>Internal reset by power-on-reset</li> <li>Internal reset by voltage detector</li> <li>Internal reset by illegal instruction execution <small>Note</small></li> <li>Internal reset by RAM parity error</li> <li>Internal reset by illegal-memory access</li> </ul>	
Power-on-reset circuit	<ul style="list-style-type: none"> <li>Power-on-reset: 1.51 ±0.04 V (TA = -40 to +85°C) 1.51 ±0.06 V (TA = -40 to +105°C)</li> <li>Power-down-reset: 1.50 ±0.04 V (TA = -40 to +85°C) 1.50 ±0.06 V (TA = -40 to +105°C)</li> </ul>	
Voltage detector	1.63 V to 4.06 V (14 stages)	
On-chip debug function	Provided	
Power supply voltage	VDD = 1.6 to 5.5 V (TA = -40 to +85°C) VDD = 2.4 to 5.5 V (TA = -40 to +105°C)	
Operating ambient temperature	TA = -40 to +85°C (A: Consumer applications, D: Industrial applications), TA = -40 to +105°C (G: Industrial applications)	

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution is not issued by emulation with the in-circuit emulator or on-chip debug emulator.

[80-pin, 100-pin products (code flash memory 384 KB to 512 KB)]

**Caution This outline describes the functions at the time when Peripheral I/O redirection register 0, 1 (PIOR0, 1) are set to 00H.**

(1/2)

Item		80-pin	100-pin		
		R5F104Mx (x = K, L)	R5F104Px (x = K, L)		
Code flash memory (KB)		384 to 512			
Data flash memory (KB)		8			
RAM (KB)		32 to 48 Note			
Address space		1 MB			
Main system clock	High-speed system clock	X1 (crystal/ceramic) oscillation, external main system clock input (EXCLK) HS (high-speed main) mode: 1 to 20 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
	High-speed on-chip oscillator clock ( $f_{IH}$ )	HS (high-speed main) mode: 1 to 32 MHz ( $V_{DD} = 2.7$ to 5.5 V), HS (high-speed main) mode: 1 to 16 MHz ( $V_{DD} = 2.4$ to 5.5 V), LS (low-speed main) mode: 1 to 8 MHz ( $V_{DD} = 1.8$ to 5.5 V), LV (low-voltage main) mode: 1 to 4 MHz ( $V_{DD} = 1.6$ to 5.5 V)			
Subsystem clock		XT1 (crystal) oscillation, external subsystem clock input (EXCLKS) 32.768 kHz			
Low-speed on-chip oscillator clock		15 kHz (TYP.): $V_{DD} = 1.6$ to 5.5 V			
General-purpose register		8 bits × 32 registers (8 bits × 8 registers × 4 banks)			
Minimum instruction execution time	0.03125 µs (High-speed on-chip oscillator clock: $f_{IH} = 32$ MHz operation)				
	0.05 µs (High-speed system clock: $f_{MX} = 20$ MHz operation)				
	30.5 µs (Subsystem clock: $f_{SUB} = 32.768$ kHz operation)				
Instruction set		<ul style="list-style-type: none"> <li>• Data transfer (8/16 bits)</li> <li>• Adder and subtractor/logical operation (8/16 bits)</li> <li>• Multiplication (8 bits × 8 bits, 16 bits × 16 bits), Division (16 bits ÷ 16 bits, 32 bits ÷ 32 bits)</li> <li>• Multiplication and Accumulation (16 bits × 16 bits + 32 bits)</li> <li>• Rotate, barrel shift, and bit manipulation (Set, reset, test, and Boolean operation), etc.</li> </ul>			
I/O port	Total	74	92		
	CMOS I/O	64	82		
	CMOS input	5	5		
	CMOS output	1	1		
	N-ch open-drain I/O (6 V tolerance)	4	4		
Timer	16-bit timer	12 channels (TAU: 8 channels, Timer RJ: 1 channel, Timer RD: 2 channels, Timer RG: 1 channel)			
	Watchdog timer	1 channel			
	Real-time clock (RTC)	1 channel			
	12-bit interval timer	1 channel			
	Timer output	Timer outputs: 18 channels PWM outputs: 12 channels			
	RTC output	1 • 1 Hz (subsystem clock: $f_{SUB} = 32.768$ kHz)			

**Note** In the case of the 48 KB, this is about 47 KB when the self-programming function and data flash function are used (For details, see **CHAPTER 3**).

(2/2)

Item	80-pin	100-pin
	R5F104Mx (x = K, L)	R5F104Px (x = K, L)
Clock output/buzzer output	2	2
	• 2.44 kHz, 4.88 kHz, 9.76 kHz, 1.25 MHz, 2.5 MHz, 5 MHz, 10 MHz (Main system clock: fMAIN = 20 MHz operation) • 256 Hz, 512 Hz, 1,024 kHz, 2,048 kHz, 4,096 kHz, 8,192 kHz, 16,384 kHz, 32,768 kHz (Subsystem clock: fSUB = 32.768 kHz operation)	
8/10-bit resolution A/D converter	17 channels	20 channels
D/A converter	2 channels	2 channels
Comparator	2 channels	2 channels
Serial interface	[80-pin, 100-pin products] • CSI: 2 channels/UART (UART supporting LIN-bus): 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels • CSI: 2 channels/UART: 1 channel/simplified I <sup>2</sup> C: 2 channels	
I <sup>2</sup> C bus	2 channels	2 channels
Data transfer controller (DTC)	39 sources	39 sources
Event link controller (ELC)	Event input: 26 Event trigger output: 9	
Vecored interrupt sources	Internal External	32 13
Key interrupt		8
Reset	• Reset by <u>RESET</u> pin • Internal reset by watchdog timer • Internal reset by power-on-reset • Internal reset by voltage detector • Internal reset by illegal instruction execution Note • Internal reset by RAM parity error • Internal reset by illegal-memory access	32 13
Power-on-reset circuit	• Power-on-reset: 1.51 ±0.04 V (TA = -40 to +85°C) 1.51 ±0.06 V (TA = -40 to +105°C) • Power-down-reset: 1.50 ±0.04 V (TA = -40 to +85°C) 1.50 ±0.06 V (TA = -40 to +105°C)	
Voltage detector	1.63 V to 4.06 V (14 stages)	
On-chip debug function	Provided	
Power supply voltage	VDD = 1.6 to 5.5 V (TA = -40 to +85°C) VDD = 2.4 to 5.5 V (TA = -40 to +105°C)	
Operating ambient temperature	TA = -40 to +85°C (A: Consumer applications, D: Industrial applications), TA = -40 to +105°C (G: Industrial applications)	

**Note**

The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution is not issued by emulation with the in-circuit emulator or on-chip debug emulator.

## CHAPTER 2 PIN FUNCTIONS

### 2.1 Port Functions

Pin I/O buffer power supplies depend on the product. The relationship between these power supplies and the pins is shown below.

**Table 2 - 1 Pin I/O Buffer Power Supplies**

- (1) 30-pin, 32-pin, 36-pin, 40-pin, 44-pin, 48-pin, 52-pin products

Power Supply	Corresponding Pins
VDD	All pins

- (2) 64-pin products

Power Supply	Corresponding Pins
EVDD0	Port pins other than P20 to P27, P121 to P124, and P137
VDD	<ul style="list-style-type: none"> <li>• P20 to P27, P121 to P124, and P137</li> <li>• <u>RESET</u> and REGC</li> </ul>

- (3) 80-pin products

Power Supply	Corresponding Pins
EVDD0	Port pins other than P20 to P27, P121 to P124, P137, and P150 to P153
VDD	<ul style="list-style-type: none"> <li>• P20 to P27, P121 to P124, P137, and P150 to P153</li> <li>• <u>RESET</u> and REGC</li> </ul>

- (4) 100-pin products

Power Supply	Corresponding Pins
EVDD0, EVDD1	Port pins other than P20 to P27, P121 to P124, P137, and P150 to P156
VDD	<ul style="list-style-type: none"> <li>• P20 to P27, P121 to P124, P137, and P150 to P156</li> <li>• <u>RESET</u> and REGC</li> </ul>

**Caution** Set EVDD0 and EVDD1 to the same potential.

Set in each port I/O, buffer, pull-up resistor is also valid for alternate functions.

### 2.1.1 30-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	AN117/TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output ( $V_{DD}$ tolerance). P00 and P01 can be set to analog input Note 1.
P01	8-1-3			AN116/TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRD1OD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P11	7-1-8			SI11/SDA11/TRD1OC1	
P12	7-1-7			SO11/TRD1OB1	
P13	7-1-8			TxD2/SO20/TRD1OA1	
P14	8-1-8			RxD2/SI20/SDA20/TRD1OD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRD1OB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRD1OC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRD1OA0/TRD1CLK/(TxD0)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 4-bit I/O port. Input/output can be specified in 1-bit units. Can be set to analog input Note 2.
P21				AN11/AVREFM	
P22				AN12	
P23				AN13	
P30	8-1-4	I/O	Input port	INTP3/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/SSI00/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 2-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 and P61 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 2-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified.
P121	2-2-1	Input	Input port	X1	
P122				X2/EXCLK	
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.
P147	7-3-3	I/O	Analog function	ANI18	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-2-1	Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.2 32-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	ANI17/TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P00 and P01 can be set to analog input Note 1.
P01	8-1-3			ANI16/TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRD1OD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P11	7-1-8			SI11/SDA11/TRD1OC1	
P12	7-1-7			SO11/TRD1OB1/(INTP5)	
P13	7-1-8			TxD2/SO20/TRD1OA1	
P14	8-1-8			RxD2/SI20/SDA20/TRD1OD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRD1OB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRD1OC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRD1OA0/TRD1CLK/(TxD0)	
P20	4-3-3	I/O	Analog function	ANIO/AVREFP	Port 2. 4-bit I/O port. Input/output can be specified in 1-bit units. Can be set to analog input Note 2
P21				AN1/AVREFM	
P22				AN12	
P23				AN13	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P62 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P62				SSI00	
P70	7-1-3	I/O	Input port	—	Port 7. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 2-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified.
P121	2-2-1	Input	Input port	X1	
P122				X2/EXCLK	
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.
P147	7-3-3	I/O	Analog function	ANI18	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.3 36-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/(TRJ00)	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P11	7-1-8			SI11/SDA11/TRDIOC1	
P12	7-1-7			SO11/TRDIOB1/(INTP5)	
P13	7-1-8			TxD2/SO20/TRDIOA1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIODO0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(TxD0)	
P20	4-3-3	I/O	Analog function	ANIO0/AV <sub>REFP</sub>	Port 2. 6-bit I/O port. Input/output can be specified in 1-bit units. Can be set to analog input Note 2.
P21				ANIO1/AV <sub>REFM</sub>	
P22				ANIO2	
P23				ANIO3	
P24				ANIO4	
P25				ANIO5	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P62 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P62				SSI00	
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P71	7-1-4			KR1/SI21/SDA21	
P72	7-1-3			KR2/SO21	
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 2-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified.
P121	2-2-1	Input	Input port	X1	Port 13. 1-bit input-only port.
P122				X2/EXCLK	
P137	2-1-2	Input	Input port	INTP0	Port 14. 1-bit I/O port. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
P147	7-3-3	I/O	Analog function	ANI18	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.
RESET	2-2-1	Input	—	—	

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.4 40-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/(TRJ00)	Port 0. 2-bit I/O port.
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port.
P11	7-1-8			SI11/SDA11/TRDIOC1	Input/output can be specified in 1-bit units.
P12	7-1-7			SO11/TRDIOB1/(INTP5)	Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P13	7-1-8			TxD2/SO20/TRDIOA1	Input of P10 and P14 to P17 can be set to TTL input buffer.
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(TxD0)	
P20	4-3-3	I/O	Analog function	ANIO/AVREFP	Port 2. 7-bit I/O port.
P21				ANI1/AVREFM	Input/output can be specified in 1-bit units.
P22				ANI2	Can be set to analog input Note 2.
P23				ANI3	
P24				ANI4	
P25				ANI5	
P26				ANI6	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port.
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P62 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3			
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog input.		
P121	2-2-1	Input	Input port	X1	For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.		
P147	7-3-3	I/O	Analog function	ANI18	Port 14. 1-bit I/O port. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.		
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.5 44-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port.
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port.
P11	7-1-8			SI11/SDA11/TRDIOC1	Input/output can be specified in 1-bit units.
P12	7-1-7			SO11/TRDIOB1/(INTP5)	Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P13	7-1-8			TxD2/SO20/TRDIOA1	Input of P10 and P14 to P17 can be set to TTL input buffer.
P14	8-1-8			RxD2/SDA20/TRDIOD0/(SCLA0)	Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(TxDO)	
P20	4-3-3	I/O	Analog function	ANIO/AVREFP	Port 2. 8-bit I/O port.
P21				ANI1/AVREFM	Input/output can be specified in 1-bit units.
P22				ANI2	Can be set to analog input Note 2.
P23				ANI3	
P24				ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port.
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P40	7-1-3	I/O	Input port	TOOL0 (TRJ00)	Port 4. 2-bit I/O port.
P41					Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P63				—			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3			
P120	7-3-3	I/O	Analog function	ANI19	Port 12.		
P121	2-2-1	Input	Input port	X1	1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.		
P146	7-1-3	I/O	Input port	—	Port 14. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P147	7-3-3		Analog function	ANI18			
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.6 48-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port.
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port.
P11	7-1-8			SI11/SDA11/TRDIOC1	Input/output can be specified in 1-bit units.
P12	7-1-7			SO11/TRDIOB1/(INTP5)	Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P13	7-1-8			TxD2/SO20/TRDIOA1	Input of P10 and P14 to P17 can be set to TTL input buffer.
P14	8-1-8			RxD2/SDA20/TRDIO0/(SCLA0)	Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(RxD0)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(TxD0)	
P20	4-3-3	I/O	Analog function	ANIO/AVREFP	Port 2. 8-bit I/O port.
P21				ANI1/AVREFM	Input/output can be specified in 1-bit units.
P22				ANI2	Can be set to analog input Note 2.
P23				ANI3	
P24				ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port.
P31	7-1-3			TI03/TO03/INTP4/(PCLBUZ0)/(TRJ00)	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P40	7-1-3	I/O	Input port	TOOL0 (TRJ00)	Port 4. 2-bit I/O port.
P41					Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P62				SSI00	
P63				—	
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P71	7-1-4			KR1/SI21/SDA21	
P72	7-1-3			KR2/SO21	
P73	KR3/SO01				
P74	7-1-4			KR4/INTP8/SI01/SDA01	
P75	7-1-3			KR5/INTP9/SCK01/SCL01	
P120	7-3-3	I/O	Analog function	ANI19	Port 12.
P121	2-2-1	Input	Input port	X1	1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.
P122				X2/EXCLK	
P123				XT1	
P124				XT2/EXCLKS	
P130	1-1-1	Output	Output port	—	Port 13.
P137	2-1-2	Input	Input port	INTP0	1-bit output-only port and 1-bit input-only port.
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P146				—	
P147	7-3-3	Analog function	ANI18	P147 can be set to analog input Note.	Port 14. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-1-1	Input	—	—	
					Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.7 52-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/TRJ00	Port 0. 4-bit I/O port.
P01	8-1-3			TO00/TRGCLKB/TRJIO0	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P02	7-3-4		Analog function	ANI17/TxD1	Input of P01 and P03 can be set to TTL input buffer. Output of P00, P02, and P03 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P03	8-3-4			ANI16/RxD1	P02 and P03 can be set to analog input Note 1.
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port.
P11	7-1-8			SI11/SDA11/TRDIOC1	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P12	7-1-7			SO11/TRDIOB1/(INTP5)	Input of P10 and P14 to P17 can be set to TTL input buffer.
P13	7-1-8			TxD2/SO20/TRDIOA1	Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P14	8-1-8			RxD2/SDA20/TRDIO0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(RxDO)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(TxDO)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 8-bit I/O port.
P21				AN11/AVREFM	Input/output can be specified in 1-bit units. Can be set to analog input Note 2.
P22				AN12	
P23				AN13	
P24				AN14	
P25				AN15	
P26				AN16	
P27				AN17	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port.
P31	7-1-3			TI03/TO03/INTP4/(TRJIO0)/(PCLBUZ0)	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 4-bit I/O port.
P41				(TRJIO0)	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62	—			SSI00			
P63				—			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3/SO01			
P74	7-1-4			KR4/INTP8/SI01/SDA01			
P75	7-1-3			KR5/INTP9/SCK01/SCL01			
P76				KR6/INTP10/(RxD2)			
P77				KR7/INTP11/(TxD2)			
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog input.		
P121	2-2-1	Input	Input port	X1	For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.		
P137	2-1-2	Input	Input port	INTP0			
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P146				—			
P147	7-3-3	Input	Analog function	ANI18	P147 can be set to analog input Note. Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.		
RESET	2-1-1		—	—			

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.8 64-pin (Code Flash Memory 16 KB to 64 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/TRJ00	Port 0. 7-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01, P03 and P04 can be set to TTL input buffer.
P01	8-1-3			TO00/TRGCLKB/TRJIO0	
P02	7-3-4		Analog function	ANI17/SO10/TxD1	
P03	8-3-4			ANI16/SI10/RxD1/SDA10	
P04	8-1-4		Input port	SCK10/SCL10	
P05	7-1-3			(INTP10)	
P06				(INTP11)/(TRJIO0)	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer.
P11	7-1-8			SI11/SDA11/TRDIOC1	
P12	7-1-7			SO11/TRDIOB1/(INTP5)	
P13	7-1-8			TxD2/SO20/TRDIOA1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	
P15				SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-1-7			TI01/TO01/INTP5/TRDIOC0/(SI00/RxD0)	
P17	8-1-8			TI02/TO02/TRDIOA0/TRDCLK/(SO00/TxD0)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. Can be set to analog input Note 2.
P21				ANI1/AVREFM	
P22				ANI2	
P23				ANI3	
P24				ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer.
P31	7-1-3			TI03/TO03/INTP4/(TRJIO0)/(PCLBUZ0)	
					Output of P30 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P41				(TRJIO0)			
P42				(INTP8)			
P43				(INTP9)			
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 and P55 can be set to TTL input buffer. Output of P50, P51, and P55 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P51	7-1-4	INTP2/SO00/TxD0/TOOLTxD/TRGIOB					
P52	7-1-3			(INTP1)			
P53				(INTP2)			
P54				(INTP3)			
P55	8-1-4	(INTP4)/(PCLBUZ1)/(SCK00)					
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P63				—			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3/SO01			
P74	7-1-4			KR4/INTP8/SI01/SDA01			
P75	7-1-3			KR5/INTP9/SCK01/SCL01			
P76				KR6/INTP10/(Rx2D)			
P77				KR7/INTP11/(Tx2D)			
P120	7-3-3	I/O	Analog function	ANI19	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P121	2-2-1	Input	Input port	X1			
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.		
P137	2-1-2	Input	Input port	INTP0			
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.		
P141				PCLBUZ1/INTP7			
P146				—			
P147	7-3-3		Analog function	ANI18			
RESET	2-2-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.9 30-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	ANI17/TI00/TxD1/TRGCLKA/(TRJ00)	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output ( $V_{DD}$ tolerance). P00 and P01 can be set to analog input Note 1.
P01	8-1-3			ANI16/TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRD1OD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output ( $V_{DD}$ tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDI0C1	
P12	7-6-6			SO11/TRD1OB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRD1OA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRD1OD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRD1OB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRD1OC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRD1OA0/TRD1CLK/IVC1P0/(TxD0)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 4-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P23 to P27 can be set to analog input Note 2. P22 can be set to analog input or analog output Note 2.
P21				AN11/AVREFM	
P22	4-8-1			AN12/ANO0	
P23	4-3-3			AN13	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/SSI00/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 2-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 and P61 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 2-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified.
P121	2-2-1	Input	Input port	X1	
P122				X2/EXCLK	
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.
P147	7-3-3	I/O	Analog function	ANI18/VCOUT1	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.10 32-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	ANI17/TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P00 and P01 can be set to analog input Note 1.
P01	8-1-3			ANI16/TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRD1OD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDI0C1	
P12	7-6-6			SO11/TRD1OB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRD1OA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRD1OD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRD1OB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRD1OC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRD1OA0/TRD1CLK/IVCMPO/(TxDO)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 4-bit I/O port. Input/output can be specified in 1-bit units. P20 and P21 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				ANI1/AVREFM	
P22	4-8-1			ANI2/AN00	
P23				ANI3/AN01	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P62 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P62				SSI00	
P70	7-1-3	I/O	Input port	—	Port 7. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 2-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified.
P121	2-2-1	Input	Input port	X1	
P122				X2/EXCLK	
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.
P147	7-3-3	I/O	Analog function	ANI18/VCOUT1	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.11 36-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ100	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDIOC1	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMP0/(TxD0)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 6-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, P24, and P25 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				AN11/AVREFM	
P22	4-8-1			AN12/ANO0	
P23				AN13/ANO1	
P24	4-3-3			AN14	
P25				AN15	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ100)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. Input/output can be specified. 1-bit I/O port. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port.
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port.
P61				SDAA0	Input/output can be specified in 1-bit units.
P62				SSI00	Output of P60 to P62 is N-ch open-drain output (6 V tolerance).
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 3-bit I/O port.
P71	7-1-4			KR1/SI21/SDA21	Input/output can be specified in 1-bit units.
P72	7-1-3			KR2/SO21	Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 2-bit input-only port.
P121	2-2-1	Input	Input port	X1	P120 can be set to analog input.
P122				X2/EXCLK	For only P120, input/output can be specified.
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.
P147	7-3-3	I/O	Analog function	ANI18/VCOUT1	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-2-1	Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.12 40-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (Vdd tolerance).
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (Vdd tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDIOC1	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SDA20/TRDIODO/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMP0/(TxDO)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 7-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P26 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				ANI1/AVREFM	
P22	4-8-1			ANI2/AN00	
P23				ANI3/AN01	
P24	4-3-3			ANI4	
P25				ANI5	
P26				ANI6	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (Vdd tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 3-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P62 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3			
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog input.		
P121	2-2-1	Input	Input port	X1	For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.		
P147	7-3-3	I/O	Analog function	ANI18/VCOUT1	Port 14. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.		
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.13 44-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDIOC1	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SDA20/TRDIODO/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMP0/(TxDO)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				AN11/AVREFM	
P22	4-8-1			AN12/AN00	
P23				AN13/AN01	
P24	4-3-3			AN14	
P25				AN15	
P26				AN16	
P27				AN17	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/PCLBUZ0/(TRJ00)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P41				(TRJ00)	

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P63				—			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3			
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12.		
P121	2-2-1	Input	Input port	X1	1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P137	2-1-2	Input	Input port	INTP0	Port 13. 1-bit input-only port.		
P146	7-1-3	I/O	Input port	—	Port 14. P147 can be set to analog input. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P147	7-3-3		Analog function	ANI18/VCOUT1			
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.14 48-pin (Code Flash Memory 96 KB to 512 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TxD1/TRGCLKA/TRJ00	Port 0. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 can be set to TTL input buffer. Output of P00 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P01	8-1-3			TO00/RxD1/TRGCLKB/TRJ00	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRD1OD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRD1OC1/(RxD0_1) Note 3	
P12	7-6-6			SO11/TRD1OB1/IVREF1/(INTP5)/(TxD0_1) Note 3	
P13	7-6-8			TxD2/SO20/TRD1OA1/IVC1MP1	
P14	8-1-8			RxD2/SDA20/SDA20/TRD1OD0/(SCLA0)	
P15				PCLBUZ1/SCK20/SCL20/TRD1OB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRD1OC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRD1OA0/TRD1CLK/IVC1MP0/(TxD0)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				AN11/AVREFM	
P22	4-8-1			AN12/ANO0	
P23				AN13/ANO1	
P24	4-3-3			AN14	
P25				AN15	
P26				AN16	
P27				AN17	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (V <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/(TRJ00)/(PCLBUZ0)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P41				(TRJ00)	

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Note 3.** Mounted on the 384 KB or more code flash memory products.

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).
P61				SDAA0	
P62				SSI00	
P63				—	
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P71	7-1-4			KR1/SI21/SDA21	
P72	7-1-3			KR2/SO21	
P73				KR3/SO01	
P74	7-1-4			KR4/INTP8/SI01/SDA01	
P75	7-1-3			KR5/INTP9/SCK01/SCL01	
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12.
P121	2-2-1	Input	Input port	X1	1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.
P122				X2/EXCLK	
P123				XT1	
P124				XT2/EXCLKS	
P130	1-1-1	Output	Output port	—	Port 13.
P137	2-1-2	Input	Input port	INTP0	1-bit output-only port and 1-bit input-only port.
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P146				—	
P147	7-3-3	Analog function	ANI18/VCOUT1	—	P147 can be set to analog input Note.
RESET	2-1-1			—	
		Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.15 52-pin (Code Flash Memory 96 KB to 256 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/(TRJ00)	Port 0. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01 and P03 can be set to TTL input buffer. Output of P00, P02, and P03 can be set to N-ch open-drain output ( $V_{DD}$ tolerance). P02 and P03 can be set to analog input Note 1
P01	8-1-3			TO00/TRGCLKB/TRJIO0	
P02	7-3-4		Analog function	ANI17/TxD1	
P03	8-3-4			ANI16/RxD1	
P10	8-1-8		Input port	SCK11/SCL11/TRDIOD1	
P11	7-1-8			SI11/SDA11/TRDI0C1	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output ( $V_{DD}$ tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P15				PCLBUZ1/SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMPO/(TxD0)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				ANI1/AVREFM	
P22	4-8-1			ANI2/ANO0	
P23				ANI3/ANO1	
P24	4-3-3			ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P31	7-1-3			TI03/TO03/INTP4/(TRJIO0)/(PCLBUZ0)	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P41				(TRJIO0)	

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/ TRGIOA/(TRJ00)	Port 5. 2-bit I/O port.
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB	Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 can be set to TTL input buffer. Output of P50 and P51 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port.
P61				SDAA0	Input/output can be specified in 1-bit units.
P62				SSI00	Output of P60 to P63 is N-ch open-drain output (6 V tolerance).
P63				—	
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port.
P71	7-1-4			KR1/SI21/SDA21	Input/output can be specified in 1-bit units.
P72	7-1-3			KR2/SO21	Use of an on-chip pull-up resistor can be specified by a software setting at input port.
P73				KR3/SO01	
P74	7-1-4			KR4/INTP8/SI01/SDA01	Output of P71 and P74 can be set to N-ch open-drain output ( $V_{DD}$ tolerance).
P75	7-1-3			KR5/INTP9/SCK01/SCL01	
P76				KR6/INTP10/(RxD2)	
P77				KR7/INTP11/(TxD2)	
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 4-bit input-only port.
P121	2-2-1	Input	Input port	X1	P120 can be set to analog input.
P122				X2/EXCLK	For only P120, input/output can be specified.
P123				XT1	For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.
P124				XT2/EXCLKS	
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.
P137	2-1-2	Input	Input port	INTP0	
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 3-bit I/O port.
P146				—	Input/output can be specified in 1-bit units.
P147	7-3-3			ANI18/VCOUT1	Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to $V_{DD}$ directly or via a resistor when external reset is not used.

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.16 64-pin (Code Flash Memory 96 KB to 512 KB)

(1/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/(TRJ00)	Port 0. 7-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01, P03 and P04 can be set to TTL input buffer. Output of P00 and P02 to P04 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance). P02 and P03 can be set to analog input Note 1.
P01	8-1-3			TO00/TRGCLKB/TRJIO0	
P02	7-3-4		Analog function	ANI17/SO10/TxD1	
P03	8-3-4			ANI16/SI10/RxD1/SDA10	
P04	8-1-4		Input port	SCK10/SCL10	
P05	7-1-3			(INTP10)	
P06				(INTP11)/(TRJIO0)	
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDIOC1/(RxD0_1) Note 3	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)/(TxD0_1) Note 3	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SDA20/TRDIOD0/(SCLA0)	
P15				SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(SI00/RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVC�MP0/(SO00/TxD0)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				ANI1/AVREFM	
P22	4-8-1			ANI2/ANO0	
P23				ANI3/ANO1	
P24	4-3-3			ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units.
P31	7-1-3			TI03/TO03/INTP4/(TRJIO0)/(PCLBUZ0)	

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Note 3.** Mounted on the 384 KB or more code flash memory products.

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/2)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P41				(TRJIO0)			
P42				(INTP8)			
P43				(INTP9)			
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRx/SDA00/ TRGIOA/(TRJ00)	Port 5. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 and P55 can be set to TTL input buffer. Output of P50, P51, and P55 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P51	7-1-4	INTP2/SO00/TxD0/TOOLTx/DRGIOB					
P52	7-1-3			(INTP1)			
P53				(INTP2)			
P54				(INTP3)			
P55	8-1-4	(INTP4)/(PCLBUZ1)/(SCK00)					
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 4-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance).		
P61				SDAA0			
P62				SSI00			
P63				—			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3/SO01			
P74	7-1-4			KR4/INTP8/SI01/SDA01			
P75	7-1-3			KR5/INTP9/SCK01/SCL01			
P76				KR6/INTP10/(Rx2D)			
P77				KR7/INTP11/(Tx2D)			
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog input. For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P121	2-2-1	Input	Input port	X1			
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.		
P137	2-1-2	Input	Input port	INTP0			
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 4-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P147 can be set to analog input Note.		
P141				PCLBUZ1/INTP7			
P146				—			
P147	7-3-3		Analog function	ANI18/VCOUT1			
RESET	2-2-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.17 80-pin

(1/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/(TRJ00)	Port 0. 7-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01, P03 and P04 can be set to TTL input buffer.
P01	8-1-3			TO00/TRGCLKB/TRJ00	
P02	7-3-4		Analog function	ANI17/SO10/TxD1	
P03	8-3-4			ANI16/SI10/RxD1/SDA10	
P04	8-1-4		Input port	SCK10/SCL10	
P05	7-1-3			—	
P06				(TRJ00)	P02 and P03 can be set to analog input Note 1.
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer. Output of P10, P11, P13 to P15, and P17 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance). P12, P13, P16, and P17 can be set to analog input Note 1.
P11	7-1-8			SI11/SDA11/TRDI0C1/(RxDO_1) Note 3	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)/(TxDO_1) Note 3	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	
P15				SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDI0C0/IVREF0/(SI00/RxDO)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMP0/(SO00/TxD0)	
P20	4-3-3	I/O	Analog function	AN10/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				AN11/AVREFM	
P22	4-8-1			AN12/ANO0	
P23				AN13/ANO1	
P24	4-3-3			AN14	
P25				AN15	
P26				AN16	
P27				AN17	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer. Output of P30 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).
P31	7-1-3			TI03/TO03/INTP4/(TRJ00)/(PCLBUZ0)	

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Note 3.** Mounted on the 384 KB or more code flash memory products.

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P43 and P44 can be set to TTL input buffer. Output of P43 to P45 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P41				(TRJIO0)			
P42				(INTP8)			
P43				SCK01/SCL01/(INTP9)			
P44				SI01/SDA01			
P45				SO01			
P50	8-1-4	I/O	Input port	INTP1/SI00/RxD0/TOOLRxD/SDA00/TRGIOA/(TRJOO)	Port 5. 6-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 and P53 to P55 can be set to TTL input buffer. Output of P50 to P55 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P51	7-1-4			INTP2/SO00/TxD0/TOOLTxD/TRGIOB			
P52				SO31/(INTP1)			
P53	8-1-4			SI31/SDA31/(INTP2)			
P54				SCK31/SCL31/(INTP3)			
P55				(INTP4)/(PCLBUZ1)/(SCK00)			
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 8-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance). For P64 to P67, use of an on-chip pull-up resistor can be specified by a software setting at the input port.		
P61				SDAA0			
P62				SSI00/SCLA1			
P63				SDAA1			
P64	7-1-3			TI10/TO10			
P65				TI11/TO11			
P66				TI12/TO12			
P67				TI13/TO13			
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P71	7-1-4			KR1/SI21/SDA21			
P72	7-1-3			KR2/SO21			
P73				KR3			
P74	7-1-4			KR4/INTP8			
P75	7-1-3			KR5/INTP9			
P76				KR6/INTP10/(RxD2)			
P77				KR7/INTP11/(TxD2)			
P100	7-3-3	I/O	Analog function	ANI20/(INTP10)	Port 10. 1-bit I/O port. Input/output can be specified. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P100 can be set to analog input Note.		
P110	7-1-3	I/O	Input port	(INTP11)	Port 11. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P111				—			

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(3/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog I/O.
P121	2-2-1	Input	Input port	X1	For only P120, input/output can be specified.
P122				X2/EXCLK	For only P120, use of an on-chip pull-up resistor
P123				XT1	can be specified by a software setting at input port.
P124				XT2/EXCLKS	
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.
P137	2-1-2	Input	Input port	INTP0	
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 7-bit I/O port.
P141				PCLBUZ1/INTP7	Input/output can be specified in 1-bit units.
P142				SCK30/SCL30	Use of an on-chip pull-up resistor can be specified
P143				SI30/RxD3/SDA30	by a software setting at input port.
P144	7-1-4			SO30/TxD3	Input of P142 and P143 can be set to TTL input
P146	7-1-3			—	buffer. Output of P142 to P144 can be set to N-ch open-
P147	7-3-3		Analog function	ANI18/VCOUT1	drain output (EVDD tolerance). P147 can be set to analog I/O Note 1.
P150	4-3-3	I/O	Analog function	ANI8	Port 15. 4-bit I/O port.
P151				ANI9	Input/output can be specified in 1-bit units.
P152				ANI10	Can be set to analog input Note 2.
P153				ANI11	
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to Vdd directly or via a resistor when
					external reset is not used.

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

### 2.1.18 100-pin

(1/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function
P00	7-1-4	I/O	Input port	TI00/TRGCLKA/TRJ00	Port 0. 7-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P01, P03 and P04 can be set to TTL input buffer.
P01	8-1-3			TO00/TRGCLKB/TRJIO0	
P02	7-3-4		Analog function	ANI17/SO10/TxD1	
P03	8-3-4			ANI16/SI10/RxD1/SDA10	
P04	8-1-4		Input port	SCK10/SCL10	
P05	7-1-3			—	
P06				(TRJIO0)	P02 and P03 can be set to analog input Note 1.
P10	8-1-8	I/O	Input port	SCK11/SCL11/TRDIOD1	Port 1. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P10 and P14 to P17 can be set to TTL input buffer.
P11	7-1-8			SI11/SDA11/TRDIOC1/(RxDO_1) Note 3	
P12	7-6-6			SO11/TRDIOB1/IVREF1/(INTP5)/(TxDO_1) Note 3	
P13	7-6-8			TxD2/SO20/TRDIOA1/IVCMP1	
P14	8-1-8			RxD2/SI20/SDA20/TRDIOD0/(SCLA0)	
P15				SCK20/SCL20/TRDIOB0/(SDAA0)	
P16	8-6-6			TI01/TO01/INTP5/TRDIOC0/IVREF0/(SI00/RxD0)	
P17	8-6-8			TI02/TO02/TRDIOA0/TRDCLK/IVCMPO/(SO00/TxD0)	
P20	4-3-3	I/O	Analog function	ANI0/AVREFP	Port 2. 8-bit I/O port. Input/output can be specified in 1-bit units. P20, P21, and P24 to P27 can be set to analog input Note 2. P22 and P23 can be set to analog input or analog output Note 2.
P21				ANI1/AVREFM	
P22	4-8-1			ANI2/AN00	
P23				ANI3/AN01	
P24	4-3-3			ANI4	
P25				ANI5	
P26				ANI6	
P27				ANI7	
P30	8-1-4	I/O	Input port	INTP3/RTC1HZ/SCK00/SCL00/TRJ00	Port 3. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P30 can be set to TTL input buffer.
P31	7-1-3			TI03/TO03/INTP4/(TRJIO0)/(PCLBUZ0)	
					Output of P30 can be set to N-ch open-drain output (EVDD tolerance).

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Note 3.** Mounted on the 384 KB or more code flash memory products.

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(2/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function	
P40	7-1-3	I/O	Input port	TOOL0	Port 4. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P43 and P44 can be set to TTL input buffer. Output of P43 to P45 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).	
P41				(TRJIO0)		
P42				—		
P43				SCK01/SCL01		
P44				SI01/SDA01		
P45				SO01		
P46				INTP1		
P47				INTP2		
P50	8-1-4	I/O	Input port	SI00/RxD0/TOOLRxD/SDA00/TRGIOA(TRJ00)	Port 5. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P50 and P53 to P55 can be set to TTL input buffer. Output of P50 to P55 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).	
P51	7-1-4			SO00/TxD0/TOOLTxD/TRGIOB		
P52				SO31		
P53	8-1-4			SI31/SDA31		
P54				SCK31/SCL31		
P55				(PCLBUZ1)/(SCK00)		
P56	7-1-3			(INTP1)		
P57				(INTP3)		
P60	12-1-5	I/O	Input port	SCLA0	Port 6. 8-bit I/O port. Input/output can be specified in 1-bit units. Output of P60 to P63 is N-ch open-drain output (6 V tolerance). For P64 to P67, use of an on-chip pull-up resistor can be specified by a software setting at the input port.	
P61				SDAA0		
P62				SSI00/SCLA1		
P63				SDAA1		
P64		7-1-3		TI10/TO10		
P65				TI11/TO11		
P66				TI12/TO12		
P67				TI13/TO13		
P70	7-1-3	I/O	Input port	KR0/SCK21/SCL21	Port 7. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Output of P71 and P74 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).	
P71	7-1-4			KR1/SI21/SDA21		
P72	7-1-3			KR2/SO21		
P73				KR3		
P74	7-1-4			KR4/INTP8		
P75	7-1-3			KR5/INTP9		
P76				KR6/INTP10/(RxD2)		
P77				KR7/INTP11/(TxD2)		

**Note** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

(3/3)

Function Name	Pin Type	I/O	After Reset Release	Alternate Function	Function		
P80	8-1-4	I/O	Input port	(SCK10/SCL10)	Port 8. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P80 and P81 can be set to TTL input buffer. Output of P80 to P82 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance).		
P81				(SI10/RxD1/SDA10)			
P82				(SO10/TxD1)			
P83				—			
P84				(INTP6)			
P85				(INTP7)			
P86				(INTP8)			
P87				(INTP9)			
P100		I/O	Analog function	ANI20/(INTP10)	Port 10. 3-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. P100 can be set to analog input Note 1.		
P101	7-1-3			—			
P102				—			
P110	7-1-3	I/O	Input port	(INTP11)	Port 11. 2-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P111				—			
P120	7-3-3	I/O	Analog function	ANI19/VCOUT0	Port 12. 1-bit I/O port and 4-bit input-only port. P120 can be set to analog I/O.		
P121	2-2-1	Input	Input port	X1	For only P120, input/output can be specified. For only P120, use of an on-chip pull-up resistor can be specified by a software setting at input port.		
P122				X2/EXCLK			
P123				XT1			
P124				XT2/EXCLKS			
P130	1-1-1	Output	Output port	—	Port 13. 1-bit output-only port and 1-bit input-only port.		
P137	2-1-2	Input	Input port	INTP0			
P140	7-1-3	I/O	Input port	PCLBUZ0/INTP6	Port 14. 8-bit I/O port. Input/output can be specified in 1-bit units. Use of an on-chip pull-up resistor can be specified by a software setting at input port. Input of P142 and P143 can be set to TTL input buffer. Output of P142 to P144 can be set to N-ch open-drain output (EV <sub>DD</sub> tolerance). P147 can be set to analog I/O Note 1.		
P141				PCLBUZ1/INTP7			
P142	8-1-4			SCK30/SCL30			
P143				SI30/RxD3/SDA30			
P144	7-1-4			SO30/TxD3			
P145	7-1-3			—			
P146				(INTP4)			
P147	7-3-3			ANI18/VCOUT1			
P150	4-3-3	I/O	Analog function	ANI8	Port 15. 7-bit I/O port. Input/output can be specified in 1-bit units. Can be set to analog input Note 2.		
P151				ANI9			
P152				ANI10			
P153				ANI11			
P154				ANI12			
P155				ANI13			
P156				ANI14			
RESET	2-1-1	Input	—	—	Input-only pin for external reset. Connect to V <sub>DD</sub> directly or via a resistor when external reset is not used.		

**Note 1.** Each pin can be specified as either digital or analog by setting port mode control register x (PMCx) (Can be specified in 1-bit units).

**Note 2.** Each pin can be specified as either digital or analog by setting the A/D port configuration register (ADPC).

**Remark** Functions in parentheses in the above figure can be assigned via settings in the peripheral I/O redirection registers 0, 1 (PIOR0, 1).

## 2.2 Functions other than port pins

### 2.2.1 Functions for each product

(1/5)

Function Name	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
ANI0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI4	✓	✓	✓	✓	✓	✓	✓	✓	—	—
ANI5	✓	✓	✓	✓	✓	✓	✓	✓	—	—
ANI6	✓	✓	✓	✓	✓	✓	✓	—	—	—
ANI7	✓	✓	✓	✓	✓	✓	—	—	—	—
ANI8	✓	✓	—	—	—	—	—	—	—	—
ANI9	✓	✓	—	—	—	—	—	—	—	—
ANI10	✓	✓	—	—	—	—	—	—	—	—
ANI11	✓	✓	—	—	—	—	—	—	—	—
ANI12	✓	—	—	—	—	—	—	—	—	—
ANI13	✓	—	—	—	—	—	—	—	—	—
ANI14	✓	—	—	—	—	—	—	—	—	—
ANI16	✓	✓	✓	✓	—	—	—	—	✓	✓
ANI17	✓	✓	✓	✓	—	—	—	—	✓	✓
ANI18	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI19	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ANI20	✓	✓	—	—	—	—	—	—	—	—
ANO0	✓	✓	Note							
ANO1	✓	✓	Note	—						
INTP0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
INTP6	✓	✓	✓	✓	✓	—	—	—	—	—
INTP7	✓	✓	✓	—	—	—	—	—	—	—
INTP8	✓	✓	✓	✓	✓	—	—	—	—	—
INTP9	✓	✓	✓	✓	✓	—	—	—	—	—
INTP10	✓	✓	✓	✓	—	—	—	—	—	—
INTP11	✓	✓	✓	✓	—	—	—	—	—	—
IVCMP0	✓	✓	Note							
IVCMP1	✓	✓	Note							
IVREF0	✓	✓	Note							
IVREF1	✓	✓	Note							

**Note** Mounted on the 96 KB or more code flash memory products.

(2/5)

Function Name	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
KR0	✓	✓	✓	✓	✓	✓	✓	—	—	—
KR1	✓	✓	✓	✓	✓	✓	✓	—	—	—
KR2	✓	✓	✓	✓	✓	✓	✓	—	—	—
KR3	✓	✓	✓	✓	✓	✓	✓	—	—	—
KR4	✓	✓	✓	✓	✓	—	—	—	—	—
KR5	✓	✓	✓	✓	✓	—	—	—	—	—
KR6	✓	✓	✓	✓	—	—	—	—	—	—
KR7	✓	✓	✓	✓	—	—	—	—	—	—
PCLBUZ0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
PCLBUZ1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
REGC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RTC1HZ	✓	✓	✓	✓	✓	✓	✓	—	—	—
<u>RESET</u>	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RxD0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RxD1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RxD2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
RxD3	✓	✓	—	—	—	—	—	—	—	—
SCK00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCK01	✓	✓	✓	✓	✓	—	—	—	—	—
SCK10	✓	✓	✓	—	—	—	—	—	—	—
SCK11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCK20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCK21	✓	✓	✓	✓	✓	✓	✓	✓	—	—
SCK30	✓	✓	—	—	—	—	—	—	—	—
SCK31	✓	✓	—	—	—	—	—	—	—	—
SCLA0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCLA1	✓	✓	—	—	—	—	—	—	—	—

(3/5)

Function Name	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
SCL00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCL01	✓	✓	✓	✓	✓	—	—	—	—	—
SCL10	✓	✓	✓	—	—	—	—	—	—	—
SCL11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCL20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SCL21	✓	✓	✓	✓	✓	✓	✓	✓	—	—
SCL30	✓	✓	—	—	—	—	—	—	—	—
SCL31	✓	✓	—	—	—	—	—	—	—	—
SDAA00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SDAA10	✓	✓	—	—	—	—	—	—	—	—
SDA00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SDA01	✓	✓	✓	✓	✓	—	—	—	—	—
SDA10	✓	✓	✓	—	—	—	—	—	—	—
SDA11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SDA20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SDA21	✓	✓	✓	✓	✓	✓	✓	✓	—	—
SDA30	✓	✓	—	—	—	—	—	—	—	—
SDA31	✓	✓	—	—	—	—	—	—	—	—
SI00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SI01	✓	✓	✓	✓	✓	—	—	—	—	—
SI10	✓	✓	✓	—	—	—	—	—	—	—
SI11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SI20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SI21	✓	✓	✓	✓	✓	✓	✓	✓	—	—
SI30	✓	✓	—	—	—	—	—	—	—	—
SI31	✓	✓	—	—	—	—	—	—	—	—
SO00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SO01	✓	✓	✓	✓	✓	—	—	—	—	—
SO10	✓	✓	✓	—	—	—	—	—	—	—
SO11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SO20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SO21	✓	✓	✓	✓	✓	✓	✓	✓	—	—
SO30	✓	✓	—	—	—	—	—	—	—	—
SO31	✓	✓	—	—	—	—	—	—	—	—
SSI00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

(4/5)

Function Name	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
TI00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TI01	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TI02	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TI03	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TI10	✓	✓	—	—	—	—	—	—	—	—
TI11	✓	✓	—	—	—	—	—	—	—	—
TI12	✓	✓	—	—	—	—	—	—	—	—
TI13	✓	✓	—	—	—	—	—	—	—	—
TO00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TO01	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TO02	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TO03	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TO10	✓	✓	—	—	—	—	—	—	—	—
TO11	✓	✓	—	—	—	—	—	—	—	—
TO12	✓	✓	—	—	—	—	—	—	—	—
TO13	✓	✓	—	—	—	—	—	—	—	—
TRJIO0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRJO0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDCLK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOA0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOB0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOC0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOD0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOA1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOB1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOC1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRDIOD1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRGIOA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRGIQB	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRGCLKA	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TRGCLKB	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

(5/5)

Function Name	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
TxD0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TxD1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TxD2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TxD3	✓	✓	—	—	—	—	—	—	—	—
VCOOUT0	✓	✓	Note							
VCOOUT1	✓	✓	Note							
X1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
X2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EXCLK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EXCLKS	✓	✓	✓	✓	✓	✓	✓	—	—	—
XT1	✓	✓	✓	✓	✓	✓	✓	—	—	—
XT2	✓	✓	✓	✓	✓	✓	✓	—	—	—
VDD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EVDD0	✓	✓	✓	—	—	—	—	—	—	—
EVDD1	✓	—	—	—	—	—	—	—	—	—
AVREFP	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
AVREFM	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Vss	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EVSS0	✓	✓	✓	—	—	—	—	—	—	—
EVSS1	✓	—	—	—	—	—	—	—	—	—
TOOLRxD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TOOLTxD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
TOOL0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

**Note** Mounted on the 96 KB or more code flash memory products.

## 2.2.2 Pins for each product (pins other than port pins)

(1/2)

Function Name	I/O	Function
ANI0 to ANI14, ANI16 to ANI20	Input	A/D converter analog input (see <b>Figure 14 - 46 Analog Input Pin Connection</b> )
ANO0, ANO1	Output	D/A converter output
INTP0 to INTP11	Input	External interrupt request input pin for which the valid edge (rising edge, falling edge, or both rising and falling edges) can be specified.
IVCMP0, IVCMP1	Input	Comparator analog voltage input
IVREF0, IVREF1	Input	Comparator reference voltage input
VCOOUT0, VCOOUT1	Output	Comparator output
KR0 to KR7	Input	Key interrupt input
PCLBUZ0, PCLBUZ1	Output	Clock output/buzzer output
REGC	—	Pin for connecting regulator output stabilization capacitance for internal operation. Connect this pin to Vss via a capacitor (0.47 to 1 $\mu$ F). Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.
RTC1HZ	Output	Real-time clock correction clock (1 Hz) output
RESET	Input	This is the active-low system reset input pin. When the external reset pin is not used, connect this pin directly or via a resistor to VDD.
RxD0 to RxD3	Input	Serial data input pins of serial interface UART0 to UART3
TxD0 to TxD3	Output	Serial data output pins of serial interface UART0 to UART3
SCK00, SCK01, SCK10, SCK11, SCK20, SCK21, SCK30, SCK31	I/O	Serial clock I/O pins of serial interface CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, and CSI31
SCL00, SCL01, SCL10, SCL11, SCL20, SCL21, SCL30, SCL31	Output	Serial clock output pins of serial interface IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, and IIC31
SDA00, SDA01, SDA10, SDA11, SDA20, SDA21, SDA30, SDA31	I/O	Serial data I/O pins of serial interface IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, and IIC31
SI00, SI01, SI10, SI11, SI20, SI21, SI30, SI31	Input	Serial data input pins of serial interface CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, and CSI31
SSI00	Input	Chip select input pin of serial interface CSI00
SO00, SO01, SO10, SO11, SO20, SO21, SO30, SO31	Output	Serial data output pins of serial interface CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, and CSI31
SCLA0, SCLA1	I/O	Serial clock I/O pins of serial interface IICA0, IICA1
SDAA0, SDAA1	I/O	Serial data I/O pins of serial interface IICA0, IICA1
TI00 to TI03, TI10 to TI13	Input	The pins for inputting an external count clock/capture trigger to 16-bit timers 00 to 03, 10 to 13
TO00 to TO03, TO10 to TO13	Output	Timer output pins of 16-bit timers 00 to 03, 10 to 13
TRJIO0	I/O	Timer RJ input/output
TRJO0	Output	Timer RJ output
TRDCLK	Input	Timer RD external clock input

(2/2)

Function Name	I/O	Function
TRDIOA0, TRDIOB0, TRDIOC0, TRDIOD0, TRDIOA1, TRDIOB1, TRDIOC1, TRDIOD1	I/O	Timer RD input/output
TRGIOA, TRGIOB	I/O	Timer RG input/output
TRGCLKA, TRGCLKB	Input	Timer RG external clock input
X1, X2	—	Resonator connection for main system clock
EXCLK	Input	External clock input for main system clock
XT1, XT2	—	Resonator connection for subsystem clock
EXCLKS	Input	External clock input for subsystem clock
VDD	—	<30-pin, 32-pin, 36-pin, 40-pin, 44-pin, 48-pin, 52-pin> Positive power supply for all pins <64-pin, 80-pin, 100-pin > Positive power supply for P20 to P27, P121 to P124, P137, P150 to P156 and other than ports
EVDD0, EVDD1	—	Positive power supply for ports (other than P20 to P27, P121 to P124, P137, P150 to P156)
AVREFP	Input	A/D converter reference potential (+ side) input
AVREFM	Input	A/D converter reference potential (- side) input
Vss	—	<30-pin, 32-pin, 36-pin, 40-pin, 44-pin, 48-pin, 52-pin > Ground potential for all pins <64-pin, 80-pin, 100-pin > Ground potential for P20 to P27, P121 to P124, P137, P150 to P156 and other than ports
EVSS0, EVSS1	—	Ground potential for ports (other than P20 to P27, P121 to P124, P137, P150 to P156)
TOOLRxD	Input	UART reception pin for the external device connection used during flash memory programming
TOOLTxD	Output	UART transmission pin for the external device connection used during flash memory programming
TOOL0	I/O	Data I/O for flash memory programmer/debugger

**Caution** After reset release, the relationships between P40/TOOL0 and the operating mode are as follows.

**Table 2 - 2 Relationships Between P40/TOOL0 and Operation Mode After Reset Release**

P40/TOOL0	Operating mode
EVDD	Normal operation mode
0 V	Flash memory programming mode

For details, see 30.4 Programming Method.

**Remark** Use bypass capacitors (about 0.1  $\mu$ F) as noise and latch up countermeasures with relatively thick wires at the shortest distance to VDD to Vss, EVDD0 to EVSS0 and EVDD1 to EVSS1 lines.

## 2.3 Connection of Unused Pins

Table 2 - 3 shows the Connection of Unused Pins.

**Remark** The mounted pins depend on the product. Refer to **1.3 Pin Configuration (Top View)** and **2.1 Port Functions**.

**Table 2 - 3 Connection of Unused Pins**

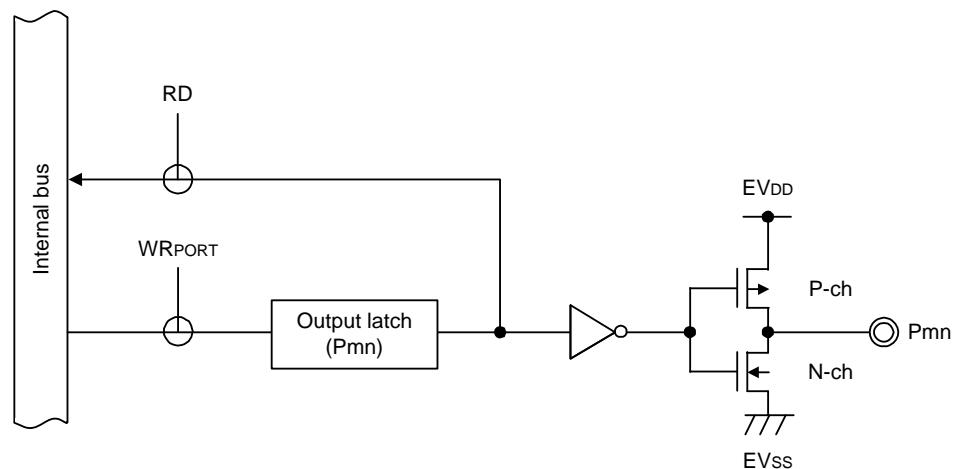
Pin Name	I/O	Recommended Connection of Unused Pins
P00 to P06	I/O	Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Leave open.
P10 to P17		
P20 to P27		Input: Independently connect to Vdd or Vss via a resistor. Output: Leave open.
P30, P31		Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Leave open.
P40/TOOL0		Input: Independently connect to EVdd0 via a resistor, or leave open. Output: Leave open.
P41 to P47		Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Leave open.
P50 to P57		
P60 to P63		Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Set the port's output latch to 0 and leave the pins open, or set the port's output latch to 1 and independently connect the pins to EVdd0 and EVdd1 or EVss0 and EVss1 via a resistor.
P64 to P67		Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Leave open.
P70 to P77		
P80 to P87		
P100 to P102		
P110, P111		
P120		
P121 to P124	Input	Independently connect to Vdd or Vss via a resistor.
P130	Output	Leave open.
P137	Input	Independently connect to Vdd or Vss via a resistor.
P140 to P147	I/O	Input: Independently connect to EVdd0, EVdd1 or EVss0, EVss1 via a resistor. Output: Leave open.
P150 to P156		Input: Independently connect to Vdd or Vss via a resistor. Output: Leave open.
RESET	Input	Connect to Vdd directly or via a resistor.
REGC	—	Connect to Vss via a capacitor (0.47 to 1 $\mu$ F).

**Remark** With products not provided with an EVdd0, EVdd1, EVss0, or EVss1 pin, replace EVdd0 and EVdd1 with Vdd, or replace EVss0 and EVss1 with Vss.

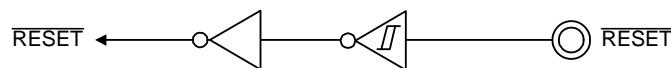
## 2.4 Pin Block Diagrams

For the pin types listed in 2.1.1 30-pin (Code Flash Memory 16 KB to 64 KB) to 2.1.18 100-pin, pin block diagrams are shown in Figures 2 - 1 to 2 - 22.

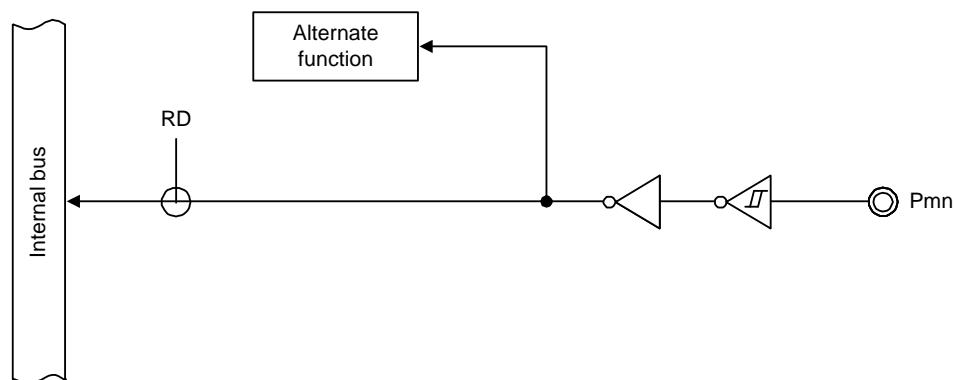
**Figure 2 - 1 Pin Block Diagram of Pin Type 1-1-1**



**Figure 2 - 2 Pin Block Diagram of Pin Type 2-1-1**

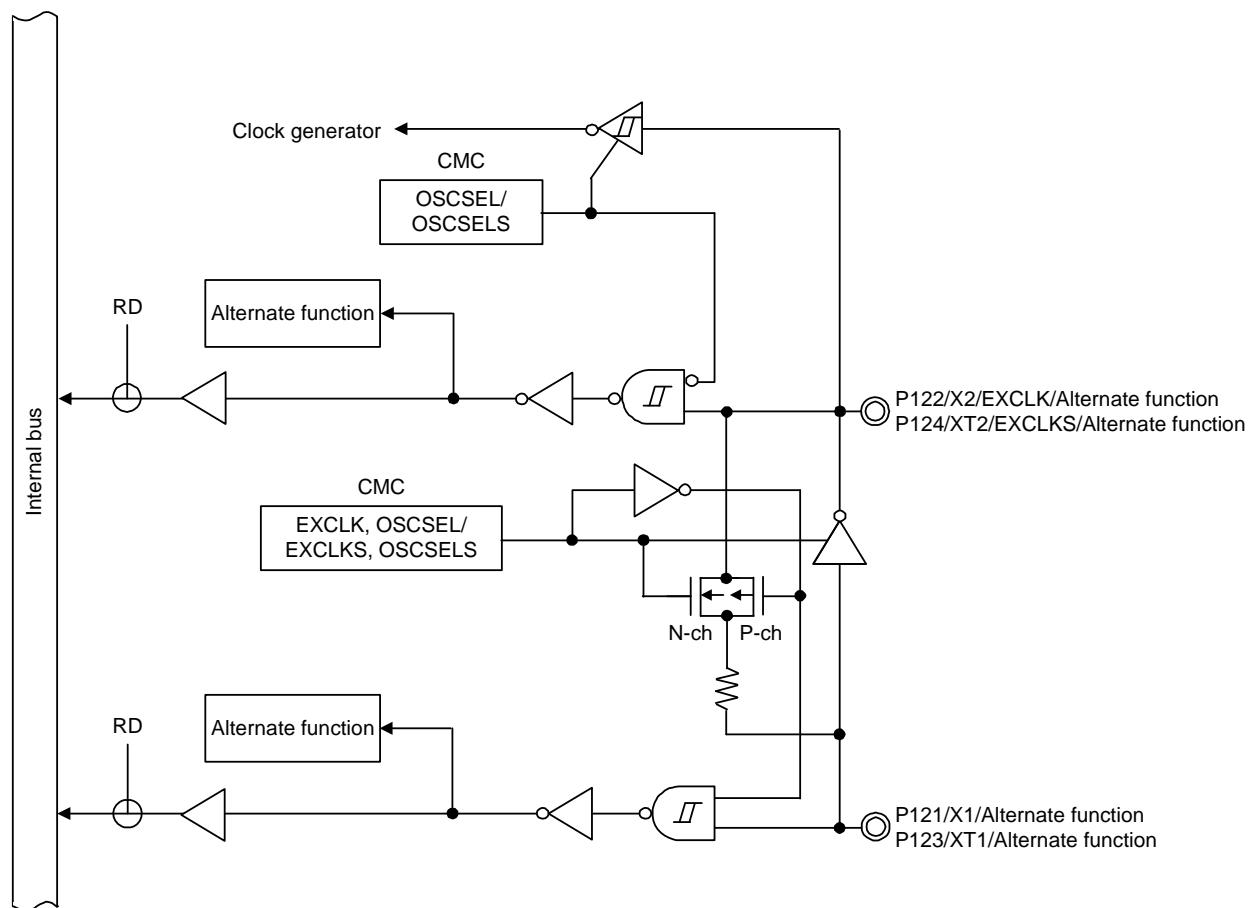


**Figure 2 - 3 Pin Block Diagram of Pin Type 2-1-2**



**Remark** Refer to **2.1 Port Functions** for alternate functions.

Figure 2 - 4 Pin Block Diagram of Pin Type 2-2-1



**Remark** Refer to **2.1 Port Functions** for alternate functions.

Figure 2 - 5 Pin Block Diagram of Pin Type 4-3-3

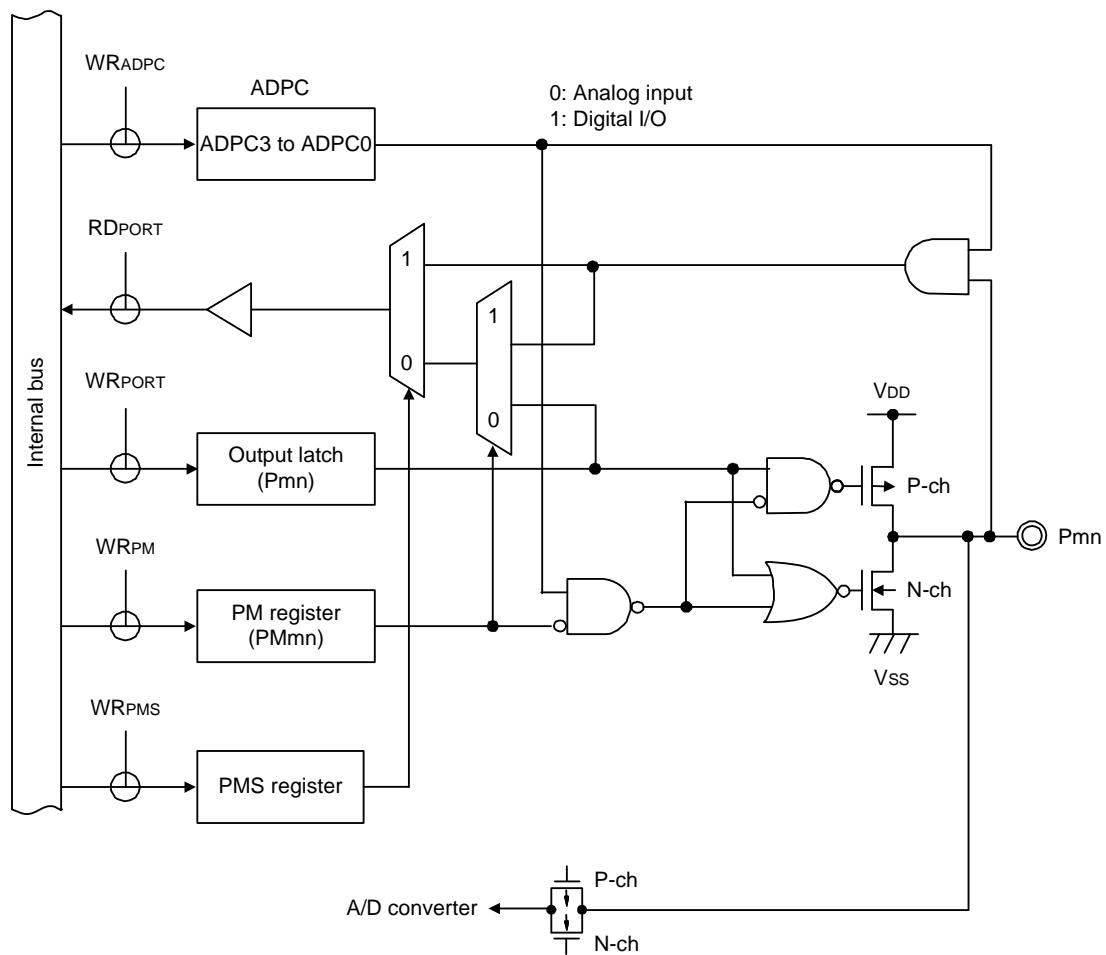


Figure 2 - 6 Pin Block Diagram of Pin Type 4-8-1

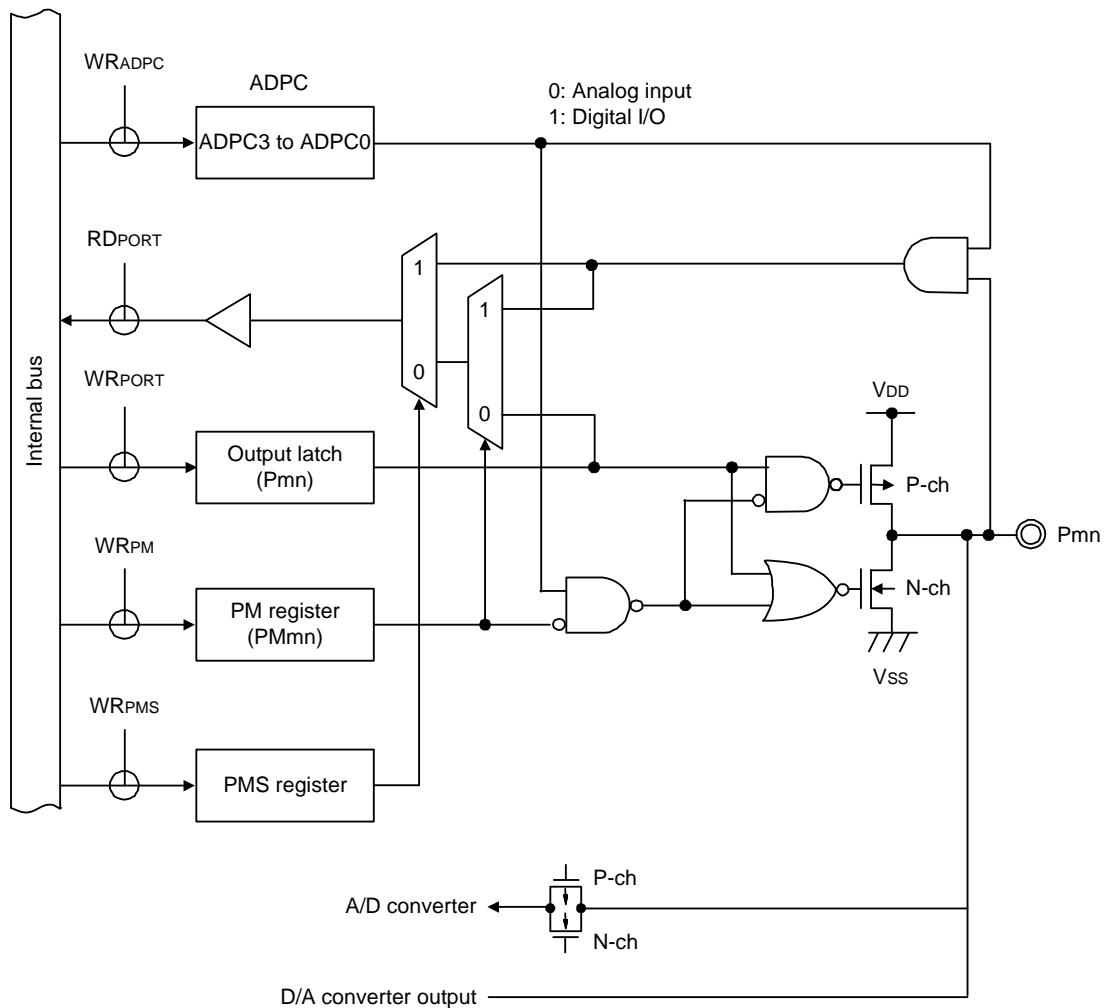
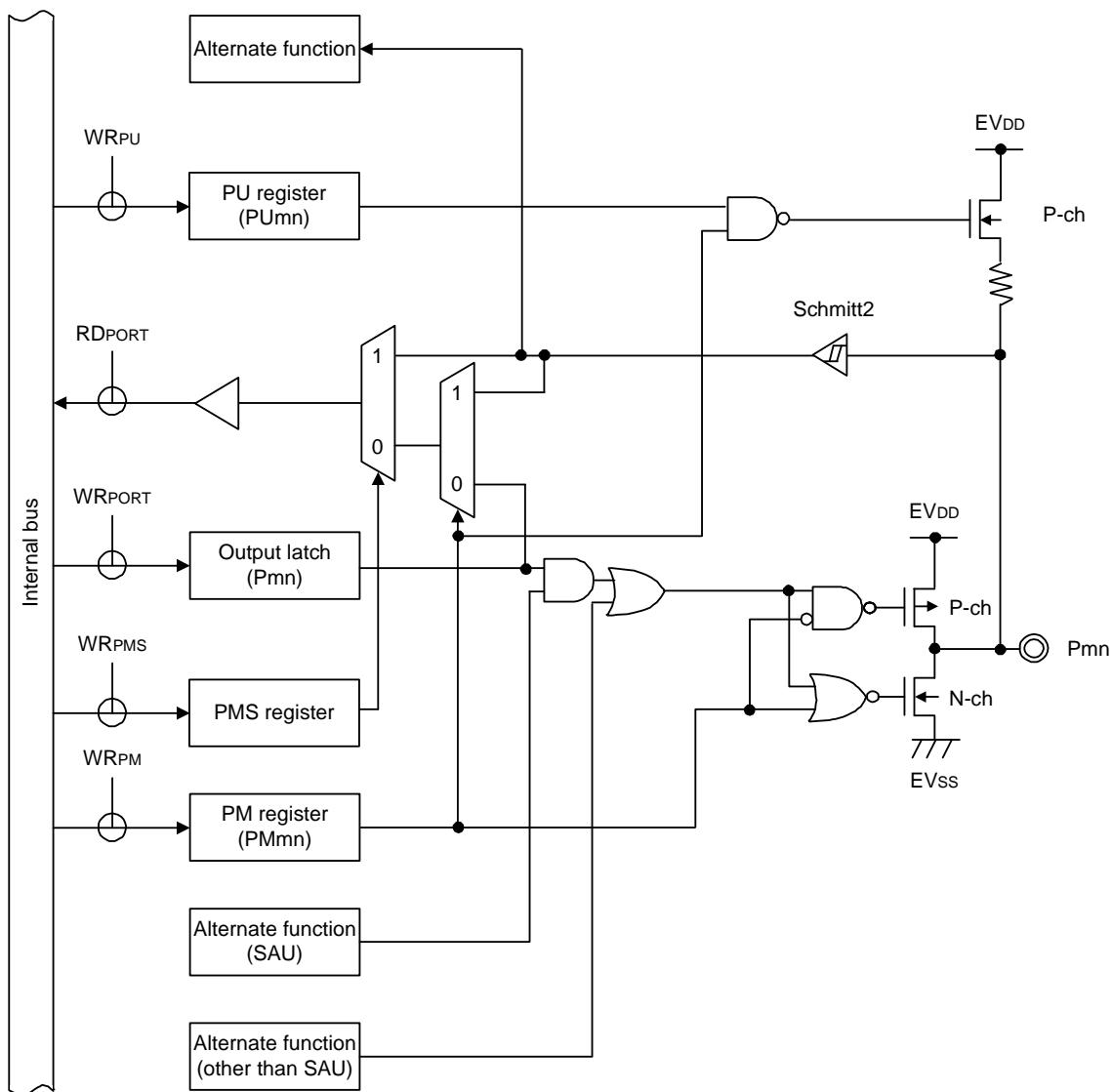


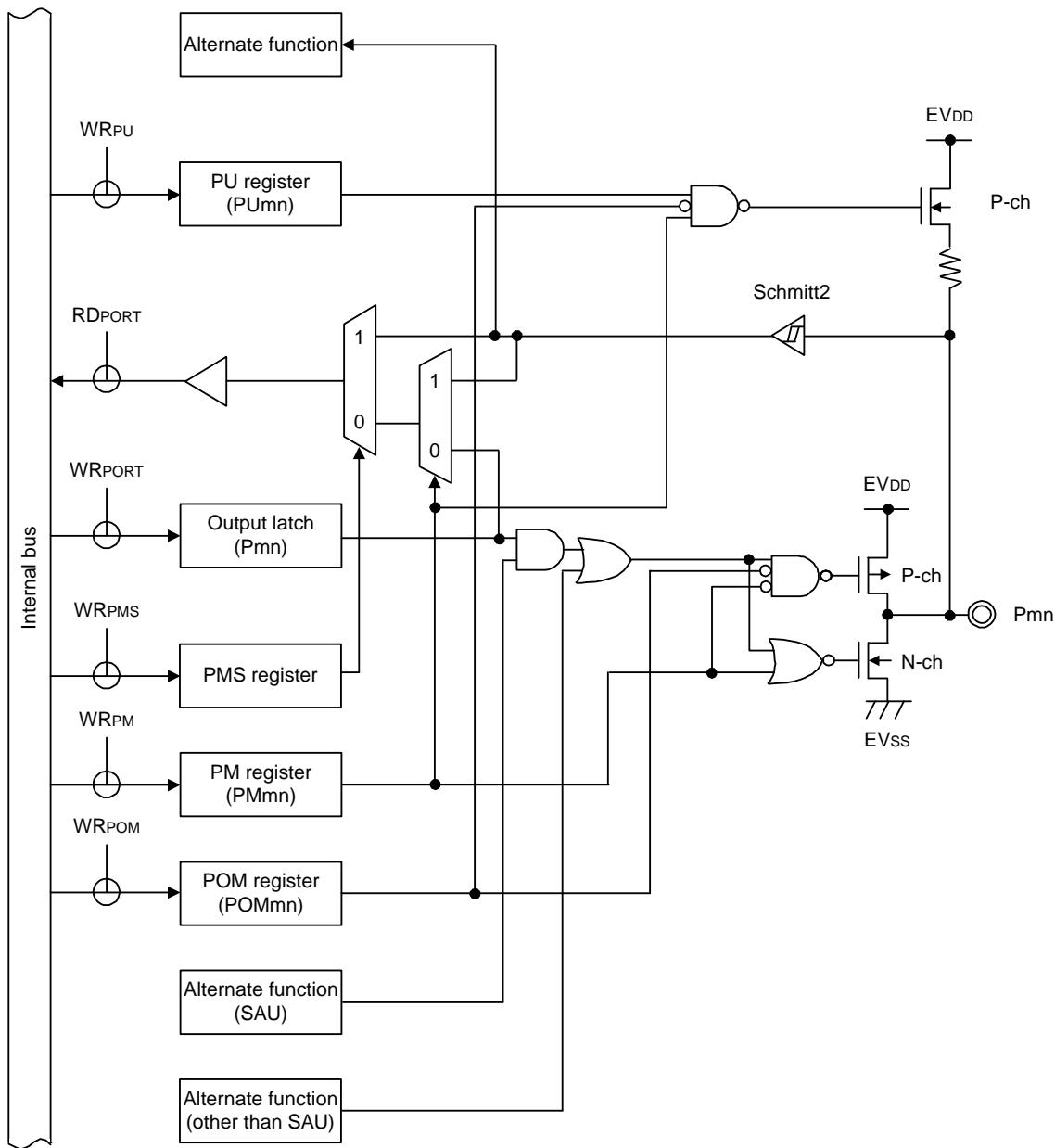
Figure 2 - 7 Pin Block Diagram of Pin Type 7-1-3



**Remark 1.** Refer to **2.1 Port Functions** for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 8 Pin Block Diagram of Pin Type 7-1-4

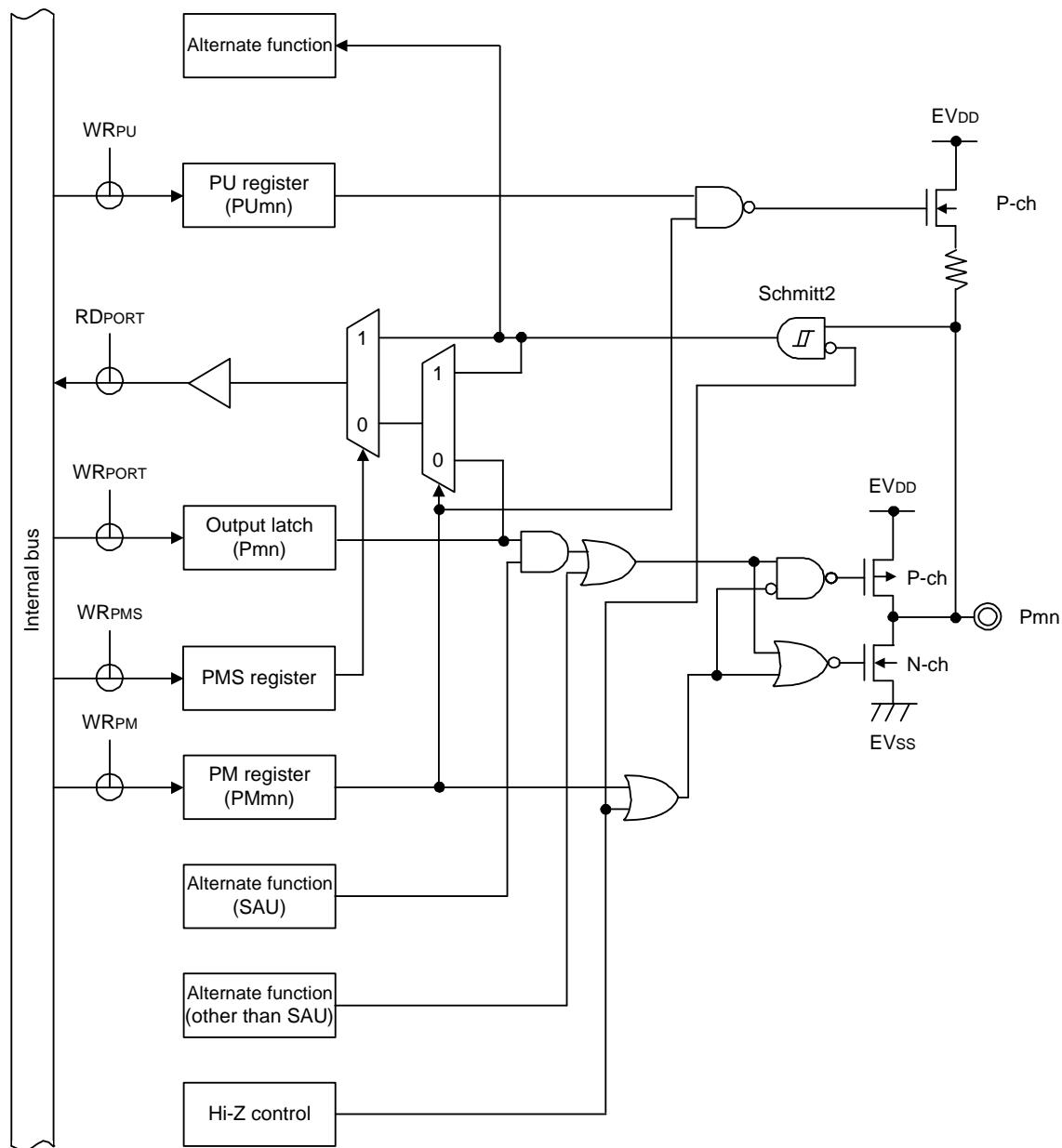


**Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

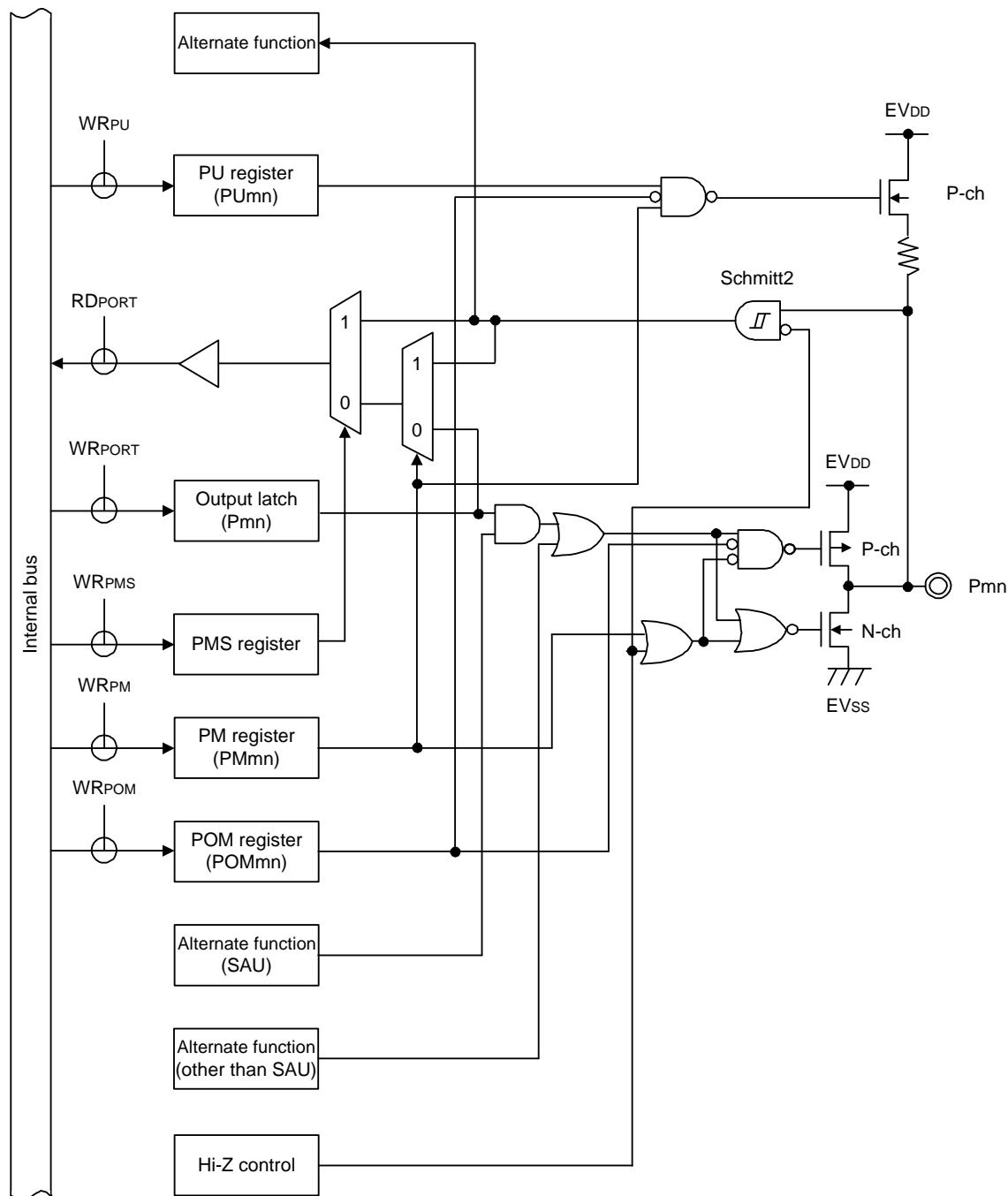
Figure 2 - 9 Pin Block Diagram of Pin Type 7-1-7



**Remark 1.** Refer to **2.1 Port Functions** for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 10 Pin Block Diagram of Pin Type 7-1-8

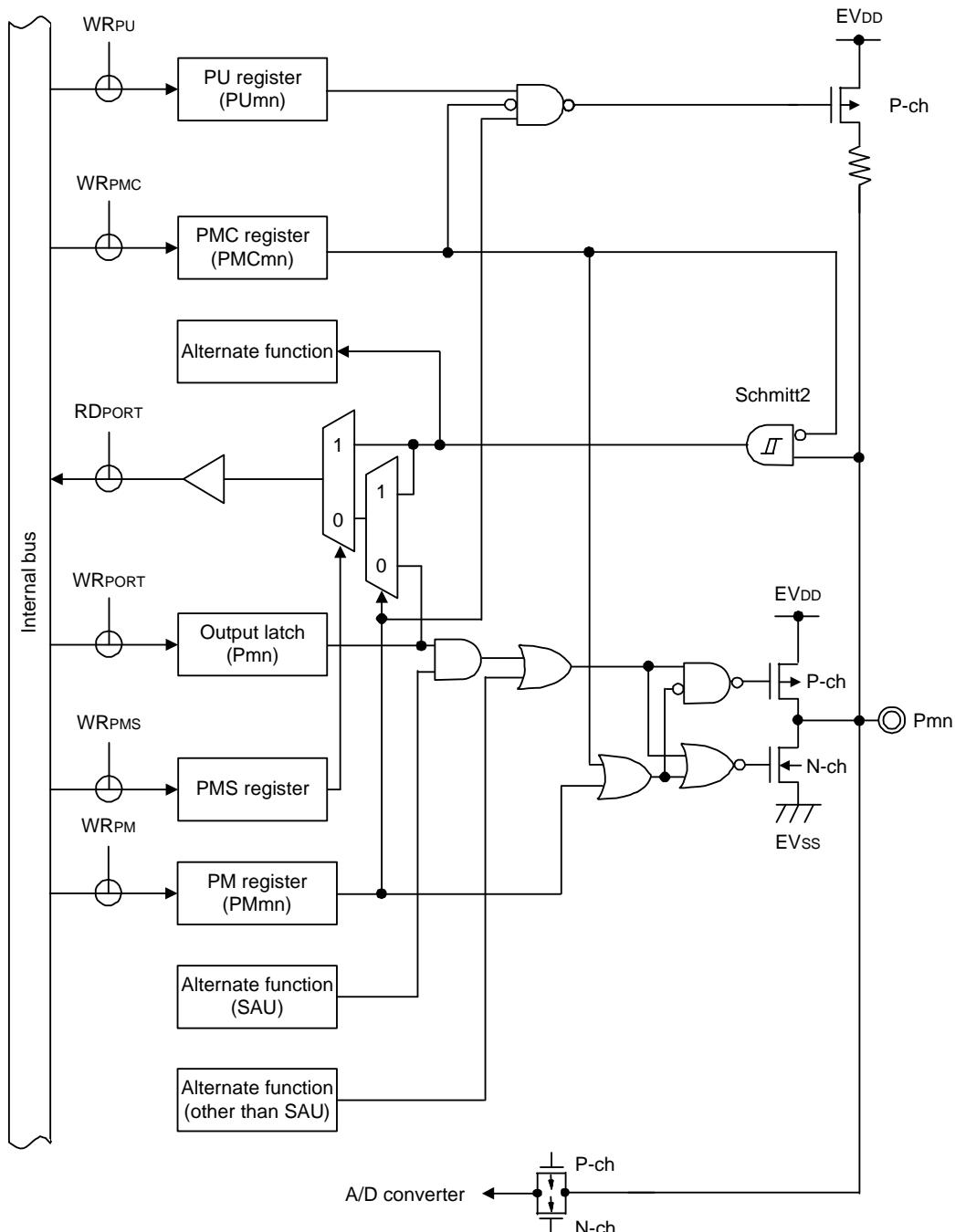


**Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Remark 1.** Refer to **2.1 Port Functions** for alternate functions.

**Remark 2.** SAU: Serial array unit

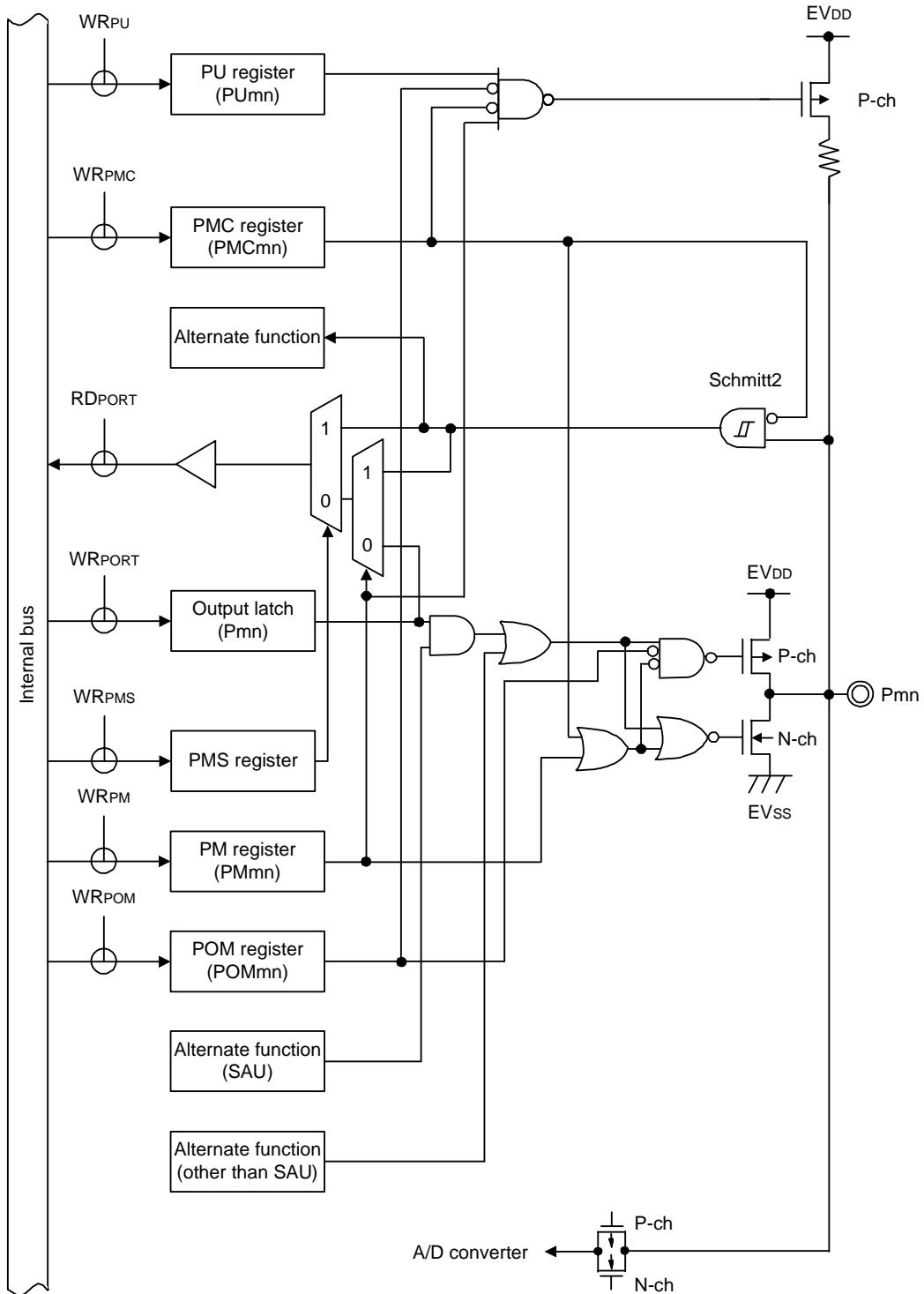
Figure 2 - 11 Pin Block Diagram of Pin Type 7-3-3



**Remark 1.** Refer to **2.1 Port Functions** for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 12 Pin Block Diagram of Pin Type 7-3-4

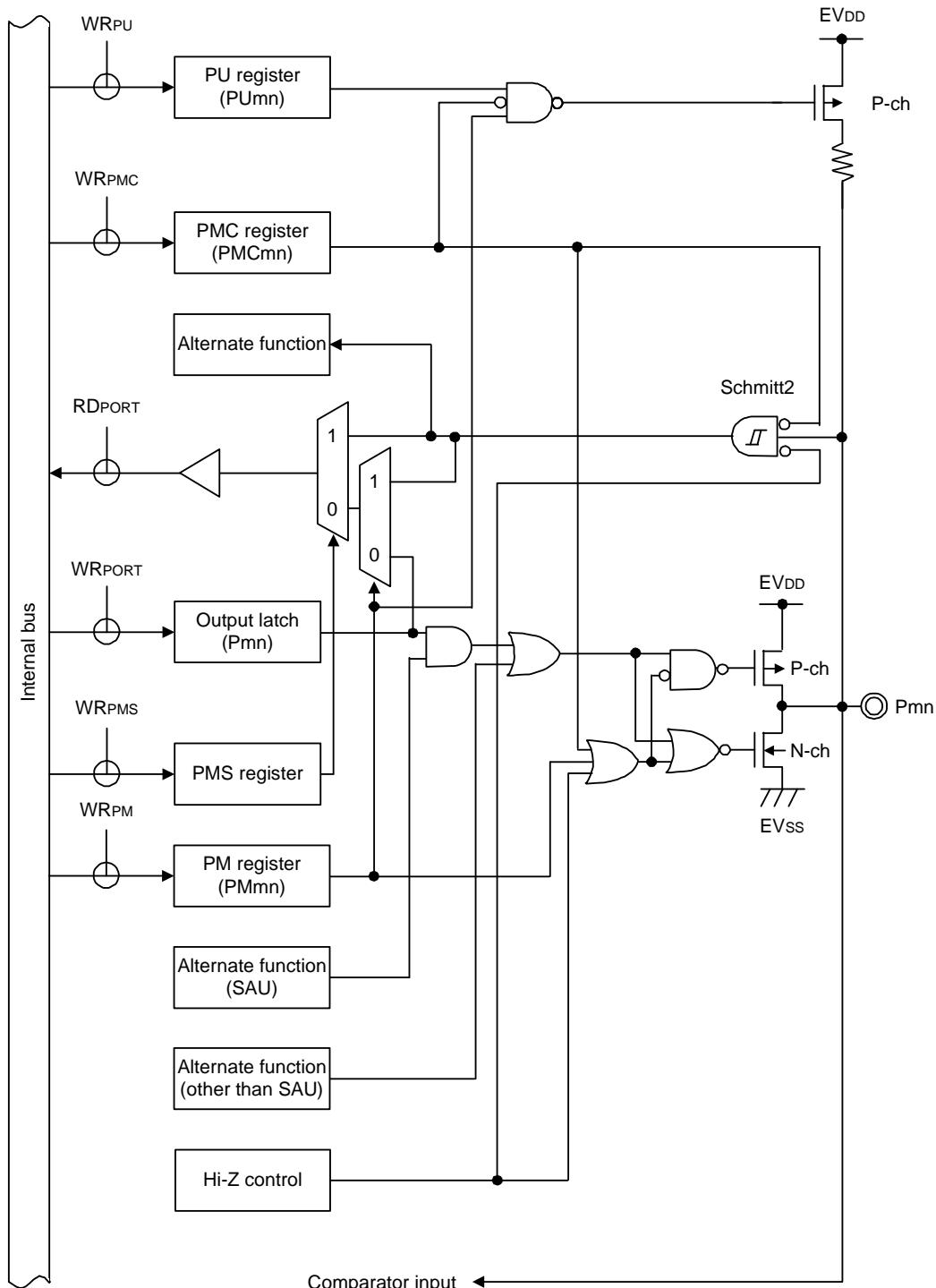


**Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

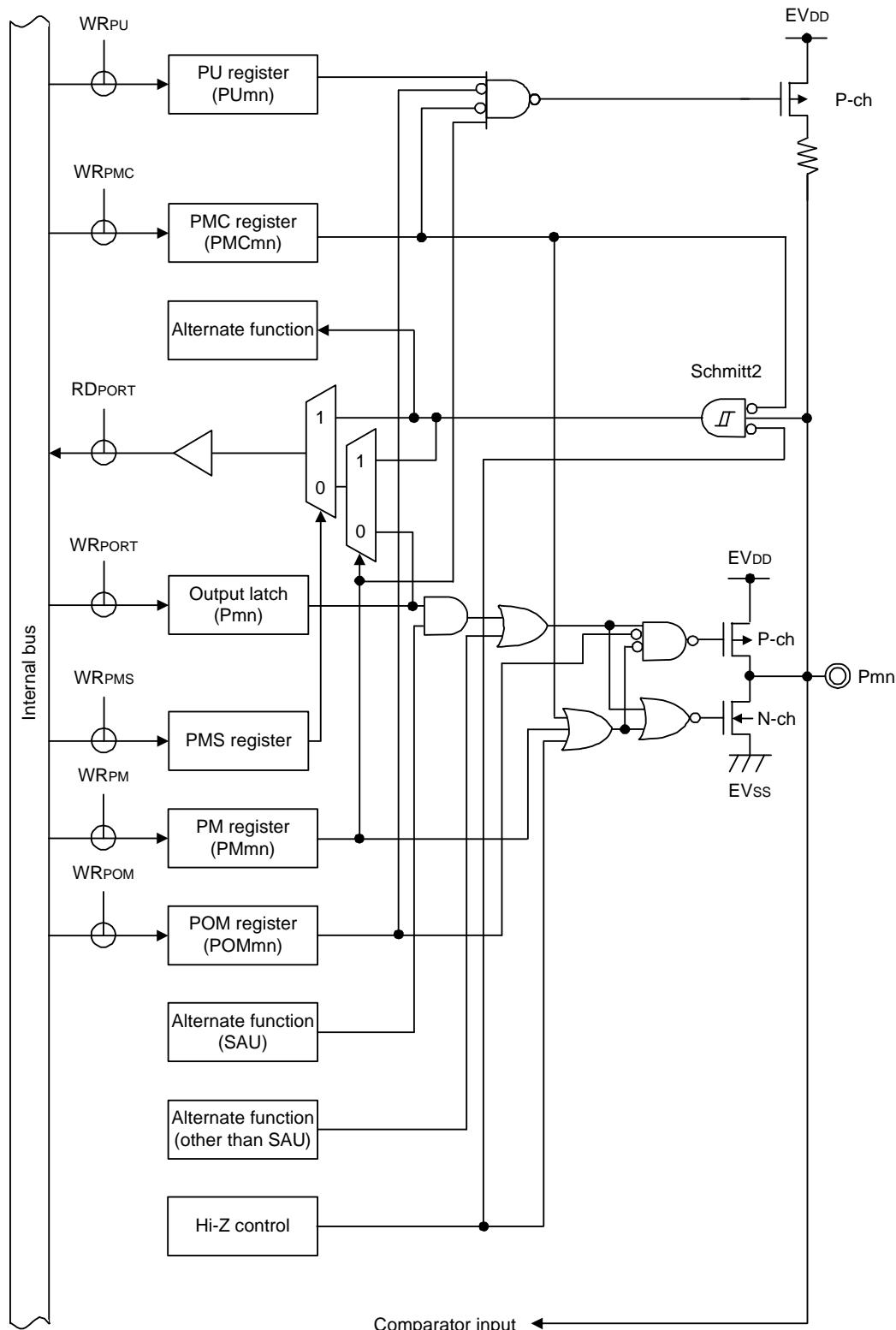
Figure 2 - 13 Pin Block Diagram of Pin Type 7-6-6



**Remark 1.** Refer to **2.1 Port Functions** for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 14 Pin Block Diagram of Pin Type 7-6-8

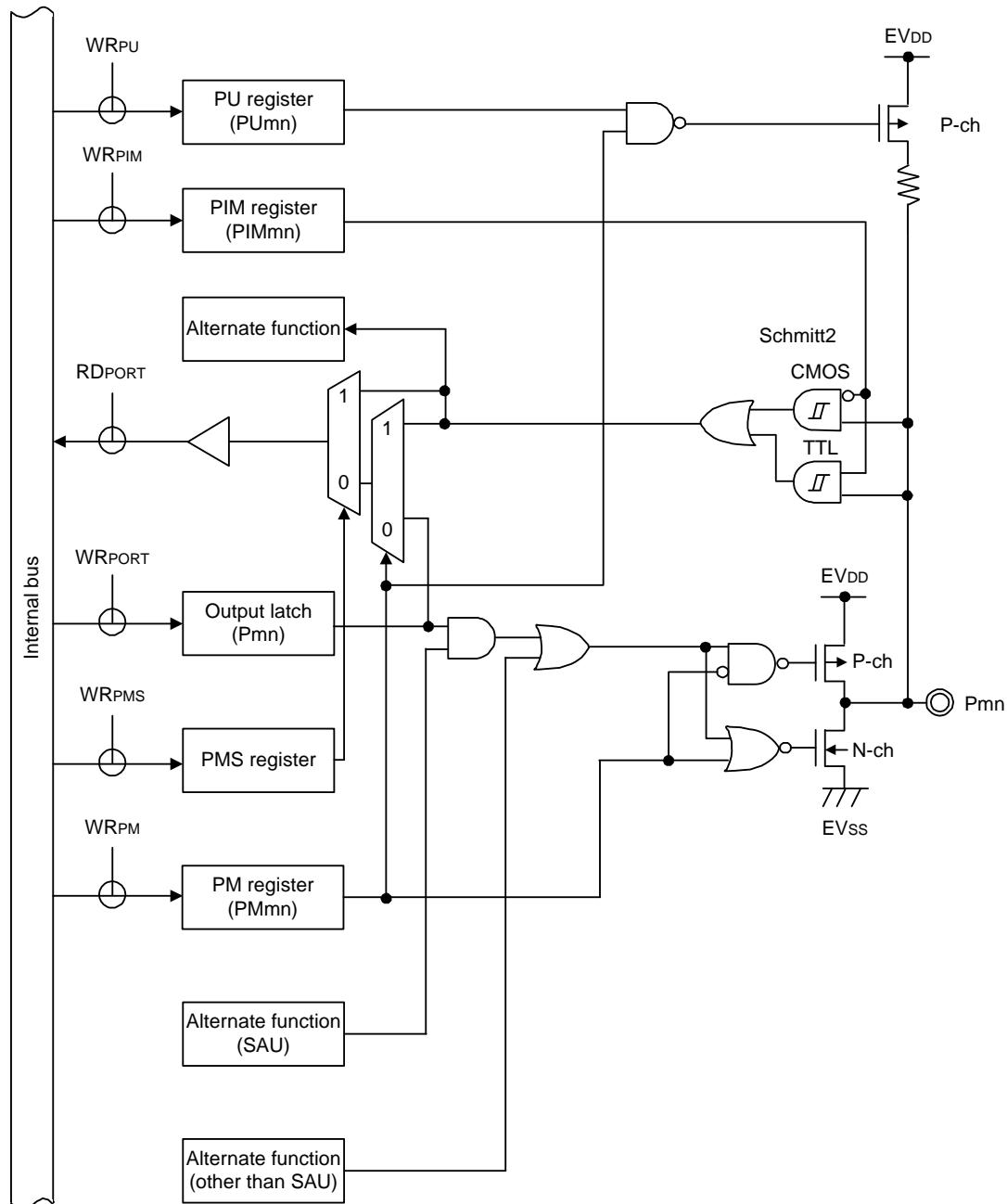


**Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 15 Pin Block Diagram of Pin Type 8-1-3

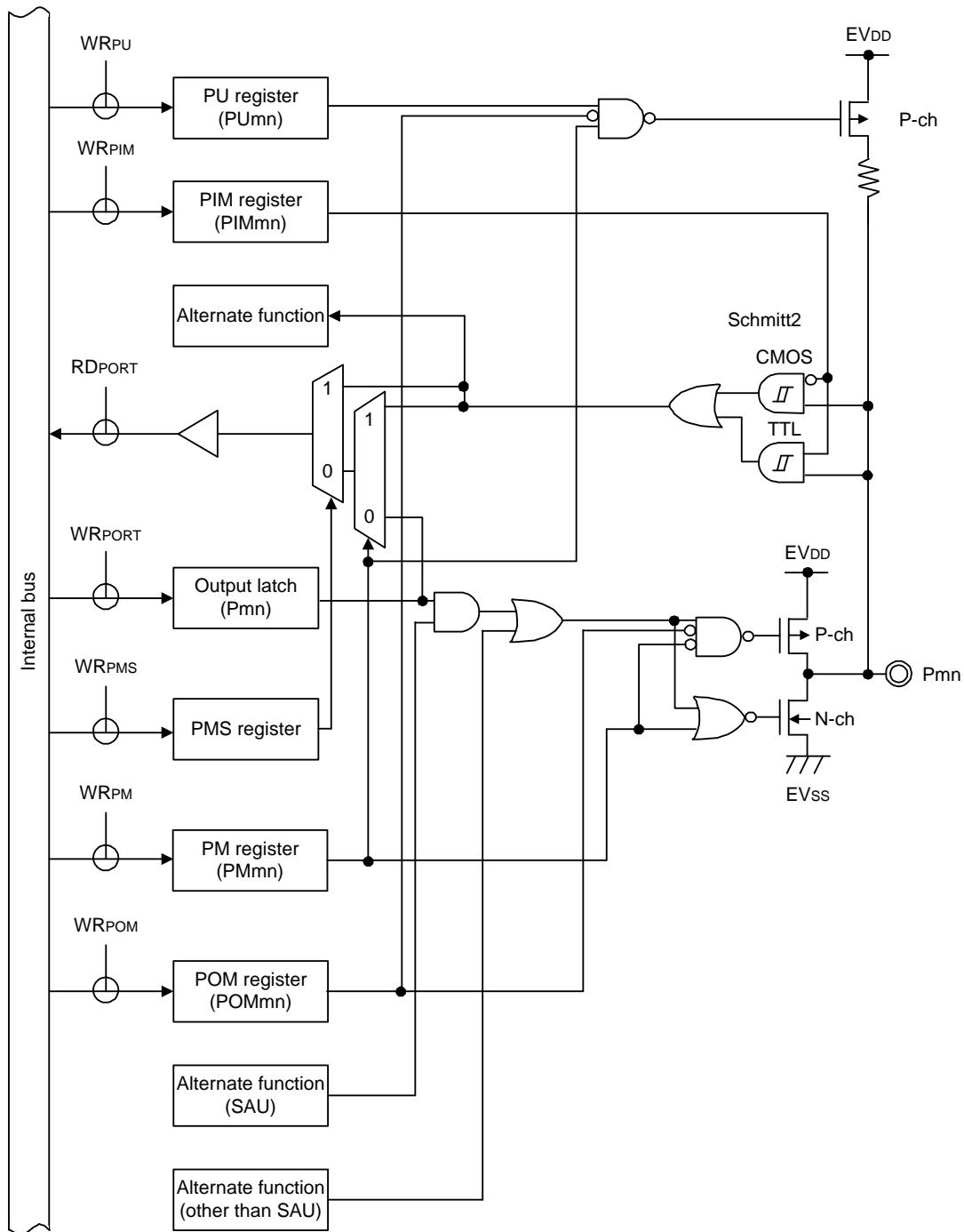


**Caution** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 16 Pin Block Diagram of Pin Type 8-1-4



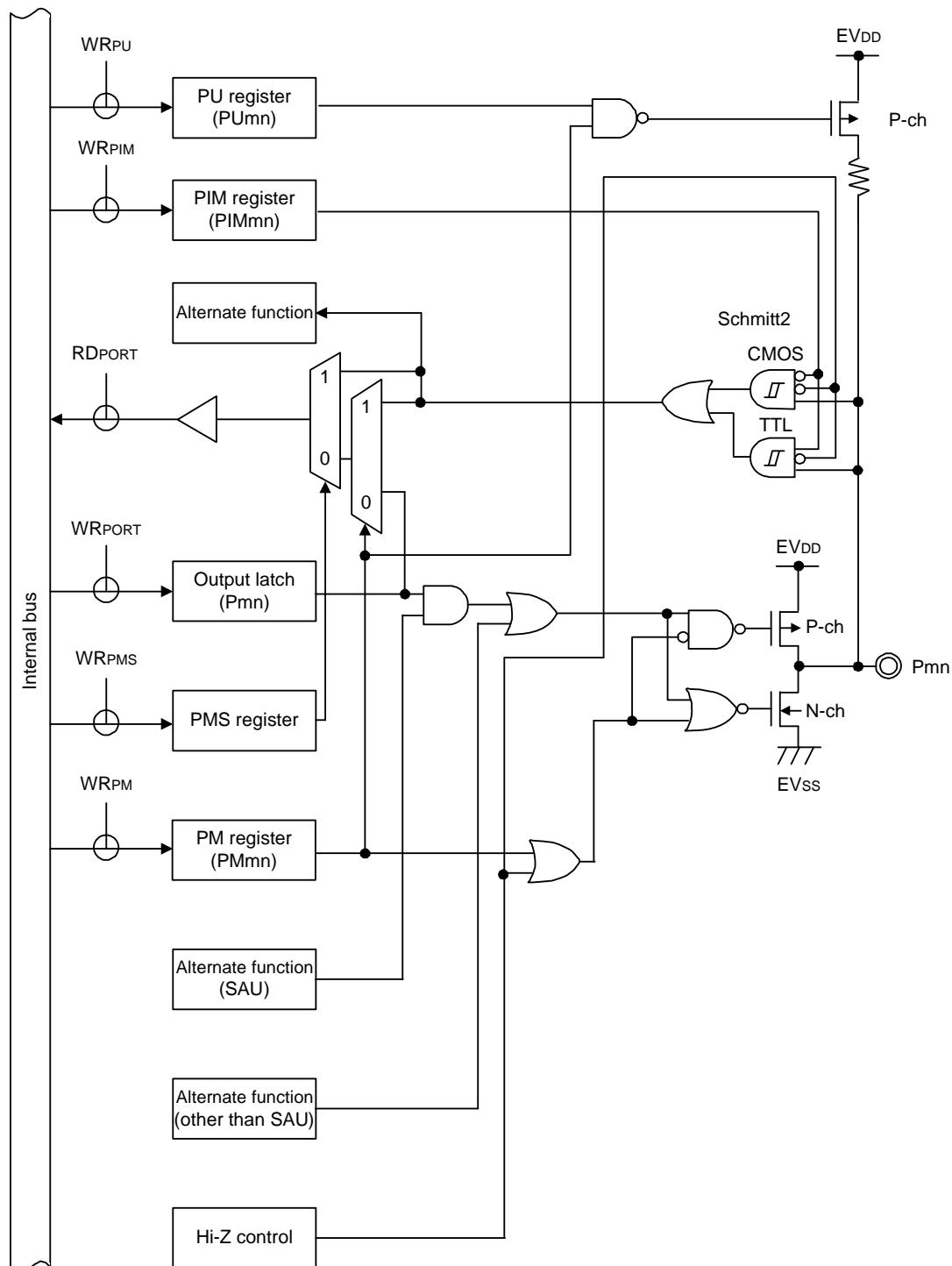
**Caution 1.** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Caution 2.** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 17 Pin Block Diagram of Pin Type 8-1-7

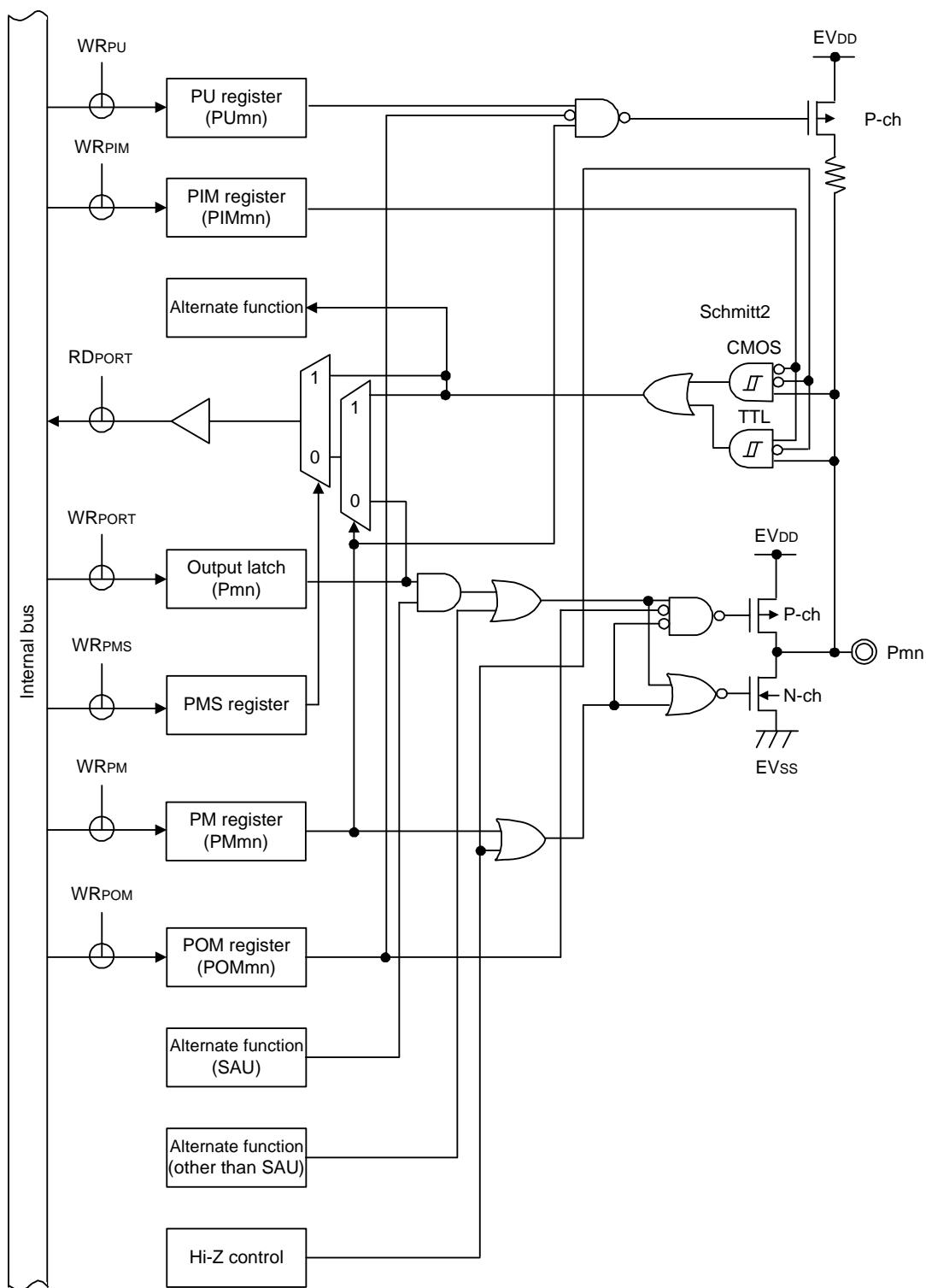


**Caution** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 18 Pin Block Diagram of Pin Type 8-1-8



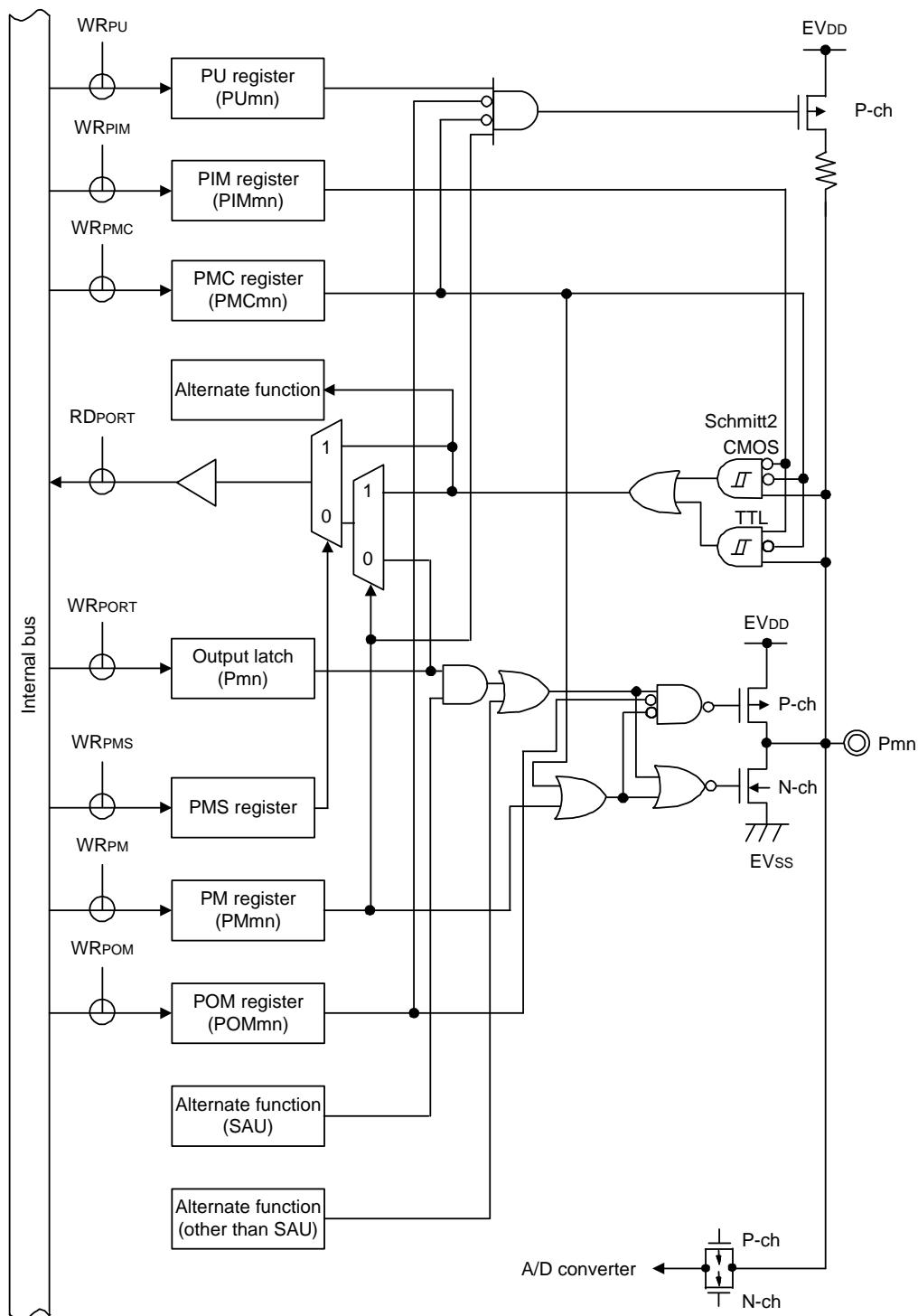
**Caution 1.** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Caution 2.** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

**Figure 2 - 19 Pin Block Diagram of Pin Type 8-3-4**



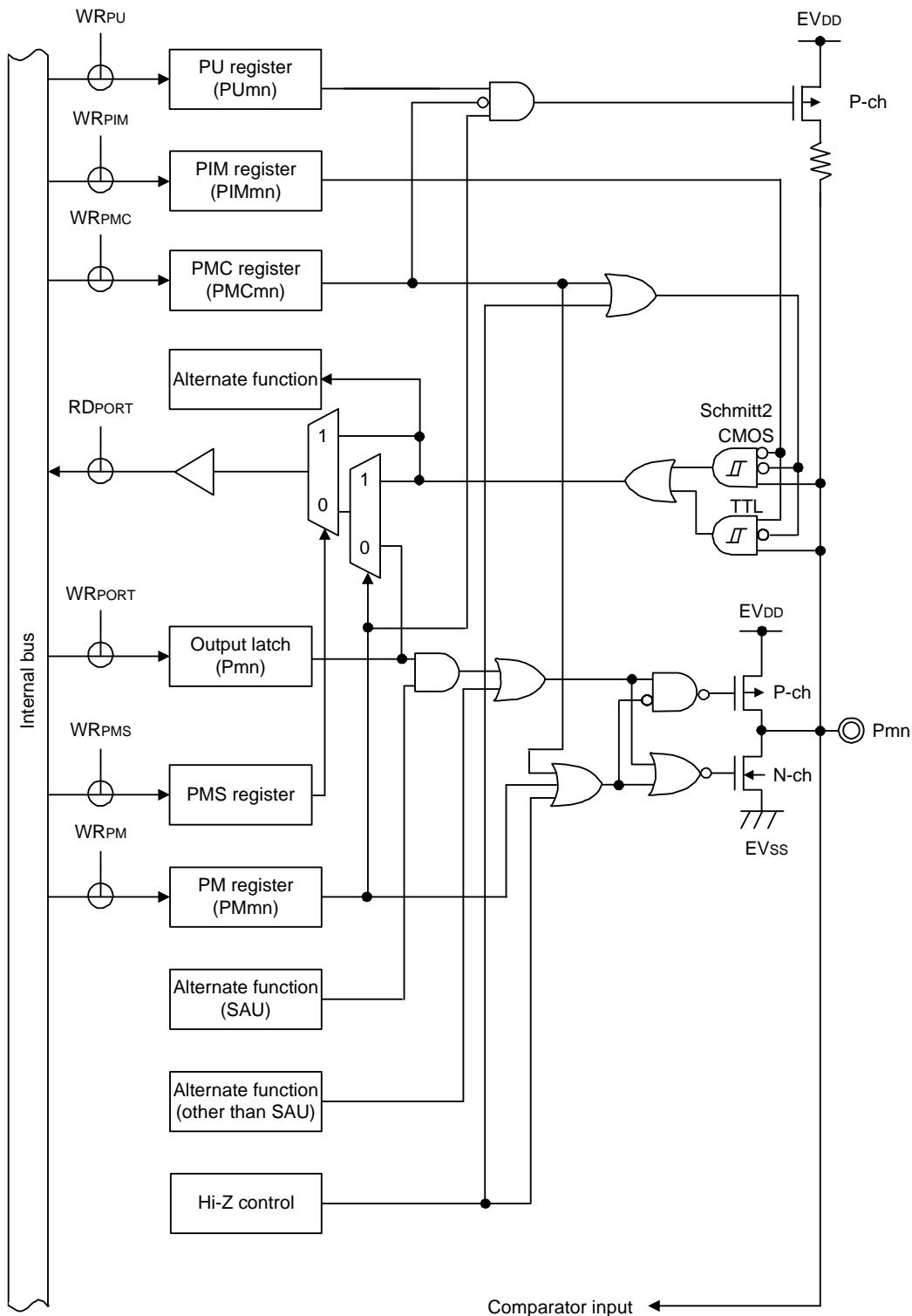
**Caution 1.** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Caution 2.** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to [2.1 Port Functions](#) for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 20 Pin Block Diagram of Pin Type 8-6-6

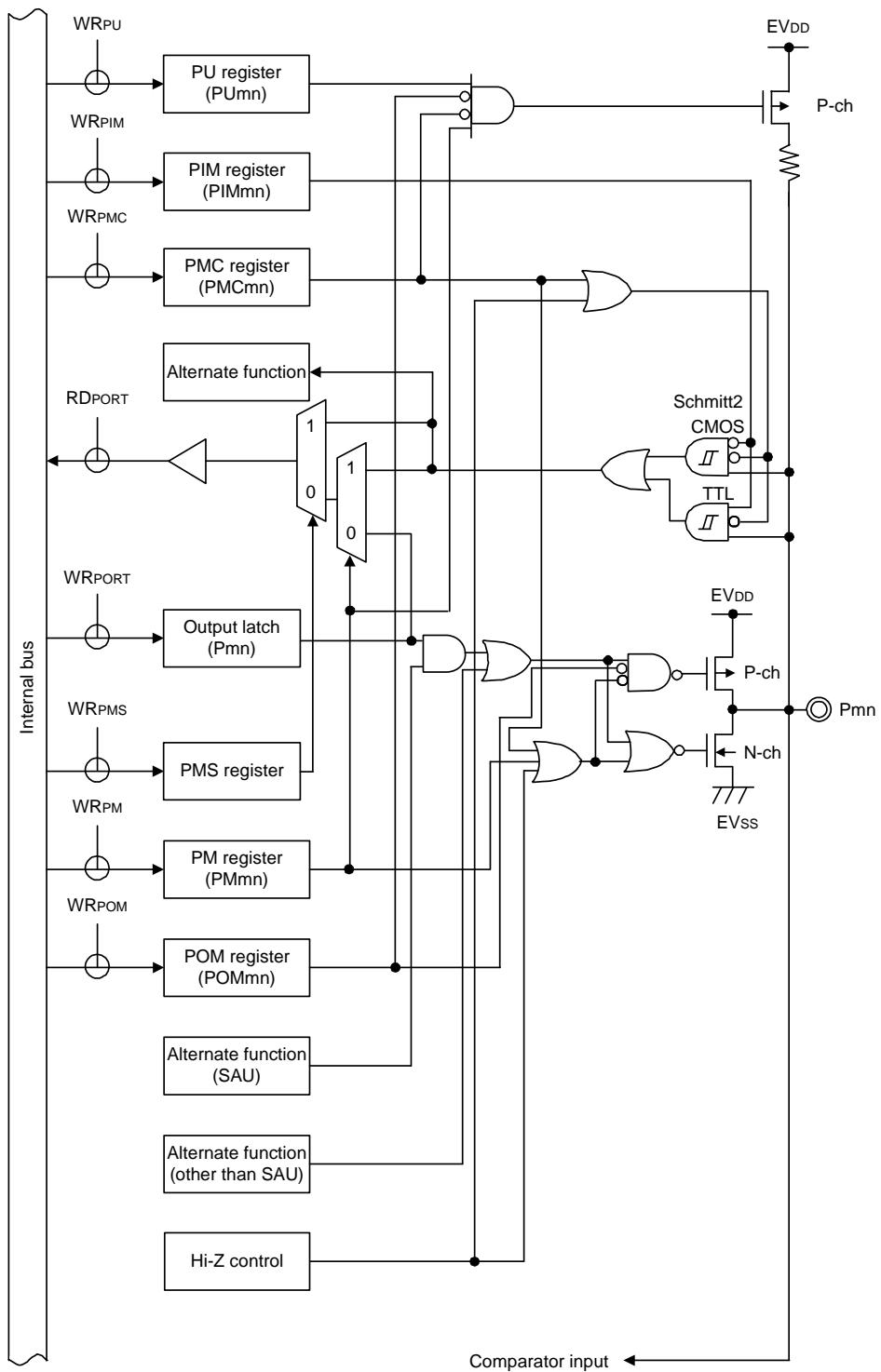


**Caution** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 21 Pin Block Diagram of Pin Type 8-6-8



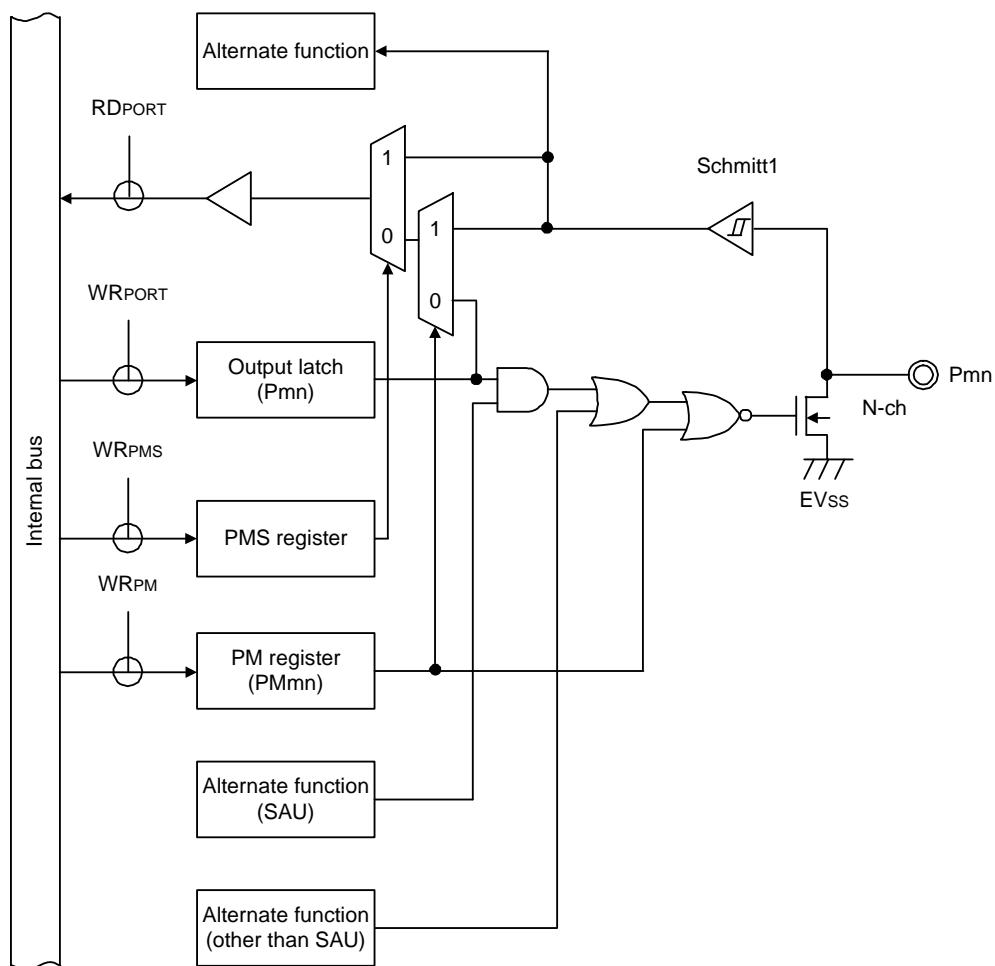
**Caution 1.** A through current may flow through if the pin is in the intermediate potential, because the input buffer is also turned on when the pin is in N-ch open-drain output mode by port output mode register (POMx).

**Caution 2.** Because of TTL input buffer structure, if the port input mode register (PIMx) is set in TTL input buffer, a through current may flow through in the case of high level input. It is recommended to input a low level to prevent a through current.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

**Remark 2.** SAU: Serial array unit

Figure 2 - 22 Pin Block Diagram of Pin Type 12-1-5



**Caution** A through current may flow through if the pin is in the intermediate potential, because the input buffer is turned on when the pin is in output mode.

**Remark 1.** Refer to 2.1 Port Functions for alternate functions.

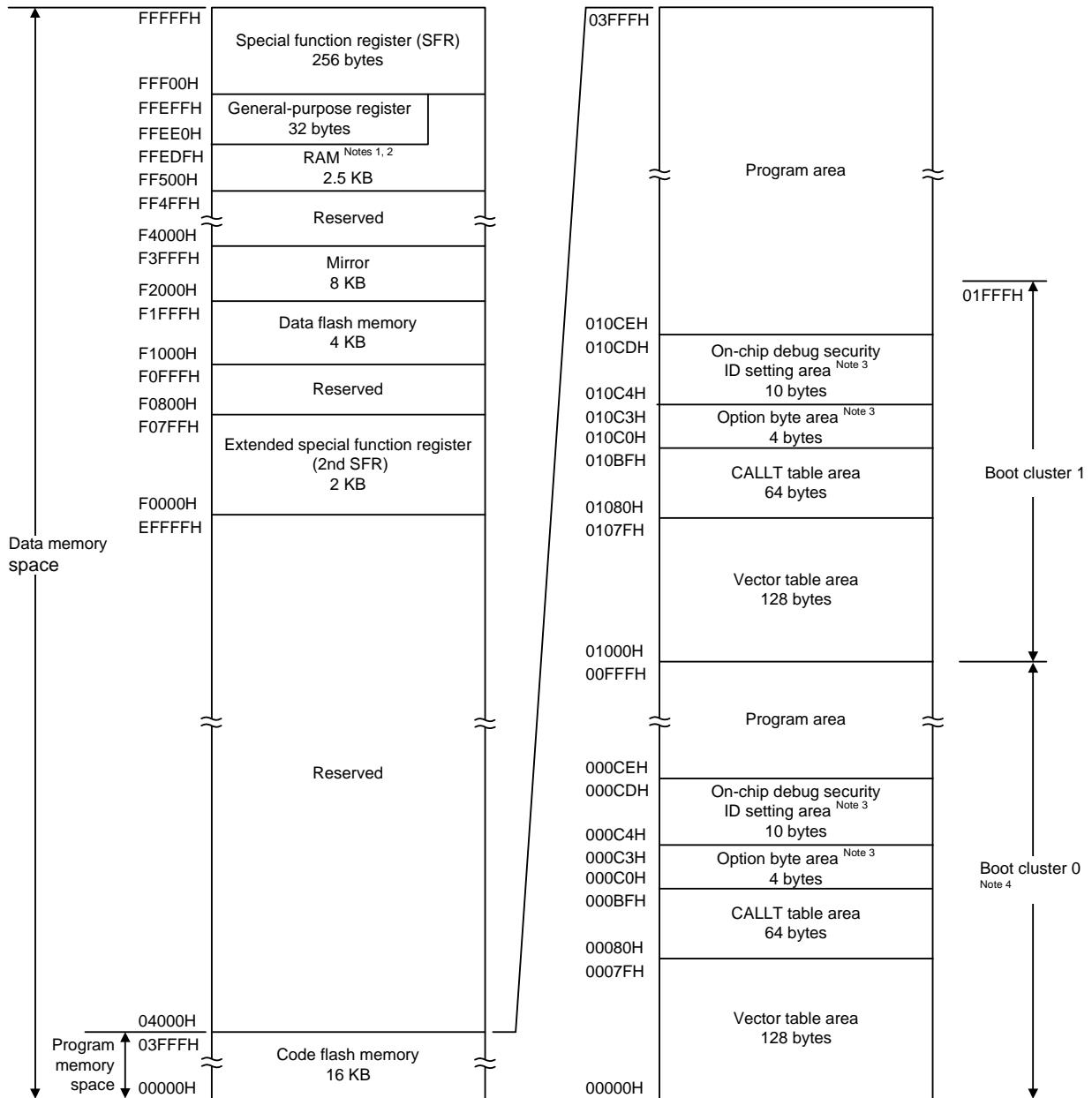
**Remark 2.** SAU: Serial array unit

## CHAPTER 3 CPU ARCHITECTURE

### 3.1 Memory Space

Products in the RL78/G14 can access a 1 MB address space. Figures 3 - 1 to 3 - 10 show the memory maps.

Figure 3 - 1 Memory Map (R5F104xA (x = A to C, E to G))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

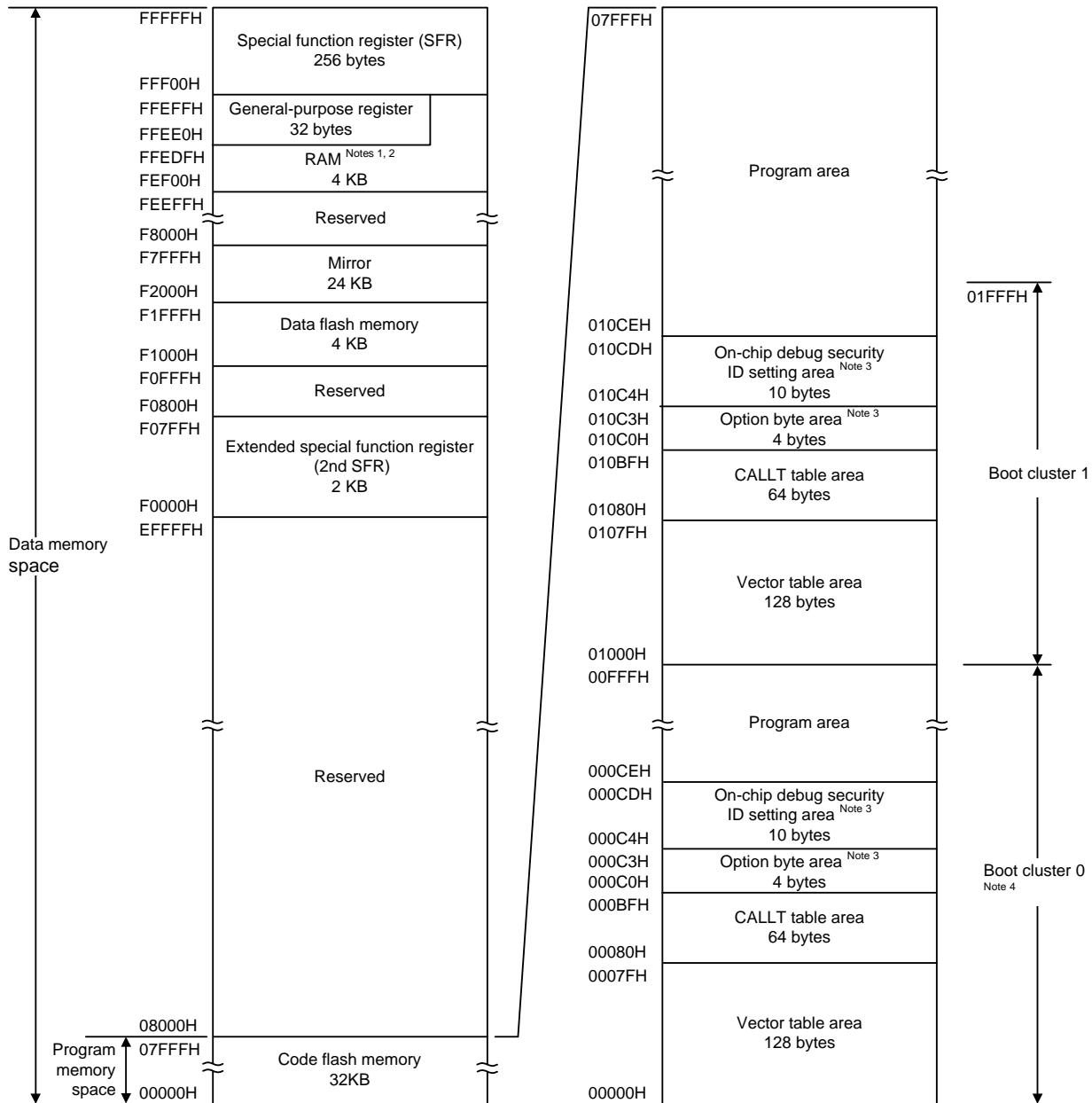
**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

**Figure 3 - 2 Memory Map (R5F104xC (x = A to C, E to G, J, L))**

**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

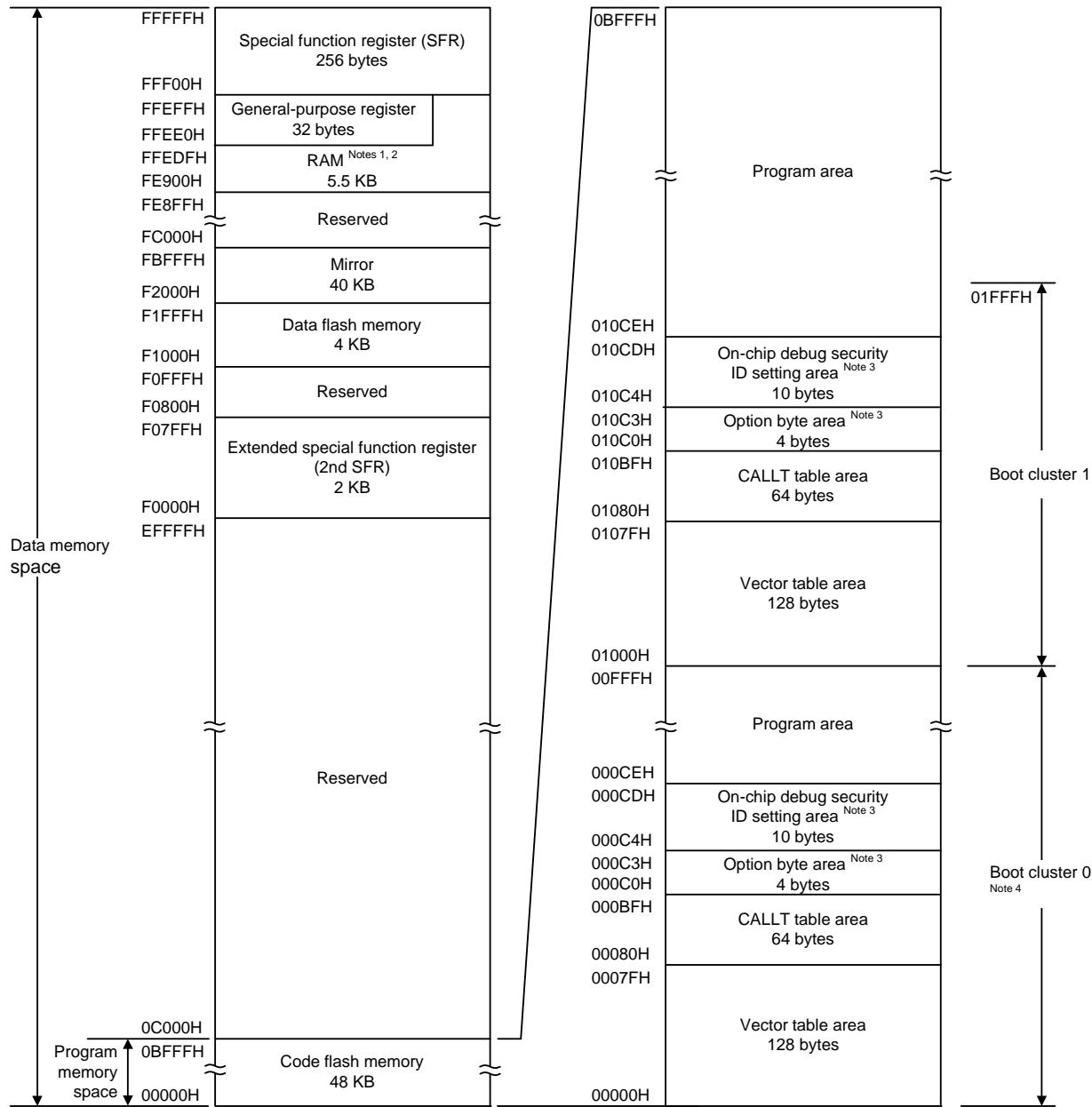
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 3 Memory Map (R5F104xD (x = A to C, E to G, J, L))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

The RAM area used by the flash library starts at FE900H. For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

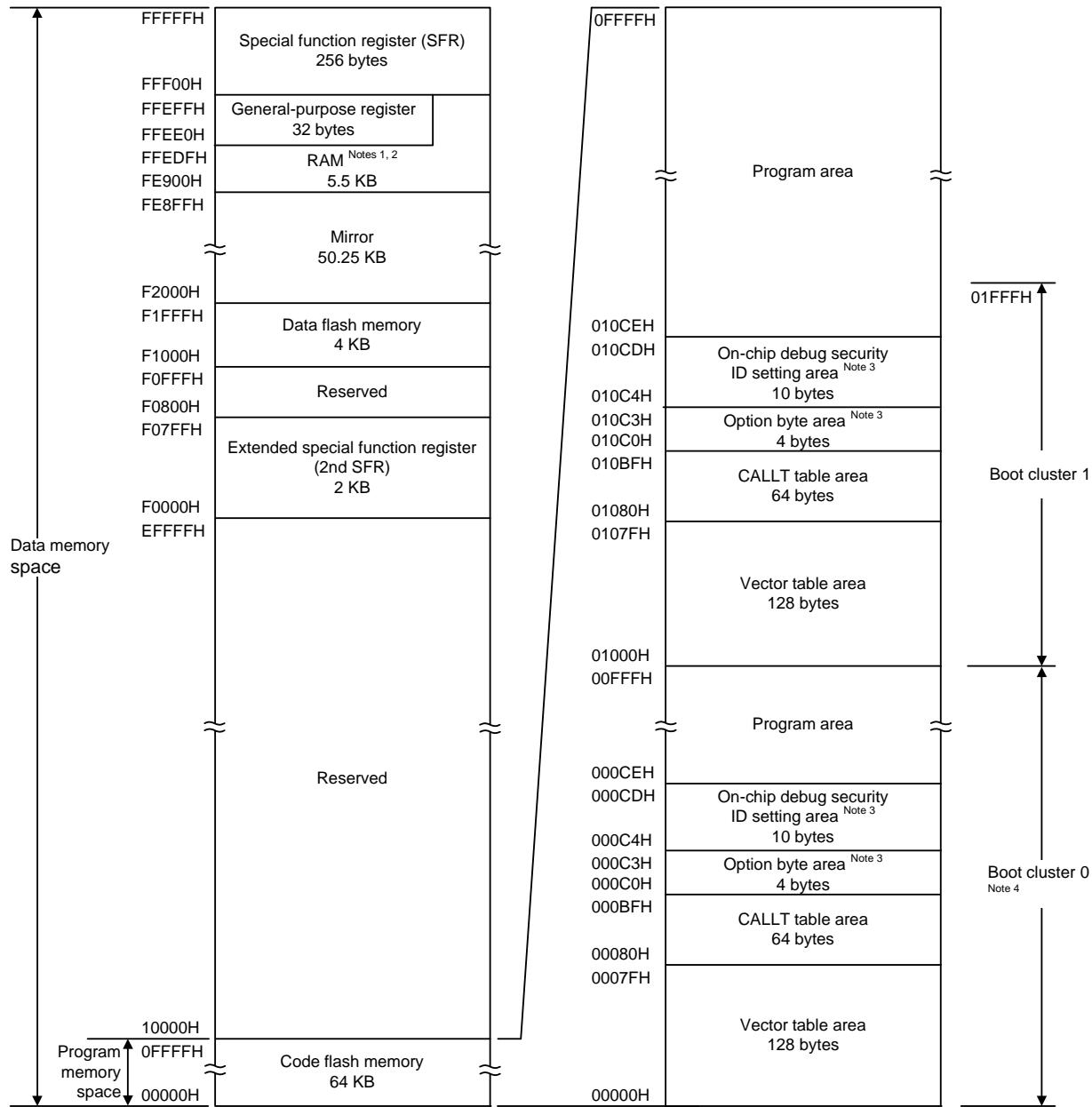
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 4 Memory Map (R5F104xE (x = A to C, E to G, J, L))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

The RAM area used by the flash library starts at FE900H. For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

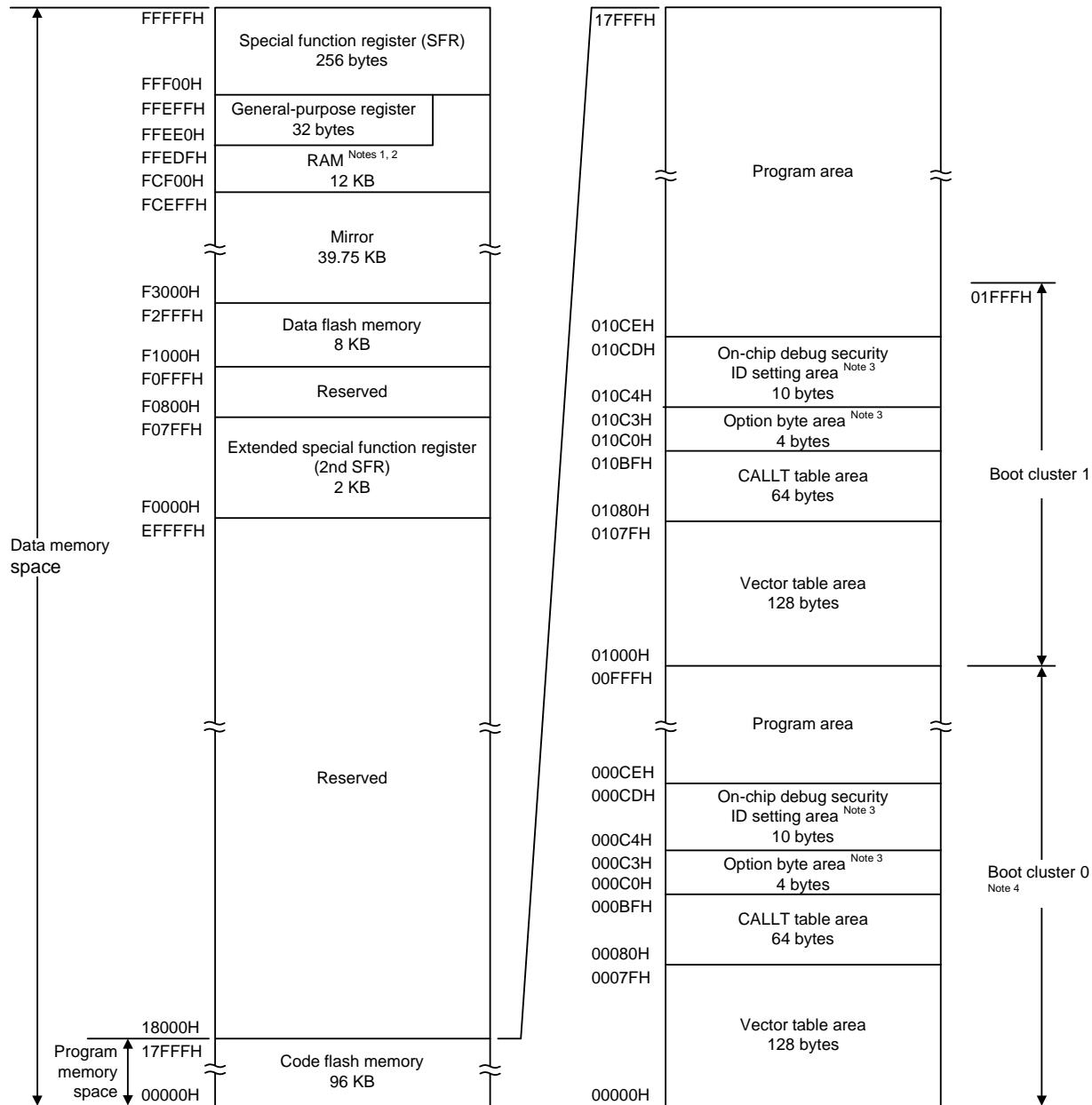
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 5 Memory Map (R5F104xF (x = A to C, E to G, J, L, M, P))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

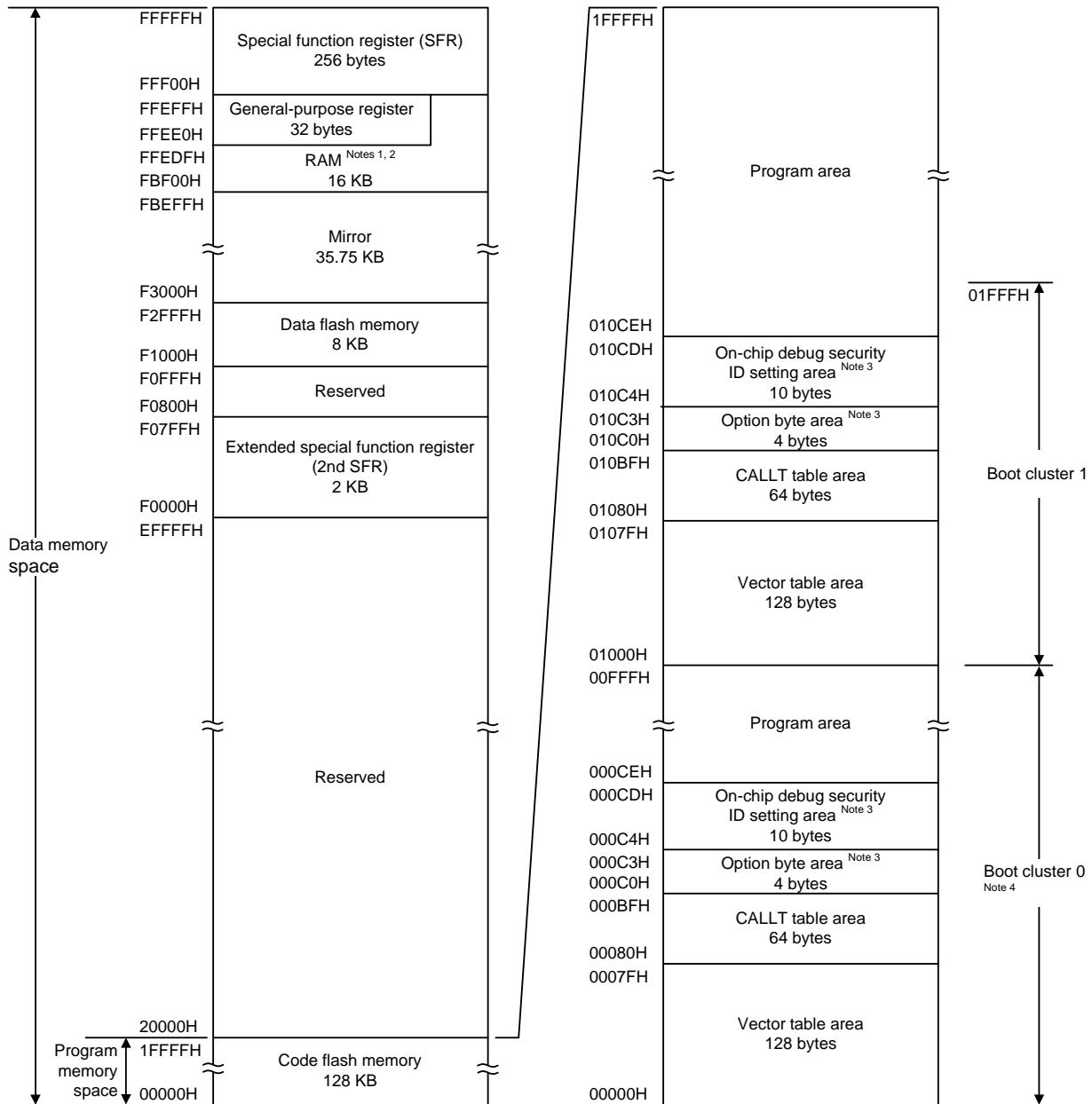
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 6 Memory Map (R5F104xG (x = A to C, E to G, J, L, M, P))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

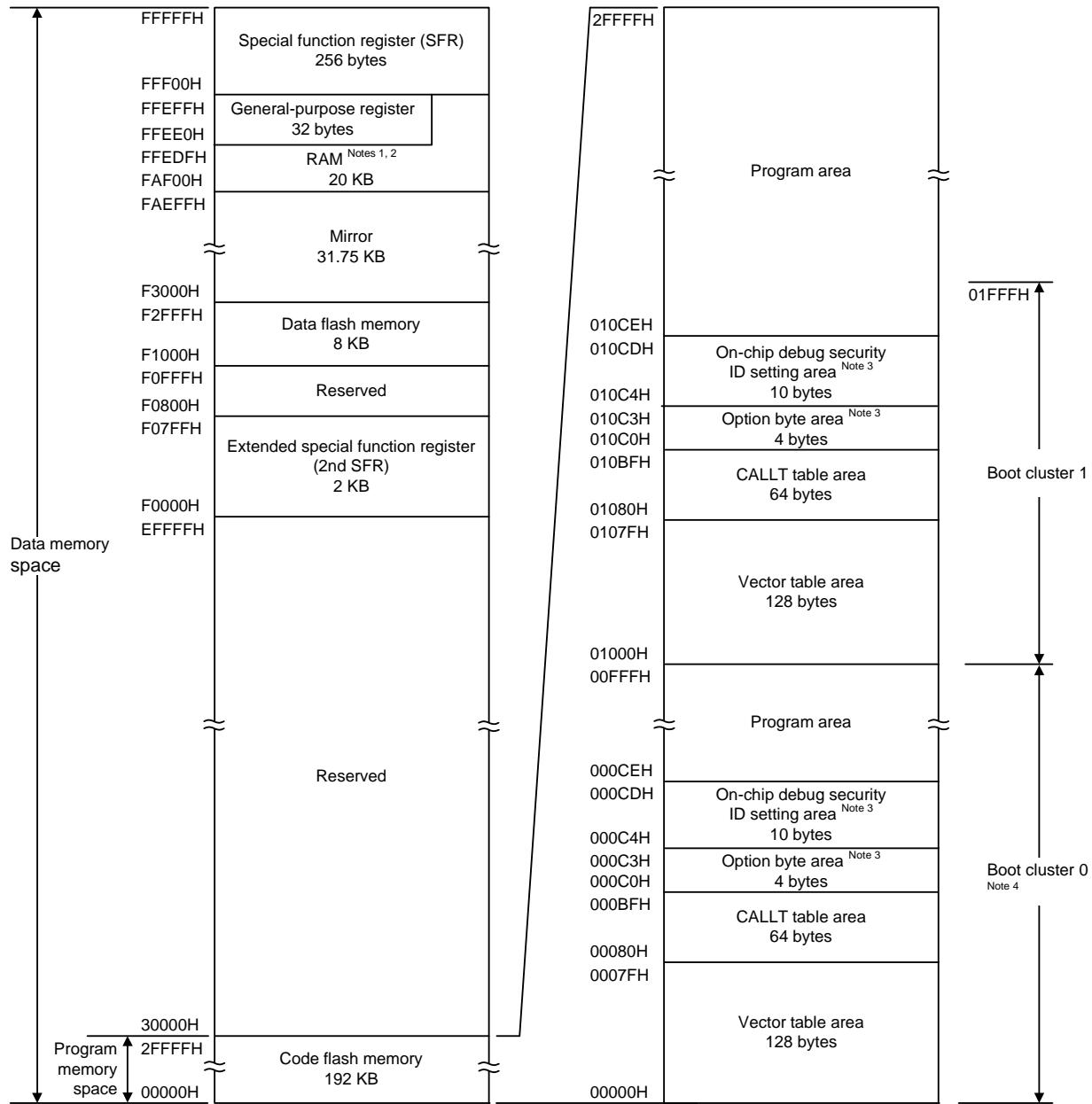
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 7 Memory Map (R5F104xH (x = E to G, J, L, M, P))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see 30.7 Security Settings).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see 27.3.3 RAM parity error detection function.

Figure 3 - 8 Memory Map (R5F104xJ (x = F, G, J, L, M, P))

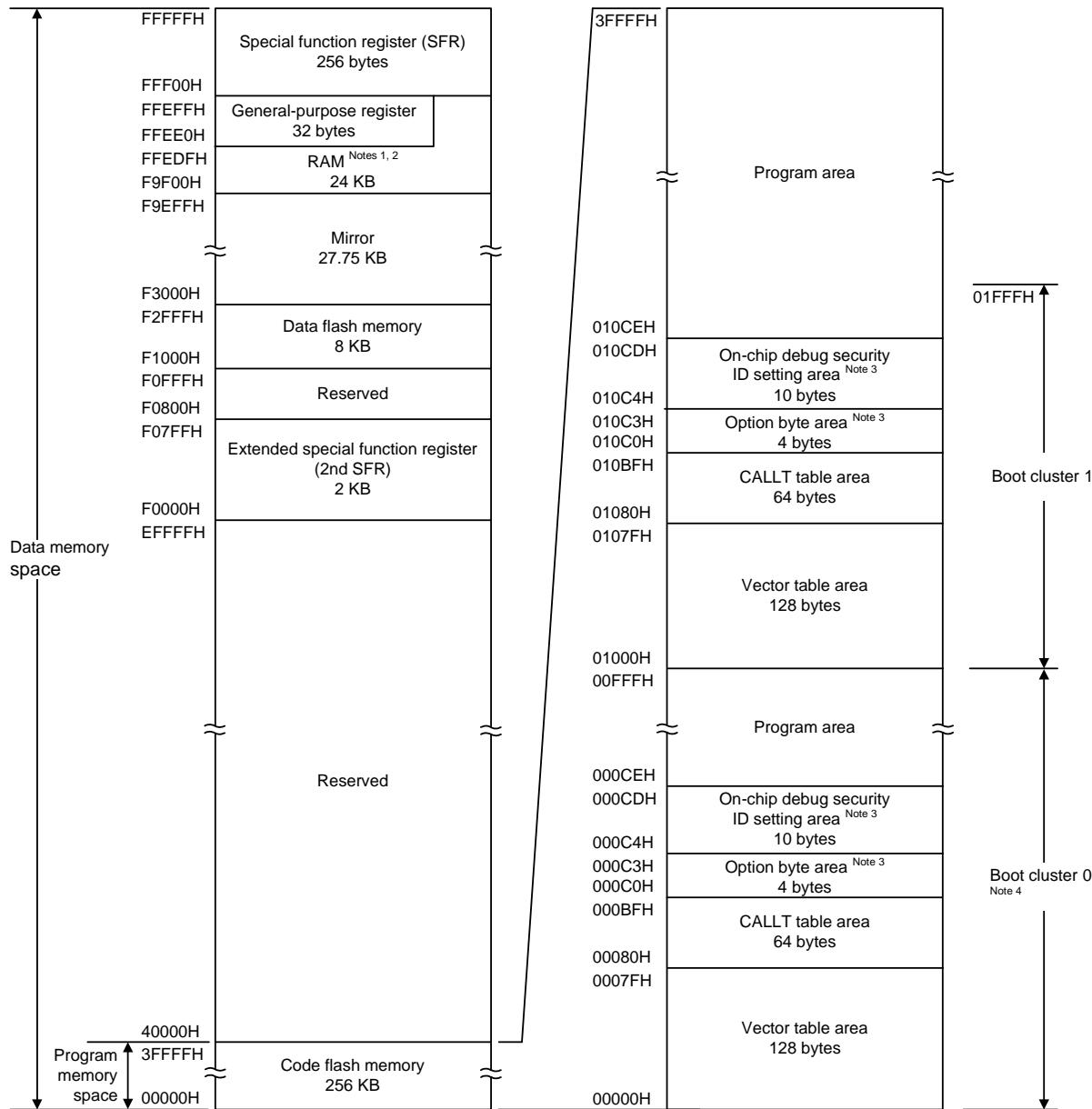
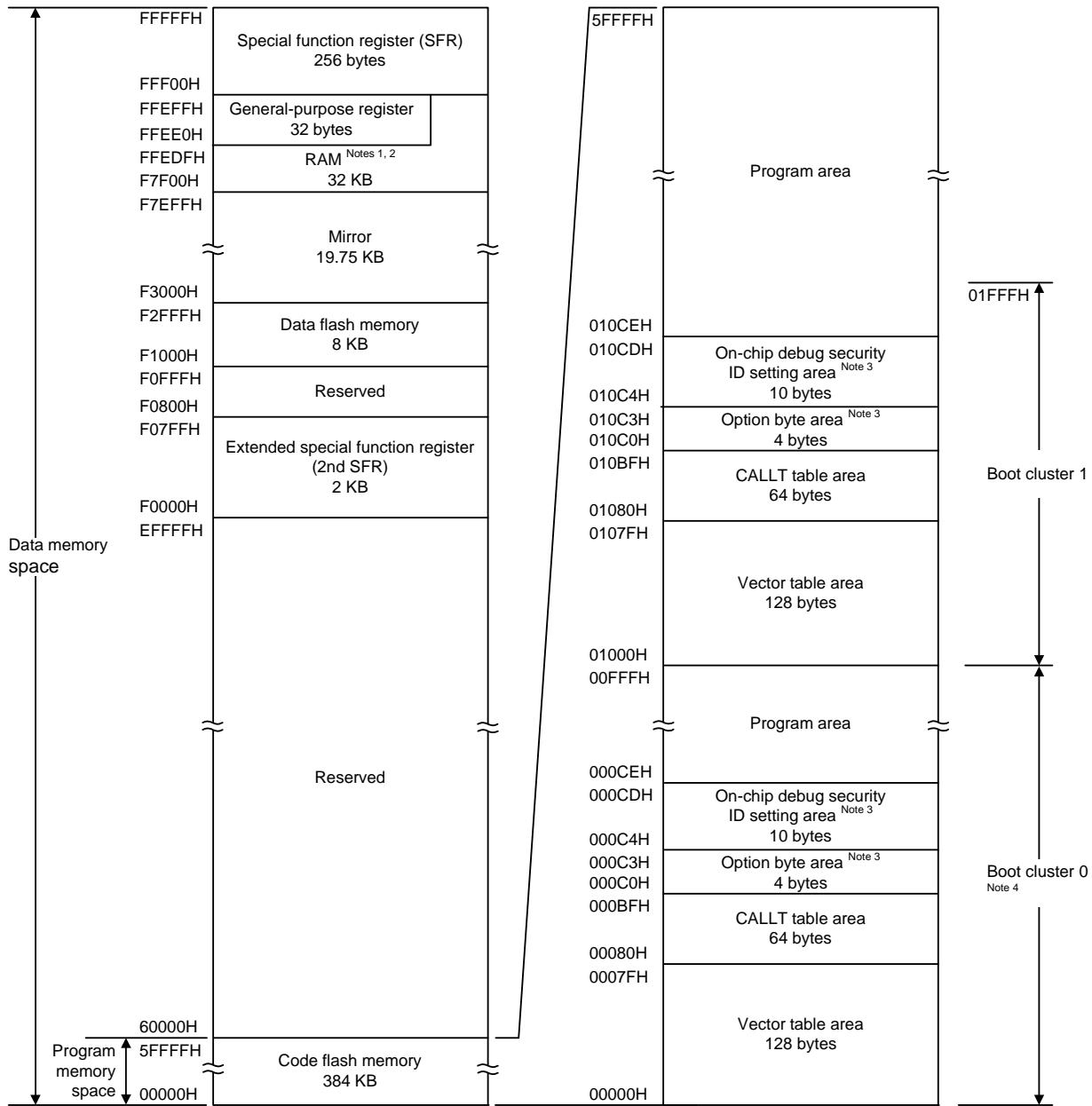


Figure 3 - 9 Memory Map (R5F104xK (x = G, L, M, P))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

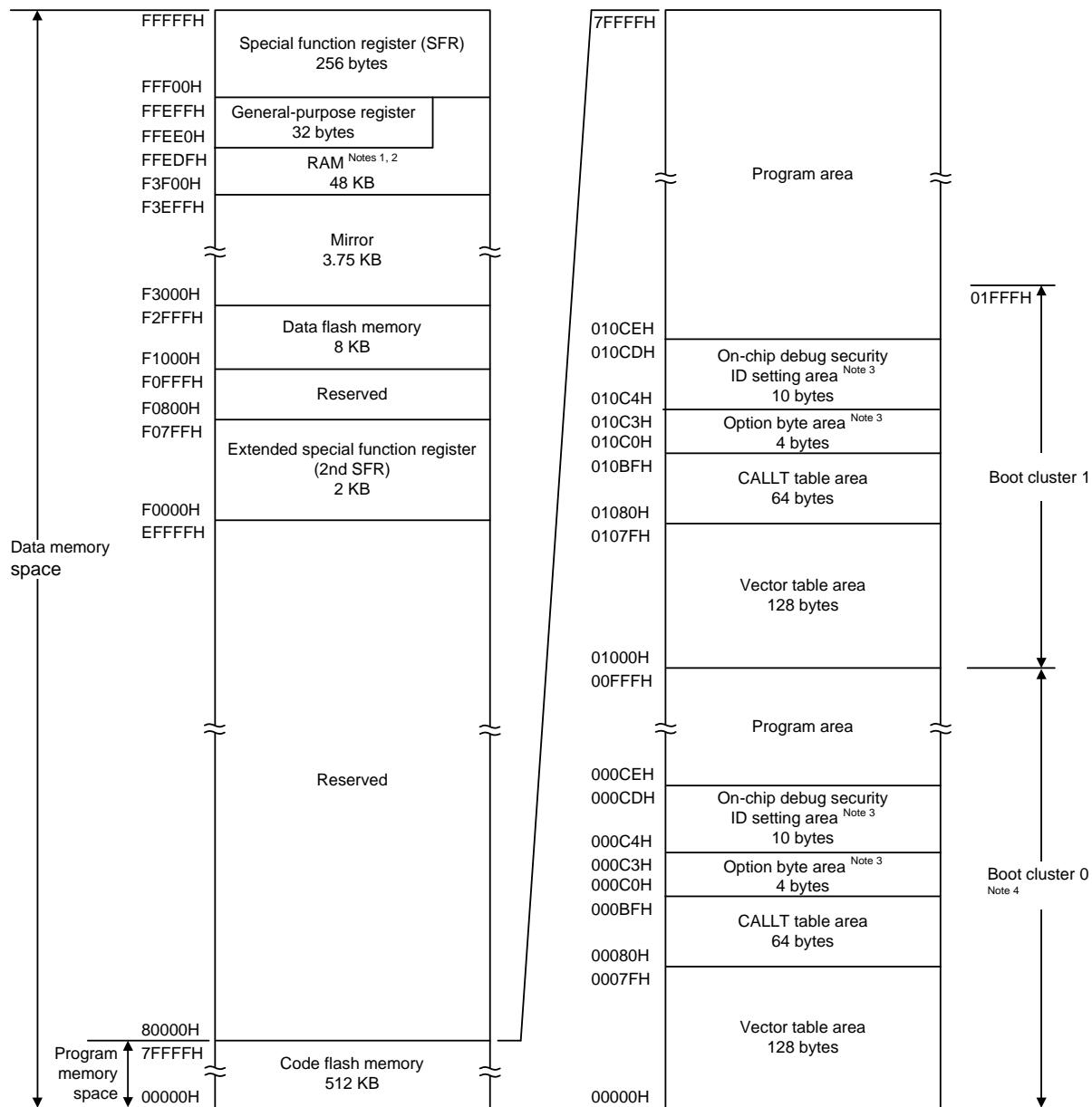
When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

Figure 3 - 10 Memory Map (R5F104xL (x = G, L, M, P))



**Note 1.** Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.

The RAM area used by the flash library starts at F3F00H. For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

**Note 2.** Instructions can be executed from the RAM area excluding the general-purpose register area.

**Note 3.** When boot swap is not used: Set the option bytes to 000C0H to 000C3H, and the on-chip debug security IDs to 000C4H to 000CDH.

When boot swap is used: Set the option bytes to 000C0H to 000C3H and 010C0H to 010C3H, and the on-chip debug security IDs to 000C4H to 000CDH and 010C4H to 010CDH.

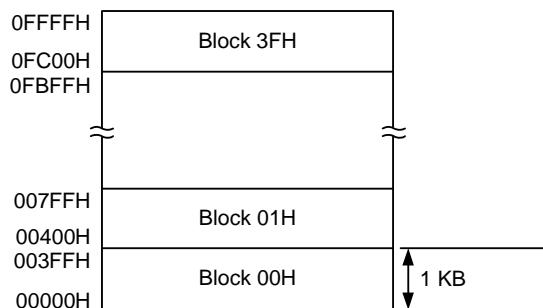
**Note 4.** Writing boot cluster 0 can be prohibited depending on the setting of security (see **30.7 Security Settings**).

**Note 5.** Use of the area F4300H to F46FFH is prohibited when using the on-chip debugging trace function.

**Caution** While RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed and the RAM area + 10 bytes when instructions are fetched from RAM areas, respectively.

Reset signal generation sets RAM parity error resets to enabled (RPERDIS = 0). For details, see **27.3.3 RAM parity error detection function**.

**Remark** The flash memory is divided into blocks (one block = 1 KB). For the address values and block numbers, see **Tables 3 - 1 to 3 - 4 Correspondence Between Address Values and Block Numbers in Flash Memory**.



(R5F104xE (x = A to C, E to G, J, L))

Correspondence between the address values and block numbers in the flash memory are shown below.

**Table 3 - 1 Correspondence Between Address Values and Block Numbers in Flash Memory (1/4)**

Address Value	Block Number						
00000H to 003FFH	00H	08000H to 083FFH	20H	10000H to 103FFH	40H	18000H to 183FFH	60H
00400H to 007FFH	01H	08400H to 087FFH	21H	10400H to 107FFH	41H	18400H to 187FFH	61H
00800H to 00BFFH	02H	08800H to 08BFFH	22H	10800H to 10BFFH	42H	18800H to 18BFFH	62H
00C00H to 00FFFH	03H	08C00H to 08FFFH	23H	10C00H to 10FFFH	43H	18C00H to 18FFFH	63H
01000H to 013FFH	04H	09000H to 093FFH	24H	11000H to 113FFH	44H	19000H to 193FFH	64H
01400H to 017FFH	05H	09400H to 097FFH	25H	11400H to 117FFH	45H	19400H to 197FFH	65H
01800H to 01BFFH	06H	09800H to 09BFFH	26H	11800H to 11BFFH	46H	19800H to 19BFFH	66H
01C00H to 01FFFH	07H	09C00H to 09FFFH	27H	11C00H to 11FFFH	47H	19C00H to 19FFFH	67H
02000H to 023FFH	08H	0A000H to 0A3FFH	28H	12000H to 123FFH	48H	1A000H to 1A3FFH	68H
02400H to 027FFH	09H	0A400H to 0A7FFH	29H	12400H to 127FFH	49H	1A400H to 1A7FFH	69H
02800H to 02BFFH	0AH	0A800H to 0ABFFH	2AH	12800H to 12BFFH	4AH	1A800H to 1ABFFH	6AH
02C00H to 02FFFH	0BH	0AC00H to 0AFFFH	2BH	12C00H to 12FFFH	4BH	1AC00H to 1AFFFH	6BH
03000H to 033FFH	0CH	0B000H to 0B3FFH	2CH	13000H to 133FFH	4CH	1B000H to 1B3FFH	6CH
03400H to 037FFH	0DH	0B400H to 0B7FFH	2DH	13400H to 137FFH	4DH	1B400H to 1B7FFH	6DH
03800H to 03BFFH	0EH	0B800H to 0BBFFH	2EH	13800H to 13BFFH	4EH	1B800H to 1BBFFH	6EH
03C00H to 03FFFH	0FH	0BC00H to 0BFFFH	2FH	13C00H to 13FFFH	4FH	1BC00H to 1BFFFH	6FH
04000H to 043FFH	10H	0C000H to 0C3FFH	30H	14000H to 143FFH	50H	1C000H to 1C3FFH	70H
04400H to 047FFH	11H	0C400H to 0C7FFH	31H	14400H to 147FFH	51H	1C400H to 1C7FFH	71H
04800H to 04BFFH	12H	0C800H to 0CBFFH	32H	14800H to 14BFFH	52H	1C800H to 1CBFFH	72H
04C00H to 04FFFH	13H	0CC00H to 0CFFFH	33H	14C00H to 14FFFH	53H	1CC00H to 1CFFFH	73H
05000H to 053FFH	14H	0D000H to 0D3FFH	34H	15000H to 153FFH	54H	1D000H to 1D3FFH	74H
05400H to 057FFH	15H	0D400H to 0D7FFH	35H	15400H to 157FFH	55H	1D400H to 1D7FFH	75H
05800H to 05BFFH	16H	0D800H to 0DBFFH	36H	15800H to 15BFFH	56H	1D800H to 1DBFFH	76H
05C00H to 05FFFH	17H	0DC00H to 0DFFFH	37H	15C00H to 15FFFH	57H	1DC00H to 1DFFFH	77H
06000H to 063FFH	18H	0E000H to 0E3FFH	38H	16000H to 163FFH	58H	1E000H to 1E3FFH	78H
06400H to 067FFH	19H	0E400H to 0E7FFH	39H	16400H to 167FFH	59H	1E400H to 1E7FFH	79H
06800H to 06BFFH	1AH	0E800H to 0EBFFH	3AH	16800H to 16BFFH	5AH	1E800H to 1EBFFH	7AH
06C00H to 06FFFH	1BH	0EC00H to 0EFFFH	3BH	16C00H to 16FFFH	5BH	1EC00H to 1EFFFH	7BH
07000H to 073FFH	1CH	0F000H to 0F3FFH	3CH	17000H to 173FFH	5CH	1F000H to 1F3FFH	7CH
07400H to 077FFH	1DH	0F400H to 0F7FFH	3DH	17400H to 177FFH	5DH	1F400H to 1F7FFH	7DH
07800H to 07BFFH	1EH	0F800H to 0FBFFH	3EH	17800H to 17BFFH	5EH	1F800H to 1FBFFH	7EH
07C00H to 07FFFH	1FH	0FC00H to 0FFFFH	3FH	17C00H to 17FFFH	5FH	1FC00H to 1FFFH	7FH

- Remark**
- R5F104xA (x = A to C, E to G): Block numbers 00H to 0FH
  - R5F104xC (x = A to C, E to G, J, L): Block numbers 00H to 1FH
  - R5F104xD (x = A to C, E to G, J, L): Block numbers 00H to 2FH
  - R5F104xE (x = A to C, E to G, J, L): Block numbers 00H to 3FH
  - R5F104xF (x = A to C, E to G, J, L, M, P): Block numbers 00H to 5FH
  - R5F104xG (x = A to C, E to G, J, L, M, P): Block numbers 00H to 7FH

**Table 3 - 2 Correspondence Between Address Values and Block Numbers in Flash Memory (2/4)**

Address Value	Block Number						
20000H to 203FFH	80H	28000H to 283FFH	A0H	30000H to 303FFH	C0H	38000H to 383FFH	E0H
20400H to 207FFH	81H	28400H to 287FFH	A1H	30400H to 307FFH	C1H	38400H to 387FFH	E1H
20800H to 20BFFH	82H	28800H to 28BFFH	A2H	30800H to 30BFFH	C2H	38800H to 38BFFH	E2H
20C00H to 20FFFH	83H	28C00H to 28FFFH	A3H	30C00H to 30FFFH	C3H	38C00H to 38FFFH	E3H
21000H to 213FFH	84H	29000H to 293FFH	A4H	31000H to 313FFH	C4H	39000H to 393FFH	E4H
21400H to 217FFH	85H	29400H to 297FFH	A5H	31400H to 317FFH	C5H	39400H to 397FFH	E5H
21800H to 21BFFH	86H	29800H to 29BFFH	A6H	31800H to 31BFFH	C6H	39800H to 39BFFH	E6H
21C00H to 21FFFH	87H	29C00H to 29FFFH	A7H	31C00H to 31FFFH	C7H	39C00H to 39FFFH	E7H
22000H to 223FFH	88H	2A000H to 2A3FFH	A8H	32000H to 323FFH	C8H	3A000H to 3A3FFH	E8H
22400H to 227FFH	89H	2A400H to 2A7FFH	A9H	32400H to 327FFH	C9H	3A400H to 3A7FFH	E9H
22800H to 22BFFH	8AH	2A800H to 2ABFFH	AAH	32800H to 32BFFH	CAH	3A800H to 3ABFFH	EAH
22C00H to 22FFFH	8BH	2AC00H to 2AFFFH	ABH	32C00H to 32FFFH	CBH	3AC00H to 3AFFFH	EBH
23000H to 233FFH	8CH	2B000H to 2B3FFH	ACH	33000H to 333FFH	CCH	3B000H to 3B3FFH	ECH
23400H to 237FFH	8DH	2B400H to 2B7FFH	ADH	33400H to 337FFH	CDH	3B400H to 3B7FFH	EDH
23800H to 23BFFH	8EH	2B800H to 2BBFFH	AEH	33800H to 33BFFH	CEH	3B800H to 3BBFFH	EEH
23C00H to 23FFFH	8FH	2BC00H to 2BFFFH	AFH	33C00H to 33FFFH	CFH	3BC00H to 3BFFFH	EFH
24000H to 243FFH	90H	2C000H to 2C3FFH	B0H	34000H to 343FFH	D0H	3C000H to 3C3FFH	F0H
24400H to 247FFH	91H	2C400H to 2C7FFH	B1H	34400H to 347FFH	D1H	3C400H to 3C7FFH	F1H
24800H to 24BFFH	92H	2C800H to 2CBFFH	B2H	34800H to 34BFFH	D2H	3C800H to 3CBFFH	F2H
24C00H to 24FFFH	93H	2CC00H to 2CFFFH	B3H	34C00H to 34FFFH	D3H	3CC00H to 3CFFFH	F3H
25000H to 253FFH	94H	2D000H to 2D3FFH	B4H	35000H to 353FFH	D4H	3D000H to 3D3FFH	F4H
25400H to 257FFH	95H	2D400H to 2D7FFH	B5H	35400H to 357FFH	D5H	3D400H to 3D7FFH	F5H
25800H to 25BFFH	96H	2D800H to 2DBFFH	B6H	35800H to 35BFFH	D6H	3D800H to 3DBFFH	F6H
25C00H to 25FFFH	97H	2DC00H to 2DFFFH	B7H	35C00H to 35FFFH	D7H	3DC00H to 3DFFFH	F7H
26000H to 263FFH	98H	2E000H to 2E3FFH	B8H	36000H to 363FFH	D8H	3E000H to 3E3FFH	F8H
26400H to 267FFH	99H	2E400H to 2E7FFH	B9H	36400H to 367FFH	D9H	3E400H to 3E7FFH	F9H
26800H to 26BFFH	9AH	2E800H to 2EBFFH	BAH	36800H to 36BFFH	DAH	3E800H to 3EBFFH	FAH
26C00H to 26FFFH	9BH	2EC00H to 2EFFFH	BBH	36C00H to 36FFFH	DBH	3EC00H to 3EFFFH	FBH
27000H to 273FFH	9CH	2F000H to 2F3FFH	BCH	37000H to 373FFH	DCH	3F000H to 3F3FFH	FCH
27400H to 277FFH	9DH	2F400H to 2F7FFH	BDH	37400H to 377FFH	DDH	3F400H to 3F7FFH	FDH
27800H to 27BFFH	9EH	2F800H to 2FBFFH	BEH	37800H to 37BFFH	DEH	3F800H to 3FBFFH	FEH
27C00H to 27FFFH	9FH	2FC00H to 2FFFFH	BFH	37C00H to 37FFFH	DFH	3FC00H to 3FFFH	FFH

**Remark** R5F104xH (x = E to G, J, L, M, P): Block numbers 00H to BFH  
 R5F104xJ (x = F, G, J, L, M, P): Block numbers 00H to FFH

**Table 3 - 3 Correspondence Between Address Values and Block Numbers in Flash Memory (3/4)**

Address Value	Block Number						
40000H to 403FFH	100H	48000H to 483FFH	120H	50000H to 503FFH	140H	58000H to 583FFH	160H
40400H to 407FFH	101H	48400H to 487FFH	121H	50400H to 507FFH	141H	58400H to 587FFH	161H
40800H to 40BFFH	102H	48800H to 48BFFH	122H	50800H to 50BFFH	142H	58800H to 58BFFH	162H
40C00H to 40FFFH	103H	48C00H to 48FFFH	123H	50C00H to 50FFFH	143H	58C00H to 58FFFH	163H
41000H to 413FFH	104H	49000H to 493FFH	124H	51000H to 513FFH	144H	59000H to 593FFH	164H
41400H to 417FFH	105H	49400H to 497FFH	125H	51400H to 517FFH	145H	59400H to 597FFH	165H
41800H to 41BFFH	106H	49800H to 49BFFH	126H	51800H to 51BFFH	146H	59800H to 59BFFH	166H
41C00H to 41FFFH	107H	49C00H to 49FFFH	127H	51C00H to 51FFFH	147H	59C00H to 59FFFH	167H
42000H to 423FFH	108H	4A000H to 4A3FFH	128H	52000H to 523FFH	148H	5A000H to 5A3FFH	168H
42400H to 427FFH	109H	4A400H to 4A7FFH	129H	52400H to 527FFH	149H	5A400H to 5A7FFH	169H
42800H to 42BFFH	10AH	4A800H to 4ABFFH	12AH	52800H to 52BFFH	14AH	5A800H to 5ABFFH	16AH
42C00H to 42FFFH	10BH	4AC00H to 4AFFFH	12BH	52C00H to 52FFFH	14BH	5AC00H to 5AFFFH	16BH
43000H to 433FFH	10CH	4B000H to 4B3FFH	12CH	53000H to 533FFH	14CH	5B000H to 5B3FFH	16CH
43400H to 437FFH	10DH	4B400H to 4B7FFH	12DH	53400H to 537FFH	14DH	5B400H to 5B7FFH	16DH
43800H to 43BFFH	10EH	4B800H to 4BBFFH	12EH	53800H to 53BFFH	14EH	5B800H to 5BBFFH	16EH
43C00H to 43FFFH	10FH	4BC00H to 4BFFFH	12FH	53C00H to 53FFFH	14FH	5BC00H to 5BFFFH	16FH
44000H to 443FFH	110H	4C000H to 4C3FFH	130H	54000H to 543FFH	150H	5C000H to 5C3FFH	170H
44400H to 447FFH	111H	4C400H to 4C7FFH	131H	54400H to 547FFH	151H	5C400H to 5C7FFH	171H
44800H to 44BFFH	112H	4C800H to 4CBFFH	132H	54800H to 54BFFH	152H	5C800H to 5CBFFH	172H
44C00H to 44FFFH	113H	4CC00H to 4CFFFH	133H	54C00H to 54FFFH	153H	5CC00H to 5CFFFH	173H
45000H to 453FFH	114H	4D000H to 4D3FFH	134H	55000H to 553FFH	154H	5D000H to 5D3FFH	174H
45400H to 457FFH	115H	4D400H to 4D7FFH	135H	55400H to 557FFH	155H	5D400H to 5D7FFH	175H
45800H to 45BFFH	116H	4D800H to 4DBFFH	136H	55800H to 55BFFH	156H	5D800H to 5DBFFH	176H
45C00H to 45FFFH	117H	4DC00H to 4DFFFH	137H	55C00H to 55FFFH	157H	5DC00H to 5DFFFH	177H
46000H to 463FFH	118H	4E000H to 4E3FFH	138H	56000H to 563FFH	158H	5E000H to 5E3FFH	178H
46400H to 467FFH	119H	4E400H to 4E7FFH	139H	56400H to 567FFH	159H	5E400H to 5E7FFH	179H
46800H to 46BFFH	11AH	4E800H to 4EBFFH	13AH	56800H to 56BFFH	15AH	5E800H to 5EBFFH	17AH
46C00H to 46FFFH	11BH	4EC00H to 4EFFFH	13BH	56C00H to 56FFFH	15BH	5EC00H to 5EFFFH	17BH
47000H to 473FFH	11CH	4F000H to 4F3FFH	13CH	57000H to 573FFH	15CH	5F000H to 5F3FFH	17CH
47400H to 477FFH	11DH	4F400H to 4F7FFH	13DH	57400H to 577FFH	15DH	5F400H to 5F7FFH	17DH
47800H to 47BFFH	11EH	4F800H to 4FBFFH	13EH	57800H to 57BFFH	15EH	5F800H to 5FBFFH	17EH
47C00H to 47FFFH	11FH	4FC00H to 4FFFFH	13FH	57C00H to 57FFFH	15FH	5FC00H to 5FFFFH	17FH

**Remark** R5F104xK (x = G, L, M, P): Block numbers 00H to 17FH

**Table 3 - 4 Correspondence Between Address Values and Block Numbers in Flash Memory (4/4)**

Address Value	Block Number						
60000H to 603FFH	180H	68000H to 683FFH	1A0H	70000H to 703FFH	1C0H	78000H to 783FFH	1E0H
60400H to 607FFH	181H	68400H to 687FFH	1A1H	70400H to 707FFH	1C1H	78400H to 787FFH	1E1H
60800H to 60BFFH	182H	68800H to 68BFFH	1A2H	70800H to 70BFFH	1C2H	78800H to 78BFFH	1E2H
60C00H to 60FFFH	183H	68C00H to 68FFFH	1A3H	70C00H to 70FFFH	1C3H	78C00H to 78FFFH	1E3H
61000H to 613FFH	184H	69000H to 693FFH	1A4H	71000H to 713FFH	1C4H	79000H to 793FFH	1E4H
61400H to 617FFH	185H	69400H to 697FFH	1A5H	71400H to 717FFH	1C5H	79400H to 797FFH	1E5H
61800H to 61BFFH	186H	69800H to 69BFFH	1A6H	71800H to 71BFFH	1C6H	79800H to 79BFFH	1E6H
61C00H to 61FFFH	187H	69C00H to 69FFFH	1A7H	71C00H to 71FFFH	1C7H	79C00H to 79FFFH	1E7H
62000H to 623FFH	188H	6A000H to 6A3FFH	1A8H	72000H to 723FFH	1C8H	7A000H to 7A3FFH	1E8H
62400H to 627FFH	189H	6A400H to 6A7FFH	1A9H	72400H to 727FFH	1C9H	7A400H to 7A7FFH	1E9H
62800H to 62BFFH	18AH	6A800H to 6ABFFH	1AAH	72800H to 72BFFH	1CAH	7A800H to 7ABFFH	1EAH
62C00H to 62FFFH	18BH	6AC00H to 6AFFFH	1ABH	72C00H to 72FFFH	1CBH	7AC00H to 7AFFFH	1EBH
63000H to 633FFH	18CH	6B000H to 6B3FFH	1ACH	73000H to 733FFH	1CCH	7B000H to 7B3FFH	1ECH
63400H to 637FFH	18DH	6B400H to 6B7FFH	1ADH	73400H to 737FFH	1CDH	7B400H to 7B7FFH	1EDH
63800H to 63BFFH	18EH	6B800H to 6BBFFH	1AEH	73800H to 73BFFH	1CEH	7B800H to 7BBFFH	1EEH
63C00H to 63FFFH	18FH	6BC00H to 6BFFFH	1AFH	73C00H to 73FFFH	1CFH	7BC00H to 7BFFFH	1EFH
64000H to 643FFH	190H	6C000H to 6C3FFH	1B0H	74000H to 743FFH	1D0H	7C000H to 7C3FFH	1F0H
64400H to 647FFH	191H	6C400H to 6C7FFH	1B1H	74400H to 747FFH	1D1H	7C400H to 7C7FFH	1F1H
64800H to 64BFFH	192H	6C800H to 6CBFFH	1B2H	74800H to 74BFFH	1D2H	7C800H to 7CBFFH	1F2H
64C00H to 64FFFH	193H	6CC00H to 6CFFFH	1B3H	74C00H to 74FFFH	1D3H	7CC00H to 7CFFFH	1F3H
65000H to 653FFH	194H	6D000H to 6D3FFH	1B4H	75000H to 753FFH	1D4H	7D000H to 7D3FFH	1F4H
65400H to 657FFH	195H	6D400H to 6D7FFH	1B5H	75400H to 757FFH	1D5H	7D400H to 7D7FFH	1F5H
65800H to 65BFFH	196H	6D800H to 6DBFFH	1B6H	75800H to 75BFFH	1D6H	7D800H to 7DBFFH	1F6H
65C00H to 65FFFH	197H	6DC00H to 6DFFFH	1B7H	75C00H to 75FFFH	1D7H	7DC00H to 7DFFFH	1F7H
66000H to 663FFH	198H	6E000H to 6E3FFH	1B8H	76000H to 763FFH	1D8H	7E000H to 7E3FFH	1F8H
66400H to 667FFH	199H	6E400H to 6E7FFH	1B9H	76400H to 767FFH	1D9H	7E400H to 7E7FFH	1F9H
66800H to 66BFFH	19AH	6E800H to 6EBFFH	1BAH	76800H to 76BFFH	1DAH	7E800H to 7EBFFH	1FAH
66C00H to 66FFFH	19BH	6EC00H to 6EFFFH	1BBH	76C00H to 76FFFH	1DBH	7EC00H to 7EFFFH	1FBH
67000H to 673FFH	19CH	6F000H to 6F3FFH	1BCH	77000H to 773FFH	1DCH	7F000H to 7F3FFH	1FCH
67400H to 677FFH	19DH	6F400H to 6F7FFH	1BDH	77400H to 777FFH	1DDH	7F400H to 7F7FFH	1FDH
67800H to 67BFFH	19EH	6F800H to 6FBFFH	1BEH	77800H to 77BFFH	1DEH	7F800H to 7FBFFH	1FEH
67C00H to 67FFFH	19FH	6FC00H to 6FFFFH	1BFH	77C00H to 77FFFH	1DFH	7FC00H to 7FFFFH	1FFH

**Remark** R5F104xL (x = G, L, M, P): Block numbers 00H to 1FFH

### 3.1.1 Internal program memory space

The internal program memory space stores the program and table data.

The RL78/G14 products incorporate internal ROM (flash memory), as shown below.

**Table 3 - 5 Internal ROM Capacity**

Part Number	Internal ROM	
	Structure	Capacity
R5F104xA (x = A to C, E to G)	Flash memory	16384 × 8 bits (00000H to 03FFFFH)
R5F104xC (x = A to C, E to G, J, L)		32768 × 8 bits (00000H to 07FFFFH)
R5F104xD (x = A to C, E to G, J, L)		49152 × 8 bits (00000H to 0BFFFFH)
R5F104xE (x = A to C, E to G, J, L)		65536 × 8 bits (00000H to 0FFFFFH)
R5F104xF (x = A to C, E to G, J, L, M, P)		98304 × 8 bits (00000H to 17FFFFH)
R5F104xG (x = A to C, E to G, J, L, M, P)		131072 × 8 bits (00000H to 1FFFFFFH)
R5F104xH (x = E to G, J, L, M, P)		196608 × 8 bits (00000H to 2FFFFFFH)
R5F104xJ (x = F, G, J, L, M, P)		262144 × 8 bits (00000H to 3FFFFFFH)
R5F104xK (x = G, L, M, P)		393216 × 8 bits (00000H to 5FFFFFFH)
R5F104xL (x = G, L, M, P)		524288 × 8 bits (00000H to 7FFFFFFH)

The internal program memory space is divided into the following areas.

(1) Vector table area

The 128-byte area 00000H to 0007FH is reserved as a vector table area. The program start addresses for branch upon reset or generation of each interrupt request are stored in the vector table area. Furthermore, the interrupt jump address is a 64 K address of 00000H to 0FFFFH, because the vector code is assumed to be 2 bytes.

Of the 16-bit address, the lower 8 bits are stored at even addresses and the higher 8 bits are stored at odd addresses.

To use the boot swap function, set a vector table also at 01000H to 0107FH.

Tables 3 - 6 to 3 - 7 list the vector table. “/” indicates an interrupt source which is supported. “—” indicates an interrupt source which is not supported.

**Table 3 - 6 Vector Table (1/2)**

Vector Table Address	Interrupt Source	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
00000H	RESET, POR, LVD, WDT, TRAP, IAW, RPE	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00004H	INTWDTI	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00006H	INTLVI	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00008H	INTP0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0000AH	INTP1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0000CH	INTP2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0000EH	INTP3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00010H	INTP4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00012H	INTP5	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00014H	INTST2/INTCSI20/INTIIC20	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00016H	INTSR2/INTCSI21/INTIIC21	✓	✓	✓	✓	✓	✓	✓	✓	Note 1	Note 1
00018H	INTSRE2	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	INTTM11H	✓	✓	—	—	—	—	—	—	—	—
0001EH	INTST0/INTCSI00/INTIIC00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00020H	INTSR0/INTCSI01/INTIIC01	✓	✓	✓	✓	✓	✓	Note 2	Note 2	Note 2	Note 2
00022H	INTSRE0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	INTTM01H	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00024H	INTST1/INTCSI10/INTIIC10	✓	✓	✓	✓	Note 3					
00026H	INTSR1/INTCSI11/INTIIC11	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00028H	INTSRE1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	INTTM03H	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0002AH	INTIICA0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0002CH	INTTM00	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0002EH	INTTM01	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00030H	INTTM02	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00032H	INTTM03	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00034H	INTAD	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00036H	INTRTC	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00038H	INTIT	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0003AH	INTKR	✓	✓	✓	✓	✓	✓	✓	—	—	—
0003CH	INTST3/INTCSI30/INTIIC30	✓	✓	—	—	—	—	—	—	—	—
0003EH	INTSR3/INTCSI31/INTIIC31	✓	✓	—	—	—	—	—	—	—	—
00040H	INTTRJ0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00042H	INTTM10	✓	✓	—	—	—	—	—	—	—	—
00044H	INTTM11	✓	✓	—	—	—	—	—	—	—	—
00046H	INTTM12	✓	✓	—	—	—	—	—	—	—	—
00048H	INTTM13	✓	✓	—	—	—	—	—	—	—	—

**Note 1.** Only INTSR2 is supported.

**Note 2.** Only INTSR0 is supported.

**Note 3.** Only INTST1 is supported.

**Table 3 - 7 Vector Table (2/2)**

Vector Table Address	Interrupt Source	100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
0004AH	INTP6	✓	✓	✓	✓	✓	—	—	—	—	—
0004CH	INTP7	✓	✓	✓	—	—	—	—	—	—	—
0004EH	INTP8	✓	✓	✓	✓	✓	—	—	—	—	—
00050H	INTP9	✓	✓	✓	✓	✓	—	—	—	—	—
00052H	INTP10	✓	✓	✓	✓	—	—	—	—	—	—
	INTCMP0	✓	✓	Note							
00054H	INTP11	✓	✓	✓	✓	—	—	—	—	—	—
	INTCMP1	✓	✓	Note							
00056H	INTTRD0	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
00058H	INTTRD1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0005AH	INTTRG	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0005CH	INTSRE3	✓	✓	—	—	—	—	—	—	—	—
	INTTM13H	✓	✓	—	—	—	—	—	—	—	—
00060H	INTIICA1	✓	✓	—	—	—	—	—	—	—	—
00062H	INTFL	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
0007EH	BRK	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

**Note** Supported only in the products with 96 KB or more code flash memory.

(2) CALLT instruction table area

The 64-byte area 00080H to 000BFH can store the subroutine entry address of a 2-byte call instruction (CALLT). Set the subroutine entry address to a value in a range of 00000H to 0FFFFH (because an address code is 2 bytes).

To use the boot swap function, set a CALLT instruction table also at 01080H to 010BFH.

(3) Option byte area

A 4-byte area of 000C0H to 000C3H can be used as an option byte area. Set the option byte at 010C0H to 010C3H when the boot swap is used. For details, see **CHAPTER 29 OPTION BYTE**.

(4) On-chip debug security ID setting area

A 10-byte area of 000C4H to 000CDH and 010C4H to 010CDH can be used as an on-chip debug security ID setting area. Set the on-chip debug security ID of 10 bytes at 000C4H to 000CDH when the boot swap is not used and at 000C4H to 000CDH and at 010C4H to 010CDH when the boot swap is used. For details, see **CHAPTER 31 ON-CHIP DEBUG FUNCTION**.

### 3.1.2 Mirror area

The RL78/G14 mirrors the code flash area of 00000H to 0FFFFH, to F0000H to FFFFFH. The products with 96 KB or more flash memory mirror the code flash area of 00000H to 0FFFFH or 10000H to 1FFFFH, to F0000H to FFFFFH (the code flash area to be mirrored is set by the processor mode control register (PMC)).

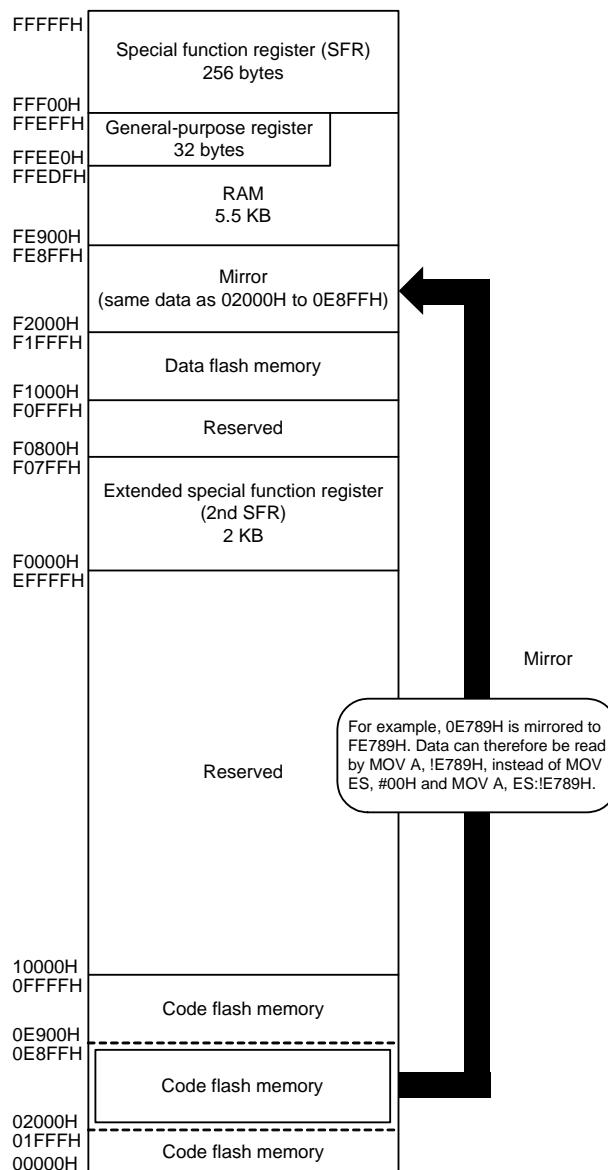
By reading data from F0000H to FFFFFH, an instruction that does not have the ES register as an operand can be used, and thus the contents of the code flash can be read with the shorter code. However, the code flash area is not mirrored to the special function register (SFR), extended special function register (2nd SFR), RAM, data flash memory, and use prohibited areas.

See **3.1 Memory Space** for the mirror area of each product.

The mirror area can only be read and no instruction can be fetched from this area.

The following show examples.

Example R5F104xE (x = A to C, E to G, J, L) (Flash memory: 64 KB, RAM: 5.5 KB)



The PMC register is described below.

- Processor mode control register (PMC)

This register sets the flash memory space for mirroring to area from F0000H to FFFFFH.

The PMC register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Figure 3 - 11 Format of Configuration of Processor mode control register (PMC)**

Address: FFFFEH      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	<0>
--------	---	---	---	---	---	---	---	-----

PMC	0	0	0	0	0	0	0	MAA
-----	---	---	---	---	---	---	---	-----

MAA	Selection of flash memory space for mirroring to area from F0000H to FFFFFH
0	00000H to 0FFFFH is mirrored to F0000H to FFFFFH
1	10000H to 1FFFFH is mirrored to F0000H to FFFFFH

**Caution 1.** In products with 64 KB or less flash memory, be sure to clear bit 0 (MAA) of this register to 0 (default value).

**Caution 2.** After setting the PMC register, wait for at least one instruction and access the mirror area.

### 3.1.3 Internal data memory space

The RL78/G14 products incorporate the following RAMs.

**Table 3 - 8 Internal RAM Capacity**

Part Number	Internal RAM
R5F104xA (x = A to C, E to G)	2560 × 8 bits (FF500H to FFEFFFH)
R5F104xC (x = A to C, E to G, J, L)	4096 × 8 bits (FEF00H to FFEFFFH)
R5F104xD (x = A to C, E to G, J, L)	5632 × 8 bits (FE900H to FFEFFFH)
R5F104xE (x = A to C, E to G, J, L)	
R5F104xF (x = A to C, E to G, J, L, M, P)	12288 × 8 bits (FCF00H to FFEFFFH)
R5F104xG (x = A to C, E to G, J, L, M, P)	16384 × 8 bits (FBF00H to FFEFFFH)
R5F104xH (x = E to G, J, L, M, P)	20480 × 8 bits (FAF00H to FFEFFFH)
R5F104xJ (x = F, G, J, L, M, P)	24576 × 8 bits (F9F00H to FFEFFFH)
R5F104xK (x = G, L, M, P)	32768 × 8 bits (F7F00H to FFEFFFH)
R5F104xL (x = G, L, M, P)	49152 × 8 bits (F3F00H to FFEFFFH)

The internal RAM can be used as a data area and a program area where instructions are fetched (it is prohibited to use the general-purpose register area for fetching instructions). Four general-purpose register banks consisting of eight 8-bit registers per bank are assigned to the 32-byte area of FFEE0H to FFEFFFH of the internal RAM area.

The internal RAM is used as stack memory.

**Caution 1. It is prohibited to use the general-purpose register (FFEE0H to FFEFFFH) space for fetching instructions or as a stack area.**

**Caution 2. Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.**

**Caution 3. The flash library uses RAM in self-programming and rewriting of the data flash memory.**

The target products and start address of the RAM areas used by the flash library are shown below.

R5F104xD (x = A to C, E to G, J, L): Start address FE900H

R5F104xE (x = A to C, E to G, J, L): Start address FE900H

R5F104xJ (x = F, G, J, L, M, P): Start address F9F00H

R5F104xL (x = G, L, M, P): Start address F3F00H

For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

**Caution 4. The internal RAM area in the following products cannot be used as stack memory when using the on-chip debugging trace function.**

R5F104xJ (x = A to C, E to G, J, L): FA300H to FA6FFH

R5F104xL (x = G, L, M, P): F4300H to F46FFH

### 3.1.4 Special function register (SFR) area

On-chip peripheral hardware special function registers (SFRs) are allocated in the area FFF00H to FFFFFH (see **Tables 3 - 9 to 3 - 13** in **3.2.4 Special function registers (SFRs)**).

**Caution** Do not access addresses to which SFRs are not assigned.

### 3.1.5 Extended special function register (2nd SFR: 2nd Special Function Register) area

On-chip peripheral hardware special function registers (2nd SFRs) are allocated in the area F0000H to F07FFH (see **Tables 3 - 14 to 3 - 22** in **3.2.5 Extended special function registers (2nd SFRs: 2nd Special Function Registers)**).

**Caution 1.** Do not access addresses to which extended SFRs are not assigned.

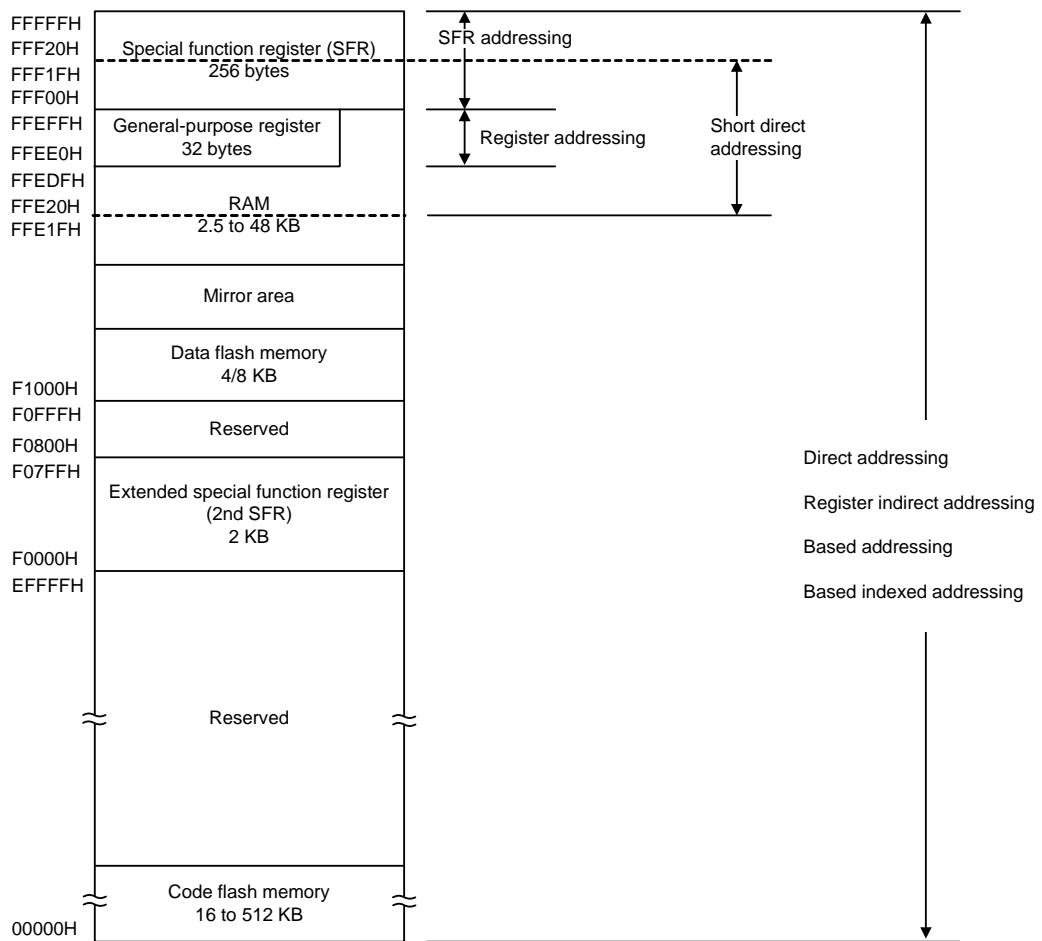
**Caution 2.** When accessing timer RJ counter register 0 (TRJ0) allocated in F0500H of the extended SFR (2nd SFR), the CPU does not proceed to the next instruction processing but enters the wait state for CPU processing. For this reason, if this wait state occurs, the number of instruction execution clocks is increased by the number of wait clocks. The number of wait clocks for access to timer RJ counter register 0 (TRJ0) is one clock for both writing and reading.

### 3.1.6 Data memory addressing

Addressing refers to the method of specifying the address of the instruction to be executed next or the address of the register or memory relevant to the execution of instructions.

Several addressing modes are provided for addressing the memory relevant to the execution of instructions for the RL78/G14, based on operability and other considerations. For areas containing data memory in particular, special addressing methods designed for the functions of the special function registers (SFR) and general-purpose registers are available for use. Figure 3 - 12 shows correspondence between data memory and addressing. For details of each addressing, see **3.4 Addressing for Processing Data Addresses**.

**Figure 3 - 12 Correspondence Between Data Memory and Addressing**



## 3.2 Processor Registers

The RL78/G14 products incorporate the following processor registers.

### 3.2.1 Control registers

The control registers control the program sequence, statuses and stack memory. The control registers consist of a program counter (PC), a program status word (PSW) and a stack pointer (SP).

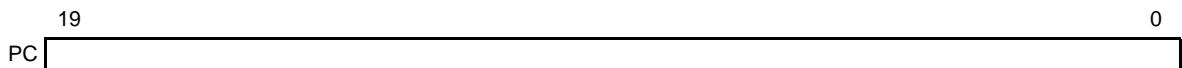
#### (1) Program counter (PC)

The program counter is a 20-bit register that holds the address information of the next program to be executed.

In normal operation, PC is automatically incremented according to the number of bytes of the instruction to be fetched. When a branch instruction is executed, immediate data and register contents are set.

Reset signal generation sets the reset vector table values at addresses 00000H and 00001H to the program counter.

**Figure 3 - 13 Format of Program Counter**

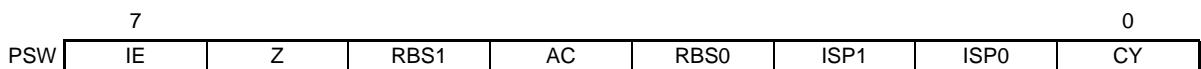


#### (2) Program status word (PSW)

The program status word is an 8-bit register consisting of various flags set/reset by instruction execution.

Program status word contents are stored in the stack area upon vectored interrupt request is acknowledged or PUSH PSW instruction execution and are restored upon execution of the RETB, RETI and POP PSW instructions. Reset signal generation sets the PSW register to 06H.

**Figure 3 - 14 Format of Program Status Word**



#### (a) Interrupt enable flag (IE)

This flag controls the interrupt request acknowledge operations of the CPU.

When 0, the IE flag is set to the interrupt disabled (DI) state, and all maskable interrupt requests are disabled.

When 1, the IE flag is set to the interrupt enabled (EI) state and maskable interrupt requests acknowledgment is controlled with an in-service priority flag (ISP1, ISP0), an interrupt mask flag for various interrupt sources, and a priority specification flag.

The IE flag is reset (0) upon DI instruction execution or interrupt acknowledgment and is set (1) upon EI instruction execution.

## (b) Zero flag (Z)

When the operation result is zero or equal, this flag is set (1). It is reset (0) in all other cases.

## (c) Register bank select flags (RBS0, RBS1)

These are 2-bit flags to select one of the four register banks.

In these flags, the 2-bit information that indicates the register bank selected by SEL RBn instruction execution is stored.

## (d) Auxiliary carry flag (AC)

If the operation result has a carry from bit 3 or a borrow at bit 3, this flag is set (1). It is reset (0) in all other cases.

## (e) In-service priority flags (ISP1, ISP0)

This flag manages the priority of acknowledgeable maskable vectored interrupts. Vectored interrupt requests specified lower than the value of ISP0 and ISP1 flags by the priority specification flag registers (PRn0L, PRn0H, PRn1L, PRn1H, PRn2L, PRn2H) (see 21.3.3) can not be acknowledged. Actual vectored interrupt requests acknowledgment is controlled by the interrupt enable flag (IE).

**Remark** n = 0, 1

## (f) Carry flag (CY)

This flag stores overflow and underflow upon add/subtract instruction execution. It stores the shift-out value upon rotate instruction execution and functions as a bit accumulator during bit operation instruction execution.

## (3) Stack pointer (SP)

This is a 16-bit register to hold the start address of the memory stack area. Only the internal RAM area can be set as the stack area.

**Figure 3 - 15 Format of Stack Pointer**

15																0
SP	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8	SP7	SP6	SP5	SP4	SP3	SP2	SP1	0

In stack addressing through a stack pointer, the SP is decremented ahead of write (save) to the stack memory and is incremented after read (restore) from the stack memory.

**Caution 1. Since reset signal generation makes the SP contents undefined, be sure to initialize the SP before using the stack.**

**Caution 2. It is prohibited to use the general-purpose register (FFEE0H to FFEFFFH) space for fetching instructions or as a stack area.**

**Caution 3. Do not allocate the stack area, data buffers for use by the flash library, arguments of library functions, branch destinations in the processing of vectored interrupts, or destinations or sources for DTC transfer to the area from FFE20H to FFEDFH when performing self-programming or rewriting of the data flash memory.**

**Caution 4. The flash library uses RAM in self-programming and rewriting of the data flash memory. The target products and start address of the RAM areas used by the flash library are shown below.**

R5F104xD (x = A to C, E to G, J, L): Start address FE900H

R5F104xE (x = A to C, E to G, J, L): Start address FE900H

R5F104xJ (x = F, G, J, L, M, P): Start address F9F00H

R5F104xL (x = G, L, M, P): Start address F3F00H

For the RAM areas used by the flash library, see **Self RAM list of Flash Self-Programming Library for RL78 Family (R20UT2944)**.

**Caution 5. The internal RAM area in the following products cannot be used as stack memory when using the on-chip debugging trace function.**

R5F104xJ (x = A to C, E to G, J, L): FA300H to FA6FFH

R5F104xL (x = G, L, M, P): F4300H to F46FFH

### 3.2.2 General-purpose registers

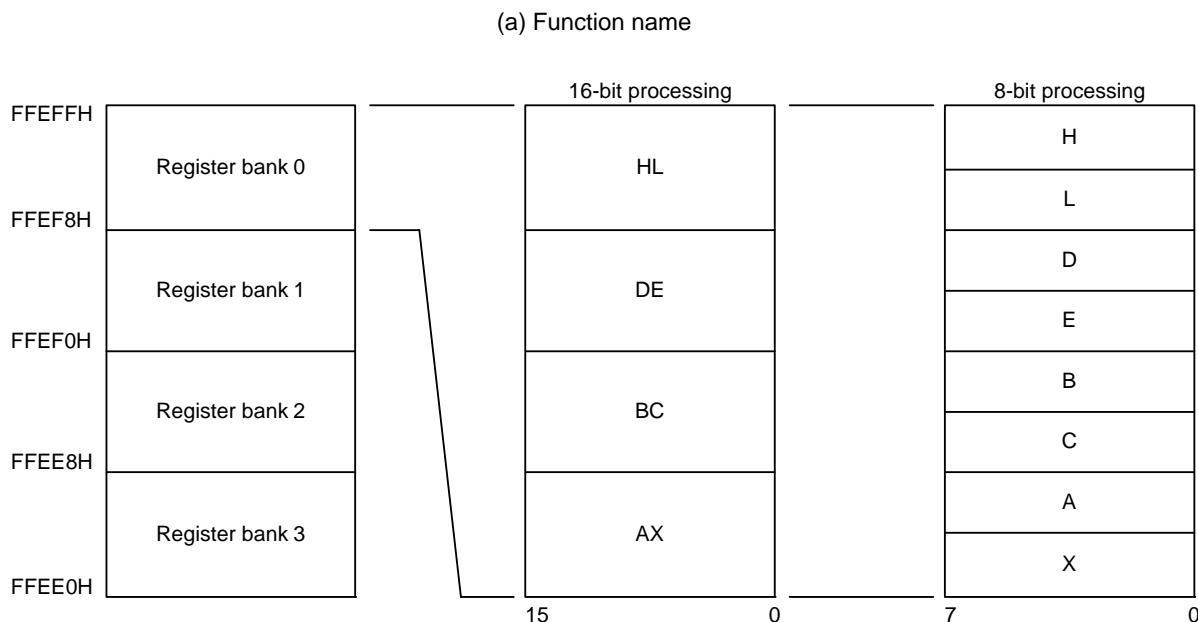
General-purpose registers are mapped at particular addresses (FFEE0H to FFEFFFH) of the data memory. The general-purpose registers consists of 4 banks, each bank consisting of eight 8-bit registers (X, A, C, B, E, D, L, and H).

Each register can be used as an 8-bit register, and two 8-bit registers can also be used in a pair as a 16-bit register (AX, BC, DE, and HL).

Register banks to be used for instruction execution are set by the CPU control instruction (SEL RBn). Because of the 4-register bank configuration, an efficient program can be created by switching between a register for normal processing and a register for interrupt processing for each bank.

**Caution** It is prohibited to use the general-purpose register (FFEE0H to FFEFFFH) space for fetching instructions or as a stack area.

Figure 3 - 16 Configuration of General-Purpose Registers



### 3.2.3 ES and CS registers

The ES register and CS register are used to specify the higher address for data access and when a branch instruction is executed (register direct addressing), respectively.

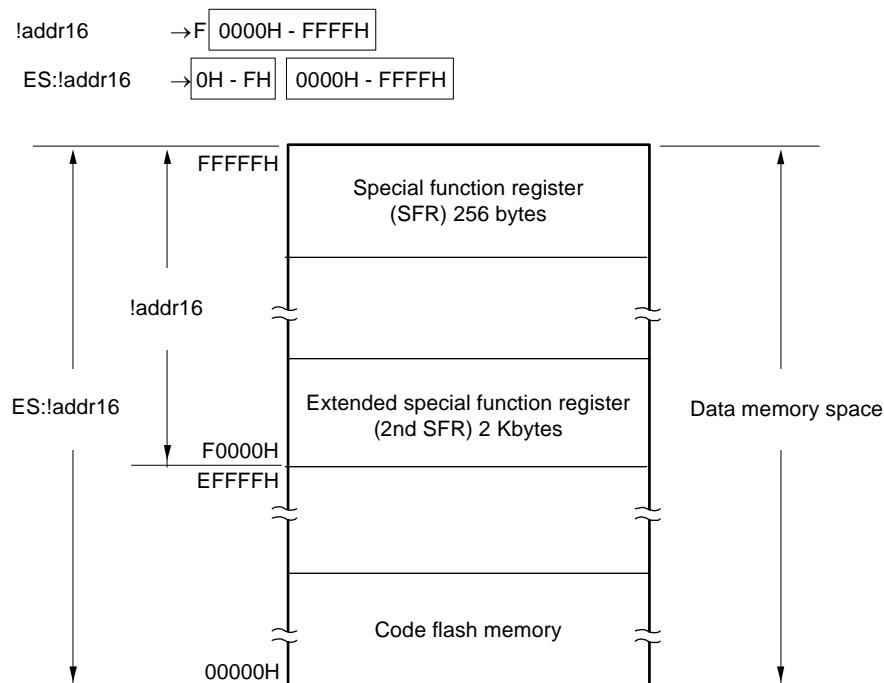
The default value of the ES register after reset is 0FH, and that of the CS register is 00H.

**Figure 3 - 17 Configuration of ES and CS Registers**

7	6	5	4	3	2	1	0	
ES	0	0	0	0	ES3	ES2	ES1	ES0
CS	0	0	0	0	CS3	CS2	CS1	CS0

Though the data area which can be accessed with 16-bit addresses is the 64 Kbytes from F0000H to FFFFFH, using the ES register as well extends this to the 1 Mbyte from 00000H to FFFFFH.

**Figure 3 - 18 Extension of Data Area Which Can Be Accessed**



### 3.2.4 Special function registers (SFRs)

Unlike a general-purpose register, each SFR has a special function.

SFRs are allocated to the FFF00H to FFFFFH area.

SFRs can be manipulated like general-purpose registers, using operation, transfer, and bit manipulation instructions. The manipulable bit units, 1, 8, and 16, depend on the SFR type.

Each manipulation bit unit can be specified as follows.

- 1-bit manipulation

Describe as follows for the 1-bit manipulation instruction operand (sfr.bit).

When the bit name is defined: <Bit name>

When the bit name is not defined: <Register name>, <Bit number> or <Address>, <Bit number>

- 8-bit manipulation

Describe the symbol defined by the assembler for the 8-bit manipulation instruction operand (sfr). This manipulation can also be specified with an address.

- 16-bit manipulation

Describe the symbol defined by the assembler for the 16-bit manipulation instruction operand (sfrp). When specifying an address, describe an even address.

Tables 3 - 9 to 3 - 13 give lists of the SFRs. The meanings of items in the table are as follows.

- Symbol

This item indicates the address of a special function register. It is a reserved word in the assembler, and is defined as an sfr variable using the #pragma sfr directive in the compiler. When using the assembler, debugger, and simulator, symbols can be written as an instruction operand.

- R/W

This item indicates whether the corresponding SFR can be read or written.

R/W: Read/write enable

R: Read only

W: Write only

- Manipulable bit units

“√” indicates the manipulable bit unit (1, 8, or 16). “—” indicates a bit unit for which manipulation is not possible.

- After reset

This item indicates each register status upon reset signal generation.

**Caution** Do not access addresses to which SFRs are not assigned.

**Remark** For extended SFRs (2nd SFRs), see **3.2.5 Extended special function registers (2nd SFRs: 2nd Special Function Registers)**.

**Table 3 - 9 Special Function Register (SFR) List (1/5)**

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
FFF00H	Port register 0	P0	R/W	✓	✓	—	00H
FFF01H	Port register 1	P1	R/W	✓	✓	—	00H
FFF02H	Port register 2	P2	R/W	✓	✓	—	00H
FFF03H	Port register 3	P3	R/W	✓	✓	—	00H
FFF04H	Port register 4	P4	R/W	✓	✓	—	00H
FFF05H	Port register 5	P5	R/W	✓	✓	—	00H
FFF06H	Port register 6	P6	R/W	✓	✓	—	00H
FFF07H	Port register 7	P7	R/W	✓	✓	—	00H
FFF08H	Port register 8	P8	R/W	✓	✓	—	00H
FFF0AH	Port register 10	P10	R/W	✓	✓	—	00H
FFF0BH	Port register 11	P11	R/W	✓	✓	—	00H
FFF0CH	Port register 12	P12	R/W	✓	✓	—	Undefined
FFF0DH	Port register 13	P13	R/W	✓	✓	—	Undefined
FFF0EH	Port register 14	P14	R/W	✓	✓	—	00H
FFF0FH	Port register 15	P15	R/W	✓	✓	—	00H
FFF10H	Serial data register 00	TXD0/ SIO00	R/W	—	✓	✓	0000H
FFF11H		—		—	—	—	
FFF12H	Serial data register 01	RXD0/ SIO01	R/W	—	✓	✓	0000H
FFF13H		—		—	—	—	
FFF14H	Serial data register 12	RXD3/ SIO30	R/W	—	✓	✓	0000H
FFF15H		—		—	—	—	
FFF16H	Serial data register 13	RXD3/ SIO31	R/W	—	✓	✓	0000H
FFF17H		—		—	—	—	
FFF18H	Timer data register 00	TDR00	R/W	—	—	✓	0000H
FFF19H							
FFF1AH	Timer data register 01	TDR01L	R/W	—	✓	✓	00H
FFF1BH		TDR01H		—	✓	✓	
FFF1EH	10-bit A/D conversion result register	ADCR	R	—	—	✓	0000H
FFF1FH	8-bit A/D conversion result register	ADCRH	R	—	✓	—	00H
FFF20H	Port mode register 0	PM0	R/W	✓	✓	—	FFH
FFF21H	Port mode register 1	PM1	R/W	✓	✓	—	FFH
FFF22H	Port mode register 2	PM2	R/W	✓	✓	—	FFH
FFF23H	Port mode register 3	PM3	R/W	✓	✓	—	FFH
FFF24H	Port mode register 4	PM4	R/W	✓	✓	—	FFH
FFF25H	Port mode register 5	PM5	R/W	✓	✓	—	FFH
FFF26H	Port mode register 6	PM6	R/W	✓	✓	—	FFH
FFF27H	Port mode register 7	PM7	R/W	✓	✓	—	FFH

**Table 3 - 10 Special Function Register (SFR) List (2/5)**

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset	
				1-bit	8-bit	16-bit		
FFF28H	Port mode register 8	PM8	R/W	✓	✓	—	FFH	
FFF2AH	Port mode register 10	PM10	R/W	✓	✓	—	FFH	
FFF2BH	Port mode register 11	PM11	R/W	✓	✓	—	FFH	
FFF2CH	Port mode register 12	PM12	R/W	✓	✓	—	FFH	
FFF2EH	Port mode register 14	PM14	R/W	✓	✓	—	FFH	
FFF2FH	Port mode register 15	PM15	R/W	✓	✓	—	FFH	
FFF30H	A/D converter mode register 0	ADM0	R/W	✓	✓	—	00H	
FFF31H	Analog input channel specification register	ADS	R/W	✓	✓	—	00H	
FFF32H	A/D converter mode register 1	ADM1	R/W	✓	✓	—	00H	
FFF34H	D/A conversion value setting register 0	DACS0	R/W	—	✓	—	00H	
FFF35H	D/A conversion value setting register 1	DACS1	R/W	—	✓	—	00H	
FFF36H	D/A converter mode register	DAM	R/W	✓	✓	—	00H	
FFF37H	Key return mode register	KRM	R/W	✓	✓	—	00H	
FFF38H	External interrupt rising edge enable register 0	EGP0	R/W	✓	✓	—	00H	
FFF39H	External interrupt falling edge enable register 0	EGN0	R/W	✓	✓	—	00H	
FFF3AH	External interrupt rising edge enable register 1	EGP1	R/W	✓	✓	—	00H	
FFF3BH	External interrupt falling edge enable register 1	EGN1	R/W	✓	✓	—	00H	
FFF44H	Serial data register 02	TXD1/ SIO10	SDR02	R/W	—	✓	✓	0000H
FFF45H		—			—	—		
FFF46H	Serial data register 03	RXD1/ SIO11	SDR03	R/W	—	✓	✓	0000H
FFF47H		—			—	—		
FFF48H	Serial data register 10	TXD2/ SIO20	SDR10	R/W	—	✓	✓	0000H
FFF49H		—			—	—		
FFF4AH	Serial data register 11	RXD2/ SIO21	SDR11	R/W	—	✓	✓	0000H
FFF4BH		—			—	—		
FFF50H	IICA shift register 0	IICA0	R/W	—	✓	—	00H	
FFF51H	IICA status register 0	IICS0	R	✓	✓	—	00H	
FFF52H	IICA flag register 0	IICF0	R/W	✓	✓	—	00H	
FFF54H	IICA shift register 1	IICA1	R/W	—	✓	—	00H	
FFF55H	IICA status register 1	IICS1	R	✓	✓	—	00H	
FFF56H	IICA flag register 1	IICF1	R/W	✓	✓	—	00H	

**Table 3 - 11 Special Function Register (SFR) List (3/5)**

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
FFF58H	Timer RD general register C0	TRDGRC0	R/W	—	—	✓	FFFFH Note 1
FFF59H				—	—	✓	
FFF5AH	Timer RD general register D0	TRDGRD0	R/W	—	—	✓	FFFFH Note 1
FFF5BH				—	—	✓	
FFF5CH	Timer RD general register C1	TRDGRC1	R/W	—	—	✓	FFFFH Note 1
FFF5DH				—	—	✓	
FFF5EH	Timer RD general register D1	TRDGRD1	R/W	—	—	✓	FFFFH Note 1
FFF5FH				—	—	✓	
FFF60H	Timer RD general register C	TRGGRC	R/W	—	—	✓	FFFFH
FFF61H				—	—	✓	
FFF62H	Timer RD general register D	TRGGRD	R/W	—	—	✓	FFFFH
FFF63H				—	—	✓	
FFF64H	Timer data register 02	TDR02	R/W	—	—	✓	0000H
FFF65H				—	—	✓	
FFF66H	Timer data register 03	TDR03L	R/W	—	✓	✓	00H
FFF67H		TDR03H		—	✓	✓	
FFF70H	Timer data register 10	TDR10	R/W	—	—	✓	0000H
FFF71H				—	—	✓	
FFF72H	Timer data register 11	TDR11L	R/W	—	✓	✓	00H
FFF73H		TDR11H		—	✓	✓	
FFF74H	Timer data register 12	TDR12	R/W	—	—	✓	0000H
FFF75H				—	—	✓	
FFF76H	Timer data register 13	TDR13L	R/W	—	✓	✓	00H
FFF77H		TDR13H		—	✓	✓	
FFF90H	12-bit interval timer control register	ITMC	R/W	—	—	✓	0FFFH
FFF91H				—	—	✓	
FFF92H	Second count register	SEC	R/W	—	✓	—	00H
FFF93H	Minute count register	MIN	R/W	—	✓	—	00H
FFF94H	Hour count register	HOUR	R/W	—	✓	—	12H Note 2
FFF95H	Week count register	WEEK	R/W	—	✓	—	00H
FFF96H	Day count register	DAY	R/W	—	✓	—	01H
FFF97H	Month count register	MONTH	R/W	—	✓	—	01H
FFF98H	Year count register	YEAR	R/W	—	✓	—	00H
FFF99H	Watch error correction register	SUBCUD	R/W	—	✓	—	00H
FFF9AH	Alarm minute register	ALARMWM	R/W	—	✓	—	00H
FFF9BH	Alarm hour register	ALARMWH	R/W	—	✓	—	12H
FFF9CH	Alarm week register	ALARMWW	R/W	—	✓	—	00H

**Note 1.** The timer RD SFRs are undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** The value of this register is 00H if the AMPM bit (bit 3 of real-time clock control register 0 (RTCC0)) is set to 1 after reset.

**Table 3 - 12 Special Function Register (SFR) List (4/5)**

Address	Special Function Register (SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
FFF9DH	Real-time clock control register 0	RTCC0	R/W	✓	✓	—	00H
FFF9EH	Real-time clock control register 1	RTCC1	R/W	✓	✓	—	00H
FFFA0H	Clock operation mode control register	CMC	R/W	—	✓	—	00H
FFFA1H	Clock operation status control register	CSC	R/W	✓	✓	—	C0H
FFFA2H	Oscillation stabilization time counter status register	OSTC	R	✓	✓	—	00H
FFFA3H	Oscillation stabilization time select register	OSTS	R/W	—	✓	—	07H
FFFA4H	System clock control register	CKC	R/W	✓	✓	—	00H
FFFA5H	Clock output select register 0	CKS0	R/W	✓	✓	—	00H
FFFA6H	Clock output select register 1	CKS1	R/W	✓	✓	—	00H
FFFA8H	Reset control flag register	RESF	R	—	✓	—	Undefined Note 1
FFFA9H	Voltage detection register	LVIM	R/W	✓	✓	—	00H Note 1
FFFAAH	Voltage detection level register	LVIS	R/W	✓	✓	—	00H/01H/81H Note 1
FFFBABH	Watchdog timer enable register	WDTE	R/W	—	✓	—	9AH/1AH Note 2
FFFACH	CRC input register	CRCIN	R/W	—	✓	—	00H
FFFD0H	Interrupt request flag register 2L	IF2L	IF2	R/W	✓	✓	00H
FFFD1H	Interrupt request flag register 2H	IF2H		R/W	✓	✓	
FFFD4H	Interrupt mask flag register 2L	MK2L	MK2	R/W	✓	✓	FFH
FFFD5H	Interrupt mask flag register 2H	MK2H		R/W	✓	✓	
FFFD8H	Priority specification flag register 02L	PR02L	PR02	R/W	✓	✓	FFH
FFFD9H	Priority specification flag register 02H	PR02H		R/W	✓	✓	
FFFDCH	Priority specification flag register 12L	PR12L	PR12	R/W	✓	✓	FFH
FFFDDH	Priority specification flag register 12H	PR12H		R/W	✓	✓	

**Note 1.** These values vary depending on the reset source.

Reset Source Register		RESET Input	Reset by POR	Reset by Execution of Illegal Instruction	Reset by WDT	Reset by RAM parity error	Reset by illegal- memory access	Reset by LVD	
RESF		Cleared (0)		Set (1)	Held			Held	
TRAP	Held			Set (1)	Held				
WDTRF	Held			Set (1)	Held				
RPERF	Held			Set (1)					
IAWRF	Held			Set (1)					
LVIM	LVIRF			Held					
	LVISEN	Cleared (0)					Held		
	LVIOMSK	Held							
LVIS		Cleared (00H/01H/81H)							

**Note 2.** The reset value of the WDTE register is determined by the setting of the option byte.

**Table 3 - 13 Special Function Register (SFR) List (5/5)**

Address	Special Function Register (SFR) Name	Symbol		R/W	Manipulable Bit Range			After Reset	
					1-bit	8-bit	16-bit		
FFFE0H	Interrupt request flag register 0L	IF0L	IF0	R/W	✓	✓	✓	00H	
FFFE1H	Interrupt request flag register 0H	IF0H		R/W	✓	✓		00H	
FFFE2H	Interrupt request flag register 1L	IF1L	IF1	R/W	✓	✓	✓	00H	
FFFE3H	Interrupt request flag register 1H	IF1H		R/W	✓	✓		00H	
FFFE4H	Interrupt mask flag register 0	MK0L	MK0	R/W	✓	✓	✓	FFH	
FFFE5H		MK0H		R/W	✓	✓		FFH	
FFFE6H	Interrupt mask flag register 1	MK1L	MK1	R/W	✓	✓	✓	FFH	
FFFE7H		MK1H		R/W	✓	✓		FFH	
FFFE8H	Priority specification flag register 00	PR00L	PR00	R/W	✓	✓	✓	FFH	
FFFE9H		PR00H		R/W	✓	✓		FFH	
FFFEAH	Priority specification flag register 01	PR01L	PR01	R/W	✓	✓	✓	FFH	
FFFEBH		PR01H		R/W	✓	✓		FFH	
FFFECH	Priority specification flag register 10	PR10L	PR10	R/W	✓	✓	✓	FFH	
FFFEDH		PR10H		R/W	✓	✓		FFH	
FFFEHH	Priority specification flag register 11	PR11L	PR11	R/W	✓	✓	✓	FFH	
FFFEFH		PR11H		R/W	✓	✓		FFH	
FFFF0H	Multiply and accumulation register (L)	MACRL		R/W	—	—	✓	0000H	
FFFF1H									
FFFF2H	Multiply and accumulation register (H)	MACRH		R/W	—	—	✓	0000H	
FFFF3H									
FFFFEH	Processor mode control register	PMC		R/W	✓	✓	—	00H	

**Remark** For extended SFRs (2nd SFRs), see **Tables 3 - 14 to 3 - 22 Extended Special Function Register (2nd SFR) List**.

### 3.2.5 Extended special function registers (2nd SFRs: 2nd Special Function Registers)

Unlike a general-purpose register, each extended SFR (2<sup>nd</sup> SFR) has a special function.

Extended SFRs are allocated to the F0000H to F07FFH area. SFRs other than those in the SFR area (FFF00H to FFFFFH) are allocated to this area. An instruction that accesses the extended SFR area, however, is 1 byte longer than an instruction that accesses the SFR area.

Extended SFRs can be manipulated like general-purpose registers, using operation, transfer, and bit manipulation instructions. The manipulable bit units, 1, 8, and 16, depend on the SFR type.

Each manipulation bit unit can be specified as follows.

- 1-bit manipulation

Describe as follows for the 1-bit manipulation instruction operand (!addr16.bit)

When the bit name is defined: <Bit name>

When the bit name is not defined: <Register name>, <Bit number> or <Address>, <Bit number>

- 8-bit manipulation

Describe the symbol defined by the assembler for the 8-bit manipulation instruction operand (!addr16). This manipulation can also be specified with an address.

- 16-bit manipulation

Describe the symbol defined by the assembler for the 16-bit manipulation instruction operand (!addr16). When specifying an address, describe an even address.

Tables 3 - 14 to 3 - 22 give lists of the extended SFRs. The meanings of items in the table are as follows.

- Symbol

This item indicates the address of an extended SFR. It is a reserved word in the assembler, and is defined as an sfr variable using the #pragma sfr directive in the compiler. When using the assembler, debugger, and simulator, symbols can be written as an instruction operand.

- R/W

This item indicates whether the corresponding extended SFR can be read or written.

R/W:Read/write enable

R:Read only

W:Write only

- Manipulable bit units

“√” indicates the manipulable bit unit (1, 8, or 16). “—” indicates a bit unit for which manipulation is not possible.

- After reset

This item indicates each register status upon reset signal generation.

**Caution** Do not access addresses to which extended SFRs are not assigned.

**Remark** For SFRs in the SFR area, see [3.2.4 Special function registers \(SFRs\)](#).

**Table 3 - 14 Extended Special Function Register (2nd SFR) List (1/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F0010H	A/D converter mode register 2	ADM2	R/W	✓	✓	—	00H
F0011H	Conversion result comparison upper limit setting register	ADUL	R/W	—	✓	—	FFH
F0012H	Conversion result comparison lower limit setting register	ADLL	R/W	—	✓	—	00H
F0013H	A/D test register	ADTES	R/W	—	✓	—	00H
F0030H	Pull-up resistor option register 0	PU0	R/W	✓	✓	—	00H
F0031H	Pull-up resistor option register 1	PU1	R/W	✓	✓	—	00H
F0033H	Pull-up resistor option register 3	PU3	R/W	✓	✓	—	00H
F0034H	Pull-up resistor option register 4	PU4	R/W	✓	✓	—	01H
F0035H	Pull-up resistor option register 5	PU5	R/W	✓	✓	—	00H
F0036H	Pull-up resistor option register 6	PU6	R/W	✓	✓	—	00H
F0037H	Pull-up resistor option register 7	PU7	R/W	✓	✓	—	00H
F0038H	Pull-up resistor option register 8	PU8	R/W	✓	✓	—	00H
F003AH	Pull-up resistor option register 10	PU10	R/W	✓	✓	—	00H
F003BH	Pull-up resistor option register 11	PU11	R/W	✓	✓	—	00H
F003CH	Pull-up resistor option register 12	PU12	R/W	✓	✓	—	00H
F003EH	Pull-up resistor option register 14	PU14	R/W	✓	✓	—	00H
F0040H	Port input mode register 0	PIM0	R/W	✓	✓	—	00H
F0041H	Port input mode register 1	PIM1	R/W	✓	✓	—	00H
F0043H	Port input mode register 3	PIM3	R/W	✓	✓	—	00H
F0044H	Port input mode register 4	PIM4	R/W	✓	✓	—	00H
F0045H	Port input mode register 5	PIM5	R/W	✓	✓	—	00H
F0048H	Port input mode register 8	PIM8	R/W	✓	✓	—	00H
F004EH	Port input mode register 14	PIM14	R/W	✓	✓	—	00H
F0050H	Port output mode register 0	POM0	R/W	✓	✓	—	00H
F0051H	Port output mode register 1	POM1	R/W	✓	✓	—	00H
F0053H	Port output mode register 3	POM3	R/W	✓	✓	—	00H
F0054H	Port output mode register 4	POM4	R/W	✓	✓	—	00H
F0055H	Port output mode register 5	POM5	R/W	✓	✓	—	00H
F0057H	Port output mode register 7	POM7	R/W	✓	✓	—	00H
F0058H	Port output mode register 8	POM8	R/W	✓	✓	—	00H
F005EH	Port output mode register 14	POM14	R/W	✓	✓	—	00H
F0060H	Port mode control register 0	PMC0	R/W	✓	✓	—	FFH
F0061H	Port mode control register 1	PMC1	R/W	✓	✓	—	00H
F006AH	Port mode control register 10	PMC10	R/W	✓	✓	—	FFH
F006CH	Port mode control register 12	PMC12	R/W	✓	✓	—	FFH
F006EH	Port mode control register 14	PMC14	R/W	✓	✓	—	FFH
F0070H	Noise filter enable register 0	NFEN0	R/W	✓	✓	—	00H
F0071H	Noise filter enable register 1	NFEN1	R/W	✓	✓	—	00H
F0072H	Noise filter enable register 2	NFEN2	R/W	✓	✓	—	00H
F0073H	Input switch control register	ISC	R/W	✓	✓	—	00H

**Table 3 - 15 Extended Special Function Register (2nd SFR) List (2/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset	
				1-bit	8-bit	16-bit		
F0074H	Timer input select register 0	TIS0	R/W	—	✓	—	00H	
F0076H	A/D port configuration register	ADPC	R/W	—	✓	—	00H	
F0077H	Peripheral I/O redirection register 0	PIOR0	R/W	—	✓	—	00H	
F0078H	Invalid memory access detection control register	IAWCTL	R/W	—	✓	—	00H	
F0079H	Peripheral I/O redirection register 1	PIOR1	R/W	—	✓	—	00H	
F007AH	Peripheral enable register 1	PER1	R/W	✓	✓	—	00H	
F007BH	Port mode select register	PMS	R/W	✓	✓	—	00H	
F007DH	Global digital input disable register	GIDDIS	R/W	✓	✓	—	00H	
F0090H	Data flash control register	DFLCTL	R/W	✓	✓	—	00H	
F00A0H	High-speed on-chip oscillator trimming register	HIOTRM	R/W	—	✓	—	Undefined Note 1	
F00A8H	High-speed on-chip oscillator frequency select register	HOCODIV	R/W	—	✓	—	Undefined Note 2	
F00F0H	Peripheral enable register 0	PER0	R/W	✓	✓	—	00H	
F00F3H	Subsystem clock supply mode control register	OSMC	R/W	—	✓	—	00H	
F00F5H	RAM parity error control register	RPECTL	R/W	✓	✓	—	00H	
F00FEH	BCD correction result register	BCDADJ	R	—	✓	—	Undefined	
F0100H	Serial status register 00	SSR00L	SSR00	R	—	✓	✓	0000H
F0101H		—			—	—		
F0102H	Serial status register 01	SSR01L	SSR01	R	—	✓	✓	0000H
F0103H		—			—	—		
F0104H	Serial status register 02	SSR02L	SSR02	R	—	✓	✓	0000H
F0105H		—			—	—		
F0106H	Serial status register 03	SSR03L	SSR03	R	—	✓	✓	0000H
F0107H		—			—	—		
F0108H	Serial flag clear trigger register 00	SIR00L	SIR00	R/W	—	✓	✓	0000H
F0109H		—			—	—		
F010AH	Serial flag clear trigger register 01	SIR01L	SIR01	R/W	—	✓	✓	0000H
F010BH		—			—	—		
F010CH	Serial flag clear trigger register 02	SIR02L	SIR02	R/W	—	✓	✓	0000H
F010DH		—			—	—		
F010EH	Serial flag clear trigger register 03	SIR03L	SIR03	R/W	—	✓	✓	0000H
F010FH		—			—	—		

**Note 1.** The value after a reset is adjusted at the time of shipment.

**Note 2.** The value after a reset is a value set in FRQSEL2 to FRQSEL0 of the option byte (000C2H).

**Table 3 - 16 Extended Special Function Register (2nd SFR) List (3/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset	
				1-bit	8-bit	16-bit		
F0110H	Serial mode register 00	SMR00	R/W	—	—	✓	0020H	
F0111H				—	—	✓		
F0112H	Serial mode register 01	SMR01	R/W	—	—	✓	0020H	
F0113H				—	—	✓		
F0114H	Serial mode register 02	SMR02	R/W	—	—	✓	0020H	
F0115H				—	—	✓		
F0116H	Serial mode register 03	SMR03	R/W	—	—	✓	0020H	
F0117H				—	—	✓		
F0118H	Serial communication operation setting register 00	SCR00	R/W	—	—	✓	0087H	
F0119H				—	—	✓		
F011AH	Serial communication operation setting register 01	SCR01	R/W	—	—	✓	0087H	
F011BH				—	—	✓		
F011CH	Serial communication operation setting register 02	SCR02	R/W	—	—	✓	0087H	
F011DH				—	—	✓		
F011EH	Serial communication operation setting register 03	SCR03	R/W	—	—	✓	0087H	
F011FH				—	—	✓		
F0120H	Serial channel enable status register 0	SE0L	SE0	R	✓	✓	✓	
F0121H		—			—	—		
F0122H	Serial channel start register 0	SS0L	SS0	R/W	✓	✓	✓	
F0123H		—			—	—		
F0124H	Serial channel stop register 0	ST0L	ST0	R/W	✓	✓	✓	
F0125H		—			—	—		
F0126H	Serial clock select register 0	SPS0L	SPS0	R/W	—	✓	✓	
F0127H		—			—	—		
F0128H	Serial output register 0	SO0		R/W	—	—	✓	
F0129H					—	—		
F012AH	Serial output enable register 0	SOE0L	SOE0	R/W	✓	✓	✓	
F012BH		—			—	—		
F0134H	Serial output level register 0	SOL0L	SOL0	R/W	—	✓	✓	
F0135H		—			—	—		
F0138H	Serial standby control register 0	SSC0L	SSC0	R/W	—	✓	✓	
F0139H		—			—	—		
F0140H	Serial status register 10	SSR10L	SSR10	R	—	✓	✓	
F0141H		—			—	—		
F0142H	Serial status register 11	SSR11L	SSR11	R	—	✓	✓	
F0143H		—			—	—		
F0144H	Serial status register 12	SSR12L	SSR12	R	—	✓	✓	
F0145H		—			—	—		
F0146H	Serial status register 13	SSR13L	SSR13	R	—	✓	✓	
F0147H		—			—	—		

**Table 3 - 17 Extended Special Function Register (2nd SFR) List (4/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset		
				1-bit	8-bit	16-bit			
F0148H	Serial flag clear trigger register 10	SIR10L	SIR10	R/W	—	✓	✓	0000H	
F0149H		—			—	—			
F014AH	Serial flag clear trigger register 11	SIR11L	SIR11	R/W	—	✓	✓	0000H	
F014BH		—			—	—			
F014CH	Serial flag clear trigger register 12	SIR12L	SIR12	R/W	—	✓	✓	0000H	
F014DH		—			—	—			
F014EH	Serial flag clear trigger register 13	SIR13L	SIR13	R/W	—	✓	✓	0000H	
F014FH		—			—	—			
F0150H	Serial mode register 10	SMR10		R/W	—	—	✓	0020H	
F0151H					—	—			
F0152H	Serial mode register 11	SMR11		R/W	—	—	✓	0020H	
F0153H					—	—			
F0154H	Serial mode register 12	SMR12		R/W	—	—	✓	0020H	
F0155H					—	—			
F0156H	Serial mode register 13	SMR13		R/W	—	—	✓	0020H	
F0157H					—	—			
F0158H	Serial communication operation setting register 10	SCR10		R/W	—	—	✓	0087H	
F0159H					—	—			
F015AH	Serial communication operation setting register 11	SCR11		R/W	—	—	✓	0087H	
F015BH					—	—			
F015CH	Serial communication operation setting register 12	SCR12		R/W	—	—	✓	0087H	
F015DH					—	—			
F015EH	Serial communication operation setting register 13	SCR13		R/W	—	—	✓	0087H	
F015FH					—	—			
F0160H	Serial channel enable status register 1	SE1L	SE1	R	✓	✓	✓	0000H	
F0161H		—			—	—			
F0162H	Serial channel start register 1	SS1L	SS1	R/W	✓	✓	✓	0000H	
F0163H		—			—	—			
F0164H	Serial channel stop register 1	ST1L	ST1	R/W	✓	✓	✓	0000H	
F0165H		—			—	—			
F0166H	Serial clock select register 1	SPS1L	SPS1	R/W	—	✓	✓	0000H	
F0167H		—			—	—			
F0168H	Serial output register 1	SO1		R/W	—	—	✓	0F0FH	
F0169H					—	—			
F016AH	Serial output enable register 1	SOE1L	SOE1	R/W	✓	✓	✓	0000H	
F016BH		—			—	—			
F0174H	Serial output level register 1	SOL1L	SOL1	R/W	—	✓	✓	0000H	
F0175H		—			—	—			
F0178H	Serial standby control register 1	SSC1L	SSC1	R/W	—	✓	✓	0000H	
F0179H		—			—	—			

**Table 3 - 18 Extended Special Function Register (2nd SFR) List (5/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F0180H	Timer counter register 00	TCR00	R	—	—	✓	FFFFH
F0181H							
F0182H	Timer counter register 01	TCR01	R	—	—	✓	FFFFH
F0183H							
F0184H	Timer counter register 02	TCR02	R	—	—	✓	FFFFH
F0185H							
F0186H	Timer counter register 03	TCR03	R	—	—	✓	FFFFH
F0187H							
F0190H	Timer mode register 00	TMR00	R/W	—	—	✓	0000H
F0191H							
F0192H	Timer mode register 01	TMR01	R/W	—	—	✓	0000H
F0193H							
F0194H	Timer mode register 02	TMR02	R/W	—	—	✓	0000H
F0195H							
F0196H	Timer mode register 03	TMR03	R/W	—	—	✓	0000H
F0197H							
F01A0H	Timer status register 00	TSR00L	TSR00	R	—	✓	0000H
F01A1H		—			—	—	
F01A2H	Timer status register 01	TSR01L	TSR01	R	—	✓	0000H
F01A3H		—			—	—	
F01A4H	Timer status register 02	TSR02L	TSR02	R	—	✓	0000H
F01A5H		—			—	—	
F01A6H	Timer status register 03	TSR03L	TSR03	R	—	✓	0000H
F01A7H		—			—	—	
F01B0H	Timer channel enable status register 0	TE0L	TE0	R	✓	✓	0000H
F01B1H		—			—	—	
F01B2H	Timer channel start register 0	TS0L	TS0	R/W	✓	✓	0000H
F01B3H		—			—	—	
F01B4H	Timer channel stop register 0	TT0L	TT0	R/W	✓	✓	0000H
F01B5H		—			—	—	
F01B6H	Timer clock select register 0	TPS0	R/W	R/W	—	—	0000H
F01B7H							
F01B8H	Timer output register 0	TO0L	TO0	R/W	—	✓	0000H
F01B9H		—			—	—	
F01BAH	Timer output enable register 0	TOE0L	TOE0	R/W	✓	✓	0000H
F01BBH		—			—	—	
F01BCH	Timer output level register 0	TOL0L	TOL0	R/W	—	✓	0000H
F01BDH		—			—	—	
F01BEH	Timer output mode register 0	TOM0L	TOM0	R/W	—	✓	0000H
F01BFH		—			—	—	

**Table 3 - 19 Extended Special Function Register (2nd SFR) List (6/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F01C0H	Timer counter register 10	TCR10	R	—	—	✓	FFFFH
F01C1H							
F01C2H	Timer counter register 11	TCR11	R	—	—	✓	FFFFH
F01C3H							
F01C4H	Timer counter register 12	TCR12	R	—	—	✓	FFFFH
F01C5H							
F01C6H	Timer counter register 13	TCR13	R	—	—	✓	FFFFH
F01C7H							
F01D0H	Timer mode register 10	TMR10	R/W	—	—	✓	0000H
F01D1H							
F01D2H	Timer mode register 11	TMR11	R/W	—	—	✓	0000H
F01D3H							
F01D4H	Timer mode register 12	TMR12	R/W	—	—	✓	0000H
F01D5H							
F01D6H	Timer mode register 13	TMR13	R/W	—	—	✓	0000H
F01D7H							
F01E0H	Timer status register 10	TSR10L	R/W	—	✓	✓	0000H
F01E1H		—		—	—		
F01E2H	Timer status register 11	TSR11L	R/W	—	✓	✓	0000H
F01E3H		—		—	—		
F01E4H	Timer status register 12	TSR12L	R/W	—	✓	✓	0000H
F01E5H		—		—	—		
F01E6H	Timer status register 13	TSR13L	R/W	—	✓	✓	0000H
F01E7H		—		—	—		
F01F0H	Timer channel enable status register 1	TE1L	R/W	—	✓	✓	0000H
F01F1H		—		—	—		
F01F2H	Timer channel start register 1	TS1L	R/W	—	✓	✓	0000H
F01F3H		—		—	—		
F01F4H	Timer channel stop register 1	TT1L	R/W	—	✓	✓	0000H
F01F5H		—		—	—		
F01F6H	Timer clock select register 1	TPS1	R/W	—	—	✓	0000H
F01F7H							
F01F8H	Timer output register 1	TO1L	R/W	—	✓	✓	0000H
F01F9H		—		—	—		
F01FAH	Timer output enable register 1	TOE1L	R/W	—	✓	✓	0000H
F01FBH		—		—	—		
F01FCH	Timer output level register 1	TOL1L	R/W	—	✓	✓	0000H
F01FDH		—		—	—		
F01FEH	Timer output mode register 1	TOM1L	R/W	—	✓	✓	0000H
F01FFH		—		—	—		

**Table 3 - 20 Extended Special Function Register (2nd SFR) List (7/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F0230H	IICA control register 00	IICCTL00	R/W	✓	✓	—	00H
F0231H	IICA control register 01	IICCTL01	R/W	✓	✓	—	00H
F0232H	IICA low-level width setting register 0	IICWL0	R/W	—	✓	—	FFH
F0233H	IICA high-level width setting register 0	IICWH0	R/W	—	✓	—	FFH
F0234H	Slave address register 0	SVA0	R/W	—	✓	—	00H
F0238H	IICA control register 10	IICCTL10	R/W	✓	✓	—	00H
F0239H	IICA control register 11	IICCTL11	R/W	✓	✓	—	00H
F023AH	IICA low-level width setting register 1	IICWL1	R/W	—	✓	—	FFH
F023BH	IICA high-level width setting register 1	IICWH1	R/W	—	✓	—	FFH
F023CH	Slave address register 1	SVA1	R/W	—	✓	—	00H
F0240H	Timer RJ control register 0	TRJCR0	R/W	—	✓	—	00H
F0241H	Timer RJ I/O control register 0	TRJIOC0	R/W	✓	✓	—	00H
F0242H	Timer RJ mode register 0	TRJMR0	R/W	✓	✓	—	00H
F0243H	Timer RJ event pin select register 0	TRJISR0	R/W	✓	✓	—	00H
F0250H	Timer RG mode register	TRGMR	R/W	✓	✓	—	00H
F0251H	Timer RG count control register	TRGCNTC	R/W	✓	✓	—	00H
F0252H	Timer RG control register	TRGCR	R/W	✓	✓	—	00H
F0253H	Timer RG interrupt enable register	TRGIER	R/W	✓	✓	—	00H
F0254H	Timer RG status register	TRGSR	R/W	✓	✓	—	00H
F0255H	Timer RG I/O control register	TRGIOR	R/W	✓	✓	—	00H
F0256H	Timer RG counter	TRG	RW	—	—	✓	0000H
F0257H				—	—	✓	0000H
F0258H	Timer RG general register A	TRGGRA	RW	—	—	✓	FFFFH
F0259H				—	—	✓	FFFFH
F025AH	Timer RG general register B	TRGGRB	RW	—	—	✓	FFFFH
F025BH				—	—	✓	FFFFH
F0260H	Timer RD ELC register	TRDELC	R/W	✓	✓	—	00H Note
F0263H	Timer RD start register	TRDSTR	R/W	—	✓	—	0CH Note
F0264H	Timer RD mode register	TRDMR	R/W	✓	✓	—	00H Note
F0265H	Timer RD PWM function select register	TRDPMR	R/W	✓	✓	—	00H Note
F0266H	Timer RD function control register	TRDFCR	R/W	✓	✓	—	80H Note
F0267H	Timer RD output master enable register 1	TRDOER1	R/W	✓	✓	—	FFH Note
F0268H	Timer RD output master enable register 2	TRDOER2	R/W	✓	✓	—	00H Note
F0269H	Timer RD output control register	TRDOCR	R/W	✓	✓	—	00H Note

**Note** The timer RD SFRs are undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 3 - 21 Extended Special Function Register (2nd SFR) List (8/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F026AH	Timer RD digital filter function select register 0	TRDDF0	R/W	✓	✓	—	00H
F026BH	Timer RD digital filter function select register 1	TRDDF1	R/W	✓	✓	—	00H
F0270H	Timer RD control register 0	TRDCR0	R/W	✓	✓	—	00H Note
F0271H	Timer RD I/O control register A0	TRDIORA0	R/W	✓	✓	—	00H Note
F0272H	Timer RD I/O control register C0	TRDIORC0	R/W	✓	✓	—	88H Note
F0273H	Timer RD status register 0	TRDSR0	R/W	✓	✓	—	00H Note
F0274H	Timer RD interrupt enable register 0	TRDIER0	R/W	✓	✓	—	00H Note
F0275H	Timer RD PWM function output level control register 0	TRDPOCR0	R/W	✓	✓	—	00H Note
F0276H	Timer RD counter 0	TRD0	R/W	—	—	✓	0000H Note
F0277H				—	—	✓	
F0278H	Timer RD general register A0	TRDGRA0	R/W	—	—	✓	FFFFH Note
F0279H				—	—	✓	
F027AH	Timer RD general register B0	TRDGRB0	R/W	—	—	✓	FFFFH Note
F027BH				—	—	✓	
F0280H	Timer RD control register 1	TRDCR1	R/W	✓	✓	—	00H Note
F0281H	Timer RD I/O control register A1	TRDIORA1	R/W	✓	✓	—	00H Note
F0282H	Timer RD I/O control register C1	TRDIORC1	R/W	✓	✓	—	88H Note
F0283H	Timer RD status register 1	TRDSR1	R/W	✓	✓	—	00H Note
F0284H	Timer RD interrupt enable register 1	TRDIER1	R/W	✓	✓	—	00H Note
F0285H	Timer RD PWM function output level control register 1	TRDPOCR1	R/W	✓	✓	—	00H Note
F0286H	Timer RD counter 1	TRD1	R/W	—	—	✓	0000H Note
F0287H				—	—	✓	
F0288H	Timer RD general register A1	TRDGRA1	R/W	—	—	✓	FFFFH Note
F0289H				—	—	✓	
F028AH	Timer RD general register B1	TRDGRB1	R/W	—	—	✓	FFFFH Note
F028BH				—	—	✓	
F02E0H	DTC base address register	DTCBAR	R/W	✓	✓	—	FDH
F02E8H	DTC activation enable register 0	DTCEN0	R/W	✓	✓	—	00H
F02E9H	DTC activation enable register 1	DTCEN1	R/W	✓	✓	—	00H
F02EAH	DTC activation enable register 2	DTCEN2	R/W	✓	✓	—	00H
F02EBH	DTC activation enable register 3	DTCEN3	R/W	✓	✓	—	00H
F02ECH	DTC activation enable register 4	DTCEN4	R/W	✓	✓	—	00H
F02F0H	Flash memory CRC control register	CRC0CTL	R/W	✓	✓	—	00H
F02F2H	Flash memory CRC operation result register	PGCRCL	R/W	—	—	✓	0000H
F02FAH	CRC data register	CRCD	R/W	—	—	✓	0000H

**Note** The timer RD SFRs are undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 3 - 22 Extended Special Function Register (2nd SFR) List (9/9)**

Address	Extended Special Function Register (2nd SFR) Name	Symbol	R/W	Manipulable Bit Range			After Reset
				1-bit	8-bit	16-bit	
F0300H	Event output destination select register 00	ELSELR00	R/W	—	✓	—	00H
F0301H	Event output destination select register 01	ELSELR01	R/W	—	✓	—	00H
F0302H	Event output destination select register 02	ELSELR02	R/W	—	✓	—	00H
F0303H	Event output destination select register 03	ELSELR03	R/W	—	✓	—	00H
F0304H	Event output destination select register 04	ELSELR04	R/W	—	✓	—	00H
F0305H	Event output destination select register 05	ELSELR05	R/W	—	✓	—	00H
F0306H	Event output destination select register 06	ELSELR06	R/W	—	✓	—	00H
F0307H	Event output destination select register 07	ELSELR07	R/W	—	✓	—	00H
F0308H	Event output destination select register 08	ELSELR08	R/W	—	✓	—	00H
F0309H	Event output destination select register 09	ELSELR09	R/W	—	✓	—	00H
F030AH	Event output destination select register 10	ELSELR10	R/W	—	✓	—	00H
F030BH	Event output destination select register 11	ELSELR11	R/W	—	✓	—	00H
F030CH	Event output destination select register 12	ELSELR12	R/W	—	✓	—	00H
F030DH	Event output destination select register 13	ELSELR13	R/W	—	✓	—	00H
F030EH	Event output destination select register 14	ELSELR14	R/W	—	✓	—	00H
F030FH	Event output destination select register 15	ELSELR15	R/W	—	✓	—	00H
F0310H	Event output destination select register 16	ELSELR16	R/W	—	✓	—	00H
F0311H	Event output destination select register 17	ELSELR17	R/W	—	✓	—	00H
F0312H	Event output destination select register 18	ELSELR18	R/W	—	✓	—	00H
F0313H	Event output destination select register 19	ELSELR19	R/W	—	✓	—	00H
F0314H	Event output destination select register 20	ELSELR20	R/W	—	✓	—	00H
F0315H	Event output destination select register 21	ELSELR21	R/W	—	✓	—	00H
F0316H	Event output destination select register 22	ELSELR22	R/W	—	✓	—	00H
F0317H	Event output destination select register 23	ELSELR23	R/W	—	✓	—	00H
F0318H	Event output destination select register 24	ELSELR24	R/W	—	✓	—	00H
F0319H	Event output destination select register 25	ELSELR25	R/W	—	✓	—	00H
F0340H	Comparator mode setting register	COMPMDR	R/W	✓	✓	—	00H
F0341H	Comparator filter control register	COMPFIR	R/W	✓	✓	—	00H
F0342H	Comparator output control register	COMPOCR	R/W	✓	✓	—	00H
F0500H	Timer RJ counter register 0	TRJ0	R/W	—	—	✓	FFFFH
F0501H				—	—	✓	

**Remark** For SFRs in the SFR area, see **Tables 3 - 9 to 3 - 13 Special Function Register (SFR) List**.

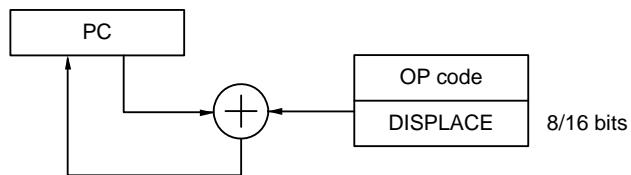
### 3.3 Instruction Address Addressing

#### 3.3.1 Relative addressing

[Function]

Relative addressing stores in the program counter (PC) the result of adding a displacement value included in the instruction word (signed complement data: -128 to +127 or -32768 to +32767) to the program counter (PC)'s value (the start address of the next instruction), and specifies the program address to be used as the branch destination. Relative addressing is applied only to branch instructions.

**Figure 3 - 19 Outline of Relative Addressing**



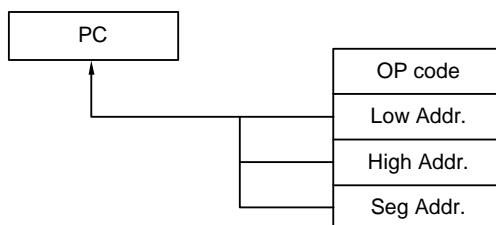
#### 3.3.2 Immediate addressing

[Function]

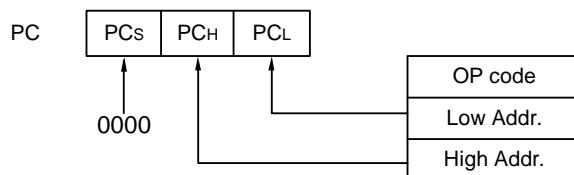
Immediate addressing stores immediate data of the instruction word in the program counter, and specifies the program address to be used as the branch destination.

For immediate addressing, CALL !!addr20 or BR !!addr20 is used to specify 20-bit addresses and CALL !addr16 or BR !addr16 is used to specify 16-bit addresses. 0000 is set to the higher 4 bits when specifying 16-bit addresses.

**Figure 3 - 20 Example of CALL !!addr20/BR !!addr20**



**Figure 3 - 21 Example of CALL !addr16/BR !addr16**



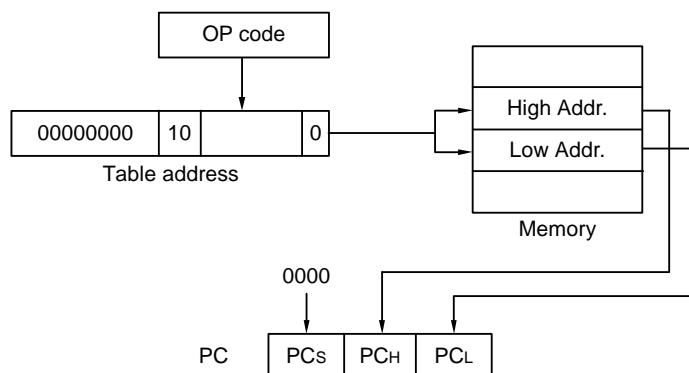
### 3.3.3 Table indirect addressing

[Function]

Table indirect addressing specifies a table address in the CALLT table area (0080H to 00BFH) with the 5-bit immediate data in the instruction word, stores the contents at that table address and the next address in the program counter (PC) as 16-bit data, and specifies the program address. Table indirect addressing is applied only for CALLT instructions.

In the RL78 microcontrollers, branching is enabled only to the 64 KB space from 00000H to 0FFFFH.

Figure 3 - 22 Outline of Table Indirect Addressing

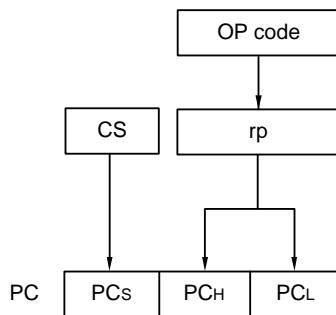


### 3.3.4 Register direct addressing

[Function]

Register direct addressing stores in the program counter (PC) the contents of a general-purpose register pair (AX/BC/DE HL) and CS register of the current register bank specified with the instruction word as 20-bit data, and specifies the program address. Register direct addressing can be applied only to the CALL AX, BC, DE, HL, and BR AX instructions.

Figure 3 - 23 Outline of Register Direct Addressing



## 3.4 Addressing for Processing Data Addresses

### 3.4.1 Implied addressing

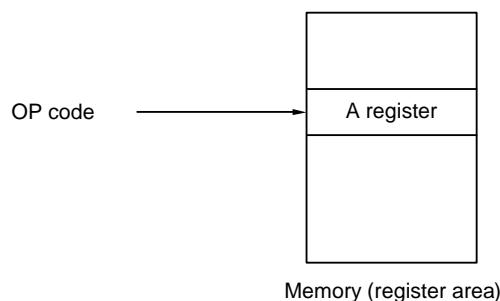
[Function]

Instructions for accessing registers (such as accumulators) that have special functions are directly specified with the instruction word, without using any register specification field in the instruction word.

[Operand format]

Implied addressing can be applied only to MULU X.

**Figure 3 - 24 Outline of Implied Addressing**



### 3.4.2 Register addressing

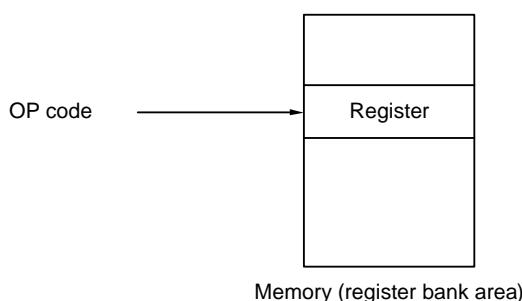
[Function]

Register addressing accesses a general-purpose register as an operand. The instruction word of 3-bit long is used to select an 8-bit register and the instruction word of 2-bit long is used to select a 16-bit register.

[Operand format]

Identifier	Description
r	X, A, C, B, E, D, L, H
rp	AX, BC, DE, HL

**Figure 3 - 25 Outline of Register Addressing**



### 3.4.3 Direct addressing

[Function]

Direct addressing uses immediate data in the instruction word as an operand address to directly specify the target address.

[Operand format]

Identifier	Description
!addr16	Label or 16-bit immediate data (only the space from F0000H to FFFFFH is specifiable)
ES:!addr16	Label or 16-bit immediate data (higher 4-bit addresses are specified by the ES register)

Figure 3 - 26 Example of !addr16

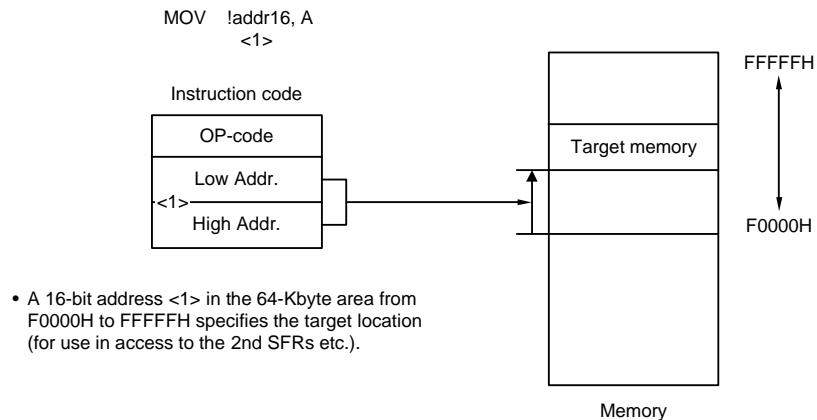
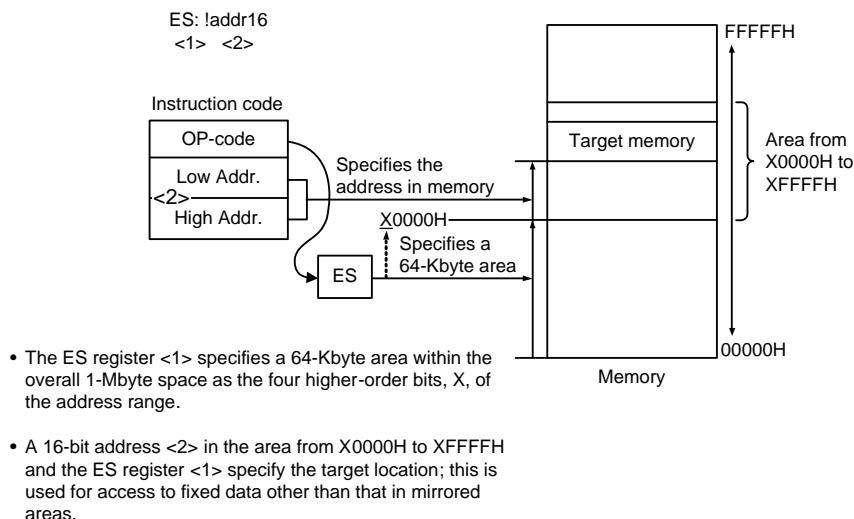


Figure 3 - 27 Example of ES:!addr16



### 3.4.4 Short direct addressing

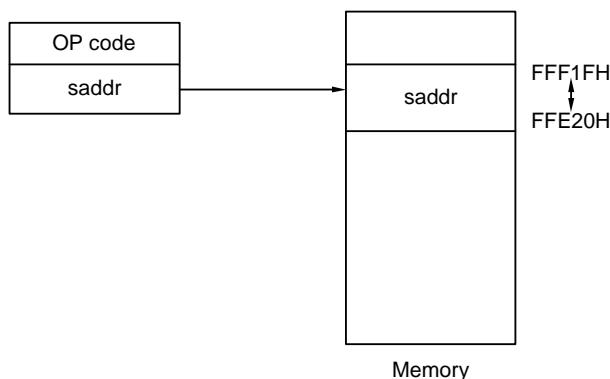
[Function]

Short direct addressing directly specifies the target addresses using 8-bit data in the instruction word. This type of addressing is applied only to the space from FFE20H to FFF1FH.

[Operand format]

Identifier	Description
SADDR	Label, FFE20H to FFF1FH immediate data, or 0FE20H to 0FF1FH immediate data (only the space from FFE20H to FFF1FH is specifiable)
SADDRP	Label, FFE20H to FFF1FH immediate data, or 0FE20H to 0FF1FH immediate data (even address only) (only the space from FFE20H to FFF1FH is specifiable)

Figure 3 - 28 Outline of Short Direct Addressing



**Remark** SADDR and SADDRP are used to describe the values of addresses FE20H to FF1FH with 16-bit immediate data (higher 4 bits of actual address are omitted), and the values of addresses FFE20H to FFF1FH with 20-bit immediate data.  
Regardless of whether SADDR or SADDRP is used, addresses within the space from FFE20H to FFF1FH are specified for the memory.

### 3.4.5 SFR addressing

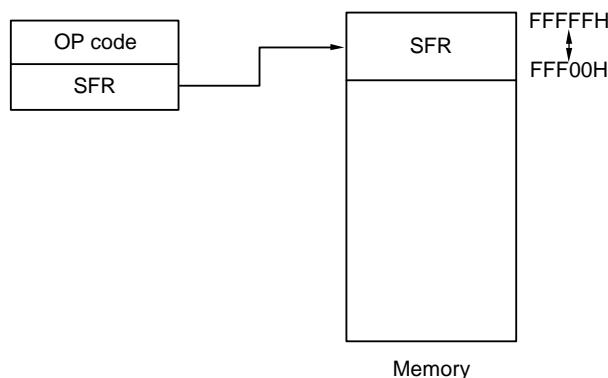
[Function]

SFR addressing directly specifies the target SFR addresses using 8-bit data in the instruction word. This type of addressing is applied only to the space from FFF00H to FFFFFH.

[Operand format]

Identifier	Description
SFR	SFR name
SFRP	16-bit-manipulatable SFR name (even address)

Figure 3 - 29 Outline of SFR Addressing



### 3.4.6 Register indirect addressing

[Function]

Register indirect addressing directly specifies the target addresses using the contents of the register pair specified with the instruction word as an operand address.

[Operand format]

Identifier	Description
—	[DE], [HL] (only the space from F0000H to FFFFFH is specifiable)
—	ES:[DE], ES:[HL] (higher 4-bit addresses are specified by the ES register)

Figure 3 - 30 Example of [DE], [HL]

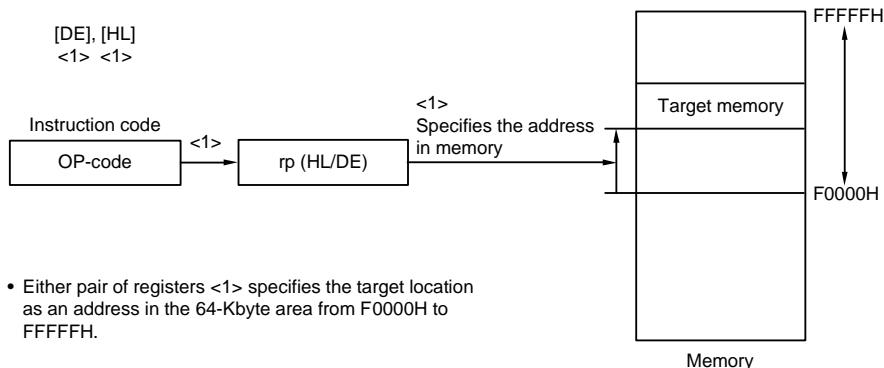
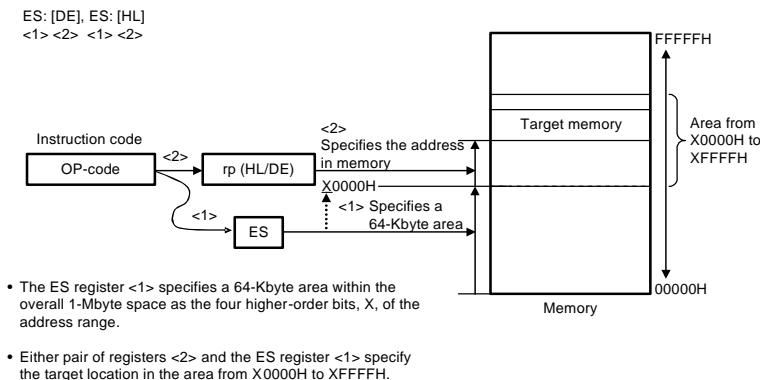


Figure 3 - 31 Example of ES:[DE], ES:[HL]



### 3.4.7 Based addressing

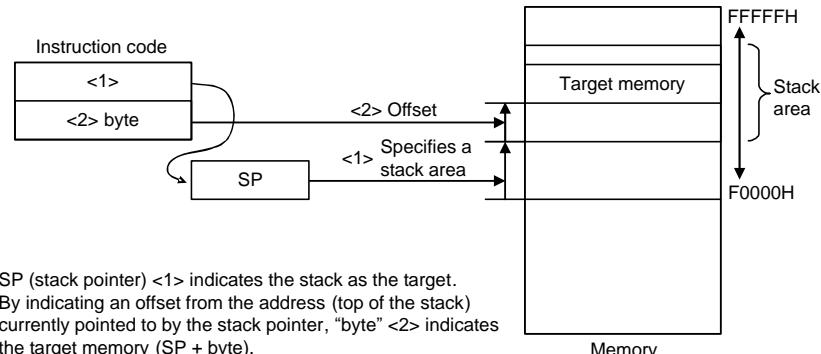
#### [Function]

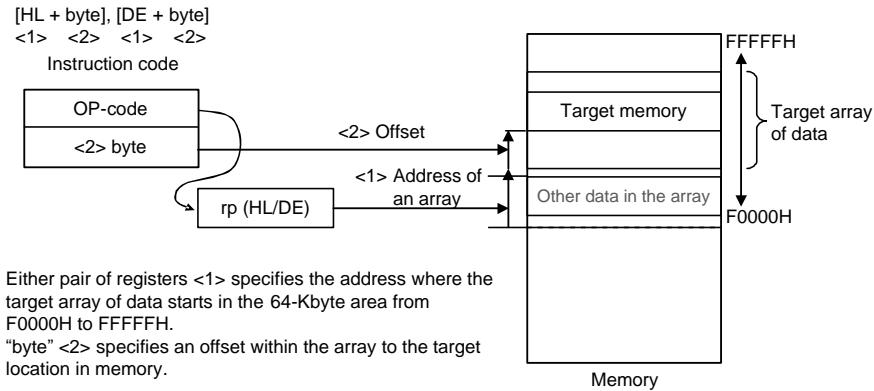
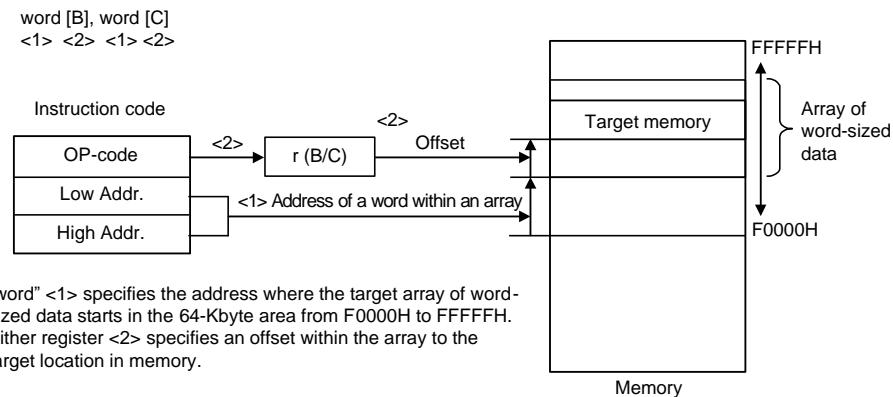
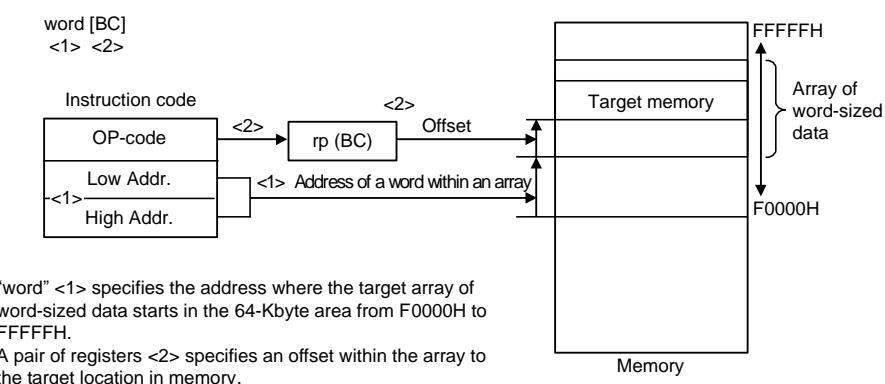
Based addressing uses the contents of a register pair specified with the instruction word or 16-bit immediate data as a base address, and 8-bit immediate data or 16-bit immediate data as offset data. The sum of these values is used to specify the target address.

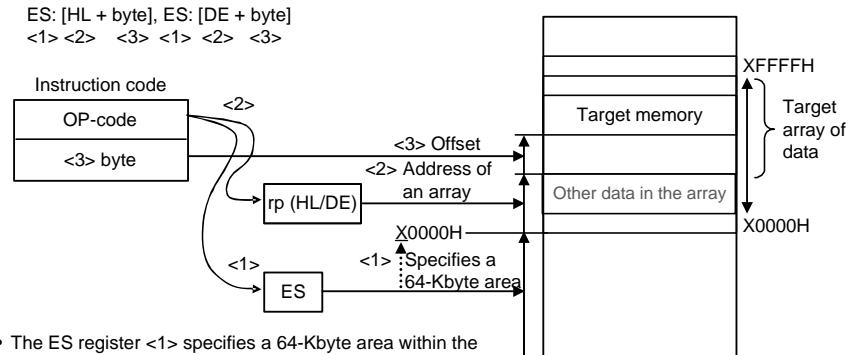
#### [Operand format]

Identifier	Description
—	[HL + byte], [DE + byte], [SP + byte] (only the space from F0000H to FFFFFH is specifiable)
—	word[B], word[C] (only the space from F0000H to FFFFFH is specifiable)
—	word[BC] (only the space from F0000H to FFFFFH is specifiable)
—	ES:[HL + byte], ES:[DE + byte] (higher 4-bit addresses are specified by the ES register)
—	ES:word[B], ES:word[C] (higher 4-bit addresses are specified by the ES register)
—	ES:word[BC] (higher 4-bit addresses are specified by the ES register)

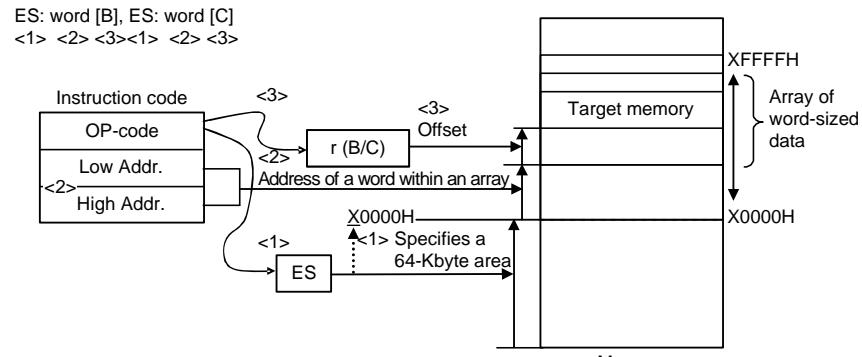
Figure 3 - 32 Example of [SP+byte]



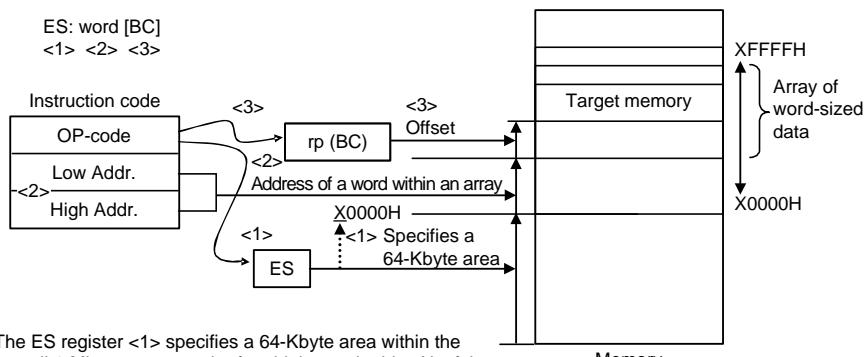
**Figure 3 - 33 Example of [HL + byte], [DE + byte]]****Figure 3 - 34 Example of word [B], word [C]****Figure 3 - 35 Example of word [BC]**

**Figure 3 - 36 Example of [HL + byte], [DE + byte]**

- The ES register <1> specifies a 64-Kbyte area within the overall 1-Mbyte space as the four higher-order bits, X, of the address range.
- Either pair of registers <2> specifies the address where the target array of data starts in the 64-Kbyte area specified in the ES register <1>.
- "byte" <3> specifies an offset within the array to the target location in memory.

**Figure 3 - 37 Example of word [B], word [C]**

- The ES register <1> specifies a 64-Kbyte area within the overall 1-Mbyte space as the four higher-order bits, X, of the address range.
- "word" <2> specifies the address where the target array of word-sized data starts in the 64-Kbyte area specified in the ES register <1>.
- Either register <3> specifies an offset within the array to the target location in memory.

**Figure 3 - 38 Example of word [BC]**

- The ES register <1> specifies a 64-Kbyte area within the overall 1-Mbyte space as the four higher-order bits, X, of the address range.
- "word" <2> specifies the address where the target array of word-sized data starts in the 64-Kbyte area specified in the ES register <1>.
- A pair of registers <3> specifies an offset within the array to the target location in memory.

### 3.4.8 Based indexed addressing

[Function]

Based indexed addressing uses the contents of a register pair specified with the instruction word as the base address, and the content of the B register or C register similarly specified with the instruction word as offset address. The sum of these values is used to specify the target address.

[Operand format]

Identifier	Description
—	[HL+B], [HL+C] (only the space from F0000H to FFFFFH is specifiable)
—	ES:[HL+B], ES:[HL+C] (higher 4-bit addresses are specified by the ES register)

Figure 3 - 39 Example of [HL + B], [HL + C]

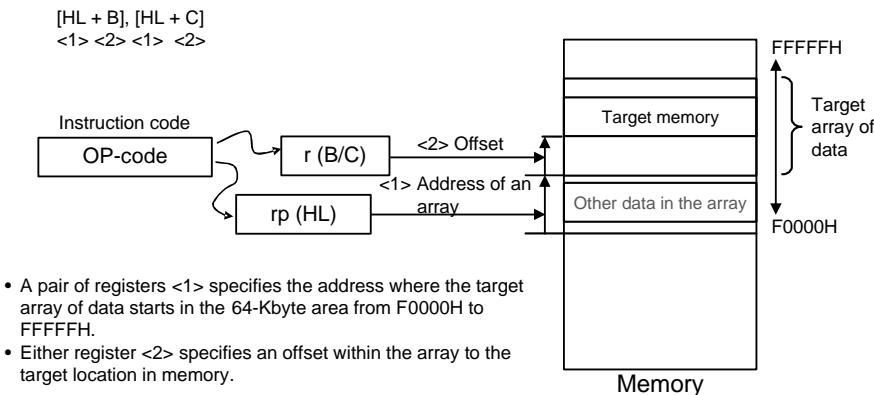
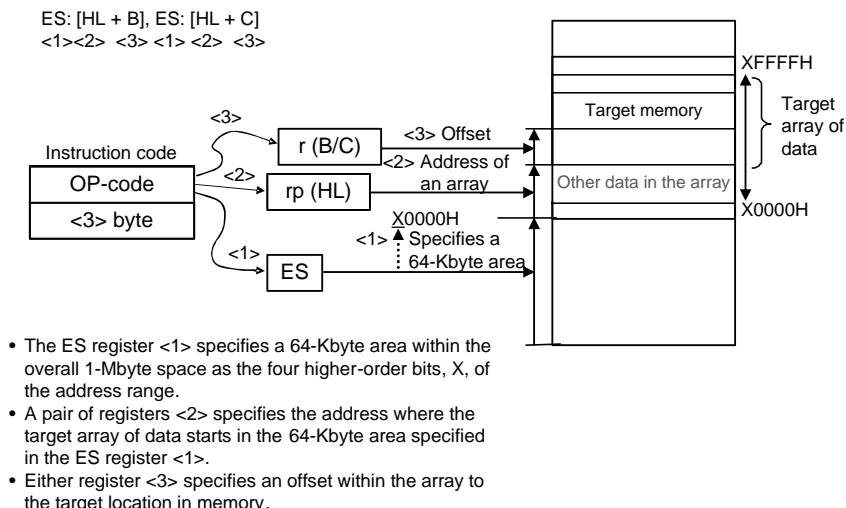


Figure 3 - 40 Example of ES:[HL + B], ES:[HL + C]



### 3.4.9 Stack addressing

[Function]

The stack area is indirectly addressed with the stack pointer (SP) values. This addressing is automatically employed when the PUSH, POP, subroutine call, and return instructions are executed or the register is saved/restored upon generation of an interrupt request.

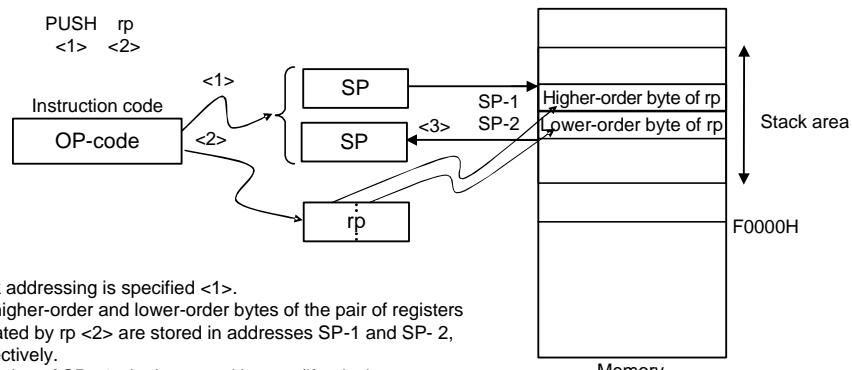
Only the internal RAM area can be set as the stack area.

[Operand format]

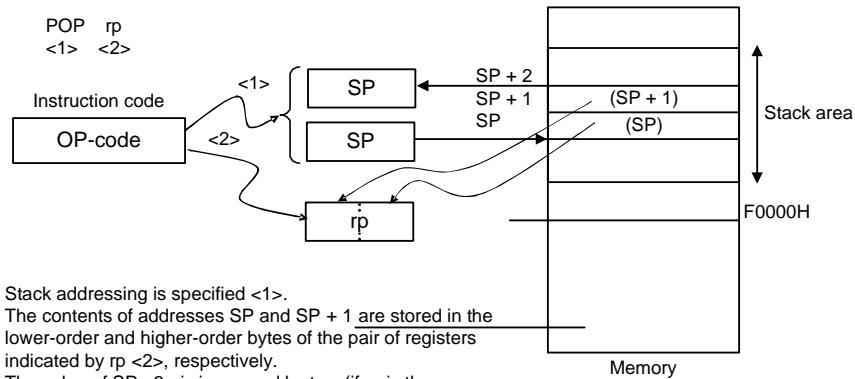
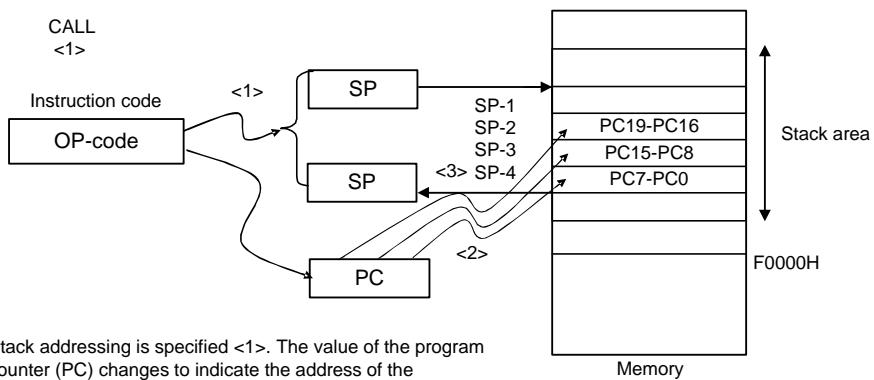
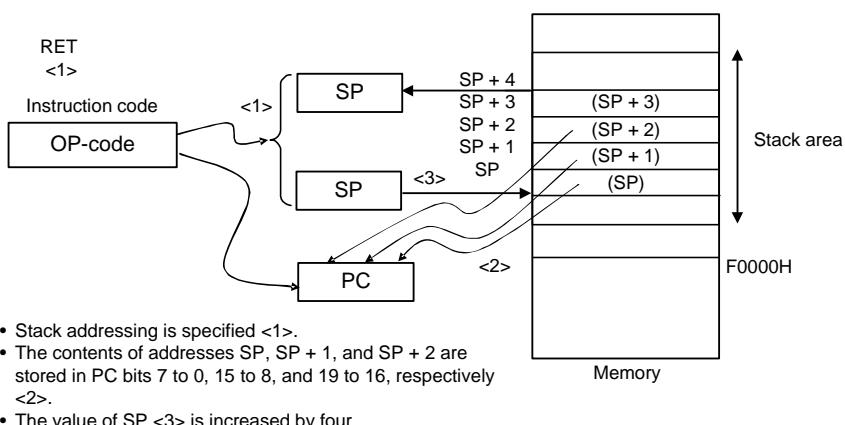
Identifier	Description
—	PUSH PSW AX/BC/DE/HL POP PSW AX/BC/DE/HL CALL/CALLT RET BRK RETB (Interrupt request generated) RETI

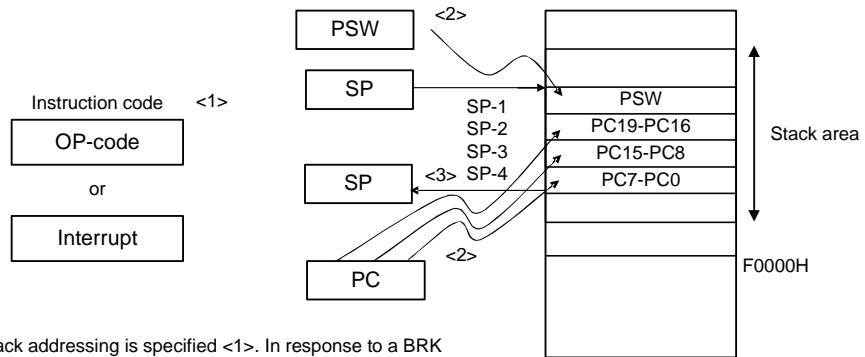
The data to be saved/restored by each stack operation is shown in Figures 3 - 41 to 3 - 46.

**Figure 3 - 41 Example of PUSH rp**

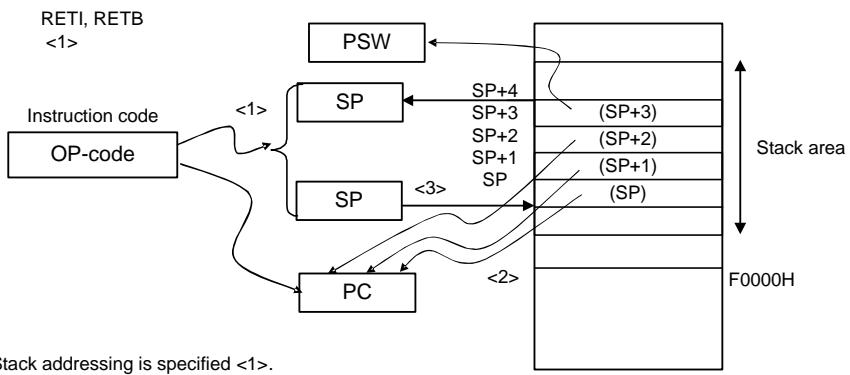


- Stack addressing is specified **<1>**.
- The higher-order and lower-order bytes of the pair of registers indicated by **rp <2>** are stored in addresses **SP-1** and **SP-2**, respectively.
- The value of **SP <3>** is decreased by two (if **rp** is the program status word (PSW), the value of the PSW is stored in **SP-1** and 0 is stored in **SP-2**).

**Figure 3 - 42 Example of POP****Figure 3 - 43 Example of CALL, CALLT****Figure 3 - 44 Example of RET**

**Figure 3 - 45 Example of Interrupt, BRK**

- Stack addressing is specified <1>. In response to a BRK instruction or acceptance of an interrupt, the value of the program counter (PC) changes to indicate the address of the next instruction.
- The values of the PSW, PC bits 19 to 16, 15 to 8, and 7 to 0 are stored in addresses SP-1, SP-2, SP-3, and SP-4, respectively <2>.
- The value of the SP <3> is decreased by 4.

**Figure 3 - 46 Example of RETI, RETB**

- Stack addressing is specified <1>.
- The contents of addresses SP, SP + 1, SP + 2, and SP + 3 are stored in PC bits 7 to 0, 15 to 8, 19 to 16, and the PSW, respectively <2>.
- The value of SP <3> is increased by four.

## CHAPTER 4 PORT FUNCTIONS

### 4.1 Port Functions

The RL78/G14 microcontrollers are provided with digital I/O ports, which enable variety of control operations. In addition to the function as digital I/O ports, these ports have several alternate functions. For details of the alternate functions, see **CHAPTER 2 PIN FUNCTIONS**.

## 4.2 Port Configuration

Ports include the following hardware.

**Table 4 - 1 Port Configuration (1/2)**

Item	Configuration
Control registers	Port mode registers (PM0 to PM8, PM10 to PM12, PM14, PM15) Port registers (P0 to P8, P10 to P15) Pull-up resistor option registers (PU0, PU1, PU3 to PU8, PU10 to PU12, PU14) Port input mode registers (PIM0, PIM1, PIM3 to PIM5, PIM8, PIM14) Port output mode registers (POM0, POM1, POM3 to POM5, POM7, POM8, POM14) Port mode control registers (PMC0, PMC1, PMC10, PMC12, PMC14) A/D port configuration register (ADPC) Peripheral I/O redirection registers (PIOR0, PIOR1) Global digital input disable register (GDIDIS)
Port	<ul style="list-style-type: none"> <li>• 30-pin products Total: 26 (CMOS I/O: 21 (N-ch open drain I/O [VDD tolerance]: 10), CMOS input: 3, N-ch open drain I/O [6 V tolerance]: 2)</li> <li>• 32-pin products Total: 28 (CMOS I/O: 22 (N-ch open drain I/O [VDD tolerance]: 10), CMOS input: 3, N-ch open drain I/O [6 V tolerance]: 3)</li> <li>• 36-pin products Total: 32 (CMOS I/O: 26 (N-ch open drain I/O [VDD tolerance]: 11), CMOS input: 3, N-ch open drain I/O [6 V tolerance]: 3)</li> <li>• 40-pin products Total: 36 (CMOS I/O: 28 (N-ch open drain I/O [VDD tolerance]: 11), CMOS input: 5, N-ch open drain I/O [6 V tolerance]: 3)</li> <li>• 44-pin products Total: 40 (CMOS I/O: 31 (N-ch open drain I/O [VDD tolerance]: 11), CMOS input: 5, N-ch open drain I/O [6 V tolerance]: 4)</li> <li>• 48-pin products Total: 44 (CMOS I/O: 34 (N-ch open drain I/O [VDD tolerance]: 12), CMOS input: 5, CMOS output: 1, N-ch open drain I/O [6 V tolerance]: 4)</li> <li>• 52-pin products Total: 48 (CMOS I/O: 38 (N-ch open drain I/O [VDD tolerance]: 14), CMOS input: 5, CMOS output: 1, N-ch open drain I/O [6 V tolerance]: 4)</li> <li>• 64-pin products Total: 58 (CMOS I/O: 48 (N-ch open drain I/O [EVDD tolerance]: 16), CMOS input: 5, CMOS output: 1, N-ch open drain I/O [6 V tolerance]: 4)</li> <li>• 80-pin products Total: 74 (CMOS I/O: 64 (N-ch open drain I/O [EVDD tolerance]: 25), CMOS input: 5, CMOS output: 1, N-ch open drain I/O [6 V tolerance]: 4)</li> <li>• 100-pin products Total: 92 (CMOS I/O: 82 (N-ch open drain I/O [EVDD tolerance]: 28), CMOS input: 5, CMOS output: 1, N-ch open drain I/O [6 V tolerance]: 4)</li> </ul>

**Table 4 - 2 Port Configuration (2/2)**

Item	Configuration	
Pull-up resistor	• 30-pin products • 32-pin products • 36-pin products • 40-pin products • 44-pin products • 48-pin products • 52-pin products • 64-pin products • 80-pin products • 100-pin products	Total: 17 Total: 18 Total: 20 Total: 21 Total: 23 Total: 26 Total: 30 Total: 40 Total: 52 Total: 67

### 4.2.1 Port 0

Port 0 is an I/O port with an output latch. Port 0 can be set to the input mode or output mode in 1-bit units using port mode register 0 (PM0). When the P00 to P06 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 0 (PU0).

Input to the P01, P03 and P04 pins can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 0 (PIM0).

Output from the P00, and P02 to P04 pins can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 0 (POM0).

To use P00 to P03 as digital input/output pins, set them in the digital I/O mode by using port mode control register 0 (PMC0) (can be specified in 1-bit units).

This port can also be used for timer I/O, A/D converter analog input, serial interface data I/O, and clock I/O.

When reset signal is generated, the following configuration will be set.

- P00 and P01 pins of the 30 and 32-pin products..... Analog input
- P00, P01 and P04 to P07 pins of the 36 to 100-pin products..... Input port
- P02 and P03 pins of the 36 to 100-pin products..... Analog input

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

### 4.2.2 Port 1

Port 1 is an I/O port with an output latch. Port 1 can be set to the input mode or output mode in 1-bit units using port mode register 1 (PM1). When the P10 to P17 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 1 (PU1).

Input to the P10 and P14 to P17 pins can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 1 (PIM1).

Output from the P10, P11, P13 to P15 and P17 pins can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 1 (POM1).

This port can also be used for serial interface data I/O, clock I/O, timer I/O, external interrupt request input, and comparator reference voltage input.

In products with 96 KB or more code flash memory, this port can also be used for comparator analog voltage input.

To use the P12, P13, P16, and P17 pins as analog input pins, set them to analog input using port mode control register 1 (PMC1) (can be specified in 1-bit units).

Reset signal generation sets port 1 to input mode.

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

### 4.2.3 Port 2

Port 2 is an I/O port with an output latch. Port 2 can be set to the input mode or output mode in 1-bit units using port mode register 2 (PM2).

This port can also be used for A/D converter analog input and reference voltage (+ side and - side) input. In products with 96 KB or more code flash memory, this port can also be used for D/A converter output.

To use P20/ANI0, P21/ANI1, P22/ANI2/ANO0, P23/ANI3/ANO1, and P24/ANI4 to P27/ANI7 as digital I/O pins, set them to digital I/O using the A/D port configuration register (ADPC). Use these pins starting from the upper bit.

To use P20/ANI0, P21/ANI1, P22/ANI2/ANO0, P23/ANI3/ANO1, and P24/ANI4 to P27/ANI7 as analog input pins, set them to analog function using the A/D port configuration register (ADPC) and input mode using the PM2 register. Use these pins starting from the lower bit.

**Table 4 - 3 Setting Functions of P20/ANI0 to P27/ANI7 Pins (64 KB or less code flash memory products)**

ADPC Register	PM2 Register	ADS Register	P20/ANI0 to P27/ANI7 Pins
Digital I/O selection	Input mode	—	Digital input
	Output mode	—	Digital output
Analog function selection	Input mode	Selects ANI.	Analog input (to be converted)
		Does not select ANI.	Analog input (not to be converted)
	Output mode	Selects ANI.	Setting prohibited
		Does not select ANI.	

All P20/ANI0 to P27/ANI7 are set in the analog function mode when the reset signal is generated.

**Table 4 - 4 Setting Functions of P20/ANI0, P21/ANI1, P24/ANI4 to P27/ANI7 Pins  
(96 KB or more code flash memory products)**

ADPC Register	PM2 Register	ADS Register	P20/ANI0, P21/ANI1, P24/ANI4 to P27/ANI7 Pins
Digital I/O selection	Input mode	—	Digital input
	Output mode	—	Digital output
Analog function selection	Input mode	Selects ANI.	Analog input (to be converted)
		Does not select ANI.	Analog input (not to be converted)
	Output mode	Selects ANI.	Setting prohibited
		Does not select ANI.	

All P20/ANI0, P21/ANI1, P24/ANI4 to P27/ANI7 are set in the analog function mode when the reset signal is generated.

**Table 4 - 5 Setting Functions of P22/ANI2/ANO0, P23/ANI3/ANO1 Pins  
(96 KB or more code flash memory products)**

ADPC Register	PM2 Register	DAM Register	ADS Register	P22/ANI2/ANO0, P23/ANI3/ANO1 Pins
Digital I/O selection	Input mode	—	—	Digital input
	Output mode	—	—	Digital output
Analog function selection	Input mode	Enables D/A conversion operation	Selects ANI.	Setting prohibited
			Does not select ANI.	Analog output
	Output mode	Stops D/A conversion operation	Selects ANI.	Analog input (to be converted)
			Does not select ANI.	Analog input (not to be converted)

All P22/ANI2/ANO0, P23/ANI3/ANO1 are set in the analog function mode when the reset signal is generated.

#### 4.2.4 Port 3

Port 3 is an I/O port with an output latch. Port 3 can be set to the input mode or output mode in 1-bit units using port mode register 3 (PM3). When the P30 and P31 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 3 (PU3).

Input to the P31 pin can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 3 (PIM3).

Output from the P30 pin can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 3 (POM3).

This port can also be used for external interrupt request input, real-time clock correction clock output, serial interface clock I/O, and timer I/O.

Reset signal generation sets port 3 to input port.

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

#### 4.2.5 Port 4

Port 4 is an I/O port with an output latch. Port 4 can be set to the input mode or output mode in 1-bit units using port mode register 4 (PM4). When the P40 to P47 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 4 (PU4).

Input to the P43 and P44 pins can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 4 (PIM4).

Output from the P43 to P45 pins can be specified as N-ch open-drain output (EVDD tolerance) in 1-bit units using port output mode register 4 (POM4).

This port can also be used for data I/O for a flash memory programmer/debugger, serial interface data I/O, clock I/O, and external interrupt request input.

Reset signal generation sets port 4 to input port.

#### 4.2.6 Port 5

Port 5 is an I/O port with an output latch. Port 5 can be set to the input mode or output mode in 1-bit units using port mode register 5 (PM5). When the P50 to P57 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 5 (PU5).

Input to the P50, P53 to P55 pin can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 5 (PIM5).

Output from the P50 to P55 pins can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 5 (POM5).

This port can also be used for external interrupt request input, serial interface data I/O, clock I/O, programming UART transmission/reception, and timer I/O.

Reset signal generation sets port 5 to input port.

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

#### 4.2.7 Port 6

Port 6 is an I/O port with an output latch. Port 6 can be set to the input mode or output mode in 1-bit units using port mode register 6 (PM6). When the P64 to P67 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 6 (PU6).

The output of the P60 to P63 pins is N-ch open-drain output (6 V tolerance).

This port can also be used for serial interface data I/O and clock I/O, chip select input, and timer I/O.

Reset signal generation sets port 6 to input port.

#### 4.2.8 Port 7

Port 7 is an I/O port with an output latch. Port 7 can be set to the input mode or output mode in 1-bit units using port mode register 7 (PM7). When used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 7 (PU7).

Output from the P71 and P74 pins can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 7 (POM7).

This port can also be used for key interrupt input, serial interface data I/O, clock I/O, and external interrupt request input.

Reset signal generation sets port 7 to input port.

**Note 1.** For 32- to 52-pin products

**Note 2.** For 64- to 100-pin products

#### 4.2.9 Port 8

Port 8 is an I/O port with an output latch. Port 8 can be set to the input mode or output mode in 1-bit units using port mode register 8 (PM8). When the P80 to P87 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 8 (PU8).

Input to the P80 and P81 pins can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 8 (PIM8).

Output from the P80 to P82 pin can be specified as N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) in 1-bit units using port output mode register 8 (POM8).

Reset signal generation sets port 8 to input port.

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

#### 4.2.10 Port 10

Port 10 is an I/O port with an output latch. Port 10 can be set to the input mode or output mode in 1-bit units using port mode register 10 (PM10). When the P100 to P102 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 10 (PU10).

To use the P100 pin as a digital I/O port, set it to digital I/O using port mode control register 10 (PMC10).

This port can also be used for A/D converter analog input.

Reset signal generation sets P100 to analog function, P101, P102 to input port.

#### 4.2.11 Port 11

Port 11 is an I/O port with an output latch. Port 11 can be set to the input mode or output mode in 1-bit units using port mode register 11 (PM11). When the P110 and P111 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 11 (PU11).

Reset signal generation sets port 11 to input port.

#### 4.2.12 Port 12

P120 is an I/O port with an output latch. Port 12 can be set to the input mode or output mode in 1-bit units using port mode register 12 (PM12). When used as an input port, use of an on-chip pull-up resistor can be specified by pull-up resistor option register 12 (PU12).

P121 to P124 are 4-bit input ports.

To use the P120 pin as a digital I/O port, set it to digital I/O using port mode control register 12 (PMC12).

This port can also be used for A/D converter analog input, connecting a resonator for the main system clock, connecting a resonator for the subsystem clock, external clock input for the main system clock, and external clock input for the subsystem clock. In products with 96 KB or more code flash memory, this port can also be used for comparator output.

Reset signal generation sets P120 to analog function, and sets P121 to P124 to input port.

#### 4.2.13 Port 13

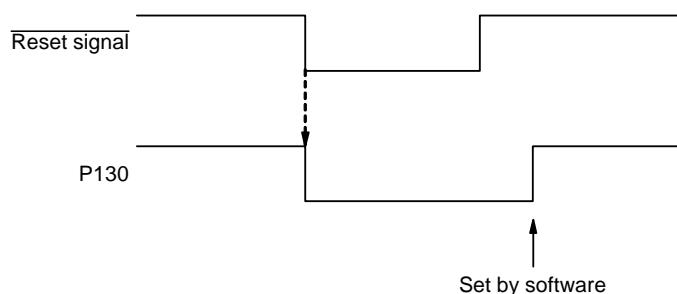
P130 is a 1-bit output-only port with an output latch.

P137 is a 1-bit input-only port.

P130 is fixed an output port, and P137 is fixed an input ports.

This port can also be used for external interrupt request input.

**Remark** When reset is effected, P130 outputs a low level. If P130 is set to output a high level before reset is effected, the output signal of P130 can be dummy-output as the CPU reset signal.



#### 4.2.14 Port 14

Port 14 is an I/O port with an output latch. Port 14 can be set to the input mode or output mode in 1-bit units using port mode register 14 (PM14). When the P140 to P147 pins are used as an input port, use of an on-chip pull-up resistor can be specified in 1-bit units by pull-up resistor option register 14 (PU14).

Input to the P142 and P143 pins can be specified through a normal input buffer or a TTL input buffer in 1-bit units using port input mode register 14 (PIM14).

Output from the P142 to P144 pin can be specified as N-ch open-drain output (EVDD tolerance) in 1-bit units using port output mode register 14 (POM14).

To use the P147 pin as a digital I/O port, set it to digital I/O using port mode control register 14 (PMC14).

This port can also be used for clock/buzzer output, external interrupt request input, A/D converter analog input, serial interface data I/O, and clock I/O. In products with 96 KB or more code flash memory, this port can also be used for comparator output.

Reset signal generation sets P140 to P146 to input port, and sets P147 to analog function.

#### 4.2.15 Port 15

Port 15 is an I/O port with an output latch. Port 15 can be set to the input mode or output mode in 1-bit units using port mode register 15 (PM15).

This port can also be used for A/D converter analog input.

To use P150/ANI8 to P156/ANI14 as digital I/O pins, set them to digital I/O using the A/D port configuration register (ADPC). Use these pins starting from the upper bit.

To use P150/ANI8 to P156/ANI14 as analog input pins, set them to analog function using the A/D port configuration register (ADPC) and input mode using the PM15 register. Use these pins starting from the lower bit.

**Table 4 - 6 Setting Functions of P150/ANI8 to P156/ANI14 Pins**

ADPC Register	PM15 Register	ADS Register	P150/ANI8 to P156/ANI14 Pins
Digital I/O selection	Input mode	—	Digital input
	Output mode	—	Digital output
Analog function selection	Input mode	Selects ANI.	Analog input (to be converted)
		Does not select ANI.	Analog input (not to be converted)
	Output mode	Selects ANI.	Setting prohibited
		Does not select ANI.	

All P150/ANI8 to P156/ANI14 are set in the analog function mode when the reset signal is generated.

## 4.3 Registers Controlling Port Function

Port functions are controlled by the following registers.

- Port mode registers (PMxx)
- Port registers (Pxx)
- Pull-up resistor option registers (PUxx)
- Port input mode registers (PIMxx)
- Port output mode registers (POMxx)
- Port mode control registers (PMCxx)
- A/D port configuration register (ADPC)
- Peripheral I/O redirection register 0, 1 (PIOR0, PIOR1)
- Global digital input disable register (GDIDIS)

**Caution** Which registers and bits are included depends on the product. For registers and bits mounted on each product, see Tables 4 - 7 to 4 - 13. Be sure to set bits that are not mounted to their initial values.

**Table 4 - 7 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(30-pin products to 64-pin products) (1/3)**

Port		Bit name					64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register								
Port 0	0	PM00	P00	PU00	—	POM00	PMC00 Note 1	✓	✓	✓	✓	✓	✓	✓
	1	PM01	P01	PU01	PIM01	—	PMC01 Note 1	✓	✓	✓	✓	✓	✓	✓
	2	PM02	P02	PU02	—	POM02	PMC02	✓	✓	—	—	—	—	—
	3	PM03	P03	PU03	PIM03	POM03	PMC03	✓	✓	—	—	—	—	—
	4	PM04	P04	PU04	PIM04	POM04	—	✓	—	—	—	—	—	—
	5	PM05	P05	PU05	—	—	—	✓	—	—	—	—	—	—
	6	PM06	P06	PU06	—	—	—	✓	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 1	0	PM10	P10	PU10	PIM10	POM10	—	✓	✓	✓	✓	✓	✓	✓
	1	PM11	P11	PU11	—	POM11	—	✓	✓	✓	✓	✓	✓	✓
	2	PM12	P12	PU12	—	—	PMC12 Note 2	✓	✓	✓	✓	✓	✓	✓
	3	PM13	P13	PU13	—	POM13	PMC13 Note 2	✓	✓	✓	✓	✓	✓	✓
	4	PM14	P14	PU14	PIM14	POM14	—	✓	✓	✓	✓	✓	✓	✓
	5	PM15	P15	PU15	PIM15	POM15	—	✓	✓	✓	✓	✓	✓	✓
	6	PM16	P16	PU16	PIM16	—	PMC16 Note 2	✓	✓	✓	✓	✓	✓	✓
	7	PM17	P17	PU17	PIM17	POM17	PMC17 Note 2	✓	✓	✓	✓	✓	✓	✓
Port 2	0	PM20	P20	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	1	PM21	P21	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	2	PM22	P22	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	3	PM23	P23	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	4	PM24	P24	—	—	—	—	✓	✓	✓	✓	✓	✓	—
	5	PM25	P25	—	—	—	—	✓	✓	✓	✓	✓	✓	—
	6	PM26	P26	—	—	—	—	✓	✓	✓	✓	✓	—	—
	7	PM27	P27	—	—	—	—	✓	✓	✓	✓	—	—	—

**Note 1.** 30-pin and 32-pin products only.

**Note 2.** R5F104xF (x = A to C, E to G, J, L, M, P), R5F104xG (x = A to C, E to G, J, L, M, P), R5F104xH (x = E to G, J, L, M, P), R5F104xJ (x = F, G, J, L, M, P), R5F104xK (x = G, L, M, P), R5F104xL (x = G, L, M, P) only.

**Table 4 - 8 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(30-pin products to 64-pin products) (2/3)**

Port		Bit name					64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register								
Port 3	0	PM30	P30	PU30	PIM30	POM30	—	✓	✓	✓	✓	✓	✓	✓
	1	PM31	P31	PU31	—	—	—	✓	✓	✓	✓	✓	✓	✓
	2	—	—	—	—	—	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 4	0	PM40	P40	PU40	—	—	—	✓	✓	✓	✓	✓	✓	✓
	1	PM41	P41	PU41	—	—	—	✓	✓	✓	✓	—	—	—
	2	PM42	P42	PU42	—	—	—	✓	—	—	—	—	—	—
	3	PM43	P43	PU43	—	—	—	✓	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 5	0	PM50	P50	PU50	PIM50	POM50	—	✓	✓	✓	✓	✓	✓	✓
	1	PM51	P51	PU51	—	POM51	—	✓	✓	✓	✓	✓	✓	✓
	2	PM52	P52	PU52	—	—	—	✓	—	—	—	—	—	—
	3	PM53	P53	PU53	—	—	—	✓	—	—	—	—	—	—
	4	PM54	P54	PU54	—	—	—	✓	—	—	—	—	—	—
	5	PM55	P55	PU55	PIM55	POM55	—	✓	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 6	0	PM60	P60	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	1	PM61	P61	—	—	—	—	✓	✓	✓	✓	✓	✓	✓
	2	PM62	P62	—	—	—	—	✓	✓	✓	✓	✓	✓	—
	3	PM63	P63	—	—	—	—	✓	✓	✓	✓	—	—	—
	4	—	—	—	—	—	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 7	0	PM70	P70	PU70	—	—	—	✓	✓	✓	✓	✓	✓	✓
	1	PM71	P71	PU71	—	POM71	—	✓	✓	✓	✓	✓	✓	—
	2	PM72	P72	PU72	—	—	—	✓	✓	✓	✓	✓	✓	—
	3	PM73	P73	PU73	—	—	—	✓	✓	✓	✓	✓	✓	—
	4	PM74	P74	PU74	—	POM74	—	✓	✓	✓	✓	—	—	—
	5	PM75	P75	PU75	—	—	—	✓	✓	✓	✓	—	—	—
	6	PM76	P76	PU76	—	—	—	✓	✓	—	—	—	—	—
	7	PM77	P77	PU77	—	—	—	✓	✓	—	—	—	—	—

**Table 4 - 9 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(30-pin products to 64-pin products) (3/3)**

Port		Bit name						64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register	PMCxx register								
Port 12	0	PM120	P120	PU120	—	—	PMC120	✓	✓	✓	✓	✓	✓	✓	✓
	1	—	P121	—	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓
	2	—	P122	—	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓
	3	—	P123	—	—	—	—	✓	✓	✓	✓	✓	✓	—	—
	4	—	P124	—	—	—	—	✓	✓	✓	✓	✓	✓	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Port 13	0	—	P130	—	—	—	—	✓	✓	✓	—	—	—	—	—
	1	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	2	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	7	—	P137	—	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓
Port 14	0	PM140	P140	PU140	—	—	—	✓	✓	✓	—	—	—	—	—
	1	PM141	P141	PU141	—	—	—	✓	—	—	—	—	—	—	—
	2	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	6	PM146	P146	PU146	—	—	—	✓	✓	✓	✓	✓	—	—	—
	7	PM147	P147	PU147	—	—	—	PMC147	✓	✓	✓	✓	✓	✓	✓

**Table 4 - 10 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(80-pin products to 100-pin products) (1/4)**

Port		Bit name						100-pin	80-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register	PMCxx register		
Port 0	0	PM00	P00	PU00	—	POM00	—	✓	✓
	1	PM01	P01	PU01	PIM01	—	—	✓	✓
	2	PM02	P02	PU02	—	POM02	PMC02	✓	✓
	3	PM03	P03	PU03	PIM03	POM03	PMC03	✓	✓
	4	PM04	P04	PU04	PIM04	POM04	—	✓	✓
	5	PM05	P05	PU05	—	—	—	✓	✓
	6	PM06	P06	PU06	—	—	—	✓	✓
	7	—	—	—	—	—	—	—	—
Port 1	0	PM10	P10	PU10	PIM10	POM10	—	✓	✓
	1	PM11	P11	PU11	—	POM11	—	✓	✓
	2	PM12	P12	PU12	—	—	PMC12	✓	✓
	3	PM13	P13	PU13	—	POM13	PMC13	✓	✓
	4	PM14	P14	PU14	PIM14	POM14	—	✓	✓
	5	PM15	P15	PU15	PIM15	POM15	—	✓	✓
	6	PM16	P16	PU16	PIM16	—	PMC16	✓	✓
	7	PM17	P17	PU17	PIM17	POM17	PMC17	✓	✓
Port 2	0	PM20	P20	—	—	—	—	✓	✓
	1	PM21	P21	—	—	—	—	✓	✓
	2	PM22	P22	—	—	—	—	✓	✓
	3	PM23	P23	—	—	—	—	✓	✓
	4	PM24	P24	—	—	—	—	✓	✓
	5	PM25	P25	—	—	—	—	✓	✓
	6	PM26	P26	—	—	—	—	✓	✓
	7	PM27	P27	—	—	—	—	✓	✓
Port 3	0	PM30	P30	PU30	PIM30	POM30	—	✓	✓
	1	PM31	P31	PU31	—	—	—	✓	✓
	2	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—

**Table 4 - 11 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(80-pin products and 100-pin products) (2/4)**

Port		Bit name						100-pin	80-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register	PMCxx register		
Port 4	0	PM40	P40	PU40	—	—	—	✓	✓
	1	PM41	P41	PU41	—	—	—	✓	✓
	2	PM42	P42	PU42	—	—	—	✓	✓
	3	PM43	P43	PU43	PIM43	POM43	—	✓	✓
	4	PM44	P44	PU44	PIM44	POM44	—	✓	✓
	5	PM45	P45	PU45	—	POM45	—	✓	✓
	6	PM46	P46	PU46	—	—	—	✓	—
	7	PM47	P47	PU47	—	—	—	✓	—
Port 5	0	PM50	P50	PU50	PIM50	POM50	—	✓	✓
	1	PM51	P51	PU51	—	POM51	—	✓	✓
	2	PM52	P52	PU52	—	POM52	—	✓	✓
	3	PM53	P53	PU53	PIM53	POM53	—	✓	✓
	4	PM54	P54	PU54	PIM54	POM54	—	✓	✓
	5	PM55	P55	PU55	PIM55	POM55	—	✓	✓
	6	PM56	P56	PU56	—	—	—	✓	—
	7	PM57	P57	PU57	—	—	—	✓	—
Port 6	0	PM60	P60	—	—	—	—	✓	✓
	1	PM61	P61	—	—	—	—	✓	✓
	2	PM62	P62	—	—	—	—	✓	✓
	3	PM63	P63	—	—	—	—	✓	✓
	4	PM64	P64	PU64	—	—	—	✓	✓
	5	PM65	P65	PU65	—	—	—	✓	✓
	6	PM66	P66	PU66	—	—	—	✓	✓
	7	PM67	P67	PU67	—	—	—	✓	✓
Port 7	0	PM70	P70	PU70	—	—	—	✓	✓
	1	PM71	P71	PU71	—	POM71	—	✓	✓
	2	PM72	P72	PU72	—	—	—	✓	✓
	3	PM73	P73	PU73	—	—	—	✓	✓
	4	PM74	P74	PU74	—	POM74	—	✓	✓
	5	PM75	P75	PU75	—	—	—	✓	✓
	6	PM76	P76	PU76	—	—	—	✓	✓
	7	PM77	P77	PU77	—	—	—	✓	✓
Port 8	0	PM80	P80	PU80	PIM80	POM80	—	✓	✓
	1	PM81	P81	PU81	PIM81	POM81	—	✓	✓
	2	PM82	P82	PU82	—	POM82	—	✓	—
	3	PM83	P83	PU83	—	—	—	✓	—
	4	PM84	P84	PU84	—	—	—	✓	—
	5	PM85	P85	PU85	—	—	—	✓	—
	6	PM86	P86	PU86	—	—	—	✓	—
	7	PM87	P87	PU87	—	—	—	✓	—

**Table 4 - 12 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(80-pin products and 100-pin products) (3/4)**

Port		Bit name						100-pin	80-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register	PMCxx register		
Port 10	0	PM100	P100	PU100	—	—	PMC100	✓	✓
	1	PM101	P101	PU101	—	—	—	✓	—
	2	PM102	P102	PU102	—	—	—	✓	—
	3	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—
Port 11	0	PM110	P110	PU110	—	—	—	✓	✓
	1	PM111	P111	PU111	—	—	—	✓	✓
	2	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—
Port 12	0	PM120	P120	PU120	—	—	PMC120	✓	✓
	1	—	P121	—	—	—	—	✓	✓
	2	—	P122	—	—	—	—	✓	✓
	3	—	P123	—	—	—	—	✓	✓
	4	—	P124	—	—	—	—	✓	✓
	5	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—
	7	—	—	—	—	—	—	—	—
Port 13	0	—	P130	—	—	—	—	✓	✓
	1	—	—	—	—	—	—	—	—
	2	—	—	—	—	—	—	—	—
	3	—	—	—	—	—	—	—	—
	4	—	—	—	—	—	—	—	—
	5	—	—	—	—	—	—	—	—
	6	—	—	—	—	—	—	—	—
	7	—	P137	—	—	—	—	✓	✓

**Table 4 - 13 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product  
(80-pin products and 100-pin products) (4/4)**

Port		Bit name						100-pin	80-pin
		PMxx register	Pxx register	PUxx register	PIMxx register	POMxx register	PMCxx register		
Port 14	0	PM140	P140	PU140	—	—	—	✓	✓
	1	PM141	P141	PU141	—	—	—	✓	✓
	2	PM142	P142	PU142	PIM142	POM142	—	✓	✓
	3	PM143	P143	PU143	PIM143	POM143	—	✓	✓
	4	PM144	P144	PU144	—	POM144	—	✓	✓
	5	PM145	P145	PU145	—	—	—	✓	—
	6	PM146	P146	PU146	—	—	—	✓	✓
	7	PM147	P147	PU147	—	—	PMC147	✓	✓
Port 15	0	PM150	P150	—	—	—	—	✓	✓
	1	PM151	P151	—	—	—	—	✓	✓
	2	PM152	P152	—	—	—	—	✓	✓
	3	PM153	P153	—	—	—	—	✓	✓
	4	PM154	P154	—	—	—	—	✓	—
	5	PM155	P155	—	—	—	—	✓	—
	6	PM156	P156	—	—	—	—	✓	—
	7	—	—	—	—	—	—	—	—

### 4.3.1 Port mode registers (PMxx)

These registers specify input or output mode for the port in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

When port pins are used as alternate-function pins, set the port mode register by referencing **4.5 Register Settings When Using Alternate Function**.

**Figure 4 - 1 Format of Port mode register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PM0	1	PM06	PM05	PM04	PM03	PM02	PM01	PM00	FFF20H	FFH	R/W
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10	FFF21H	FFH	R/W
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20	FFF22H	FFH	R/W
PM3	1	1	1	1	1	1	PM31	PM30	FFF23H	FFH	R/W
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40	FFF24H	FFH	R/W
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50	FFF25H	FFH	R/W
PM6	PM67	PM66	PM65	PM64	PM63	PM62	PM61	PM60	FFF26H	FFH	R/W
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70	FFF27H	FFH	R/W
PM8	PM87	PM86	PM85	PM84	PM83	PM82	PM81	PM80	FFF28H	FFH	R/W
PM10	1	1	1	1	1	PM102	PM101	PM100	FFF2AH	FFH	R/W
PM11	1	1	1	1	1	1	PM111	PM110	FFF2BH	FFH	R/W
PM12	1	1	1	1	1	1	1	PM120	FFF2CH	FFH	R/W
PM14	PM147	PM146	PM145	PM144	PM143	PM142	PM141	PM140	FFF2EH	FFH	R/W
PM15	1	PM156	PM155	PM154	PM153	PM152	PM151	PM150	FFF2FH	FFH	R/W
PMmn	Pmn pin I/O mode selection (m = 0 to 8, 10 to 12, 14, 15; n = 0 to 7)										
0	Output mode (the pin functions as an output port (output buffer on))										
1	Input mode (the pin functions as an input port (output buffer off))										

**Caution** Be sure to set bits that are not mounted to their initial values.

### 4.3.2 Port registers (Pxx)

These registers set the output latch value of a port.

If the data is read in the input mode, the pin level is read. If it is read in the output mode, the output latch value is read. Note.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Note** When P02, P03, P12, P13, P16, P17, P20 to P27, P100, P120, and P147 are set to the analog function, if a port is read in input mode, the read value is always 0, not the pin level.

Figure 4 - 2 Format of Port register

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
P0	0	P06	P05	P04	P03	P02	P01	P00	FFF00H	00H (output latch)	R/W
P1	P17	P16	P15	P14	P13	P12	P11	P10	FFF01H	00H (output latch)	R/W
P2	P27	P26	P25	P24	P23	P22	P21	P20	FFF02H	00H (output latch)	R/W
P3	0	0	0	0	0	0	P31	P30	FFF03H	00H (output latch)	R/W
P4	P47	P46	P45	P44	P43	P42	P41	P40	FFF04H	00H (output latch)	R/W
P5	P57	P56	P55	P54	P53	P52	P51	P50	FFF05H	00H (output latch)	R/W
P6	P67	P66	P65	P64	P63	P62	P61	P60	FFF06H	00H (output latch)	R/W
P7	P77	P76	P75	P74	P73	P72	P71	P70	FFF07H	00H (output latch)	R/W
P8	P87	P86	P85	P84	P83	P82	P81	P80	FFF08H	00H (output latch)	R/W
P10	0	0	0	0	0	P102	P101	P100	FFF0AH	00H (output latch)	R/W
P11	0	0	0	0	0	0	P111	P110	FFF0BH	00H (output latch)	R/W
P12	0	0	0	P124	P123	P122	P121	P120	FFF0CH	Undefined	R/W Note 1
P13	P137	0	0	0	0	0	0	P130	FFF0DH	Note 2	R/W Note 1
P14	P147	P146	P145	P144	P143	P142	P141	P140	FFF0EH	00H (output latch)	R/W
P15	0	P156	P155	P154	P153	P152	P151	P150	FFF0FH	00H (output latch)	R/W

Pmn	m = 0 to 8, 10 to 15; n = 0 to 7							
	Output data control (in output mode)				Input data read (in input mode)			
0	Output 0				Input low level			
1	Output 1				Input high level			

**Note 1.** P121 to P124, and P137 are read-only.

**Note 2.** P137: Undefined

P130: 0 (output latch)

**Caution** Be sure to set bits that are not mounted to their initial values.

### 4.3.3 Pull-up resistor option registers (PU $xx$ )

These registers specify whether the on-chip pull-up resistors are to be used or not. On-chip pull-up resistors can be used in 1-bit units only for the bits set to input mode ( $PM_{mn} = 1$  and  $POM_{mn} = 0$ ) for the pins to which the use of an on-chip pull-up resistor has been specified in these registers. On-chip pull-up resistors cannot be connected to bits set to output mode and bits used as alternate-function output pins, regardless of the settings of these registers. Similarly, on-chip pull-up resistors cannot be connected to the pins used as alternate-function output pins and the pins set to the analog function.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H (Only PU4 is set to 01H).

**Figure 4 - 3 Format of Pull-up resistor option register**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PU0	0	PU06	PU05	PU04	PU03	PU02	PU01	PU00	F0030H	00H	R/W
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10	F0031H	00H	R/W
PU3	0	0	0	0	0	0	PU31	PU30	F0033H	00H	R/W
PU4	PU47	PU46	PU45	PU44	PU43	PU42	PU41	PU40	F0034H	01H	R/W
PU5	PU57	PU56	PU55	PU54	PU53	PU52	PU51	PU50	F0035H	00H	R/W
PU6	PU67	PU66	PU65	PU64	0	0	0	0	F0036H	00H	R/W
PU7	PU77	PU76	PU75	PU74	PU73	PU72	PU71	PU70	F0037H	00H	R/W
PU8	PU87	PU86	PU85	PU84	PU83	PU82	PU81	PU80	F0038H	00H	R/W
PU10	0	0	0	0	0	PU102	PU101	PU100	F003AH	00H	R/W
PU11	0	0	0	0	0	0	PU111	PU110	F003BH	00H	R/W
PU12	0	0	0	0	0	0	0	PU120	F003CH	00H	R/W
PU14	PU147	PU146	PU145	PU144	PU143	PU142	PU141	PU140	F003EH	00H	R/W
PUMn	Pmn pin on-chip pull-up resistor selection ( $m = 0, 1, 3 \text{ to } 8, 10 \text{ to } 12, 14; n = 0 \text{ to } 7$ )										
0	On-chip pull-up resistor not connected										
1	On-chip pull-up resistor connected										

**Caution** Be sure to set bits that are not mounted to their initial values.

#### 4.3.4 Port input mode registers (PM<sub>xx</sub>)

These registers set the input buffer in 1-bit units.

TTL input buffer can be selected during serial communication with an external device of the different potential.

Port input mode registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 4 - 4 Format of Port input mode register (100-pin products)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PIM0	0	0	0	PIM04	PIM03	0	PIM01	0	F0040H	00H	R/W
PIM1	PIM17	PIM16	PIM15	PIM14	0	0	0	PIM10	F0041H	00H	R/W
PIM3	0	0	0	0	0	0	0	PIM30	F0043H	00H	R/W
PIM4	0	0	0	PIM44	PIM43	0	0	0	F0044H	00H	R/W
PIM5	0	0	PIM55	PIM54	PIM53	0	0	PIM50	F0045H	00H	R/W
PIM8	0	0	0	0	0	0	PIM81	PIM80	F0048H	00H	R/W
PIM14	0	0	0	0	PIM143	PIM142	0	0	F004EH	00H	R/W
PIMmn	Pmn pin input buffer selection (m = 0, 1, 3 to 5, 8, 14; n = 0 to 7)										
0	Normal input buffer										
1	TTL input buffer										

**Caution** Be sure to set bits that are not mounted to their initial values.

### 4.3.5 Port output mode registers (POMxx)

These registers set the output mode in 1-bit units.

N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) mode can be selected during serial communication with an external device of the different potential, and for the SDA00, SDA01, SDA10, SDA11, SDA20, SDA21, SDA30, and SDA31 pins during simplified I<sup>2</sup>C communication with an external device of the same potential.

In addition, POMxx register is set with PUxx register, whether or not to use the on-chip pull-up resistor.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Caution** An on-chip pull-up resistor is not connected to a bit for which N-ch open drain output (VDD tolerance Note 1/EVDD tolerance Note 2) mode is set.

**Figure 4 - 5 Format of Port output mode register (100-pin products)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
POM0	0	0	0	POM04	POM03	POM02	0	POM00	F0050H	00H	R/W
POM1	POM17	0	POM15	POM14	POM13	0	POM11	POM10	F0051H	00H	R/W
POM3	0	0	0	0	0	0	0	POM30	F0053H	00H	R/W
POM4	0	0	POM45	POM44	POM43	0	0	0	F0054H	00H	R/W
POM5	0	0	POM55	POM54	POM53	POM52	POM51	POM50	F0055H	00H	R/W
POM7	0	0	0	POM74	0	0	POM71	0	F0057H	00H	R/W
POM8	0	0	0	0	0	POM82	POM81	POM80	F0058H	00H	R/W
POM14	0	0	0	POM144	POM143	POM142	0	0	F005EH	00H	R/W
POMmn	Pmn pin output mode selection (m = 0, 1, 3 to 5, 7, 8, 14; n = 0 to 5, 7)										
0	Normal output mode										
1	N-ch open-drain output (VDD tolerance Note 1/EVDD tolerance Note 2) mode										

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

**Caution** Be sure to set bits that are not mounted to their initial values.

### 4.3.6 Port mode control registers (PMCxx)

These registers set the P02, P03, P12, P13, P16, P17, P100, P120, and P147 digital I/O/analog input in 1-bit units.

These registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to FFH.

**Figure 4 - 6 Format of Port mode control register (100-pin products)**

Symbol	7	6	5	4	3	2	1	0	Address	After reset	R/W
PMC0	1	1	1	1	PMC03 Note 2	PMC02 Note 2	PMC01 Note 1	PMC00 Note 1	F0060H	FFH	R/W
PMC1	PMC17 Note 3	PMC16 Note 3	0	0	PMC13 Note 3	PMC12 Note 3	0	0	F0061H	00H	R/W
PMC10	1	1	1	1	1	1	1	PMC100 Note 4	F006AH	FFH	R/W
PMC12	1	1	1	1	1	1	1	PMC120	F006CH	FFH	R/W
PMC14	PMC147	1	1	1	1	1	1	1	F006EH	FFH	R/W
PMCmn	Pmn pin digital I/O/analog input selection (m = 0, 1, 10, 12, 14; n = 0, 2, 3, 6, 7)										
0	Digital I/O (alternate function other than analog input)										
1	Analog input										

**Note 1.** 30-, 32-pin products only

**Note 2.** 52-, 64-, 80-, 100-pin products only

**Note 3.** 96 KB or more code flash memory products only

**Note 4.** 80-, 100-pin products only

**Caution** Be sure to set bits that are not mounted to their initial values.

### 4.3.7 A/D port configuration register (ADPC)

This register is used to switch the P20/ANI0, P21/ANI1, P22/ANI2/ANO0, P23/ANI3/ANO1, P24/ANI4 to P27/ANI7, ANI8/P150 to ANI14/P156 pins to digital I/O or analog function of A/D converter and to switch the P22/ANO0/ANI2 and P23/ANO1/ANI3 pins to digital I/O of port or analog function of D/A converter.

The ADPC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Figure 4 - 7 Format of A/D port configuration register (ADPC)**

Address: F0076H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADPC	0	0	0	0	ADPC3	ADPC2	ADPC1	ADPC0

ADPC3	ADPC2	ADPC1	ADPC0	Analog function (A)/digital I/O (D) switching														
				AN14/P156	AN13/P155	AN12/P154	AN11/P153	AN10/P152	AN19/P151	AN18/P150	AN17/P27	AN16/P26	AN15/P25	AN14/P24	AN13/ANO1/P23	AN12/ANO0/P22	AN11/P21	AN10/P20
0	0	0	0	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
0	0	0	1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D
0	0	1	0	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A
0	0	1	1	D	D	D	D	D	D	D	D	D	D	D	D	D	D	A
0	1	0	0	D	D	D	D	D	D	D	D	D	D	D	D	D	A	A
0	1	0	1	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
0	1	1	0	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
0	1	1	1	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1	0	0	0	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1	0	0	1	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1	0	1	0	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1	0	1	1	D	D	D	D	D	D	D	D	D	D	D	D	A	A	A
1	1	0	0	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A
1	1	0	1	D	D	D	A	A	A	A	A	A	A	A	A	A	A	A
1	1	1	0	D	D	A	A	A	A	A	A	A	A	A	A	A	A	A
1	1	1	1	D	A	A	A	A	A	A	A	A	A	A	A	A	A	A

**Caution 1.** Set the port to analog input by ADPC register to the input mode by using port mode registers 2, 15 (PM2, PM15).

**Caution 2.** Do not set the pin set by the ADPC register as digital I/O by the analog input channel specification register (ADS).

**Caution 3.** When using AVREFP and AVREFM, set ANI0 and ANI1 to analog input and set the port mode register to the input mode.

**Caution 4.** Do not set the pin set by the ADPC register as digital I/O by D/A converter mode register (DAM) as D/A conversion operation enable.

#### 4.3.8 Peripheral I/O redirection register 0 (PIOR0)

This register is used to specify whether to enable or disable the peripheral I/O redirect function.

This function is used to switch ports to which alternate functions are assigned.

In addition, the settings for redirection can be changed only until operation of the function is enabled.

The PIOR0 register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Figure 4 - 8 Format of Peripheral I/O redirection register 0 (PIOR0)**

Address: F0077H After reset: 00H R/W

Symbol 7 6 5 4 3 2 1 0

PIOR0	0	PIOR06 Note	PIOR05	PIOR04	PIOR03	PIOR02	PIOR01	PIOR00
-------	---	-------------	--------	--------	--------	--------	--------	--------

Bit	Function	100-pin		80-pin		64-pin		52-pin		48-pin		44-pin		40/36/32/30-pin	
		Setting value		Setting value		Setting value		Setting value		Setting value		Setting value		Setting value	
		0	1	0	1	0	1	0	1	0	1	0	1	0	1
PIOR06 Note	TxD0_1	—	P12	—	P12	—	P12	—	P12	—	P12	—	P12	—	P12
	RxD0_1	—	P11	—	P11	—	P11	—	P11	—	P11	—	P11	—	P11
PIOR05	INTP1	P46	P56	This area cannot be used. Be set to 0 (default value).											
	INTP3	P30	P57												
	INTP4	P31	P146												
	INTP6	P140	P84												
	INTP7	P141	P85												
	INTP8	P74	P86												
	INTP9	P75	P87												
	TxD1	P02	P82												
	RxD1	P03	P81												
	SCL10	P04	P80												
	SDA10	P03	P81												
PIOR04	SI10	P03	P81												
	SO10	P02	P82												
	SCK10	P04	P80												
	PCLBUZ1	P141	P55	P141	P55	P141	P55								
	INTP5	P16	P12	P16	P12	P16	P12								
	PCLBUZ0	P140	P31	P140	P31	P140	P31	P140	P31	P140	P31				
PIOR03	SCLA0	P60	P14	P60	P14	P60	P14	P60	P14	P60	P14	P60	P14	P60	P14
	SDAA0	P61	P15	P61	P15	P61	P15	P61	P15	P61	P15	P61	P15	P61	P15
	PIOR01	INTP10	P76	P100	P76	P100	P76	P05	P76	—	—	—	—	—	—
	INTP11	P77	P110	P77	P110	P77	P06	P77	—	—	—	—	—	—	—
	TxD2	P13	P77	P13	P77	P13	P77	P13	P77	P13	—	P13	—	P13	—
	RxD2	P14	P76	P14	P76	P14	P76	P14	P76	P14	—	P14	—	P14	—
	SCL20	P15	—	P15	—	P15	—	P15	—	P15	—	P15	—	P15	—
	SDA20	P14	—	P14	—	P14	—	P14	—	P14	—	P14	—	P14	—
	SI20	P14	—	P14	—	P14	—	P14	—	P14	—	P14	—	P14	—
	SO20	P13	—	P13	—	P13	—	P13	—	P13	—	P13	—	P13	—
	SCK20	P15	—	P15	—	P15	—	P15	—	P15	—	P15	—	P15	—
PIOR00	TxD0	P51	P17	P51	P17	P51	P17	P51	P17	P51	P17	P51	P17	P51	P17
	RxD0	P50	P16	P50	P16	P50	P16	P50	P16	P50	P16	P50	P16	P50	P16
	SCL00	P30	—	P30	—	P30	—	P30	—	P30	—	P30	—	P30	—
	SDA00	P50	—	P50	—	P50	—	P50	—	P50	—	P50	—	P50	—
	SI00	P50	P16	P50	P16	P50	P16	P50	—	P50	—	P50	—	P50	—
	SO00	P51	P17	P51	P17	P51	P17	P51	—	P51	—	P51	—	P51	—
	SCK00	P30	P55	P30	P55	P30	P55	P30	—	P30	—	P30	—	P30	—
	INTP1	This area cannot be used. Be set to 0 (default value).	P50	P52	P50	P52	This area cannot be used. Be set to 0 (default value).								
	INTP2		P51	P53	P51	P53									
	INTP3		P30	P54	P30	P54									
	INTP4		P31	P55	P31	P55									
	INTP8		P74	P42	P74	P42									
	INTP9		P75	P43	P75	P43									

**Note** This area cannot be used in 256 KB or less code flash memory products. Be set to 0 (default value).

In 384 KB or more code flash memory products, TxD0\_1 can be changed to P12 and RxD0\_1 can be changed to P11. The TxD0 and RxD0 set by the PIOR01 bit are disabled at this time, and TxD0\_1 and RxD0\_1 are enabled.

**Caution 1.** If bit 1 (PIOR01) of the PIOR0 register is set to 1, the TxD2 and RxD2 pins are redirected, but SCL20, SDA20, SI20, SO20, SCK20 pins are not redirected. Therefore, IIC20 and CSI20 cannot be used in its setting. However, even if the bit is set to 1, CSI21/IIC21 can be used if UART2 is not used.**Caution 2.** If bit 1 (PIOR01) of the PIOR0 register is set to 1, the SO00 and SI00 pins are redirected even in the 52-pin or less products, but the SCK00 pin is not redirected. Therefore, CSI00 cannot be used in its setting.

### 4.3.9 Peripheral I/O redirection register 1 (PIOR1)

This register is used to specify whether to enable or disable the peripheral I/O redirect function.

This function is used to switch ports to which alternate functions are assigned.

In addition, the settings for redirection can be changed only until operation of the function is enabled.

The PIOR1 register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Figure 4 - 9 Format of Peripheral I/O redirection register 1 (PIOR1)**

Address: F0079H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
PIOR1	0	0	0	0	PIOR13	PIOR12	PIOR11	PIOR10

PIOR13	PIOR12	Timer RJ TRJIO0 pin select
0	0	Multiplexed with P30/INTP3/RTC1HZ/SCK00/SCL00
0	1	Multiplexed with P50/SI00/RxD0/TOOLRxD/SDA00/TRGIOA
1	0	Multiplexed with P00/TI00/TRGCLKA
1	1	Setting prohibited

PIOR11	PIOR10	Timer RJ TRJIO0 pin select
0	0	Multiplexed with P01/TO00/TRGCLKB
0	1	Multiplexed with P31/TI03/TO03/INTP4
1	0	Multiplexed with P41 only in 44-pin, 48-pin, 52-pin, 64-pin, 80-pin, and 100-pin products
1	1	Multiplexed with P06 only in 64-pin, 80-pin and 100-pin products

### 4.3.10 Global digital input disable register (GDIDIS)

This register is used to prevent through-current flowing to the input buffers of input ports which use EVDD as the power supply when the EVDD power supply is turned off.

When not all of the I/O ports using EVDD as the power supply are used, low power consumption can be achieved by setting the GDIDIS register (setting the GDIDISO bit to 1) to turn off the EVDD power supply.

By setting the GDIDISO bit to 1, input to any input buffer using EVDD as the power supply is prohibited, preventing through-current from flowing when the EVDD power supply is turned off.

The GDIDIS register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Remark** The GDIDIS register is equipped with 64-, 80-, 100-pin products.

**Figure 4 - 10 Format of Global digital input disable register (GDIDIS)**

Address: F007DH      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
GDIDIS	0	0	0	0	0	0	0	GDIDISO
GDIDISO	Setting of input buffers using EVDD power supply							
0	Input to input buffers permitted (default)							
1	Input to input buffers prohibited. No through-current flows to the input buffers.							

Turn off the EVDD power supply with the following procedure.

1. Prohibit input to input buffers (set GDIDISO = 1).
2. Turn off the EVDD power supply.

Turn on again the EVDD power supply with the following procedure.

1. Turn on the EVDD power supply.
2. Permit input to input buffers (set GDIDISO = 0).

**Caution 1. Do not input an input voltage equal to or greater than EVDD to an input port that uses EVDD as the power supply.**

**Caution 2. When input to input buffers is prohibited (GDIDISO = 1), the value read from the port register (Pxx) of a port that uses EVDD as the power supply is 1. When 1 is set in the port output mode register (POMxx) (N-ch open drain output (EVDD tolerance) mode), the value read from the port register (Pxx) is 0.**

**Remark 1.** The GDIDIS register is equipped with 64-, 80-, 100-pin products.

**Remark 2.** Even when input to input buffers is prohibited (GDIDISO = 1), peripheral functions which do not use port functions having EVDD as the power supply can be used.

## 4.4 Port Function Operations

Port operations differ depending on whether the input or output mode is set, as shown below.

### 4.4.1 Writing to I/O port

#### (1) Output mode

A value is written to the output latch by a transfer instruction, and the output latch contents are output from the pin.

Once data is written to the output latch, it is retained until data is written to the output latch again.

The data of the output latch is cleared when a reset signal is generated.

#### (2) Input mode

A value is written to the output latch by a transfer instruction, but since the output buffer is off, the pin status does not change.

Once data is written to the output latch, it is retained until data is written to the output latch again.

The data of the output latch is cleared when a reset signal is generated.

### 4.4.2 Reading from I/O port

#### (1) Output mode

The output latch contents are read by a transfer instruction. The output latch contents do not change.

#### (2) Input mode

The pin status is read by a transfer instruction. The output latch contents do not change.

### 4.4.3 Operations on I/O port

#### (1) Output mode

An operation is performed on the output latch contents, and the result is written to the output latch. The output latch contents are output from the pins.

Once data is written to the output latch, it is retained until data is written to the output latch again.

The data of the output latch is cleared when a reset signal is generated.

#### (2) Input mode

The pin level is read and an operation is performed on its contents. The result of the operation is written to the output latch, but since the output buffer is off, the pin status does not change. Therefore, byte data can be written to the ports used for both input and output.

The data of the output latch is cleared when a reset signal is generated.

#### 4.4.4 Handling different potential (1.8 V, 2.5 V, 3 V) by using EVDD $\leq$ VDD

When connecting an external device operating on a different potential (1.8 V, 2.5 V or 3 V), it is possible to connect the I/O pins of general ports by changing EVDD to accord with the power supply of the connected device.

#### 4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers

It is possible to connect an external device operating on a different potential (1.8 V, 2.5 V or 3 V) by switching I/O buffers with the port input mode register (PIMxx) and port output mode register (POMxx).

When receiving input from an external device with a different potential (1.8 V, 2.5 V or 3 V), set the port input mode registers 0, 1, 3 to 5, 8, and 14 (PIM0, PIM1, PIM3 to PIM5, PIM8, and PIM14) on a bit-by-bit basis to enable normal input (CMOS)/TTL input buffer switching.

When outputting data to an external device with a different potential (1.8 V, 2.5 V or 3 V), set the port output mode registers 0, 1, 3 to 5, 7, 8, and 14 (POM0, POM1, POM3 to POM5, POM7, POM8, and POM14) on a bit-by-bit basis to enable normal output (CMOS)/N-ch open drain (VDD tolerance Note 1/EVDD tolerance Note 2) switching.

The connection of a serial interface is described in the following.

**Note 1.** For 30- to 52-pin products

**Note 2.** For 64- to 100-pin products

- (1) Setting procedure when using input pins of UART0 to UART3, CSI00, CSI01, CSI10, CSI20, CSI30, and CSI31 functions for the TTL input buffer

In case of UART0:	P50 (P16)
In case of UART1:	P03
In case of UART2:	P14 (P76)
In case of UART3:	P143
In case of CSI00:	P30, P50 (P55, P16)
In case of CSI01:	P43, P44
In case of CSI10:	P03, P04 (P80, P81)
In case of CSI20:	P14, P15
In case of CSI30:	P142, P143
In case of CSI31:	P53, P54

**Remark** Functions in parentheses can be assigned via settings in the peripheral I/O redirection register 0 (PIOR0).

- <1> Using an external resistor, pull up the pin to be used to the power supply of the target device (on-chip pull-up resistor cannot be used).
- <2> Set the corresponding bit of the PIM0, PIM1, PIM3 to PIM5, PIM8, and PIM14 registers to 1 to switch to the TTL input buffer. For VIH and Vil, refer to the DC characteristics when the TTL input buffer is selected.
- <3> Enable the operation of the serial array unit and set the mode to the UART/CSI mode.

- (2) Setting procedure when using output pins of UART0 to UART3, CSI00, CSI01, CSI10, CSI20, CSI30, and CSI31 functions in N-ch open-drain output mode

In case of UART0:	P51 (P17)
In case of UART1:	P02
In case of UART2:	P13 (P77)
In case of UART3:	P144
In case of CSI00:	P30, P51 (P55, P17)
In case of CSI01:	P43, P45
In case of CSI10:	P02, P04 (P82, P80)
In case of CSI20:	P13, P15
In case of CSI30:	P142, P144
In case of CSI31:	P52, P54

**Remark** Functions in parentheses can be assigned via settings in the peripheral I/O redirection register 0 (PIOR0).

- <1> Using an external resistor, pull up the pin to be used to the power supply of the target device (on-chip pull-up resistor cannot be used).
- <2> After reset release, the port mode is the input mode (Hi-Z).
- <3> Set the output latch of the corresponding port to 1.
- <4> Set the corresponding bit of the POM0, POM1, POM3 to POM5, POM7, POM8, and POM14 registers to 1 to set the N-ch open drain output (VDD tolerance Note 1/EVDD tolerance Note 2) mode.
- <5> Enable the operation of the serial array unit and set the mode to the UART/CSI mode.
- <6> Set the corresponding bit of the PM0, PM1, PM3 to PM5, PM7, and PM14 registers to the output mode. At this time, the output data is high level, so the pin is in the Hi-Z state.

Note 1. For 30- to 52-pin products

Note 2. For 64- to 100-pin products

- (3) Setting procedure when using I/O pins of IIC00, IIC01, IIC10, IIC20, IIC30, and IIC31 functions with a different potential (1.8 V, 2.5 V, 3 V)

In case of simplified IIC00: P30, P50  
In case of simplified IIC01: P43, P44  
In case of simplified IIC10: P03, P04 (P80, P81)  
In case of simplified IIC11: P10, P11  
In case of simplified IIC20: P14, P15  
In case of simplified IIC30: P142, P143  
In case of simplified IIC31: P53, P54

**Remark** Functions in parentheses can be assigned via settings in the peripheral I/O redirection register 0 (PIOR0).

- <1> Using an external resistor, pull up the pin to be used to the power supply of the target device (on-chip pull-up resistor cannot be used).
- <2> After reset release, the port mode is the input mode (Hi-Z).
- <3> Set the output latch of the corresponding port to 1.
- <4> Set the corresponding bit of the POM0, POM1, POM3 to POM5, POM8, and POM14 registers to 1 to set the N-ch open drain output (VDD tolerance Note 1/EVDD tolerance Note 2) mode.
- <5> Set the corresponding bit of the PIM0, PIM1, PIM3 to PIM5, PIM8, and PIM14 registers to 1 to switch to the TTL input buffer. For VIH and Vil, refer to the DC characteristics when the TTL input buffer is selected.
- <6> Enable the operation of the serial array unit and set the mode to the simplified I<sup>2</sup>C mode.
- <7> Set the corresponding bit of the PM0, PM1, PM3 to PM5, PM8, and PM14 registers to the output mode (data I/O is possible in the output mode). At this time, the output data is high level, so the pin is in the Hi-Z state.

Note 1. For 30- to 52-pin products

Note 2. For 64- to 100-pin products

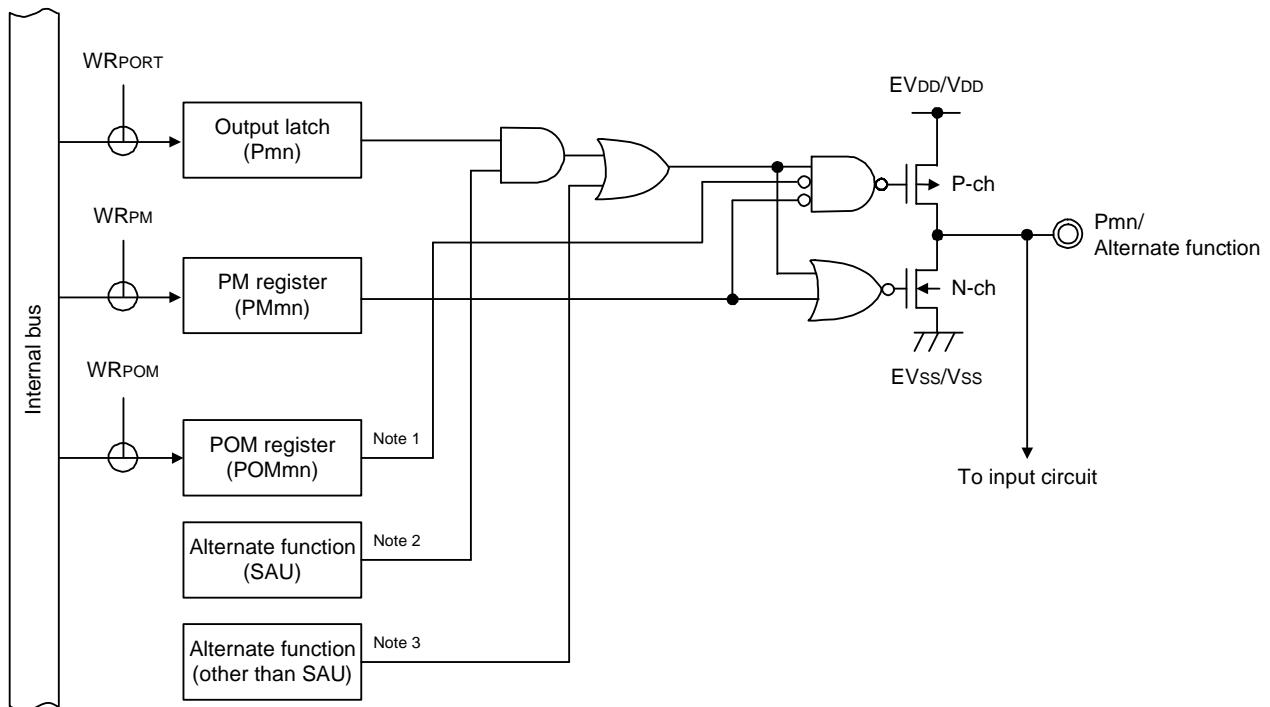
## 4.5 Register Settings When Using Alternate Function

### 4.5.1 Basic concept when using alternate function

In the beginning, for a pin also assigned to be used for analog function, use the ADPC register or port mode control register (PMC $xx$ ) to specify whether to use the pin for analog function or digital input/output.

Figure 4 - 11 shows the basic configuration of an output circuit for pins used for digital input/output. The output of the output latch for the port and the output of the alternate SAU function are input to an AND gate. The output of the AND gate is input to an OR gate. The output of an alternate function other than SAU (Timer, RTC, clock/buzzer output, IICA, etc.) is connected to the other input pin of the OR gate. When such kind of pins are used by the port function or an alternate function, the unused alternate function must not hinder the output of the function to be used. An idea of basic settings for this kind of case is shown in Table 4 - 14.

**Figure 4 - 11 Basic Configuration of Output Circuit for Pins**



**Note 1.** When there is no POM register, this signal should be considered to be low level (0).

**Note 2.** When there is no alternate function, this signal should be considered to be high level (1).

**Note 3.** When there is no alternate function, this signal should be considered to be low level (0).

**Table 4 - 14 Concept of Basic Settings**

Output Function of Used Pin	Output Settings of Unused Alternate Function		
	Output Function for Port	Output Function for SAU	Output Function for other than SAU
Output function for port	—	Output is high (1)	Output is low (0)
Output function for SAU	High (1)	—	Output is low (0)
Output function for other than SAU	Low (0)	Output is high (1)	Output is low (0) Note

**Note** Since more than one output function other than SAU may be assigned to a single pin, the output of an unused alternate function must be set to low level (0). For details on the setting method, see **4.5.2 Register settings for alternate function whose output function is not used**.

## 4.5.2 Register settings for alternate function whose output function is not used

When the output of an alternate function of the pin is not used, the following settings should be made. Note that when the peripheral I/O redirection function is the target, the output can be switched to another pin by setting the peripheral I/O redirection register 0, 1 (PIOR0, 1). This allows usage of the port function or other alternate function assigned to the target pin.

- (1) SOp = 1, TxDq = 1 (settings when the serial output (SOp/TxDq) of SAU is not used)

When the serial output (SOp/TxDq) is not used, such as, a case in which only the serial input of SAU is used, set the bit in serial output enable register m (SOEm) which corresponds to the unused output to 0 (output disabled) and set the SOmn bit in serial output register m (SOM) to 1 (high). These are the same settings as the initial state.

- (2) SCKp = 1, SDAr = 1, SCLr = 1 (settings when channel n in SAU is not used)

When SAU is not used, set bit n (SEmn) in serial channel enable status register m (SEM) to 0 (operation stopped state), set the bit in serial output enable register m (SOEm) which corresponds to the unused output to 0 (output disabled), and set the SOmn and CKOmn bits in serial output register m (SOM) to 1 (high). These are the same settings as the initial state.

- (3) TOmn = 0 (settings when the output of channel n in TAU is not used)

When the TOmn output of TAU is not used, set the bit in timer output enable register 0 (TOE0) which corresponds to the unused output to 0 (output disabled) and set the bit in timer output register 0 (TO0) to 0 (low). These are the same settings as the initial state.

- (4) SDAAn = 0, SCLAn = 0 (setting when IICA is not used)

When IICA is not used, set the IICEn bit in IICA control register n0 (IICCTLn0) to 0 (operation stopped). This is the same setting as the initial state.

- (5) PCLBUZ<sub>n</sub> = 0 (setting when clock/buzzer output is not used)

When the clock/buzzer output is not used, set the PCLOEn bit in clock output select register n (CKS<sub>n</sub>) to 0 (output disabled). This is the same setting as the initial state.

- (6) TRJIO0 = 0/TRJO0 = 0 (setting when timer RJ output is not used)

When the pulse output function of timer RJ is not used with the TRJO0 pin, set bit 2 (TOENA) in the timer RJ I/O control register (TRJIOC0) to 0 (TRJO output disabled). This is the same setting as the initial state.

When the TRJIO0 pin of timer RJ is not used for the output function, set bits 2 to 0 (TMOD2 to TMOD0) in timer RJ mode register 0 (TRJMR0) to a value other than 001b (pulse output mode). The initial value is 000b (timer mode).

- (7) TRDIOAn = 0/TRDIOBn = 0/TRDIOCn = 0/TRDIODn = 0 (setting when timer RD output is not used)

When the output function of timer RD is not used, set the pins not used for timer RD output function to "output disabled" using timer RD output master enable register 1 (TRDOER1). This is the same setting as the initial state.

- (8) TRGIOA = 0/TRGIOB = 0 (setting when timer RG output is not used)

When the output function of timer RG is not used, set the pins not used for timer RG output function to "pin output by compare match is disabled" using the timer RG I/O control register (TRGIOR). This is the same setting as the initial state.

### 4.5.3 Register setting examples for used port and alternate functions

Register setting examples for used port and alternate functions are shown in Tables 4 - 15 to 4 - 20. The registers used to control the port functions should be set as shown in Tables 4 - 15 to 4 - 20. See the following remark for legends used in Tables 4 - 15 to 4 - 20.

<b>Remark</b>	—:	Not supported
	x:	Don't care
PIORx:		Peripheral I/O redirection register
POMxx:		Port output mode register
PMCxx:		Port mode control register
PMxx:		Port mode register
Pxx:		Port output latch
		Functions in parentheses can be assigned via settings in the peripheral I/O redirection register 0, 1 (PIOR0, 1).

**Table 4 - 15 Setting Examples of Registers When Using P00 to P17 Pin Function (1/3)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P00	P00	Input	—	x	0 Note 1	1	x	x	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0 Note 1	0	0/1	TxD1 = 1 Note 2	(TRJ00) = 0 Note 3										
		N-ch OD output		1	0 Note 1	0	0/1												
	ANI17	Analog input	—	x	1	1	x	x	—	✓	✓	x	x	x	x	x	x	x	
	TI00	Input	—	x	0 Note 1	1	x	x	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TxD1	Output	x	0/1	0 Note 1	0	1	—	(TRJ00) = 0 Note 3	✓	✓	✓	✓	✓	x	x	x	x	
	TRGCLKA	Input	—	x	0 Note 1	1	x	x		✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(TRJ00)	Output	PIOR13, PIOR12 = 10B	0	0 Note 1	0	0	TxD1 = 1 Note 2	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P01	P01	Input	—	—	0 Note 1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	—	0 Note 1	0	0/1	—	TO00 = 0, TRJIO = 0 Note 4										
	ANI16	Analog input	—	—	1	1	x	—		✓	✓	x	x	x	x	x	x	x	
	TO00	Output	x	—	0 Note 1	0	0	—	TRJIO = 0 Note 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	RxD1	Input	—	—	0 Note 1	1	x	—		✓	✓	✓	✓	✓	✓	x	x	x	
	TRGCLKB	Input	—	—	0 Note 1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRJIO0	Input	PIOR13, PIOR12 = 00B	—	0 Note 1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0 Note 1	0	0	—	TO00 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P02	P02	Input	—	x	0	1	x	x	—	Tx1/SO10 = 1 Note 5	x	x	x	x	x	✓	✓	✓	
		Output	x	0	0	0	0/1												
		N-ch OD output		1	0	0	0/1												
	ANI17	Analog input	—	x	1	1	x	x	—	x	x	x	x	x	x	✓	✓	✓	
	TxD1	Output	PIOR05 = 0 Note 6	0/1	0	0	1	—	—	x	x	x	x	x	x	✓	✓	✓	
	SO10	Output		0/1	0	0	1	—	—	x	x	x	x	x	x	✓	✓	✓	
P03	P03	Input	—	x	0	1	x	x	—	SDA10 = 1 Note 5	x	x	x	x	x	✓	✓	✓	
		Output	x	0	0	0	0/1												
		N-ch OD output		1	0	0	0/1												
	ANI16	Analog input	—	x	1	1	x	x	—	x	x	x	x	x	x	✓	✓	✓	
	SI10	Input	PIOR05 = 0 Note 6	x	0	1	x	x	—	x	x	x	x	x	x	✓	✓	✓	
	RxD1	Input		x	0	1	x	x	—	x	x	x	x	x	x	✓	✓	✓	
	SDA10	I/O	PIOR05 = 0 Note 6	1	0	0	1	—	—	x	x	x	x	x	x	✓	✓	✓	
P04	P04	Input	—	x	—	1	x	—	—	SCK10/SCL 10 = 1 Note 5	x	x	x	x	x	x	✓	✓	
		Output	x	0	—	0	0/1												
		N-ch OD output		1	—	0	0/1												
	SCK10	Input	PIOR05 = 0 Note 6	x	—	1	x	—	—		x	x	x	x	x	x	✓	✓	
	Output			0/1	—	0	1	—	—		x	x	x	x	x	x	✓	✓	
	SCL10	Output	PIOR05 = 0 Note 6	0/1	—	0	1	—	—		x	x	x	x	x	x	✓	✓	
P05	P05	Input	—	—	—	1	x	—	—	(TRJIO) = 0 Note 7	x	x	x	x	x	x	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	(INTP10)	Input	PIOR01 = 1	—	—	1	x	—	—		x	x	x	x	x	x	✓	x	
P06	P06	Input	—	—	—	1	x	—	—	(TRJIO) = 0 Note 7	x	x	x	x	x	x	✓	✓	
		Output	x	—	—	0	0/1	—	—										
	(TRJIO0)	Input	PIOR13, PIOR12 = 11B	—	—	1	x	—	—	(TRJIO) = 0 Note 7	x	x	x	x	x	x	✓	✓	
	Output			—	—	0	0	—	—		x	x	x	x	x	x	✓	✓	
	(INTP11)	Input	PIOR01 = 1	—	—	1	x	—	—		x	x	x	x	x	x	✓	x	

**Table 4 - 15 Setting Examples of Registers When Using P00 to P17 Pin Function (2/3)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P10	P10	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	0	—	0	0/1	SCK11/SCL 11 = 1	TRDIOD1 = 0										
		N-ch OD output	—	1	—	0	0/1												
	SCK11	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	0/1	—	0	1	—	TRDIOD1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SCL11	Output	—	0/1	—	0	1	—	TRDIOD1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOD1	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	0	—	0	0	SCK11/SCL 11 = 1	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P11	P11	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	0	—	0	0/1	SDA11 = 1	TRDIOC1 = 0										
		N-ch OD output	—	1	—	0	0/1												
	SI11	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SDA11	I/O	—	x	—	0	1	—	TRDIOC1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOC1	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	0	—	0	0	SDA11 = 1	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(RxDO_1) Note 13	Input	PIOR06 = 1	—	—	1	x	—	—	—	—	—	—	—	✓	✓	✓	✓	
P12	P12	Input	—	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0	0	0/1	SO11 = 0	TRDIOB1 = 0										
	SO11	Output	—	—	0	0	1	—	TRDIOB1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOB1	Input	—	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0	0	0	SO11 = 0	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	IVREF1 Note 8	Input	—	—	1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P13	P13	Input	—	x	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0	0	0/1	TxD2/SO20 = 1 Note 10	—										
		N-ch OD output	—	1	0	0	0/1												
	TxD2	Output	PIOR01 = 0	0/1	0	0	1	—	TRDIOA1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SO20	Output	PIOR01 = 0	0/1	0	0	1	—	TRDIOA1 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOA1	Input	—	x	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0	0	0	TxD2/SO20 = 1 Note 10	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	IVCMP1 Note 8	Input	—	x	1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P14	P14	Input	—	x	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0	0	0/1	SDA20 = 1 Note 10	TRDIODO0 = 0, (SCLA0) = 0 Note 11	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		N-ch OD output	—	1	0	0	0/1			✓	✓	✓	✓	✓	✓	✓	✓	✓	
	RxD2	Input	PIOR01 = 0	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SI20	Input	PIOR01 = 0	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SDA20	I/O	PIOR01 = 0	1	—	0	1	—	TRDIODO0 = 0, (SCLA0) = 0 Note 11	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOD0	Input	—	x	—	1	x	—		✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	1	—	0	0	SDA20 = 1 Note 10	(SCLA0) = 0 Note 11	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(SCLA0)	I/O	PIOR02 = 1	1	—	0	0	SDA20 = 1 Note 10	TRDIOD0 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	

**Table 4 - 15 Setting Examples of Registers When Using P00 to P17 Pin Function (3/3)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P15	P15	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	—	0	0/1	SCK20/SCL 20 = 1 Note 10	TRDIOB0 = 0, (SDAA0) = 0 Note 11, PCLBUZ1 = 0 Note 9	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		N-ch OD output		1	—	0	0/1			✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SCK20	Input	PIOR01 = 0	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output		0/1	—	0	1	—	TRDIOB0 = 0, (SDAA0) = 0 Note 11, PCLBUZ1 = 0 Note 9	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	SCL20	Output	PIOR01 = 0	0/1	—	0	1	—		✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOB0	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	—	0	0	SCK20/SCL 20 = 1	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(SDAA0)	I/O	PIOR02 = 1	1	—	0	0	SCK20/SCL 20 = 1	TRDIOB0 = 0, PCLBUZ1 = 0 Note 9	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	PCLBUZ1	Output	—	0	—	0	0	SCK20/SCL 20 = 1	TRDIOB0 = 0, (SDAA0) = 0 Note 11	✓	✓	✓	✓	✓	✓	x	x	x	
P16	P16	Input	—	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0	0	0/1	—	TO01 = 0, TRDIOC0 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TI01	Input	—	—	0	1	x	—		✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TO01	Output	—	—	0	0	0	—	TRDIOC0 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	INTP5	Input	PIOR04 = 0	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIOC0	Input	—	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0	0	0	—	TO01 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	IVREF0 Note 8	Input	—	—	1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(SI00)	Input	PIOR01 = 1	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(RxD0)	Input	PIOR01 = 1, PIOR06 = 0	—	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P17	P17	Input	—	x	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0	0	0/1	(TxD0)/(SO0 0) = 1 Note 12	TO02 = 0, TRDIA0A = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		N-ch OD output		1	0	0	0/1			✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TI02	Input	—	0	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TO02	Output	x	0	0	0	0	(TxD0)/ (SO00) = 1 Note 12	TRDIA0A = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRDIA0A	Input	—	x	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	0	0	0	TO02 = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	TRDCLK	Input	—	0	0	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	IVCMPO Note 8	Input	—	0	1	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(SO00)	Output	PIOR01 = 1	0/1	0	0	1	—	TO02 = 0, TRDIA0A = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(TxD0)	Output	PIOR01 = 1, PIOR06 = 0	0/1	0	0	1	—	TO02 = 0, TRDIA0A = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	

- Note 1.** 30 and 32-pin products only.
- Note 2.** 30 to 48-pin products only.
- Note 3.** Don't care when PIOR13 and PIOR12 ≠ 10B.
- Note 4.** Don't care when PIOR13 and PIOR12 ≠ 00B.
- Note 5.** Don't care when PIOR05 = 1.
- Note 6.** 100-pin products only.
- Note 7.** Don't care when PIOR13 and PIOR12 ≠ 11B.
- Note 8.** Products with 96 KB or more code flash memory only.
- Note 9.** 30 to 52-pin products only.
- Note 10.** Don't care when PIOR01 = 1.
- Note 11.** Don't care when PIOR02 = 0.
- Note 12.** Don't care when PIOR01 = 0.
- Note 13.** Products with 384 KB or more code flash memory only.

**Table 4 - 16 Setting Examples of Registers When Using P20 to P27 Pin Function**

Pin Name	Used Function		ADPC	ADM2	PMxx	Pxx										
	Function Name	I/O					30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
P20	P20	Input	ADPC = 01H	x	1	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H	x	0	0/1										
	ANIO	Analog input	ADPC = 00H/02H to 0FH	00x0xx0x, 10x0xx0x	1	x										
	AVREFP	Reference voltage input	ADPC = 00H/02H to 0FH	01x0xx0x	1	x										
P21	P21	Input	ADPC = 01H/02H	x	1	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H/02H	x	0	0/1										
	ANII	Analog input	ADPC = 00H/03H to 0FH	xx00xx0x	1	x										
	AVREFM	Reference voltage input	ADPC = 00H/03H to 0FH	xx10xx0x	1	x										
P22	P22	Input	ADPC = 01H to 03H	x	1	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H to 03H	x	0	0/1										
	ANI2	Analog input	ADPC = 00H/04H to 0FH	x	1	x										
	ANO0 Note	Analog output	ADPC = 00H/04H to 0FH	x	1	x										
P23	P23	Input	ADPC = 01H to 04H	x	1	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H to 04H	x	0	0/1										
	ANI3	Analog input	ADPC = 00H/05H to 0FH	x	1	x										
	ANO1 Note	Analog output	ADPC = 00H/05H to 0FH	x	1	x		x	✓	✓	✓	✓	✓	✓	✓	✓
P24	P24	Input	ADPC = 01H to 05H	x	1	x	x	x	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H to 05H	x	0	0/1										
	ANI4	Analog input	ADPC = 00H/06H to 0FH	x	1	x										
	ANI5	Analog input	ADPC = 00H/07H to 0FH	x	1	x										
P25	P25	Input	ADPC = 01H to 06H	x	1	x	x	x	✓	✓	✓	✓	✓	✓	✓	✓
		Output	ADPC = 01H to 06H	x	0	0/1										
	ANI6	Analog input	ADPC = 00H/08H to 0FH	x	1	x										
	ANI7	Analog input	ADPC = 00H/09H to 0FH	x	1	x		x	x	x	x	✓	✓	✓	✓	✓

**Note** Products with 96 KB or more code flash memory only.

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (1/6)**

Pin Name	Used Function		PIORx	POMxx	PMCx x	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P30	P30	Input	—	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	x	0	—	0	0/1	SCK00/SCL00 = 1 Note 1	RTC1HZ = 0, TRJ00 = 0 Note 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		N-ch OD output		1	—	0	0/1			✓	✓	✓	✓	✓	✓	✓	✓	✓	
	INTP3	Input	PIOR00 = 0 Note 13, PIOR05 = 0 Note 14	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	RTC1HZ	Output	x	0	—	0	0	SCK00/SCL00 = 1 Note 1	—	x x x	✓	✓	✓	✓	✓	✓	✓	✓	
	SCK00	Input	PIOR01 = 0	x	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output		0/1	—	0	0	—	RTC1HZ = 0, TRJ00 = 0 Note 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P31	SCL00	Output	PIOR01 = 0	0/1	—	0	0	—	RTC1HZ = 0, TRJ00 = 0 Note 2	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TRJ00	Output	PIOR13, PIOR12 = 00B	0	—			SCK00/SCL00 = 1 Note 1	RTC1HZ = 0	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	P31	Input		—	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	Output	x	—	—	0	0/1	—	TO03 = 0, PCLBUZ0 = 0 Note 3, (PCLBUZ0) = 0 Notes 4, 5, (TRJIO0) = 0 Note 6	✓	✓	✓	✓	✓	✓	✓	✓	✓		
	TI03	Input	—	—	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	TO03	Output	x	—	—	0	0	—	PCLBUZ0 = 0 Note 3, (PCLBUZ0) = 0 Notes 4, 5, (TRJIO0) = 0 Note 6	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	INTP4	Input	PIOR00 = 0 Note 13, PIOR05 = 0 Note 14	—	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	(TRJIO0)	Input	PIOR11, PIOR10 = 01B	—	—	1	x	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		Output		—	—	0	0	—	TO03 = 0, PCLBUZ0 = 0 Note 3, (PCLBUZ0) = 0 Notes 4, 5	✓	✓	✓	✓	✓	✓	✓	✓	✓	
	PCLBUZ0	Output	—	—	0	0	—	TO03 = 0, (TRJIO0) = 0 Note 6	✓	✓	✓	✓	x x	x x	x x	x x	x x		
P40	P40	(PCLBUZ0)	Output	PIOR03 = 1	—	—	0	0	—	TO03 = 0, (TRJIO0) = 0 Note 6	x x x	x x x	x x x	✓	✓	✓	✓	✓	
		SSI00	Input	—	—	—	1	x	—	—	✓	x x	x x	x x	x x	x x	x x	x x	
	P41	P41	Input	—	—	—	1	x	—	—	x x x	x x x	x x x	✓	✓	✓	✓	✓	
		Output	x	—	—	0	0/1	—	(TRJIO0) = 0 Note 7	✓	✓	✓	✓	✓	✓	✓	✓	✓	
P42	P42	(TRJIO0)	Input	PIOR11, PIOR10 = 10B	—	—	1	x	—	—	x x x	x x x	x x x	✓	✓	✓	✓	✓	
			Output		—	—	0	0	—	x x x	x x x	x x x	✓	✓	✓	✓	✓		
	(INTP8)	Input	PIOR00 = 1	—	—	1	x	—	—	x x x	x x x	x x x	x x x	✓	✓	✓	✓	x	

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (2/6)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P43	P43	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	0	—	0	0/1	SCK01/SCL01 = 1	—										
		N-ch OD output	—	1	—	0	0/1		—							x			
	SCK01	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	0/1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
	SCL01	Output	—	0/1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
		(INTP9)	Input	PIOR00 = 1	x	—	1	x	—	—	x	x	x	x	x	x	v	v	x
P44	P44	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	0	—	0	0/1	SDA01 = 1	—										
		N-ch OD output	—	1	—	0	0/1		—										
	SI01	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		SDA01	I/O	—	1	—	0	1	—	x	x	x	x	x	x	x	v	v	v
P45	P45	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	0	—	0	0/1	SO01 = 1	—										
		N-ch OD output	—	1	—	0	0/1		—										
	SO01	Output	—	0/1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
P46	P46	Input	—	—	—	1	x	—	—	x	x	x	x	x	x	x	x	x	v
		Output	—	—	—	0	0/1	—	—										
	INTP1	Input	PIOR05 = 0	—	—	1	x	—	—	x	x	x	x	x	x	x	x	x	v
P47	P47	Input	—	—	—	1	x	—	—	x	x	x	x	x	x	x	x	x	v
		Output	—	—	—	0	0/1	—	—										
	INTP2	Input	—	—	—	1	x	—	—	x	x	x	x	x	x	x	x	x	v
P50	P50	Input	—	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	v
		Output	x	0	—	0	0/1	SDA00 = 1 Note 1	TRGIOA = 0, (TRJ00) = 0 Note 9										
		N-ch OD output	1	—	0	0/1	—												
	INTP1	Input	PIOR00 = 0	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	x
	SI00	Input	PIOR01 = 0	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	v
	RxD0	Input	PIOR01 = 0, PIOR06 = 0	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	v
	SDA00	I/O	PIOR01 = 0	1	—	0	1	—	TRGIOA = 0, (TRJ00) = 0 Note 9	v	v	v	v	v	v	v	v	v	
	TRGIOA	Input	—	x	—	1	x	—		v	v	v	v	v	v	v	v	v	v
	Output	—	0	—	0	0	0	SDA00 = 1 Note 1	(TRJ00) = 0 Note 9	v	v	v	v	v	v	v	v	v	
	(TRJ00)	Output	PIOR13, PIOR12 = 01B	0	—	0	0	SDA00 = 1 Note 1	TRGIOA = 0	v	v	v	v	v	v	v	v	v	
P51	P51	Input	—	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	v
		Output	—	0	—	0	0/1	TxDO/SO00 = 1 Note 1	TRGIOB = 0										
		N-ch OD output	1	—	0	0/1	—												
	INTP2	Input	—	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v	x
	SO00	Output	PIOR01 = 0	0/1	—	0	1	—	TRGIOB = 0	v	v	v	v	v	v	v	v	v	
	TxD0	Output	PIOR01 = 0, PIOR06 = 0	0/1	—	0	1	—	TRGIOB = 0	v	v	v	v	v	v	v	v	v	
P52	P52	TRGIOB	Input	—	x	—	1	x	—	—	v	v	v	v	v	v	v	v	v
		Output	—	0	—	0	0	0	TxDO/SO00 = 1 Note 1	—									
		N-ch OD output	1	—	0	0/1	—								x				
	SO31	Output	—	0/1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
	(INTP1)	Input	PIOR00 = 1	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	x

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (3/6)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P53	P53	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	x	0	—	0	0/1	SDA31 = 1	—										
		N-ch OD output		1	—	0	0/1									x			
	SI31	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
	SDA31	I/O	—	1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
P54	P54	(INTP2)	Input	PIOR00 = 1	x	—	1	x	—	—	x	x	x	x	x	x	v	v	x
		Input	—	x	—	1	x	—	—	x	x	x	x	x	x	v	v	v	v
		Output	—	0	—	0	0/1	SCK31/SCL31 = 1	—										
		N-ch OD output		1	—	0	0/1									x			
	SCK31	Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
	Output	—	0/1	—	0	1	—	—	—	x	x	x	x	x	x	x	v	v	v
	SCL31	Output	—	0/1	—	0	1	—	—	x	x	x	x	x	x	x	v	v	v
P55	P55	(INTP3)	Input	PIOR00 = 1	x	—	1	x	—	—	x	x	x	x	x	x	v	v	x
		Input	—	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	0	—	0	0/1	(SCK00) = 1 Note 9	(PULBUZ1) = 0 Note 10	x	x	x	x	x	x	x	v	v	v
		N-ch OD output		1	—	0	0/1			x	x	x	x	x	x	x	v	v	v
	(INTP4)	Input	PIOR00 = 1	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	x
	(PULBUZ1)	Output	PIOR04 = 1	0	—	0	0	(SCK00) = 1 Note 9	—	x	x	x	x	x	x	x	v	v	v
	(SCK00)	Input	PIOR01 = 1	x	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	0/1	—	0	1	—	—	(PULBUZ1) = 0 Note 10	x	x	x	x	x	x	x	v	v	v
P56	P56	Input	—	—	1	x	—	—	—	x	x	x	x	x	x	x	x	x	v
		Output	—	—	0	0/1	—	—	—										
		(INTP1)	Input	PIOR05 = 1	—	—	1	x	—	—	x	x	x	x	x	x	x	x	v
P57	P57	Input	—	—	1	x	—	—	—	x	x	x	x	x	x	x	x	x	v
		Output	—	—	0	0/1	—	—	—										
		(INTP3)	Input	PIOR05 = 1	—	—	1	x	—	—	x	x	x	x	x	x	x	x	v
P60	P60	Input	—	—	1	x	—	—	—	v	v	v	v	v	v	v	v	v	v
		N-ch OD output (6 V tolerance)	x	—	—	0	0/1	—	SCLA0 = 0 Note 11										
		SCLA0	I/O	PIOR02 = 0	—	—	0	0		—	—	—	—	—	—	—	—	—	—
P61	P61	Input	—	—	1	x	—	—	—	v	v	v	v	v	v	v	v	v	v
		N-ch OD output (6 V tolerance)	x	—	—	0	0/1	—	SDAA0 = 0 Note 11										
		SDAA0	I/O	PIOR02 = 0	—	—	0	0		—	—	—	—	—	—	—	—	—	—
P62	P62	Input	—	—	1	x	—	—	—	x	v	v	v	v	v	v	v	v	v
		N-ch OD output (6 V tolerance)	x	—	—	0	0/1	—	—										
	SSI00	Input	—	—	1	x	—	—	—	x	v	v	v	v	v	v	v	v	v
	SCLA1	I/O	—	—	—	0	0	—	—	x	x	x	x	x	x	x	v	v	v
P63	P63	Input	—	—	1	x	—	—	—	x	x	x	x	v	v	v	v	v	v
		N-ch OD output (6 V tolerance)	—	—	—	0	0/1	—	SDAA1 = 0										
	SDAA1	I/O	—	—	—	0	0	—		—	x	x	x	x	x	x	x	v	v
P64	P64	Input	—	—	—	1	x	—	—	x	x	x	x	x	x	x	v	v	v
		Output	—	—	—	0	0/1	—	TO10 = 0										
	TI10	Input	—	—	—	1	x	—		—	x	x	x	x	x	x	x	v	v
	TO10	Output	—	—	—	0	0	—		—	x	x	x	x	x	x	x	v	v

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (4/6)**

Pin Name	Used Function		PIORx	POMxx	PMCx x	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P65	P65	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
		Output	—	—	—	0	0/1	—	TO11 = 0										
	TI11	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
	TO11	Output	—	—	—	0	0	—	—	x	x	x	x	x	x	x	✓	✓	
P66	P66	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	TI12	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
	TO12	Output	—	—	—	0	0	—	TO12 = 0	x	x	x	x	x	x	x	✓	✓	
P67	P67	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	TI13	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	✓	
	TO13	Output	—	—	—	0	0	—	TO13 = 0	x	x	x	x	x	x	x	✓	✓	
P70	P70	Input	—	—	—	1	×	—	—	x	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	KR0	Input	—	—	—	1	×	—	—	x	x	x	✓	✓	✓	✓	✓	✓	
	SCK21	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
	SCL21	Input	—	—	—	0	1	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
P71	P71	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	SDA21 = 1	—										
		N-ch OD output	—	—	—	1	—	0	0/1										
	KR1	Input	—	—	—	1	×	—	—	x	x	x	✓	✓	✓	✓	✓	✓	
	SI21	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
	SDA21	Output	—	—	—	1	—	0	1	x	x	✓	✓	✓	✓	✓	✓	✓	
P72	P72	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	SO21 = 1	—										
	KR2	Input	—	—	—	1	×	—	—	x	x	x	○	○	○	○	○	○	
	SO21	Output	—	—	—	0	1	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
P73	P73	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	KR3	Input	—	—	—	1	×	—	—	x	x	✓	✓	✓	✓	✓	✓	✓	
	SO01	Output	—	—	—	0	1	—	—	x	x	x	x	x	✓	✓	✗	✗	
P74	P74	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	SDA01 = 1	—										
		N-ch OD output	—	—	—	1	—	0	0/1										
	KR4	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
	INTP8	Input	PIOR00 = 0 Note 13, PIOR05 = 0 Note 14	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
	SI01	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✗	
	SDA01	I/O	—	—	—	0	1	—	—	x	x	x	x	x	✓	✓	✗	✗	
P75	P75	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	SCK01/SCL01 = 1	—										
	KR5	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
	INTP9	Input	PIOR00 = 0 Note 13, PIOR05 = 0 Note 14	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
	SCK01	Input	—	—	—	1	×	—	—	x	x	x	x	x	✓	✓	✓	✓	
	Output	—	—	—	0	1	—	—	x	x	x	x	x	✓	✓	✓	✓	✓	
	SCL01	Output	—	—	—	0	1	—	—	x	x	x	x	x	✓	✓	✓	✓	

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (5/6)**

Pin Name	Used Function		PIORx	POMxx	PMCx x	PMxx	Pxx	Alternate Function Output											
	Function Name	I/O						SAU Output Function		Other than SAU									
										30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
P76	P76	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
		Output	—	—	—	0	0/1	—	—										
	KR6	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
	INTP10	Input	PIOR01 = 0	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
	(RxD2)	Input	PIOR01 = 1	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
P77	P77	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
		Output	×	—	—	0	0/1	(TxD2) = 1 Note 8	—										
	KR7	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
	INTP11	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	✓	✓	✓	
	(TxD2)	Output	PIOR01 = 1	—	—	0	1	—	—	x	x	x	x	x	x	✓	✓	✓	
P80	P80	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	(SCK01)/ (SCL01) = 1 Note 12	—										
		N-ch OD output	—	—	—	0	0/1		—										
	(SCK10)	Input	PIOR05 = 1	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output		—	—	0	1	—	—	x	x	x	x	x	x	x	x	✓	
	(SCL10)	Output	PIOR05 = 1	—	—	0	1	—	—	x	x	x	x	x	x	x	x	✓	
P81	P81	Input	—	0	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	0	—	0	0/1	(SDA10) = 1 Note 12	—										
		N-ch OD output	—	1	—	0	0/1		—										
	(SI10)	Input	PIOR05 = 1	×	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
	(RxD1)	Input	PIOR05 = 1	×	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
	(SDA10)	I/O	PIOR05 = 1	1	—	0	1	—	—	x	x	x	x	x	x	x	x	✓	
P82	P82	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	(SO10)/(TxD1) = 1 Note 12	—										
		N-ch OD output	—	—	—	0	0/1		—										
	(SO10)	Output	PIOR05 = 1	0/1	—	0	1	—	—	x	x	x	x	x	x	x	x	✓	
	(TxD1)	Output	PIOR05 = 1	0/1	—	0	1	—	—	x	x	x	x	x	x	x	x	✓	
P83	P83	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
P84	P84	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	—	—										
P85	P85	Input	PIOR05 = 1	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	—	—	x	x	x	x	x	x	x	x	✓	
P86	P86	(INTP7)	Input	PIOR05 = 1	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	
		Input	—	—	—	0	0/1	—	—	x	x	x	x	x	x	x	x	✓	
		(INTP8)	Input	PIOR05 = 1	—	—	1	×	—	—	x	x	x	x	x	x	x	✓	
P87	P87	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	—	—										
		(INTP9)	Input	PIOR05 = 1	—	0	1	×	—	x	x	x	x	x	x	x	x	✓	
P100	P100	Input	—	—	0	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	0	0	0/1	—	—										
		AN120	Analog input	—	—	1	1	×	—	x	x	x	x	x	x	x	x	✓	
		(INTP10)	Input	PIOR01 = 1	—	0	1	0/1	—	x	x	x	x	x	x	x	x	✓	
P101	P101	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	
		Output	—	—	—	0	0/1	—	—										

**Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (6/6)**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P102	P102	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	x	✓
		Output	—	—	—	0	0/1	—	—										
P110	P110	Input	—	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	✓
		Output	—	—	—	0	0/1	—	—										
P111	P111	Input	PIOR01 = 1	—	—	1	×	—	—	x	x	x	x	x	x	x	x	✓	✓
		Output	—	—	—	0	0/1	—	—										
P120	P120	Input	—	—	0	1	×	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Output	—	—	0	0	0/1	—	—										
	AN19	Analog input	—	—	1	1	×	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Note 15	VCOUT0	Analog output	—	—	0	0	0	—	—	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

**Note 1.** Don't care when PIOR01 = 1.

**Note 2.** Don't care when PIOR13 and PIOR12 ≠ 00B.

**Note 3.** 30 to 44-pin products only.

**Note 4.** 48 to 100-pin products only.

**Note 5.** Don't care when PIOR03 = 0.

**Note 6.** Don't care when PIOR11 and PIOR10 ≠ 01B.

**Note 7.** Don't care when PIOR11 and PIOR10 ≠ 10B.

**Note 8.** Don't care when PIOR01 = 0.

**Note 9.** Don't care when PIOR13 and PIOR12 ≠ 01B.

**Note 10.** Don't care when PIOR04 = 0.

**Note 11.** Don't care when PIOR02 = 1.

**Note 12.** Don't care when PIOR05 = 0.

**Note 13.** 64 and 80-pin products only.

**Note 14.** 100-pin products only.

**Note 15.** Products with 96 KB or more code flash memory only.

**Table 4 - 18 Setting Examples of Registers When Using P121 to P124 Pin Function**

Pin Name	Used Function		CMC (EXCLK, OSCSEL, EXCLKS, OSCSELS)				Pxx	30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin	
	Function Name	I/O																
P121	P121	Input	00xx/10xx/11xx				x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	X1	—	01xx				—											
P122	P122	Input	00xx/10xx/11xx				x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	X2	—	01xx				—											
	EXCLK	Input	11xx				—											
P123	P123	Input	xx00/xx10/xx11				x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	XT1	—	xx01				—											
P124	P124	Input	xx00/xx10/xx11				x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	XT2	—	xx01				—											
	EXCLKS	Input	xx11				—											

**Table 4 - 19 Setting Examples of Registers When Using P130 to P147 Pin Function**

Pin Name	Used Function		PIORx	POMxx	PMCxx	PMxx	Pxx	Alternate Function Output		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
	Function Name	I/O						SAU Output Function	Other than SAU										
P130	P130	Output	—	—	—	—	0/1	—	—	✗	✗	✗	✗	✗	✓	✓	✓	✓	
P137	P137	Input	—	—	—	—	—	✗	—	—	✓	✓	✓	✓	✓	✓	✓	✓	
	INTP0	Input	—	—	—	—	—	✗	—	—	✓	✓	✓	✓	✓	✓	✓	✓	
P140	P140	Input	—	—	—	1	✗	—	—	—	✗	✗	✗	✗	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	—	PCLBUZ0 = 0	—	—	—	—	—	✓	✓	✓	✓	
	PCLBUZ0	Output	PIOR03 = 0	—	—	0	0	—	—	—	✗	✗	✗	✗	✓	✓	✓	✓	
P141	P141	Input	—	—	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✓	✓	
		Output	—	—	—	0	0/1	—	PCLBUZ1 = 0	—	—	—	—	—	✓	✓	✓	✓	
	PCLBUZ1	Output	PIOR04 = 0	—	—	0	0	—	—	—	✗	✗	✗	✗	✗	✗	✓	✓	
	INTP7	Input	PIOR05 = 0	—	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✓	✓	
P142	P142	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
		Output	—	0	—	0	0/1	SCK30/ SCL30 = 1	—	—	—	—	—	—	—	—	—		
		N-ch OD output	—	1	—	0	0/1		—	—	—	—	—	—	—	—	—		
	SCK30	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
		Output	—	0/1	—	0	0	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
	SCL30	Output	—	0/1	—	0	1	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
P143	P143	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
		Output	—	0	—	0	0/1	SDA30 = 1	—	—	—	—	—	—	—	—	—		
		N-ch OD output	—	1	—	0	0/1		—	—	—	—	—	—	—	—	—		
	SI30	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
	RxD3	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
	SDA30	I/O	—	1	—	0	1	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
P144	P144	Input	—	✗	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
		Output	—	0	—	0	0/1	TxD3/ SO30 = 1	—	—	—	—	—	—	—	—	—		
		N-ch OD output	—	1	—	0	0/1		—	—	—	—	—	—	—	—	—		
	SO30	Output	—	0/1	—	0	1	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
	TxD3	Output	—	0/1	—	0	1	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
P145	P145	Input	—	—	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
		Output	—	—	—	0	0/1	—	—	—	—	—	—	—	—	—	—		
P146	P146	Input	—	—	—	1	✗	—	—	—	✗	✗	✗	✓	✓	✓	✓	✓	
		Output	—	—	—	0	0/1	—	—	—	—	—	—	—	—	—	—		
	(INTP4)	Input	PIOR05 = 1	—	—	1	✗	—	—	—	✗	✗	✗	✗	✗	✗	✗	✓	
P147	P147	Input	—	—	0	1	✗	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓	
		Output	—	—	0	0	0/1	—	—	—	—	—	—	—	—	—	—		
	ANI18	Analog input	—	—	1	1	✗	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓	
	VCOUT1 Note	Analog output	—	—	0	0	0	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓	

**Note** Products with 96 KB or more code flash memory only.

**Table 4 - 20 Setting Examples of Registers When Using P150 to P156 Pin Function**

Pin Name	Used Function		ADPC	PMxx	Pxx										
	Function Name	I/O				30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	52-pin	64-pin	80-pin	100-pin
P150	P150	Input	ADPC = 01H to 09H	1	x	x	x	x	x	x	x	x	x	✓	✓
		Output	ADPC = 01H to 09H	0	0/1										
	ANI8	Analog input	ADPC = 00H/0AH to 0FH	1	x										
P151	P151	Input	ADPC = 01H to 0AH	1	x	x	x	x	x	x	x	x	x	✓	✓
		Output	ADPC = 01H to 0AH	0	0/1										
	ANI9	Analog input	ADPC = 00H/0BH to 0FH	1	x										
P152	P152	Input	ADPC = 01H to 0BH	1	x	x	x	x	x	x	x	x	x	✓	✓
		Output	ADPC = 01H to 0BH	0	0/1										
	ANI10	Analog input	ADPC = 00H/0CH to 0FH	1	x										
P153	P153	Input	ADPC = 01H to 0CH	1	x	x	x	x	x	x	x	x	x	✓	✓
		Output	ADPC = 01H to 0CH	0	0/1										
	ANI11	Analog input	ADPC = 00H/0DH to 0FH	1	x										
P154	P154	Input	ADPC = 01H to 0DH	1	x	x	x	x	x	x	x	x	x	x	✓
		Output	ADPC = 01H to 0DH	0	0/1										
	ANI12	Analog input	ADPC = 00H/0EH to 0FH	1	x										
P155	P155	Input	ADPC = 01H to 0EH	1	x	x	x	x	x	x	x	x	x	x	✓
		Output	ADPC = 01H to 0EH	0	0/1										
	ANI13	Analog input	ADPC = 00H/0FH	1	x										
P156	P156	Input	ADPC = 01H to 0FH	1	x	x	x	x	x	x	x	x	x	x	✓
		Output	ADPC = 01H to 0FH	0	0/1										
	ANI14	Analog input	ADPC = 00H	1	x										

## 4.6 Cautions When Using Port Function

### 4.6.1 Cautions on 1-Bit Manipulation Instruction for Port Register n (Pn)

When a 1-bit manipulation instruction is executed on a port that provides both input and output functions, the output latch value of an input port that is not subject to manipulation may be written in addition to the targeted bit. Therefore, it is recommended to rewrite the output latch when switching a port from input mode to output mode.

<Example> When P10 is an output port, P11 to P17 are input ports (all pin statuses are high level), and the port latch value of port 1 is 00H, if the output of output port P10 is changed from low level to high level via a 1-bit manipulation instruction, the output latch value of port 1 is FFH.

Explanation: The targets of writing to and reading from the Pn register of a port whose PMnm bit is 1 are the output latch and pin status, respectively.

A 1-bit manipulation instruction is executed in the following order in the RL78/G14.

<1> The Pn register is read in 8-bit units.

<2> The targeted one bit is manipulated.

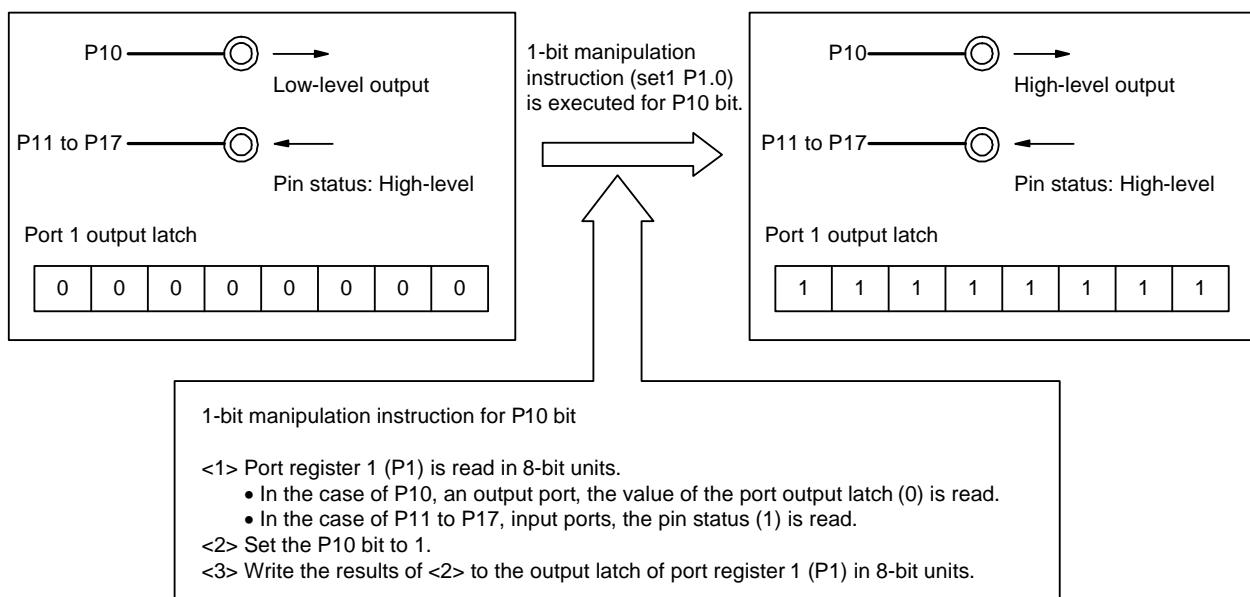
<3> The Pn register is written in 8-bit units.

In step <1>, the output latch value (0) of P10, which is an output port, is read, while the pin statuses of P11 to P17, which are input ports, are read. If the pin statuses of P11 to P17 are high level at this time, the read value is FEH.

The value is changed to FFH by the manipulation in <2>.

FFH is written to the output latch by the manipulation in <3>.

**Figure 4 - 12 Bit Manipulation Instruction (P10)**



#### 4.6.2 Notes on specifying the pin settings

For an output pin to which multiple functions are assigned, the output of the unused alternate functions must be set to its initial state so as to prevent conflicting outputs. This also applies to the functions assigned by using the peripheral I/O redirection register (PIOR). For details about the alternate function output, see [4.5 Register Settings When Using Alternate Function](#).

No specific setting is required for input pins because the output of their alternate functions is disabled (the buffer output is Hi-Z).

Disabling the unused functions, including blocks that are only used for input or do not have I/O, is recommended for lower power consumption.

## CHAPTER 5 CLOCK GENERATOR

The presence or absence of connecting resonator pin for main system clock, connecting resonator pin for subsystem clock, external clock input pin for main system clock, and external clock input pin for subsystem clock, depends on the product.

	30, 32, 36-pin products	40, 44, 48, 52, 64, 80, 100-pin products
X1, X2 pins	√	√
EXCLK pin	√	√
XT1, XT2 pins	—	√
EXCLKS pin	—	√

### 5.1 Functions of Clock Generator

The clock generator generates the clock to be supplied to the CPU and peripheral hardware.

The following three kinds of system clocks and clock oscillators are selectable.

#### (1) Main system clock

##### <1> X1 oscillator

This circuit oscillates a clock of  $f_x = 1$  to 20 MHz by connecting a resonator to X1 pin and X2 pin.

Oscillation can be stopped by executing the STOP instruction or setting of the MSTOP bit (bit 7 of the clock operation status control register (CSC)).

##### <2> High-speed on-chip oscillator (High-speed OCO)

The frequency at which to oscillate can be selected from among  $f_{HO} = 64, 48, 32, 24, 16, 12, 8, 6, 4, 3, 2,$  or 1 MHz (TYP.) by using the option byte (000C2H). When 64 MHz or 48 MHz is selected as  $f_{HO}$ ,  $f_{IH}$  is set to 32 MHz or 24 MHz, respectively. When 32 MHz or less is selected as  $f_{HO}$ ,  $f_{IH}$  is not divided and set to the same frequency as  $f_{HO}$ . After a reset release, the CPU always starts operating with this high-speed on-chip oscillator clock. Oscillation can be stopped by executing the STOP instruction or setting of the HIOSTOP bit (bit 0 of the CSC register).

The frequency specified by using an option byte can be changed by using the high-speed on-chip oscillator frequency select register (HOCODIV). For details about the frequency, see **Figure 5 - 13 Format of High-speed on-chip oscillator frequency select register (HOCODIV)**.

The frequencies that can be specified for the high-speed on-chip oscillator by using the option byte and the high-speed on-chip oscillator frequency select register (HOCODIV) are shown below.

Power Supply Voltage	Oscillation Frequency (MHz)											
	1	2	3	4	6	8	12	16	24	32	48	64
2.7 V ≤ $V_{DD} \leq 5.5$ V	√	√	√	√	√	√	√	√	√	√	√	√
2.4 V ≤ $V_{DD} \leq 5.5$ V	√	√	√	√	√	√	√	√	—	—	—	—
1.8 V ≤ $V_{DD} \leq 5.5$ V	√	√	√	√	√	√	—	—	—	—	—	—
1.6 V ≤ $V_{DD} \leq 5.5$ V	√	√	√	√	—	—	—	—	—	—	—	—

An external main system clock ( $f_{EX}$  = 1 to 20 MHz) can also be supplied from the EXCLK/X2/P122 pin. An external main system clock input can be disabled by executing the STOP instruction or setting of the MSTOP bit. As the main system clock, a high-speed system clock (X1 clock or external main system clock) or high-speed on-chip oscillator clock can be selected by setting of the MCM0 bit (bit 4 of the system clock control register (CKC)). However, note that the usable frequency range of the main system clock differs depending on the setting of the power supply voltage ( $V_{DD}$ ). The operating voltage of the flash memory must be set by using the CMODE0 and CMODE1 bits of the option byte (000C2H) (see **CHAPTER 29 OPTION BYTE**).

## (2) Subsystem clock

- XT1 clock oscillator

This circuit oscillates a clock of  $f_{XT} = 32.768$  kHz by connecting a 32.768 kHz resonator to XT1 pin and XT2 pin. Oscillation can be stopped by setting the XTSTOP bit (bit 6 of the clock operation status control register (CSC)).

An external subsystem clock ( $f_{EXS} = 32.768$  kHz) can also be supplied from the EXCLKS/XT2/P124 pin. An external subsystem clock input can be disabled by the setting of the XTSTOP bit.

## (3) Low-speed on-chip oscillator (Low-speed OCO)

This circuit oscillates a clock of  $f_{IL} = 15$  kHz (TYP.).

The low-speed on-chip oscillator clock cannot be used as the CPU clock.

Only the following peripheral hardware runs on the low-speed on-chip oscillator clock.

- Watchdog timer
- Real-time clock
- 12-bit interval timer
- Timer RJ

This clock operates when bit 4 (WDTON) of the option byte (000C0H), bit 4 (WUTMMCK0) of the subsystem clock supply mode control register (OSMC), or both are set to 1.

However, if WDTON = 1, WUTMMCK0 = 0, and bit 0 (WDSTBYON) of the option byte (000C0H) is 0, the low-speed on-chip oscillator stops oscillation when the HALT or STOP instruction is executed.

**Caution** **The low-speed on-chip oscillator clock ( $f_{IL}$ ) can only be selected as the real-time clock count clock when the fixed-cycle interrupt function is used.**

<b>Remark</b>	fx:	X1 clock oscillation frequency
	fHO CO:	High-speed on-chip oscillator clock frequency (64 MHz max.)
	fiH:	High-speed on-chip oscillator clock frequency (32 MHz max.) Note
	fEX:	External main system clock frequency
	fXT:	XT1 clock oscillation frequency
	fEXS:	External subsystem clock frequency
	fIL:	Low-speed on-chip oscillator frequency

<b>Note</b>	fiH is controlled by hardware to be set to two frequency division of fHO CO when fHO CO is set to 64 MHz or 48 MHz, and the same clock frequency as fHO CO when fHO CO is set to 32 MHz or less. When supplying 64 MHz or 48 MHz to timer RD, set fCLK to fiH.
-------------	--

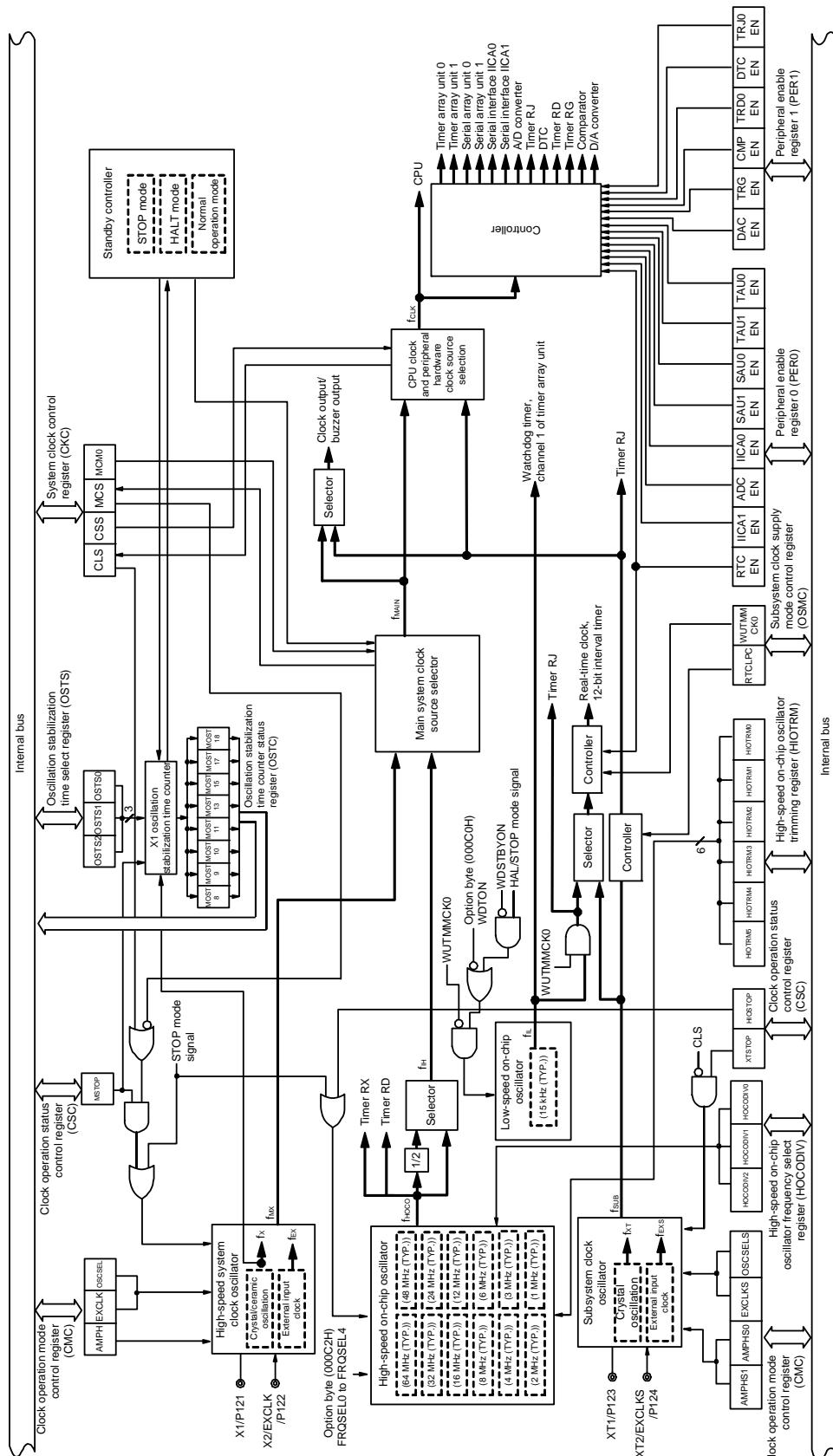
## 5.2 Configuration of Clock Generator

The clock generator includes the following hardware.

**Table 5 - 1 Configuration of Clock Generator**

Item	Configuration
Control registers	Clock operation mode control register (CMC) System clock control register (CKC) Clock operation status control register (CSC) Oscillation stabilization time counter status register (OSTC) Oscillation stabilization time select register (OSTS) Peripheral enable registers 0, 1 (PER0, PER1) Subsystem clock supply mode control register (OSMC) High-speed on-chip oscillator frequency select register (HOCODIV) High-speed on-chip oscillator trimming register (HIOTRM)
Oscillators	X1 oscillator XT1 oscillator High-speed on-chip oscillator Low-speed on-chip oscillator

**Figure 5 - 1 Block Diagram of Clock Generator**



(**Remark** and **Note** are listed on the next page after next.)

<b>Remark</b>	fx:	X1 clock oscillation frequency
	fHO CO:	High-speed on-chip oscillator clock frequency (64 MHz max.)
	fiH:	High-speed on-chip oscillator clock frequency (32 MHz max.) Note
	fEX:	External main system clock frequency
	fMX:	High-speed system clock frequency
	fMAIN:	Main system clock frequency
	fXT:	XT1 clock oscillation frequency
	fEXS:	External subsystem clock frequency
	fSUB:	Subsystem clock frequency
	fCLK:	CPU/peripheral hardware clock frequency
	fIL:	Low-speed on-chip oscillator clock frequency

**Note** fiH is controlled by hardware to be set to two frequency division of fHO CO when fHO CO is set to 64 MHz or 48 MHz, and the same clock frequency as fHO CO when fHO CO is set to 32 MHz or less. When supplying 64 MHz or 48 MHz to timer RD, set fCLK to fiH.

### 5.3 Registers Controlling Clock Generator

The following registers are used to control the clock generator.

- Clock operation mode control register (CMC)
- System clock control register (CKC)
- Clock operation status control register (CSC)
- Oscillation stabilization time counter status register (OSTC)
- Oscillation stabilization time select register (OSTS)
- Peripheral enable registers 0, 1 (PER0, PER1)
- Subsystem clock supply mode control register (OSMC)
- High-speed on-chip oscillator frequency select register (HOCODIV)
- High-speed on-chip oscillator trimming register (HIOTRM)

**Caution** Which registers and bits are included depends on the product. Be sure to set registers and bits that are not mounted in a product to their initial values.

#### 5.3.1 Clock operation mode control register (CMC)

This register is used to set the operation mode of the X1/P121, X2/EXCLK/P122, XT1/P123, and XT2/EXCLKS/P124 pins, and to select a gain of the oscillator.

The CMC register can be written only once by an 8-bit memory manipulation instruction after reset release. This register can be read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 5 - 2 Format of Clock operation mode control register (CMC)**

Address: FFFA0H After reset: 00H R/W

Symbol 7 6 5 4 3 2 1 0

CMC	EXCLK	OSCSEL	EXCLKS	OSCSELS	0	AMPHS1	AMPHS0	AMPH									
	EXCLK	OSCSEL	High-speed system clock pin operation mode		X1/P121 pin	X2/EXCLK/P122 pin											
0	0	0	Input port mode		Input port												
0	1		X1 oscillation mode		Crystal/ceramic resonator connection												
1	0	0	Input port mode		Input port												
1	1	1	External clock input mode		Input port	External clock input											
	EXCLKS	OSCSELS	Subsystem clock pin operation mode		XT1/P123 pin	XT2/EXCLKS/P124 pin											
0	0	0	Input port mode		Input port												
0	1		XT1 oscillation mode		Crystal resonator connection												
1	0	0	Input port mode		Input port												
1	1	1	External clock input mode		Input port	External clock input											
	AMPHS1	AMPHS0	XT1 oscillator oscillation mode selection														
0	0	0	Low power consumption oscillation (default)														
0	1		Normal oscillation														
1	0	0	Ultra-low power consumption oscillation														
1	1	1	Setting prohibited														
	AMPH	Control of X1 clock oscillation frequency															
0		1 MHz ≤ fx ≤ 10 MHz															
1		10 MHz < fx ≤ 20 MHz															

**Caution 1.** The CMC register can be written only once after reset release, by an 8-bit memory manipulation instruction. When using the CMC register with its initial value (00H), be sure to set the register to 00H after a reset ends in order to prevent malfunction due to a program loop. Such a malfunction becomes unrecoverable when a value other than 00H is mistakenly written.

**Caution 2.** After reset release, set the CMC register before X1 or XT1 oscillation is started as set by the clock operation status control register (CSC).

**Caution 3.** Be sure to set the AMPH bit to 1 if the X1 clock oscillation frequency exceeds 10 MHz.

**Caution 4.** Specify the settings for the AMPH, AMPHS1, and AMPHS0 bits while fIH is selected as fCLK after a reset ends (before fCLK is switched to fMX or fSUB).

**Caution 5.** Oscillation stabilization time of fXT, counting on the software.

**Caution 6.** Although the maximum system clock frequency is 32 MHz, the maximum frequency of the X1 oscillator is 20 MHz.

(Cautions and Remark are given on the next page.)

**Caution 7.** The XT1 oscillator is a circuit with low amplification in order to achieve low-power consumption. Note the following points when designing the circuit.

- Pins and circuit boards include parasitic capacitance. Therefore, perform oscillation evaluation using a circuit board to be actually used and confirm that there are no problems.
- When using the ultra-low power consumption oscillation (AMPHS1, AMPHS0 = 1, 0) as the mode of the XT1 oscillator, evaluate the resonators described in 5.7 Resonator and Oscillator Constants.
- Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance. Note this particularly when the ultra-low power consumption oscillation (AMPHS1, AMPHS0 = 1, 0) is selected.
- Configure the circuit of the circuit board, using material with little parasitic capacitance and wiring resistance.
- Place a ground pattern that has the same potential as Vss as much as possible near the XT1 oscillator.
- Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.
- The impedance between the XT1 and XT2 pins may drop and oscillation may be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.
- When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.

**Remark** fx: X1 clock frequency

### 5.3.2 System clock control register (CKC)

This register is used to select a CPU/peripheral hardware clock and a main system clock.

The CKC register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to 00H.

**Figure 5 - 3 Format of System clock control register (CKC)**

Address: FFFA4H      After reset: 00H      R/W<sup>Note 1</sup>

Symbol	<7>	<6>	<5>	<4>	3	2	1	0
CKC	CLS	CSS Note 2	MCS	MCM0 Note 2	0	0	0	0
CLS		Status of CPU/peripheral hardware clock (fCLK)						
0		Main system clock (fMAIN)						
1		Subsystem clock (fSUB)						
CSS Note 2		Selection of CPU/peripheral hardware clock (fCLK)						
0		Main system clock (fMAIN)						
1		Subsystem clock (fSUB)						
MCS		Status of Main system clock (fMAIN)						
0		High-speed on-chip oscillator clock (fIH)						
1		High-speed system clock (fMX)						
MCM0 Note 2		Main system clock (fMAIN) operation control						
0		Selects the high-speed on-chip oscillator clock (fIH) as the main system clock (fMAIN)						
1		Selects the high-speed system clock (fMX) as the main system clock (fMAIN)						

**Note 1.** Bits 7 and 5 are read-only.

**Note 2.** Changing the value of the MCM0 bit is prohibited while the CSS bit is set to 1.

**Remark** fHOOC: High-speed on-chip oscillator clock frequency (64 MHz max.)  
 fIH: High-speed on-chip oscillator clock frequency (32 MHz max.)  
 fMX: High-speed system clock frequency  
 fMAIN: Main system clock frequency  
 fSUB: Subsystem clock frequency

**Note** fIH is controlled by hardware to be set to two frequency division of fHOOC when fHOOC is set to 64 MHz or 48 MHz, and the same clock frequency as fHOOC when fHOOC is set to 32 MHz or less. When supplying 64 MHz or 48 MHz to timer RD, set fCLK to fIH.

(Cautions are listed on the next page.)

- Caution 1.** Be sure to set bits 0 to 3 of the CKC register to 0.
- Caution 2.** The clock set by the CSS bit is supplied to the CPU and peripheral hardware. If the CPU clock is changed, therefore, the clock supplied to peripheral hardware (except the real-time clock, 12-bit interval timer, clock output/buzzer output, and watchdog timer) is also changed at the same time. Consequently, stop each peripheral function when changing the CPU/peripheral hardware clock.
- Caution 3.** If the subsystem clock is used as the peripheral hardware clock, the operations of the A/D converter and IICA are not guaranteed. For the operating characteristics of the peripheral hardware, refer to the chapters describing the various peripheral hardware as well as CHAPTER 34 or CHAPTER 35 ELECTRICAL SPECIFICATIONS.
- Caution 4.** When selecting f<sub>HOCO</sub> as the count source for timer RD, set f<sub>CLK</sub> to f<sub>IH</sub> before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing f<sub>CLK</sub> to a clock other than f<sub>IH</sub>, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

### 5.3.3 Clock operation status control register (CSC)

This register is used to control the operations of the high-speed system clock, high-speed on-chip oscillator clock, and subsystem clock (except the low-speed on-chip oscillator clock).

The CSC register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to C0H.

**Figure 5 - 4 Format of Clock operation status control register (CSC)**

Address: FFFA1H      After reset: C0H      R/W

Symbol	<7>	<6>	5	4	3	2	1	<0>
--------	-----	-----	---	---	---	---	---	-----

CSC	MSTOP	XTSTOP	0	0	0	0	0	HIOSTOP													
MSTOP		High-speed system clock operation control																			
		X1 oscillation mode		External clock input mode		Input port mode															
0	X1 oscillator operating		External clock from EXCLK pin is valid		Input port																
	X1 oscillator stopped		External clock from EXCLK pin is invalid																		
XTSTOP		Subsystem clock operation control																			
		XT1 oscillation mode		External clock input mode		Input port mode															
0	XT1 oscillator operating		External clock from EXCLKS pin is valid		Input port																
	XT1 oscillator stopped		External clock from EXCLKS pin is invalid																		
HIOSTOP		High-speed on-chip oscillator clock operation control																			
		0 High-speed on-chip oscillator operating																			
		1 High-speed on-chip oscillator stopped																			

**Caution 1. After reset release, set the clock operation mode control register (CMC) before setting the CSC register.**

**Caution 2. Set the oscillation stabilization time select register (OSTS) before setting the MSTOP bit to 0 after releasing reset. Note that if the OSTS register is being used with its default settings, the OSTS register is not required to be set here.**

**Caution 3. To start X1 oscillation as set by the MSTOP bit, check the oscillation stabilization time of the X1 clock by using the oscillation stabilization time counter status register (OSTC).**

**Caution 4. When starting XT1 oscillation by setting the XTSTOP bit, wait for oscillation of the subsystem clock to stabilize by setting a wait time using software.**

**Caution 5. Do not stop the clock selected for the CPU peripheral hardware clock (fCLK) with the OSC register.**

**Caution 6. The setting of the flags of the register to stop clock oscillation (invalidate the external clock input) and the condition before clock oscillation is to be stopped are as Table 5 - 2. Before stopping the clock oscillation, check the conditions before the clock oscillation is stopped.**

**Table 5 - 2 Stopping Clock Method**

Clock	Condition Before Stopping Clock (Invalidate External Clock Input)	Setting of CSC Register Flags
X1 clock	CPU and peripheral hardware clocks operate with a clock other than the high-speed system clock. (CLS = 0 and MCS = 0, or CLS = 1)	MSTOP = 1
External main system clock		
XT1 clock	CPU and peripheral hardware clocks operate with a clock other than the subsystem clock. (CLS = 0)	XTSTOP = 1
External subsystem clock		
High-speed on-chip oscillator clock	CPU and peripheral hardware clocks operate with a clock other than the high-speed on-chip oscillator clock. (CLS = 0 and MCS = 1, or CLS = 1)	HIOSTOP = 1

### 5.3.4 Oscillation stabilization time counter status register (OSTC)

This is the register that indicates the count status of the X1 clock oscillation stabilization time counter.

The X1 clock oscillation stabilization time can be checked in the following case,

- When the X1 clock starts oscillation while the high-speed on-chip oscillator clock or subsystem clock is being used as the CPU clock.
- When the STOP mode is entered and then released while the high-speed on-chip oscillator clock is being used as the CPU clock with the X1 clock oscillating.

The OSTC register can be read by a 1-bit or 8-bit memory manipulation instruction.

The generation of reset signal, the STOP instruction and MSTOP (bit 7 of clock operation status control register (CSC)) = 1 clear the OSTC register to 00H.

**Remark** The oscillation stabilization time counter starts counting in the following cases.

- When oscillation of the X1 clock starts (EXCLK, OSCSEL = 0, 1 → MSTOP = 0)
- When the STOP mode is released

**Figure 5 - 5 Format of Oscillation stabilization time counter status register (OSTC)**

Address: FFFA2H After reset: 00H R

Symbol 7 6 5 4 3 2 1 0

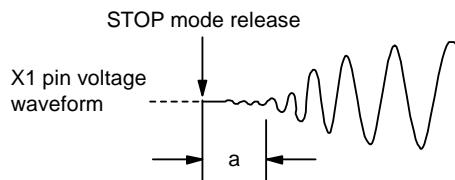
OSTC	MOST8	MOST9	MOST10	MOST11	MOST13	MOST15	MOST17	MOST18
------	-------	-------	--------	--------	--------	--------	--------	--------

MOST 8	MOST 9	MOST 10	MOST 11	MOST 13	MOST 15	MOST 17	MOST 18	Oscillation stabilization time status
								$f_x = 10 \text{ MHz}$
0	0	0	0	0	0	0	0	$2^8/f_x \text{ max.}$
1	0	0	0	0	0	0	0	$2^8/f_x \text{ min.}$
1	1	0	0	0	0	0	0	$2^9/f_x \text{ min.}$
1	1	1	0	0	0	0	0	$2^{10}/f_x \text{ min.}$
1	1	1	1	0	0	0	0	$2^{11}/f_x \text{ min.}$
1	1	1	1	1	0	0	0	$2^{13}/f_x \text{ min.}$
1	1	1	1	1	1	0	0	$2^{15}/f_x \text{ min.}$
1	1	1	1	1	1	1	0	$2^{17}/f_x \text{ min.}$
1	1	1	1	1	1	1	1	$2^{18}/f_x \text{ min.}$

**Caution 1.** After the above time has elapsed, the bits are set to 1 in order from the MOST8 bit and remain 1.**Caution 2.** The oscillation stabilization time counter counts up to the oscillation stabilization time set by the oscillation stabilization time select register (OSTS).

In the following cases, set the oscillation stabilization time of the OSTS register to the value greater than the count value which is to be checked by the OSTC register.

- When the X1 clock starts oscillation while the high-speed on-chip oscillator clock or subsystem clock is being used as the CPU clock.
- When the STOP mode is entered and then released while the high-speed on-chip oscillator clock is being used as the CPU clock with the X1 clock oscillating.  
(Note, therefore, that only the status up to the oscillation stabilization time set by the OSTS register is set to the OSTC register after the STOP mode is released.)

**Caution 3.** The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).**Remark** fx: X1 clock oscillation frequency

### 5.3.5 Oscillation stabilization time select register (OSTS)

This register is used to select the X1 clock oscillation stabilization wait time.

When the X1 clock is made to oscillate by clearing the MSTOP bit to start the X1 oscillation circuit operating, actual operation is automatically delayed for the time set in the OSTS register.

When switching the CPU clock from the high-speed on-chip oscillator clock or the subsystem clock to the X1 clock, and when using the high-speed on-chip oscillator clock for switching the X1 clock from the oscillating state to STOP mode, use the oscillation stabilization time counter status register (OSTC) to confirm that the desired oscillation stabilization time has elapsed after release from the STOP mode. That is, use the OSTC register to check that the oscillation stabilization time corresponding to its setting has been reached.

The OSTS register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets the OSTS register to 07H.

**Figure 5 - 6 Format of Oscillation stabilization time select register (OSTS)**

Address: FFFA3H After reset: 07H R/W

Symbol	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2	OSTS1	OSTS0
Oscillation stabilization time selection								
						$f_x = 10 \text{ MHz}$	$f_x = 20 \text{ MHz}$	
0	0	0	$2^8/f_x$		25.6 $\mu\text{s}$	12.8 $\mu\text{s}$		
0	0	1	$2^9/f_x$		51.2 $\mu\text{s}$	25.6 $\mu\text{s}$		
0	1	0	$2^{10}/f_x$		102 $\mu\text{s}$	51.2 $\mu\text{s}$		
0	1	1	$2^{11}/f_x$		204 $\mu\text{s}$	102 $\mu\text{s}$		
1	0	0	$2^{13}/f_x$		819 $\mu\text{s}$	409 $\mu\text{s}$		
1	0	1	$2^{15}/f_x$		3.27 ms	1.63 ms		
1	1	0	$2^{17}/f_x$		13.1 ms	6.55 ms		
1	1	1	$2^{18}/f_x$		26.2 ms	13.1 ms		

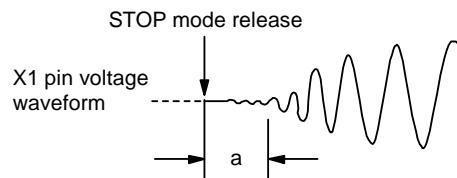
**Caution 1.** Change the setting of the OSTS register before setting the MSTOP bit of the clock operation status control register (CSC) to 0.

**Caution 2.** The oscillation stabilization time counter counts up to the oscillation stabilization time set by the OSTS register.

In the following cases, set the oscillation stabilization time of the OSTS register to the value greater than the count value which is to be checked by the OSTC register after the oscillation starts.

- When the X1 clock starts oscillation while the high-speed on-chip oscillator clock or subsystem clock is being used as the CPU clock.
- When the STOP mode is entered and then released while the high-speed on-chip oscillator clock is being used as the CPU clock with the X1 clock oscillating. (Note, therefore, that only the status up to the oscillation stabilization time set by the OSTS register is set to the OSTC register after the STOP mode is released.)

**Caution 3.** The X1 clock oscillation stabilization wait time does not include the time until clock oscillation starts ("a" below).



**Remark** fx: X1 clock oscillation frequency

### 5.3.6 Peripheral enable registers 0, 1 (PER0, PER1)

These registers are used to enable or disable supplying the clock to the peripheral hardware. Clock supply to the hardware that is not used is also stopped so as to decrease the power consumption and noise.

To use the peripheral functions below, which are controlled by these registers, set (1) the bit corresponding to each function before specifying the initial settings of the peripheral functions.

- Real-time clock and 12-bit interval timer
- Serial interface IICA1
- A/D converter
- Serial interface IICA0
- Serial array unit 1
- Serial array unit 0
- Timer array unit 1
- Timer array unit 0
- D/A converter
- Timer RG
- Comparator
- Timer RD
- DTC
- Timer RJ

The PER0 and PER1 registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 5 - 7 Format of Peripheral enable register 0 (PER0) (1/3)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PER0	RTCEN	IICA1EN	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
RTCEN	Control of supplying input clock for real-time clock (RTC) and 12-bit interval timer							
0	Stops input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer cannot be written. • The real-time clock (RTC) and 12-bit interval timer are in the reset status.							
1	Enables input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer can be read and written.							

**Caution** Be sure to clear the following bits to 0.  
 30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

**Figure 5 - 8 Format of Peripheral enable register 0 (PER0) (2/3)**

Address: F00F0H After reset: 00H R/W

Symbol &lt;7&gt; &lt;6&gt; &lt;5&gt; &lt;4&gt; &lt;3&gt; &lt;2&gt; &lt;1&gt; &lt;0&gt;

PER0	RTCEN	IICA1EN	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
------	-------	---------	-------	---------	--------	--------	--------	--------

IICA1EN	Control of serial interface IICA1 input clock supply
0	Stops input clock supply. • SFR used by the serial interface IICA1 cannot be written. • The serial interface IICA1 is in the reset status.
1	Enables input clock supply. • SFR used by the serial interface IICA1 can be read and written.

ADCEN	Control of A/D converter input clock supply
0	Stops input clock supply. • SFR used by the A/D converter cannot be written. • The A/D converter is in the reset status.
1	Enables input clock supply. • SFR used by the A/D converter can be read and written.

IICA0EN	Control of serial interface IICA0 input clock supply
0	Stops input clock supply. • SFR used by the serial interface IICA0 cannot be written. • The serial interface IICA0 is in the reset status.
1	Enables input clock supply. • SFR used by the serial interface IICA0 can be read and written.

SAU1EN	Control of serial array unit 1 input clock supply
0	Stops input clock supply. • SFR used by the serial array unit 1 cannot be written. • The serial array unit 1 is in the reset status.
1	Enables input clock supply. • SFR used by the serial array unit 1 can be read and written.

SAU0EN	Control of serial array unit 0 input clock supply
0	Stops input clock supply. • SFR used by the serial array unit 0 cannot be written. • The serial array unit 0 is in the reset status.
1	Enables input clock supply. • SFR used by the serial array unit 0 can be read and written.

**Caution Be sure to clear the following bits to 0.****30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6**

**Figure 5 - 9 Format of Peripheral enable register 0 (PER0) (3/3)**

Address: F00F0H After reset: 00H R/W

Symbol &lt;7&gt; &lt;6&gt; &lt;5&gt; &lt;4&gt; &lt;3&gt; &lt;2&gt; &lt;1&gt; &lt;0&gt;

PER0	RTCEN	IICA1EN	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN	TAU0EN
------	-------	---------	-------	---------	--------	--------	--------	--------

TAU1EN	Control of timer array unit 1 input clock supply
0	Stops input clock supply. • SFR used by timer array unit 1 cannot be written. • Timer array unit 1 is in the reset status.
1	Enables input clock supply. • SFR used by timer array unit 1 can be read and written.

TAU0EN	Control of timer array unit 0 input clock supply
0	Stops input clock supply. • SFR used by timer array unit 0 cannot be written. • Timer array unit 0 is in the reset status.
1	Enables input clock supply. • SFR used by timer array unit 0 can be read and written.

**Caution Be sure to clear the following bits to 0.**

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

**Figure 5 - 10 Format of Peripheral enable register 1 (PER1) (1/2)**

Address: F007AH After reset: 00H R/W

Symbol &lt;7&gt; &lt;6&gt; &lt;5&gt; &lt;4&gt; &lt;3&gt; &lt;2&gt; &lt;1&gt; &lt;0&gt;

PER1	DACEN	TRGEN	CMPEN	TRD0EN Note	DTCEN	0	0	TRJ0EN
------	-------	-------	-------	-------------	-------	---	---	--------

DACEN	Control of D/A converter input clock supply
0	Stops input clock supply. • SFR used by the D/A converter cannot be written. • The D/A converter is in the reset status.
1	Enables input clock supply. • SFR used by the D/A converter can be read and written.

TRGEN	Control of timer RG input clock supply
0	Stops input clock supply. • SFR used by timer RG cannot be written. • Timer RG is in the reset status.
1	Enables input clock supply. • SFR used by timer RG can be read and written.

CMPEN	Control of comparator input clock supply
0	Stops input clock supply. • SFR used by comparator cannot be written. • Comparator is in the reset status.
1	Enables input clock supply. • SFR used by comparator can be read and written.

TRD0EN Note	Control of timer RD input clock supply
0	Stops input clock supply. • SFR used by timer RD cannot be written. • Timer RD is in the reset status.
1	Enables input clock supply. • SFR used by timer RD can be read and written.

**Note** When FRQSEL4 = 1 in the user option byte (000C2H), set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Caution** Be sure to clear the following bits to 0.

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

**Figure 5 - 11 Format of Peripheral enable register 1 (PER1) (2/2)**

Address: F007AH After reset: 00H R/W

Symbol &lt;7&gt; &lt;6&gt; &lt;5&gt; &lt;4&gt; &lt;3&gt; &lt;2&gt; &lt;1&gt; &lt;0&gt;

PER1	DACEN	TRGEN	CMPEN	TRD0EN Note	DTCEN	0	0	TRJ0EN
------	-------	-------	-------	-------------	-------	---	---	--------

DTCEN	Control of DTC input clock supply
0	Stops input clock supply. • DTC cannot run.
1	Enables input clock supply. • DTC can run.

TRJ0EN	Control of timer RJ0 input clock supply
0	Stops input clock supply. • SFR used by timer RJ0 cannot be written. • Timer RJ0 is in the reset status.
1	Enables input clock supply. • SFR used by timer RJ0 can be read and written.

**Caution** Be sure to clear the following bits to 0.

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 5.3.7 Subsystem clock supply mode control register (OSMC)

This register is used to reduce power consumption by stopping unnecessary clock functions.

If the RTCLPC bit is set to 1, power consumption can be reduced, because clock supply to the peripheral functions, except the real-time clock and 12-bit interval timer, is stopped in STOP mode or HALT mode while subsystem clock is selected as CPU clock.

In addition, the OSMC register can be used to select the operation clock of the real-time clock and 12-bit interval timer.

The OSMC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 5 - 12 Format of Subsystem clock supply mode control register (OSMC)**

Address: F00F3H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0
RTCLPC	Setting in STOP mode or HALT mode while subsystem clock is selected as CPU clock							
0	Enables supply of subsystem clock to peripheral functions (See <b>Tables 23 - 1 to 23 - 4</b> for peripheral functions whose operations are enabled.)							
1	Stops supply of subsystem clock to peripheral functions other than real-time clock and 12-bit interval timer.							
WUTMMCK0	Selection of operation clock for real-time clock, 12-bit interval timer, and timer RJ							
0	<ul style="list-style-type: none"> <li>The subsystem clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>The low-speed on-chip oscillator cannot be selected as the count source for timer RJ.</li> </ul>							
1	<ul style="list-style-type: none"> <li>The low-speed on-chip oscillator clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>Either the low-speed on-chip oscillator or the subsystem clock can be selected as the count source for timer RJ.</li> </ul>							

### 5.3.8 High-speed on-chip oscillator frequency select register (HOCODIV)

The frequency of the high-speed on-chip oscillator which is set by an option byte (000C2H) can be changed by using high-speed on-chip oscillator frequency select register (HOCODIV). However, the selectable frequency depends on the FRQSEL4 and FRQSEL3 bits of the option byte (000C2H).

The HOCODIV register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to the value set by FRQSEL2 to FRQSEL0 of the option byte (000C2H).

**Figure 5 - 13 Format of High-speed on-chip oscillator frequency select register (HOCODIV)**

Address: F00A8H      After reset: the value set by FRQSEL2 to FRQSEL0 of the option byte (000C2H)      R/W

Symbol	7	6	5	4	3	2	1	0
HOCODIV	0	0	0	0	0	HOCODIV2	HOCODIV1	HOCODIV0
Selection of high-speed on-chip oscillator clock frequency								
HOCODIV2	HOCODIV1	HOCODIV0	FRQSEL4 = 0			FRQSEL4 = 1		
			FRQSEL3 = 0	FRQSEL3 = 1	FRQSEL3 = 0	FRQSEL3 = 1		
			$f_{IH} = 24 \text{ MHz}$	$f_{IH} = 32 \text{ MHz}$	$f_{IH} = 24 \text{ MHz}$ $f_{HOCO} = 48 \text{ MHz}$	$f_{IH} = 32 \text{ MHz}$ $f_{HOCO} = 64 \text{ MHz}$		
0	0	0	$f_{IH} = 12 \text{ MHz}$	$f_{IH} = 16 \text{ MHz}$	$f_{IH} = 12 \text{ MHz}$ $f_{HOCO} = 24 \text{ MHz}$	$f_{IH} = 16 \text{ MHz}$ $f_{HOCO} = 32 \text{ MHz}$		
0	0	1	$f_{IH} = 6 \text{ MHz}$	$f_{IH} = 8 \text{ MHz}$	$f_{IH} = 6 \text{ MHz}$ $f_{HOCO} = 12 \text{ MHz}$	$f_{IH} = 8 \text{ MHz}$ $f_{HOCO} = 16 \text{ MHz}$		
0	1	0	$f_{IH} = 3 \text{ MHz}$	$f_{IH} = 4 \text{ MHz}$	$f_{IH} = 3 \text{ MHz}$ $f_{HOCO} = 6 \text{ MHz}$	$f_{IH} = 4 \text{ MHz}$ $f_{HOCO} = 8 \text{ MHz}$		
1	0	0	Setting prohibited	$f_{IH} = 2 \text{ MHz}$	Setting prohibited	$f_{IH} = 2 \text{ MHz}$ $f_{HOCO} = 4 \text{ MHz}$		
1	0	1	Setting prohibited	$f_{IH} = 1 \text{ MHz}$	Setting prohibited	$f_{IH} = 1 \text{ MHz}$ $f_{HOCO} = 2 \text{ MHz}$		
Other than above			Setting prohibited					

**Caution 1.** Set the HOCODIV register within the operable voltage range of the flash operation mode set in the option byte (000C2H) before and after the frequency change.

Option Byte (000C2H) Value		Flash Operation Mode	Operating Frequency Range	Operating Voltage Range
CMODE1	CMODE0			
0	0	LV (low-voltage main) mode	1 to 4 MHz	1.6 to 5.5 V
1	0	LS (low-speed main) mode	1 to 8 MHz	1.8 to 5.5 V
1	1	HS (high-speed main) mode	1 to 16 MHz	2.4 to 5.5 V
			1 to 32 MHz	2.7 to 5.5 V
Setting prohibited		Other than above		

**Caution 2.** Set the HOCODIV register with the high-speed on-chip oscillator clock ( $f_{IH}$ ) selected as the CPU/peripheral hardware clock (fCLK).

**Caution 3.** After the frequency is changed with the HOCODIV register, the frequency is switched after the following transition time has elapsed.

- Operation for up to three clocks at the pre-change frequency
- CPU/peripheral hardware clock wait at the post-change frequency for up to three clocks

### 5.3.9 High-speed on-chip oscillator trimming register (HIOTRM)

This register is used to adjust the accuracy of the high-speed on-chip oscillator.

With self-measurement of the high-speed on-chip oscillator frequency via a timer using high-accuracy external clock input, and so on, the accuracy can be adjusted.

The HIOTRM register can be set by an 8-bit memory manipulation instruction.

**Caution** The frequency will vary if the temperature and VDD pin voltage change after accuracy adjustment. When the temperature and VDD voltage change, accuracy adjustment must be executed regularly or before the frequency accuracy is required.

**Figure 5 - 14 Format of High-speed on-chip oscillator trimming register (HIOTRM)**

Address: F00A0H      After reset: Undefined R/W

Symbol	7	6	5	4	3	2	1	0
HIOTRM	0	0	HIOTRM5	HIOTRM4	HIOTRM3	HIOTRM2	HIOTRM1	HIOTRM0
<b>High-speed on-chip oscillator</b>								
HIOTRM5	HIOTRM4	HIOTRM3	HIOTRM2	HIOTRM1	HIOTRM0			
0	0	0	0	0	0	Minimum speed		
0	0	0	0	0	1		↑	
0	0	0	0	1	0			
0	0	0	0	1	1			
0	0	0	1	0	0			
•								
•								
•								
1	1	1	1	1	0			
1	1	1	1	1	1	Maximum speed	↓	

**Note** The value after reset is the value adjusted at shipment.

**Remark 1.** The HIOTRM register can be used to adjust the high-speed on-chip oscillator clock to an accuracy within about 0.05% on 1 bit per.

**Remark 2.** For the usage example of the HIOTRM register, see the application note for RL78 MCU Series High-speed On-chip Oscillator (HOCO) Clock Frequency Correction (R01AN0464).

## 5.4 System Clock Oscillator

### 5.4.1 X1 oscillator

The X1 oscillator oscillates with a crystal resonator or ceramic resonator (1 to 20 MHz) connected to the X1 and X2 pins.

An external clock can also be input. In this case, input the clock signal to the EXCLK pin.

To use the X1 oscillator, set bits 7 and 6 (EXCLK, OSCSEL) of the clock operation mode control register (CMC) as follows.

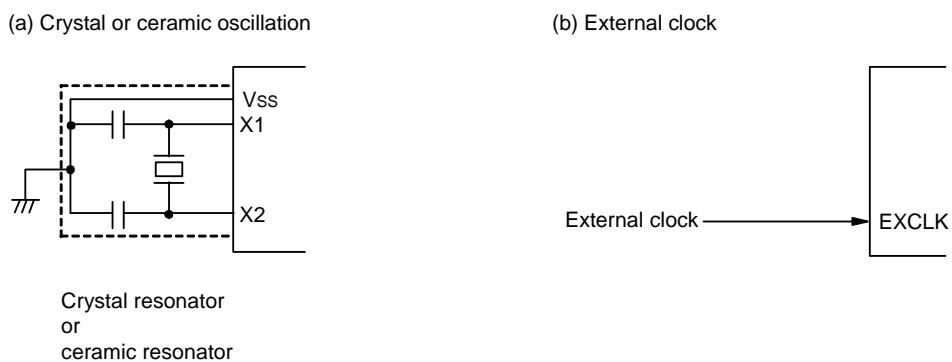
- Crystal or ceramic oscillation: EXCLK, OSCSEL = 0, 1
- External clock input: EXCLK, OSCSEL = 1, 1

When the X1 oscillator is not used, set the input port mode (EXCLK, OSCSEL = 0, 0).

When the pins are not used as input port pins, either, see **Table 2 - 3 Connection of Unused Pins**.

Figure 5 - 15 shows an example of the external circuit of the X1 oscillator.

**Figure 5 - 15 Example of External Circuit of X1 Oscillator**



**Caution** is listed on the next page.

### 5.4.2 XT1 oscillator

The XT1 oscillator oscillates with a crystal resonator (32.768 kHz (TYP.)) connected to the XT1 and XT2 pins.

To use the XT1 oscillator, set bit 4 (OSCSELS) of the clock operation mode control register (CMC) to 1.

An external clock can also be input. In this case, input the clock signal to the EXCLKS pin.

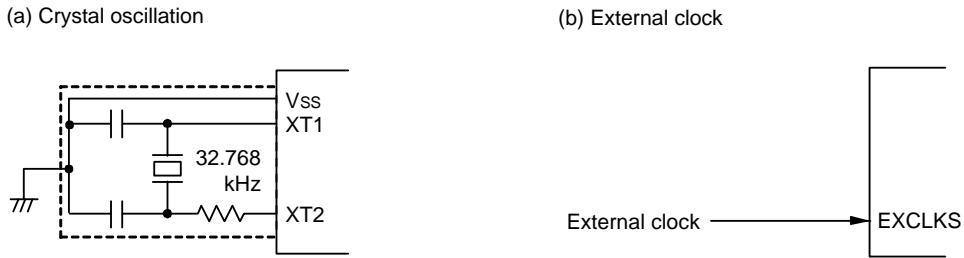
To use the XT1 oscillator, set bits 5 and 4 (EXCLKS, OSCSELS) of the clock operation mode control register (CMC) as follows.

- Crystal oscillation: EXCLKS, OSCSELS = 0, 1
- External clock input: EXCLKS, OSCSELS = 1, 1

When the XT1 oscillator is not used, set the input port mode (EXCLKS, OSCSELS = 0, 0).

When the pins are not used as input port pins, either, see **Table 2 - 3 Connection of Unused Pins**.

Figure 5 - 16 shows an example of the external circuit of the XT1 oscillator.

**Figure 5 - 16 Example of External Circuit of XT1 Oscillator**

**Caution** When using the X1 oscillator and XT1 oscillator, wire as follows in the area enclosed by the broken lines in the Figures 5 - 15 and 5 - 16 to avoid an adverse effect from wiring capacitance.

- Keep the wiring length as short as possible.
- Do not cross the wiring with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.
- Always make the ground point of the oscillator capacitor the same potential as Vss. Do not ground the capacitor to a ground pattern through which a high current flows.
- Do not fetch signals from the oscillator.

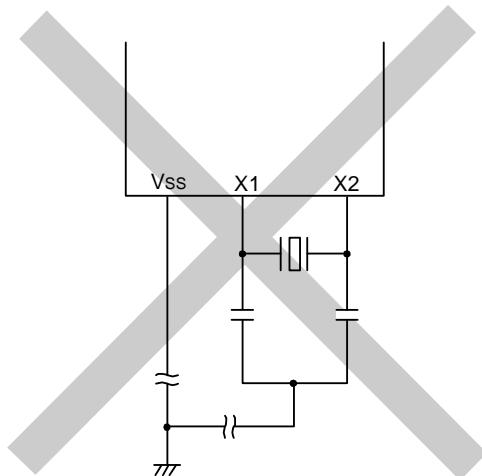
The XT1 oscillator is a circuit with low amplification in order to achieve low-power consumption. Note the following points when designing the circuit.

- Pins and circuit boards include parasitic capacitance. Therefore, perform oscillation evaluation using a circuit board to be actually used and confirm that there are no problems.
- When using the ultra-low power consumption oscillation (AMPHS1, AMPHS0 = 1, 0) as the mode of the XT1 oscillator, evaluate the resonators described in 5.7 Resonator and Oscillator Constants.
- Make the wiring between the XT1 and XT2 pins and the resonators as short as possible, and minimize the parasitic capacitance and wiring resistance. Note this particularly when the ultra-low power consumption oscillation (AMPHS1, AMPHS0 = 1, 0) is selected.
- Configure the circuit of the circuit board, using material with little wiring resistance.
- Place a ground pattern that has the same potential as Vss as much as possible near the XT1 oscillator.
- Be sure that the signal lines between the XT1 and XT2 pins, and the resonators do not cross with the other signal lines. Do not route the wiring near a signal line through which a high fluctuating current flows.
- The impedance between the XT1 and XT2 pins may drop and oscillation may be disturbed due to moisture absorption of the circuit board in a high-humidity environment or dew condensation on the board. When using the circuit board in such an environment, take measures to damp-proof the circuit board, such as by coating.
- When coating the circuit board, use material that does not cause capacitance or leakage between the XT1 and XT2 pins.

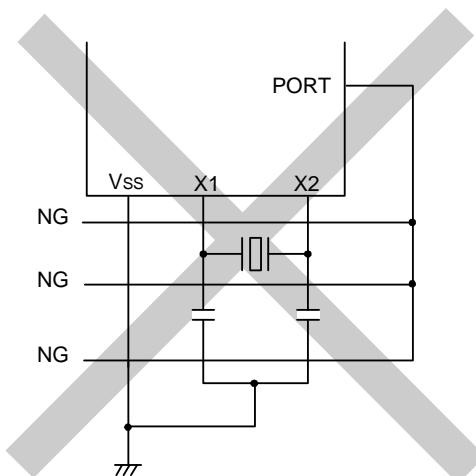
Figure 5 - 17 shows examples of incorrect resonator connection.

**Figure 5 - 17 Examples of Incorrect Resonator Connection (1/2)**

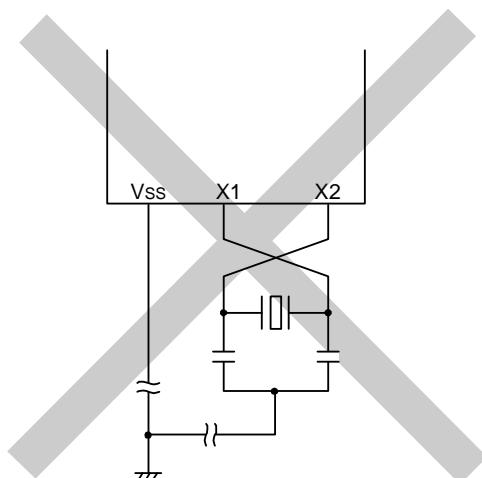
(a) Too long wiring



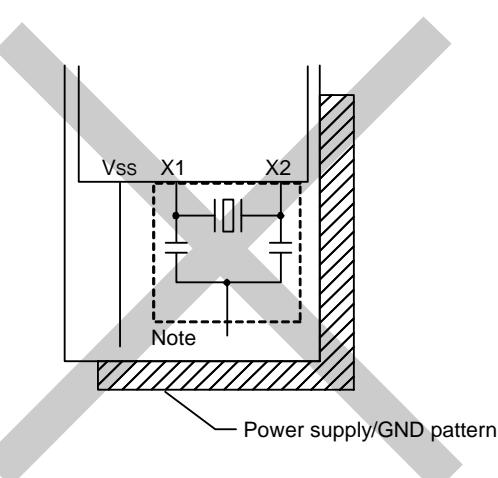
(b) Crossed signal line



(c) The X1 and X2 signal line wires cross.



(d) A power supply/GND pattern exists under the X1 and X2 wires.

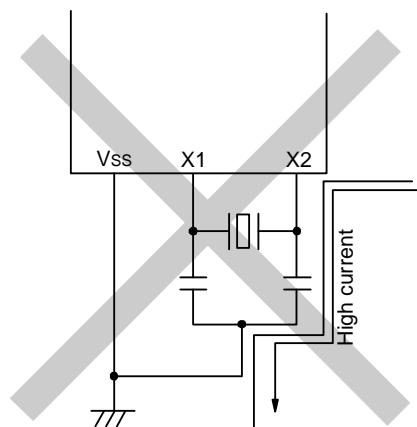
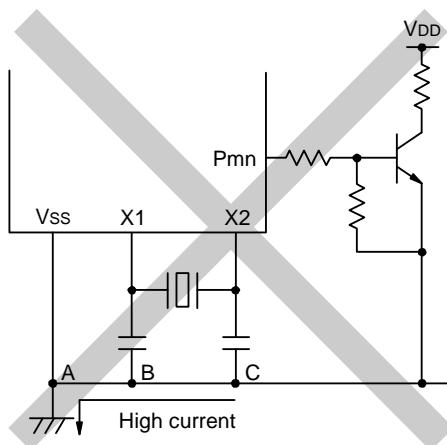


**Note** Do not place a power supply/GND pattern under the wiring section (section indicated by a broken line in the figure) of the X1 and X2 pins and the resonators in a multi-layer board or double-sided board.  
Do not configure a layout that will cause capacitance elements and affect the oscillation characteristics.

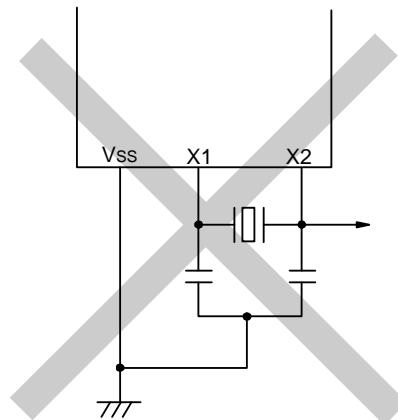
**Remark** When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

**Figure 5 - 18 Examples of Incorrect Resonator Connection (2/2)**

(e) Wiring near high alternating current

(f) Current flowing through ground line of oscillator  
(potential at points A, B, and C fluctuates)

(g) Signals are fetched



**Caution** When X2 and XT1 are wired in parallel, the crosstalk noise of X2 may increase with XT1, resulting in malfunctioning.

**Remark** When using the subsystem clock, replace X1 and X2 with XT1 and XT2, respectively. Also, insert resistors in series on the XT2 side.

### 5.4.3 High-speed on-chip oscillator

The high-speed on-chip oscillator is incorporated in the RL78/G14. The frequency can be selected from among 64, 48, 32, 24, 16, 12, 8, 6, 4, 3, 2, or 1 MHz by using the option byte (000C2H). When 64 MHz or 48 MHz is selected, the two frequency division of the selected clock is supplied to CPU clock. Oscillation can be controlled by bit 0 (HIOSTOP) of the clock operation status control register (CSC).

The high-speed on-chip oscillator automatically starts oscillating after reset release.

### 5.4.4 Low-speed on-chip oscillator

The low-speed on-chip oscillator is incorporated in the RL78/G14.

The low-speed on-chip oscillator clock is used only as the watchdog timer, real-time clock, 12-bit interval timer, and timer RJ clock. The low-speed on-chip oscillator clock cannot be used as the CPU clock.

This clock operates when bit 4 (WDTON) of the option byte (000C0H), bit 4 (WUTMMCK0) of the subsystem clock supply mode control register (OSMC), or both are set to 1.

Unless the watchdog timer is stopped and WUTMMCK0 is a value other than zero, oscillation of the low-speed on-chip oscillator continues. Note that only when the watchdog timer is operating and the WUTMMCK0 bit is 0, oscillation of the low-speed on-chip oscillator will stop while the WDSTBYON bit is 0 and operation is in the HALT, STOP, or SNOOZE mode. While the watchdog timer operates, the low-speed on-chip oscillator clock does not stop even if the program freezes.

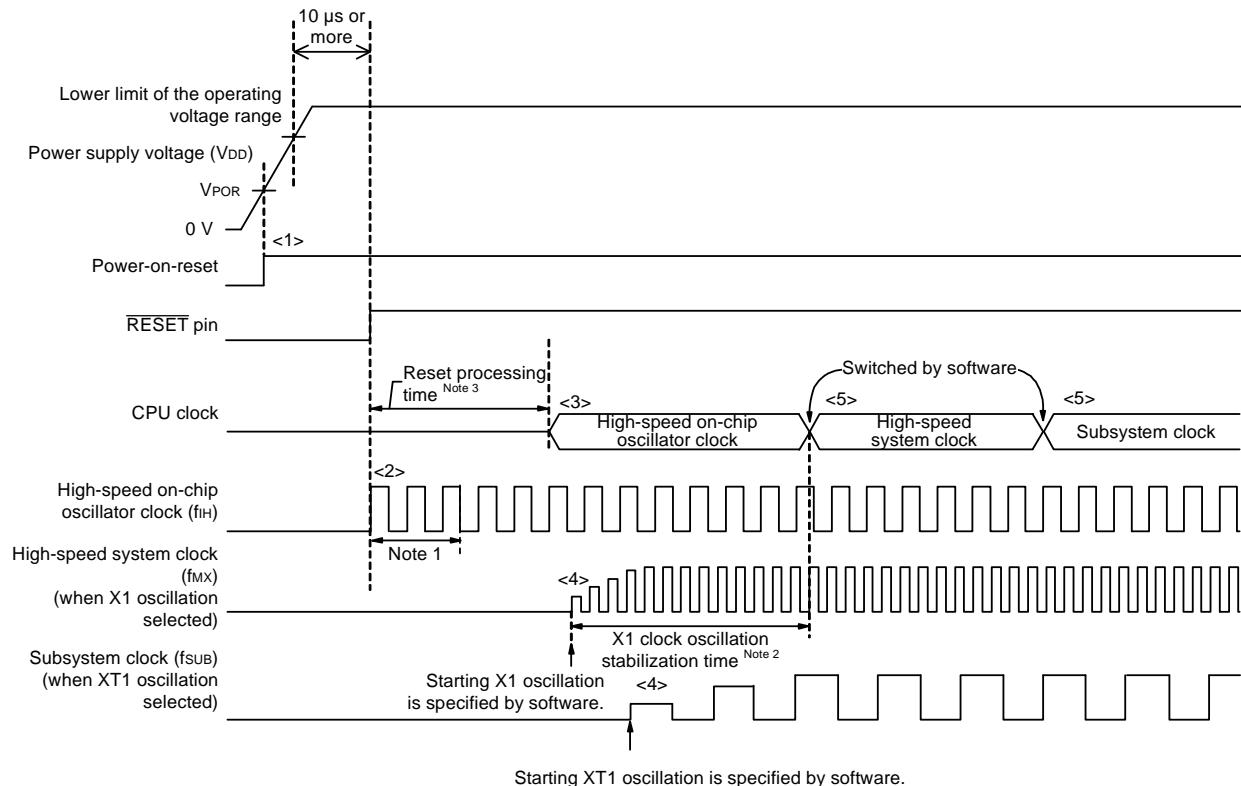
## 5.5 Clock Generator Operation

The clock generator generates the following clocks and controls the operation modes of the CPU, such as standby mode (see **Figure 5 - 1**).

- Main system clock  $f_{MAIN}$ 
  - High-speed system clock  $f_{MX}$
  - X1 clock  $f_X$
  - External main system clock  $f_{EX}$
- High-speed on-chip oscillator clock  $f_{IH}$
- Subsystem clock  $f_{SUB}$ 
  - XT1 clock  $f_{XT}$
  - External subsystem clock  $f_{EXS}$
- Low-speed on-chip oscillator clock  $f_{IL}$
- CPU/peripheral hardware clock  $f_{CLK}$

The CPU starts operation when the high-speed on-chip oscillator starts outputting after a reset release in the RL78/G14.

When the power supply voltage is turned on, the clock generator operation is shown in Figure 5 - 19.

**Figure 5 - 19 Clock Generator Operation When Power Supply Voltage Is Turned On**

- <1> When the power is turned on, an internal reset signal is generated by the power-on-reset (POR) circuit. Note that the reset state is maintained after a reset by the voltage detection circuit or an external reset until the voltage reaches the range of operating voltage described in **34.4 or 35.4 AC Characteristics** (the above figure is an example when the external reset is in use).
- <2> When the reset is released, the high-speed on-chip oscillator automatically starts oscillation.
- <3> The CPU starts operation on the high-speed on-chip oscillator clock after waiting for the voltage to stabilize and a reset processing have been performed after reset release.
- <4> Set the start of oscillation of the X1 or XT1 clock via software (see **5.6.2 Example of setting X1 oscillation clock** and **5.6.3 Example of setting XT1 oscillation clock**).
- <5> When switching the CPU clock to the X1 or XT1 clock, wait for the clock oscillation to stabilize, and then set switching via software (see **5.6.2 Example of setting X1 oscillation clock** and **5.6.3 Example of setting XT1 oscillation clock**).

- Note 1.** The internal reset processing time includes the oscillation accuracy stabilization time of the high-speed on chip oscillator clock.
- Note 2.** When releasing a reset, confirm the oscillation stabilization time for the X1 clock using the oscillation stabilization time counter status register (OSTC).
- Note 3.** For the reset processing time, see **CHAPTER 25 POWER-ON-RESET CIRCUIT**.

**Caution** It is not necessary to wait for the oscillation stabilization time when an external clock input from the EXCLK pin is used.

## 5.6 Controlling Clock

### 5.6.1 Example of setting high-speed on-chip oscillator

After a reset release, the CPU/peripheral hardware clock (fCLK) always starts operating with the high-speed on-chip oscillator clock. The frequency of the high-speed on-chip oscillator can be selected from 64, 48, 32, 24, 16, 12, 8, 6, 4, 3, 2, and 1 MHz by using FRQSEL0 to FRQSEL4 of the option byte (000C2H). In addition, Oscillation can be changed by the high-speed on-chip oscillator frequency select register (HOCODIV).

[Option byte setting]

Address: 000C2H

Option byte (000C2H)	7	6	5	4	3	2	1	0
	CMODE1 0/1	CMODE0 0/1		FRQSEL4 0/1	FRQSEL3 0/1	FRQSEL2 0/1	FRQSEL1 0/1	FRQSEL0 0/1

CMODE1	CMODE0	Setting of flash operation mode		
0	0	LV (low-voltage main) mode	VDD = 1.6 V to 5.5 V @ 1 MHz to 4 MHz	
1	0	LS (low-speed main) mode	VDD = 1.8 V to 5.5 V @ 1 MHz to 8 MHz	
1	1	HS (high-speed main) mode	VDD = 2.4 V to 5.5 V @ 1 MHz to 16 MHz VDD = 2.7 V to 5.5 V @ 1 MHz to 32 MHz	
Other than above		Setting prohibited		

FRQSEL4	FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	Frequency of the high-speed on-chip oscillator	
					f <sub>HOCO</sub>	f <sub>H</sub>
1	1	0	0	0	64 MHz	32 MHz
1	0	0	0	0	48 MHz	24 MHz
0	1	0	0	0	32 MHz	32 MHz
0	0	0	0	0	24 MHz	24 MHz
0	1	0	0	1	16 MHz	16 MHz
0	0	0	0	1	12 MHz	12 MHz
0	1	0	1	0	8 MHz	8 MHz
0	0	0	1	0	6 MHz	6 MHz
0	1	0	1	1	4 MHz	4 MHz
0	0	0	1	1	3 MHz	3 MHz
0	1	1	0	0	2 MHz	2 MHz
0	1	1	0	1	1 MHz	1 MHz
Other than above					Setting prohibited	

[High-speed on-chip oscillator frequency select register (HOCODIV) setting]

Address: F00A8H

Symbol	7	6	5	4	3	2	1	0
HOCODIV	0	0	0	0	0	HOCODIV2	HOCODIV1	HOCODIV0
Selection of high-speed on-chip oscillator clock frequency								
			FRQSEL4 = 0		FRQSEL4 = 1			
			FRQSEL3 = 0	FRQSEL3 = 1	FRQSEL3 = 0	FRQSEL3 = 1		
0	0	0	$f_{IH} = 24 \text{ MHz}$	$f_{IH} = 32 \text{ MHz}$	$f_{IH} = 24 \text{ MHz}$ $f_{HOCO} = 48 \text{ MHz}$	$f_{IH} = 32 \text{ MHz}$ $f_{HOCO} = 64 \text{ MHz}$		
0	0	1	$f_{IH} = 12 \text{ MHz}$	$f_{IH} = 16 \text{ MHz}$	$f_{IH} = 12 \text{ MHz}$ $f_{HOCO} = 24 \text{ MHz}$	$f_{IH} = 16 \text{ MHz}$ $f_{HOCO} = 32 \text{ MHz}$		
0	1	0	$f_{IH} = 6 \text{ MHz}$	$f_{IH} = 8 \text{ MHz}$	$f_{IH} = 6 \text{ MHz}$ $f_{HOCO} = 12 \text{ MHz}$	$f_{IH} = 8 \text{ MHz}$ $f_{HOCO} = 16 \text{ MHz}$		
0	1	1	$f_{IH} = 3 \text{ MHz}$	$f_{IH} = 4 \text{ MHz}$	$f_{IH} = 3 \text{ MHz}$ $f_{HOCO} = 6 \text{ MHz}$	$f_{IH} = 4 \text{ MHz}$ $f_{HOCO} = 8 \text{ MHz}$		
1	0	0	Setting prohibited	$f_{IH} = 2 \text{ MHz}$	Setting prohibited	$f_{IH} = 2 \text{ MHz}$ $f_{HOCO} = 4 \text{ MHz}$		
1	0	1	Setting prohibited	$f_{IH} = 1 \text{ MHz}$	Setting prohibited	$f_{IH} = 1 \text{ MHz}$ $f_{HOCO} = 2 \text{ MHz}$		
Other than above			Setting prohibited					

### 5.6.2 Example of setting X1 oscillation clock

After a reset release, the CPU/peripheral hardware clock (fCLK) always starts operating with the high-speed on-chip oscillator clock. To subsequently change the clock to the X1 oscillation clock, set the oscillator and start oscillation by using the oscillation stabilization time select register (OSTS), clock operation mode control register (CMC), and clock operation status control register (CSC) and wait for oscillation to stabilize by using the oscillation stabilization time counter status register (OSTC). After the oscillation stabilizes, set the X1 oscillation clock to fCLK by using the system clock control register (CKC).

[Register settings] Set the register in the order of <1> to <5> below.

- <1> Set (1) the OSCSEL bit of the CMC register, except for the cases where the fx is equal to or more than 10 MHz, in such cases set (1) the AMPH bit, to operate the X1 oscillator.

	7	6	5	4	3	2	1	0
CMC	EXCLK 0	OSCSEL 1	EXCLKS 0	OSCSELS 0	0	AMPHS1 0	AMPHS0 0	AMPH 0/1

- <2> Using the OSTS register, select the oscillation stabilization time of the X1 oscillator at releasing of the STOP mode.

Example: Setting values when a wait of at least 102 µs is set based on a 10 MHz resonator.

	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	0	OSTS2 0	OSTS1 1	OSTS0 0

- <3> Clear (0) the MSTOP bit of the CSC register to start oscillating the X1 oscillator.

	7	6	5	4	3	2	1	0
CSC	MSTOP 0	XTSTOP 1	0	0	0	0	0	HIOSTOP 0

- <4> Use the OSTC register to wait for oscillation of the X1 oscillator to stabilize.

Example: Wait until the bits reach the following values when a wait of at least 102 µs is set based on a 10 MHz resonator.

	7	6	5	4	3	2	1	0
OSTC	MOST8 1	MOST9 1	MOST10 1	MOST11 0	MOST13 0	MOST15 0	MOST17 0	MOST18 0

- <5> Use the MCM0 bit of the CKC register to specify the X1 oscillation clock as the CPU/peripheral hardware clock.

	7	6	5	4	3	2	1	0
CKC	CLS 0	CSS 0	MCS 0	MCM0 1	0	0	0	0

**Caution** Keep the operating voltage within the range that allows operation of the flash memory as set in an option byte (000C2H) before and after changes to the main system clock (fMAIN) by using the system clock control register (CKC).

Option Byte (000C2H) Value		Flash Operation Mode	Operating Frequency Range	Operating Voltage Range
CMODE1	CMODE0			
0	0	LV (low-voltage main) mode	1 MHz to 4 MHz	1.6 V to 5.5 V
1	0	LS (low-speed main) mode	1 MHz to 8 MHz	1.8 V to 5.5 V
1	1	HS (high-speed main) mode	1 MHz to 16 MHz	2.4 V to 5.5 V
			1 MHz to 32 MHz	2.7 V to 5.5 V

### 5.6.3 Example of setting XT1 oscillation clock

After a reset release, the CPU/peripheral hardware clock (fCLK) always starts operating with the high-speed on-chip oscillator clock. To subsequently change the clock to the XT1 oscillation clock, set the oscillator and start oscillation by using the subsystem clock supply mode control register (OSMC), clock operation mode control register (CMC), and clock operation status control register (CSC), set the XT1 oscillation clock to fCLK by using the system clock control register (CKC).

[Register settings] Set the register in the order of <1> to <5> below.

- <1> Set the RTCLPC bit to 1 to run only the real-time clock, and 12-bit interval timer on the subsystem clock (for ultra-low current consumption) in the STOP mode or HALT mode during CPU operation on the subsystem clock.

	7	6	5	4	3	2	1	0
OSMC	RTCLPC 0/1	0	0	WUTMMCK0 0	0	0	0	0

- <2> Set (1) the OSCSELS bit of the CMC register to operate the XT1 oscillator.

	7	6	5	4	3	2	1	0
CMC	EXCLK 0	OSCSEL 0	EXCLKS 0	OSCSELS 1	0	AMPHS1 0/1	AMPHS0 0/1	AMPH 0

AMPHS0 and AMPHS1 bits: These bits are used to specify the oscillation mode of the XT1 oscillator.

- <3> Clear (0) the XTSTOP bit of the CSC register to start oscillating the XT1 oscillator.

	7	6	5	4	3	2	1	0
CSC	MSTOP 1	XTSTOP 0	0	0	0	0	0	HIOSTOP 0

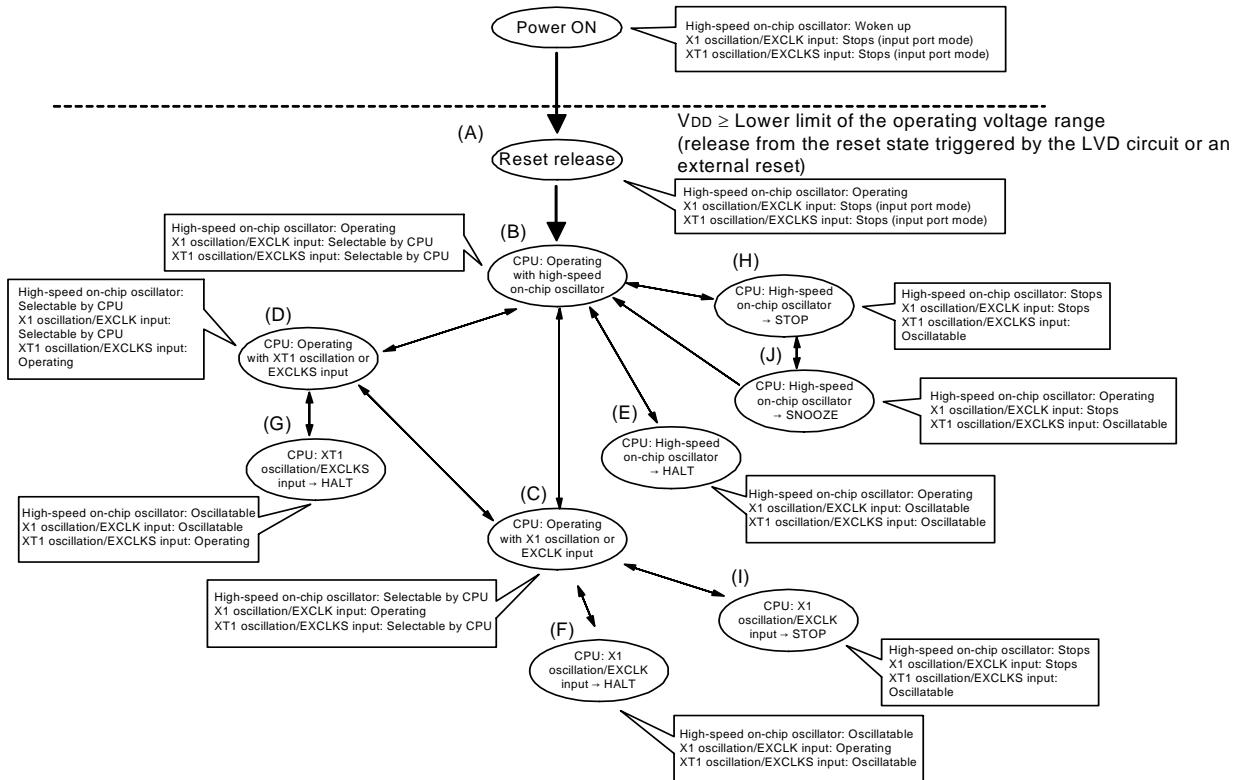
- <4> Use the timer function or another function to wait for oscillation of the subsystem clock to stabilize by using software.
- <5> Use the CSS bit of the CKC register to specify the XT1 oscillation clock as the CPU/peripheral hardware clock.

	7	6	5	4	3	2	1	0
CKC	CLS 0	CSS 1	MCS 0	MCM0 0	0	0	0	0

### 5.6.4 CPU clock status transition diagram

Figure 5 - 20 shows the CPU clock status transition diagram of this product.

**Figure 5 - 20 CPU Clock Status Transition Diagram**



Tables 5 - 3 to 5 - 7 show transition of the CPU clock and examples of setting the SFR registers.

**Table 5 - 3 CPU Clock Transition and SFR Register Setting Examples (1/5)**

- (1) CPU operating with high-speed on-chip oscillator clock (B) after reset release (A)

Status Transition	SFR Register Setting
(A) → (B)	SFR registers do not have to be set (default status after reset release).

- (2) CPU operating with high-speed system clock (C) after reset release (A)

(The CPU operates with the high-speed on-chip oscillator clock immediately after a reset release (B).)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register Status Transition	CMC Register Note 1			OSTS Register	CSC Register	CKC Register
	EXCLK	OSCSEL	AMPH		MSTOP	OSTC Register
(A) → (B) → (C) (X1 clock: 1 MHz ≤ fx ≤ 10 MHz)	0	1	0	Note 2	0	Must be checked 1
(A) → (B) → (C) (X1 clock: 10 MHz < fx ≤ 20 MHz)	0	1	1	Note 2	0	Must be checked 1
(A) → (B) → (C) (external main clock)	1	1	×	Note 2	0	Must not be checked 1

**Note 1.** The clock operation mode control register (CMC) can be written only once by an 8-bit memory manipulation instruction after reset release.

**Note 2.** Set the oscillation stabilization time as follows.

- Desired the oscillation stabilization time counter status register (OSTC) oscillation stabilization time ≤ Oscillation stabilization time set by the oscillation stabilization time select register (OSTS)

**Caution** Set the clock after the supply voltage has reached the operable voltage of the clock to be set (see CHAPTER 34 or CHAPTER 35 ELECTRICAL SPECIFICATIONS).

- (3) CPU operating with subsystem clock (D) after reset release (A)

(The CPU operates with the high-speed on-chip oscillator clock immediately after a reset release (B).)

(Setting sequence of SFR registers) →

Setting Flag of SFR Register Status Transition	CMC Register Note				CSC Register	Waiting for Oscillation Stabilization	CKC Register
	EXCLKS	OSCSELS	AMPHS1	AMPHS0	XTSTOP		CSS
(A) → (B) → (C) (XT1 clock)	0	1	0/1	0/1	0	Necessary	1
(A) → (B) → (C) (external sub clock)	1	1	×	×	0	Necessary	1

**Note** The clock operation mode control register (CMC) can be written only once by an 8-bit memory manipulation instruction after reset release.

**Remark 1.** ×: Don't care

**Remark 2.** (A) to (J) in Tables 5 - 3 to 5 - 7 correspond to (A) to (J) in Figure 5 - 20.

**Table 5 - 4 CPU Clock Transition and SFR Register Setting Examples (2/5)**

(4) CPU clock changing from high-speed on-chip oscillator clock (B) to high-speed system clock (C)

Status Transition	(Setting sequence of SFR registers)					
	CMC Register Note 1			OSTS Register	CSC Register	OSTC Register
	EXCLK	OSCSEL	AMPH			
(B) → (C) (XT1 clock: 1 MHz ≤ fx ≤ 10 MHz)	0	1	0	Note 2	0	Must be checked
(B) → (C) (XT1 clock: 10 MHz < fx ≤ 20 MHz)	0	1	1	Note 2	0	Must be checked
(B) → (C) (external main clock)	1	1	x	Note 2	0	Need not be checked

Unnecessary if these registers are already set      Unnecessary if the CPU is operating with the high-speed system clock

**Note 1.** The clock operation mode control register (CMC) can be changed only once after reset release. This setting is not necessary if it has already been set.

**Note 2.** Set the oscillation stabilization time as follows.

- Desired the oscillation stabilization time counter status register (OSTC) oscillation stabilization time ≤ Oscillation stabilization time set by the oscillation stabilization time select register (OSTS)

**Caution** Set the clock after the supply voltage has reached the operable voltage of the clock to be set (see CHAPTER 34 or CHAPTER 35 ELECTRICAL SPECIFICATIONS).

(5) CPU clock changing from high-speed on-chip oscillator clock (B) to subsystem clock (D)

Status Transition	(Setting sequence of SFR registers)					
	CMC Register Note			CSC Register	Waiting for Oscillation Stabilization	CKC Register
	EXCLKS	OSCSELS	AMPHS1,0			
(B) → (D) (XT1 clock)	0	1	00: Low power consumption oscillation 01: Normal oscillation 10: Ultra-low power consumption oscillation	0	Necessary	1
(B) → (D) (external sub clock)	1	1	x	0	Necessary	1

Unnecessary if these registers are already set      Unnecessary if the CPU is operating with the subsystem clock

**Note** The clock operation mode control register (CMC) can be written only once by an 8-bit memory manipulation instruction after reset release. This setting is not necessary if it has already been set.

**Remark 1.** x: Don't care

**Remark 2.** (A) to (J) in Tables 5 - 3 to 5 - 7 correspond to (A) to (J) in Figure 5 - 20.

**Table 5 - 5 CPU Clock Transition and SFR Register Setting Examples (3/5)**

- (6) CPU clock changing from high-speed system clock (C) to high-speed on-chip oscillator clock (B)

(Setting sequence of SFR registers)				
Status Transition	Setting Flag of SFR Register	CSC Register	Oscillation accuracy stabilization time	CKC Register
		HIOSTOP	MCM0	
(C) → (B)		0	Note	0

Unnecessary if the CPU is operating with the high-speed on-chip oscillator clock

**Note** When FRQSEL4 = 0: 18 µs to 65 µs  
When FRQSEL4 = 1: 18 µs to 135 µs

**Remark** The oscillation accuracy stabilization time changes according to the temperature conditions and the STOP mode period.

- (7) CPU clock changing from high-speed system clock (C) to subsystem clock (D)

(Setting sequence of SFR registers)				
Status Transition	Setting Flag of SFR Register	CSC Register	Waiting for Oscillation Stabilization	CKC Register
		XTSTOP	CSS	
(C) → (D)		0	Necessary	1

Unnecessary if the CPU is operating with the subsystem clock

- (8) CPU clock changing from subsystem clock (D) to high-speed on-chip oscillator clock (B)

(Setting sequence of SFR registers)				
Status Transition	Setting Flag of SFR Register	CSC Register	Waiting for Oscillation Stabilization	CKC Register
		HIOSTOP	CSS	
(D) → (B)		0	Note	0

Unnecessary if the CPU is operating with the high-speed on-chip oscillator clock

**Note** When FRQSEL4 = 0: 18 µs to 65 µs  
When FRQSEL4 = 1: 18 µs to 135 µs

**Remark 1.** (A) to (J) in Tables 5 - 3 to 5 - 7 correspond to (A) to (J) in Figure 5 - 20.

**Remark 2.** The oscillation accuracy stabilization time changes according to the temperature conditions and the STOP mode period.

**Table 5 - 6 CPU Clock Transition and SFR Register Setting Examples (4/5)**

(9) CPU clock changing from subsystem clock (D) to high-speed system clock (C)

Status Transition	Setting Flag of SFR Register	OSTS Register	CSC Register	OSTC Register	CKC Register
			MSTOP		CSS
(D) → (C) (X1 clock: 1 MHz ≤ fx ≤ 10 MHz)	Note		0	Must be checked	0
(D) → (C) (X1 clock: 10 MHz < fx ≤ 20 MHz)	Note		0	Must be checked	0
(D) → (C) (external main clock)	Note		0	Need not be checked	0

(Setting sequence of SFR registers) →

Unnecessary if the CPU is operating with the high-speed system clock

**Note** Set the oscillation stabilization time as follows.

- Desired the oscillation stabilization time counter status register (OSTC) oscillation stabilization time ≤ Oscillation stabilization time set by the oscillation stabilization time select register (OSTS)

**Caution** Set the clock after the supply voltage has reached the operable voltage of the clock to be set (see CHAPTER 34 or CHAPTER 35 ELECTRICAL SPECIFICATIONS).

- (10)
- HALT mode (E) set while CPU is operating with high-speed on-chip oscillator clock (B)
  - HALT mode (F) set while CPU is operating with high-speed system clock (C)
  - HALT mode (G) set while CPU is operating with subsystem clock (D)

Status Transition	Setting
(B) → (E) (C) → (F) (D) → (G)	Executing HALT instruction

**Remark** (A) to (J) in Tables 5 - 3 to 5 - 7 correspond to (A) to (J) in Figure 5 - 20.

**Table 5 - 7 CPU Clock Transition and SFR Register Setting Examples (5/5)**

- (11) • STOP mode (H) set while CPU is operating with high-speed on-chip oscillator clock (B)  
     • STOP mode (I) set while CPU is operating with high-speed system clock (C)

(Setting sequence) →		Setting			
Status Transition		Stopping peripheral functions that cannot operate in STOP mode	—	Executing STOP instruction	
(B) → (H)			—		
(C) → (I)			Sets the OSTS register		
In X1 oscillation			—		
External main system clock			—		

- (12) CPU changing from STOP mode (H) to SNOOZE mode (J)

For details about the setting for switching from the STOP mode to the SNOOZE mode, see **14.8 SNOOZE Mode Function**, **17.5.7 SNOOZE mode function**, and **17.7.3 SNOOZE mode function**.

**Remark** (A) to (J) in Tables 5 - 3 to 5 - 7 correspond to (A) to (J) in Figure 5 - 20.

## 5.6.5 Condition before changing CPU clock and processing after changing CPU clock

Condition before changing the CPU clock and processing after changing the CPU clock are shown below.

**Table 5 - 8 Changing CPU Clock (1/2)**

CPU Clock		Condition Before Change	Processing After Change
Before Change	After Change		
High-speed on-chip oscillator clock	X1 clock	Stabilization of X1 oscillation • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • After elapse of oscillation stabilization time	The operating current can be reduced by stopping the high-speed on-chip oscillator (HIOSTOP = 1) after checking that the CPU clock is changed.
	External main system clock	Enabling input of external clock from the EXCLK pin • OSCSEL = 1, EXCLK = 1, MSTOP = 0	
	XT1 clock	Stabilization of XT1 oscillation • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • After elapse of oscillation stabilization time	
	External subsystem clock	Enabling input of external clock from the EXCLKS pin • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	
X1 clock	High-speed on-chip oscillator clock	Enabling oscillation of high-speed on-chip oscillator • HIOSTOP = 0 • After elapse of oscillation stabilization time	X1 oscillation can be stopped (MSTOP = 1) after checking that the CPU clock is changed.
	External main system clock	Transition not possible	—
	XT1 clock	Stabilization of XT1 oscillation • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • After elapse of oscillation stabilization time	X1 oscillation can be stopped (MSTOP = 1) after checking that the CPU clock is changed.
	External subsystem clock	Enabling input of external clock from the EXCLKS pin • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	X1 oscillation can be stopped (MSTOP = 1) after checking that the CPU clock is changed.
External main system clock	High-speed on-chip oscillator clock	Enabling oscillation of high-speed on-chip oscillator • HIOSTOP = 0 • After elapse of oscillation stabilization time	External main system clock input can be disabled (MSTOP = 1).
	X1 clock	Transition not possible	—
	XT1 clock	Stabilization of XT1 oscillation • OSCSELS = 1, EXCLKS = 0, XTSTOP = 0 • After elapse of oscillation stabilization time	External main system clock input can be disabled (MSTOP = 1).
	External subsystem clock	Enabling input of external clock from the EXCLKS pin • OSCSELS = 1, EXCLKS = 1, XTSTOP = 0	External main system clock input can be disabled (MSTOP = 1).

**Table 5 - 9 Changing CPU Clock (2/2)**

CPU Clock		Condition Before Change	Processing After Change
Before Change	After Change		
XT1 clock	High-speed on-chip oscillator clock	Oscillation of high-speed on-chip oscillator and selection of high-speed on-chip oscillator clock as main system clock • HIOSTOP = 0, MCS = 0	XT1 oscillation can be stopped (XTSTOP = 1)
	X1 clock	Stabilization of X1 oscillation and selection of high-speed system clock as main system clock • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • After elapse of oscillation stabilization time • MCS = 1	
	External main system clock	Enabling input of external clock from the EXCLK pin and selection of high-speed system clock as main system clock • OSCSEL = 1, EXCLK = 1, MSTOP = 0 • MCS = 1	
	External subsystem clock	Transition not possible	
External subsystem clock	High-speed on-chip oscillator clock	Oscillation of high-speed on-chip oscillator and selection of high-speed on-chip oscillator clock as main system clock • HIOSTOP = 0, MCS = 0	External subsystem clock input can be disabled (XTSTOP = 1).
	X1 clock	Stabilization of X1 oscillation and selection of high-speed system clock as main system clock • OSCSEL = 1, EXCLK = 0, MSTOP = 0 • After elapse of oscillation stabilization time • MCS = 1	
	External main system clock	Enabling input of external clock from the EXCLK pin and selection of high-speed system clock as main system clock • OSCSEL = 1, EXCLK = 1, MSTOP = 0 • MCS = 1	
	XT1 clock	Transition not possible	

### 5.6.6 Time required for switchover of CPU clock and main system clock

By setting bits 4 and 6 (MCM0, CSS) of the system clock control register (CKC), the CPU clock can be switched (between the main system clock and the subsystem clock), and main system clock can be switched (between the high-speed on-chip oscillator clock and the high-speed system clock).

The actual switchover operation is not performed immediately after rewriting to the CKC register; operation continues on the pre-swatchover clock for several clocks (see **Tables 5 - 10 to 5 - 12**).

Whether the CPU is operating on the main system clock or the subsystem clock can be ascertained using bit 7 (CLS) of the CKC register. Whether the main system clock is operating on the high-speed system clock or high-speed on-chip oscillator clock can be ascertained using bit 5 (MCS) of the CKC register.

When the CPU clock is switched, the peripheral hardware clock is also switched.

**Table 5 - 10 Maximum Time Required for Main System Clock Switchover**

Clock A	Switching directions	Clock B	Remark
$f_{IH}$	$\leftrightarrow$	$f_{MX}$	See <b>Table 5 - 11</b>
$f_{MAIN}$	$\leftrightarrow$	$f_{SUB}$	See <b>Table 5 - 12</b>

**Table 5 - 11 Maximum Number of Clocks Required for  $f_{IH} \leftrightarrow f_{MX}$**

Set Value Before Switchover		Set Value After Switchover	
MCM0		MCM0	
		0 ( $f_{MAIN} = f_{IH}$ )	1 ( $f_{MAIN} = f_{MX}$ )
0 ( $f_{MAIN} = f_{IH}$ )	$f_{MX} \geq f_{IH}$		2 clock
	$f_{MX} < f_{IH}$		$1 + f_{IH}/f_{MX}$ clock
1 ( $f_{MAIN} = f_{IH}$ )	$f_{MX} \geq f_{IH}$	2 $f_{MX}/f_{IH}$ clock	
	$f_{MX} < f_{IH}$	2 clock	

**Table 5 - 12 Maximum Number of Clocks Required for  $f_{MAIN} \leftrightarrow f_{SUB}$**

Set Value Before Switchover		Set Value After Switchover	
CSS		CSS	
		0 ( $f_{CLK} = f_{MAIN}$ )	1 ( $f_{CLK} = f_{SUB}$ )
0 ( $f_{CLK} = f_{MAIN}$ )			$1 + 2 f_{MAIN}/f_{SUB}$ clock
1 ( $f_{CLK} = f_{SUB}$ )		3 clock	

**Remark 1.** The number of clocks listed in Tables 5 - 11 and 5 - 12 is the number of CPU clocks before switchover.

**Remark 2.** Calculate the number of clocks in Tables 5 - 11 and 5 - 12 by rounding up the number after the decimal position.

Example When switching the main system clock from the high-speed system clock to the high-speed on-chip oscillator clock (@ oscillation with  $f_{IH} = 8$  MHz,  $f_{MX} = 10$  MHz)  
 $2 f_{MX}/f_{IH} = 2 (10/8) = 2.5 \rightarrow 3$  clocks

### 5.6.7 Conditions before clock oscillation is stopped

The following lists the register flag settings for stopping the clock oscillation (disabling external clock input) and conditions before the clock oscillation is stopped. Before stopping the clock oscillation, check the conditions before the clock oscillation is stopped.

**Table 5 - 13 Conditions Before the Clock Oscillation Is Stopped and Flag Settings**

Clock	Conditions Before Clock Oscillation Is Stopped (External Clock Input Disabled)	Flag Settings of SFR Register
High-speed on-chip oscillator clock	MCS = 1 or CLS = 1 (The CPU is operating on a clock other than the high-speed on-chip oscillator clock.)	HIOSTOP = 1
X1 clock	MCS = 0 or CLS = 1	MSTOP = 1
External main system clock	(The CPU is operating on a clock other than the high-speed system clock.)	
XT1 clock	CLS = 0	XTSTOP = 1
External subsystem clock	(The CPU is operating on a clock other than the subsystem clock.)	

## 5.7 Resonator and Oscillator Constants

The resonators for which the operation is verified and their oscillator constants are shown below.

**Caution 1.** The constants for these oscillator circuits are reference values based on specific environments set up for evaluation by the manufacturers. For actual applications, request evaluation by the manufacturer of the oscillator circuit mounted on a board. Furthermore, if you are switching from a different product to this microcontroller, and whenever you change the board, again request evaluation by the manufacturer of the oscillator circuit mounted on the new board.

**Caution 2.** The oscillation voltage and oscillation frequency only indicate the oscillator characteristic. Use the RL78 microcontroller so that the internal operation conditions are within the specifications of the DC and AC characteristics.

Figure 5 - 21 Example of External Circuit



## (1) X1 oscillation

As of Mar 2016

Manufacturer	Resonator	Part Number	SMD/ Lead	Frequency (MHz)	Flash Operation Mode Note 1	Circuit Constants (Reference) Note 2			Voltage Range (V)	
						C1 (pF)	C2 (pF)	Rd (kΩ)	MIN.	MAX.
Murata Manufacturing Co., Ltd. Note 3	Crystal resonator	CSTCR4M00G55-R0	SMD	4.0	LV	(39)	(39)	0	1.6	5.5
		CSTLS4M00G53-B0	Lead			(15)	(15)	0		
		CSTCR4M00G55-R0	SMD	4.0	LS	(39)	(39)	0	1.8	5.5
		CSTLS4M00G53-B0	Lead			(15)	(15)	0		
		CSTCR5M00G53-R0	SMD	5.0		(15)	(15)	0		
		CSTLS5M00G53-B0	Lead			(15)	(15)	0		
		CSTCR6M00G53-R0	SMD	6.0		(15)	(15)	0		
		CSTLS6M00G53-B0	Lead			(15)	(15)	0		
		CSTCE8M00G52-R0	SMD	8.0		(10)	(10)	0		
		CSTLS8M00G53-B0	Lead			(15)	(15)	0		
		CSTCR5M00G53-R0	SMD	5.0	HS	(15)	(15)	0	2.4	5.5
		CSTLS5M00G53-B0	Lead			(15)	(15)	0		
		CSTCR6M00G53-R0	SMD	6.0		(15)	(15)	0		
		CSTLS6M00G53-B0	Lead			(15)	(15)	0		
		CSTCE8M00G52-R0	SMD	8.0		(10)	(10)	0		
		CSTLS8M00G53-B0	Lead			(15)	(15)	0		
		CSTCE10M0G52-R0	SMD	10.0		(10)	(10)	0		
		CSTLS10M0G53-B0	Lead			(15)	(15)	0		
		CSTCE16M0V53-R0	SMD	16.0		(15)	(15)	0		
		CSTLS16M0X51-B0	Lead			(5)	(5)	0		
		CSTCE20M0V51-R0	SMD	20.0	HS	(5)	(5)	0	2.7	5.5
		CSTLS20M0X51-B0	Lead			(5)	(5)	0		
Nihon Dempa Kogyo Co., Ltd. Note 4	Crystal resonator	NX8045GB	SMD	8	LS	Note 4			1.8	5.5
		NX8045GB	SMD	8	HS				2.4	5.5
		NX5032GA	SMD	16	HS				2.7	5.5
		NX3225HA	SMD	20	HS					
RIVER ELETEC CORPORATION	Crystal resonator	FCX-03-8.000 MHZ-J21140 Note 5	SMD	8.0	HS	3	3	0	2.4	5.5
		FCX-04C-10.000 MHZ-J21139 Note 5	SMD	10.0		4	4	0		
		FCX-05-12.000 MHZ-J21138 Note 5	SMD	12.0		6	6	0		
		FCX-06-16.000 MHZ-J21137 Note 5	SMD	16.0		4	4	0		

**Note 1.** Set the flash operation mode by using the CMODE1 and CMODE0 bits of the option byte (000C2H).

**Note 2.** Values in parentheses in the C1 and C2 columns indicate an internal capacitance.

**Note 3.** When using these resonators, contact Murata Manufacturing Company, Ltd. (<http://www.murata.co.jp>) for more information on matching.

Also, products compatible with 105°C have different part numbers. Contact Murata Manufacturing Company, Ltd. for more information.

**Note 4.** When using these resonators, contact Nihon Dempa Kogyo Co., Ltd (<http://www.ndk.com/en>) for more information on matching.

**Note 5.** When using this resonator, for details about the matching, contact RIVER ELETEC CORPORATION (<http://www.river-ele.co.jp/english/index.html>).

**Remark 1.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (High-speed main) mode:  $2.7 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$  @ 1 MHz to 32 MHz

$2.4 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$  @ 1 MHz to 16 MHz

LS (Low-speed main) mode:  $1.8 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$  @ 1 MHz to 8 MHz

LV (Low-voltage main) mode:  $1.6 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$  @ 1 MHz to 4 MHz

**Remark 2.** A list of the resonators for which the operation has most recently been verified and their oscillation constants (for reference) is provided on the page for the corresponding product at the Renesas Web site (<http://www.renesas.com>).

## (2) XT1 oscillation (crystal resonator)

As of Mar 2016

Manufacturer	Part Number	SMD/ Lead	Frequency (kHz)	Load Capacitance CL (pF)	X1 oscillation Mode Note 1	Circuit Constants (Reference)			Voltage Range (V)		
						C3 (pF)	C4 (pF)	Rd (kΩ)	MIN.	MAX.	
Seiko Instruments Inc. Note 2	SSP-T7-FL	SMD	32.768	6.0	Normal oscillation	10	9	0	1.6	5.5	
				4.4	Low power consumption oscillation	7	5	0			
				3.7	Ultra-low power consumption oscillation	6	3	0			
	VT-200-FL	Lead		6.0	Normal oscillation	10	9	0			
				4.4	Low power consumption oscillation	7	5	0			
				3.7	Ultra-low power consumption oscillation	6	3	0			
Nihon Dempa Kogyo Co., Ltd. Note 3	NX3215SA	SMD	32.768	6.0	Normal oscillation	Note 3			1.6	5.5	
RIVER ELETEC CORPORATION	TFX-02- 32.768 KHZ- J20986 Note 4	SMD	32.768	9	Low power consumption oscillation						
					Ultra-low power consumption oscillation						
	TFX-03- 32.768 KHZ- J13375 Note 4	SMD	32.768	9	Normal oscillation	12	10	0	1.6	5.5	
					Low power consumption oscillation	12	10	0			

**Note 1.** Set the XT1 oscillation mode by using the AMPHS0 and AMPHS1 bits of the clock operation mode control register (CMC).

**Note 2.** When using these resonators, contact Seiko Instruments Inc., Ltd (<http://www.sii-crystal.com>) for more information on matching. Operation has been confirmed only in RL78/G14 products with 16 to 256 Kbytes of ROM.

**Note 3.** When using these resonators, contact Nihon Dempa Kogyo Co., Ltd (<http://www.ndk.com/en>) for more information on matching.

**Note 4.** When using this resonator, for details about the matching, contact RIVER ELETEC CORPORATION (<http://www.river-ele.co.jp/english/index.html>).

**Remark** A list of the resonators for which the operation has most recently been verified and their oscillation constants (for reference) is provided on the page for the corresponding product at the Renesas Web site (<http://www.renesas.com>).

## CHAPTER 6 TIMER ARRAY UNIT

The number of units or channels of the timer array unit differs, depending on the product.

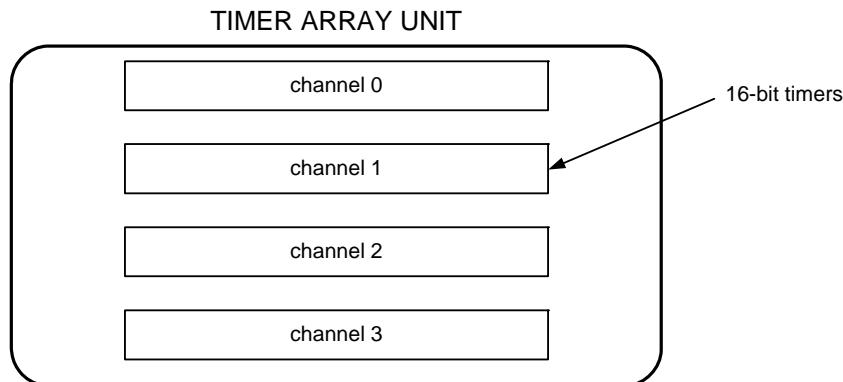
Units	Channels	30, 32, 36, 40, 44, 48, 52, 64-pin	80, 100-pin
Unit 0	Channel 0	√	√
	Channel 1	√	√
	Channel 2	√	√
	Channel 3	√	√
Unit 1	Channel 0	—	√
	Channel 1	—	√
	Channel 2	—	√
	Channel 3	—	√

**Caution 1.** The presence or absence of timer I/O pins depends on the product. See Table 6 - 2 Timer I/O Pins provided in Each Product for details.

**Caution 2.** Most of the following descriptions in this chapter use the 100-pin products as an example.

The timer array unit has four 16-bit timers.

Each 16-bit timer is called a channel and can be used as an independent timer. In addition, two or more “channels” can be used to create a high-accuracy timer.



For details about each function, see the table below.

Independent channel operation function	Simultaneous channel operation function
<ul style="list-style-type: none"> <li>• Interval timer (→ refer to <b>6.8.1</b>)</li> <li>• Square wave output (→ refer to <b>6.8.1</b>)</li> <li>• External event counter (→ refer to <b>6.8.2</b>)</li> <li>• Divider Note (→ refer to <b>6.8.3</b>)</li> <li>• Input pulse interval measurement (→ refer to <b>6.8.4</b>)</li> <li>• Measurement of high-/low-level width of input signal (→ refer to <b>6.8.5</b>)</li> <li>• Delay counter (→ refer to <b>6.8.6</b>)</li> </ul>	<ul style="list-style-type: none"> <li>• One-shot pulse output (→ refer to <b>6.9.1</b>)</li> <li>• PWM output (→ refer to <b>6.9.2</b>)</li> <li>• Multiple PWM output (→ refer to <b>6.9.3</b>)</li> </ul>

**Note** Only channel 0 of unit 0.

It is possible to use the 16-bit timer of channels 1 and 3 of the units 0 and 1 as two 8-bit timers (higher and lower). The functions that can use channels 1 and 3 as 8-bit timers are as follows:

- Interval timer (upper or lower 8-bit timer)/square wave output (lower 8-bit timer only)
- External event counter (lower 8-bit timer only)
- Delay counter (lower 8-bit timer only)

Channel 3 of unit 0 can be used to realize LIN-bus communication operating in combination with UART0 of the serial array unit.

## 6.1 Functions of Timer Array Unit

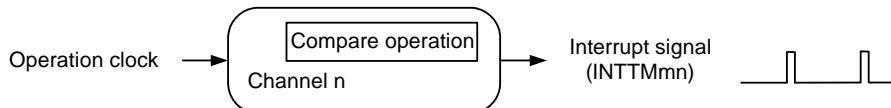
Timer array unit has the following functions.

### 6.1.1 Independent channel operation function

By operating a channel independently, it can be used for the following purposes without being affected by the operation mode of other channels.

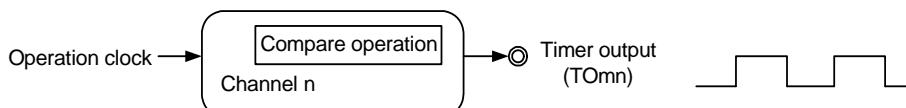
(1) Interval timer

Each timer of a unit can be used as a reference timer that generates an interrupt (INTTM<sub>Mn</sub>) at fixed intervals.



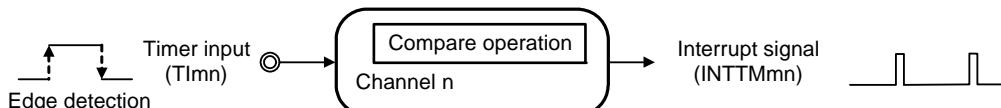
(2) Square wave output

A toggle operation is performed each time INTTM<sub>Mn</sub> interrupt is generated and a square wave with a duty factor of 50% is output from a timer output pin (TO<sub>Mn</sub>).



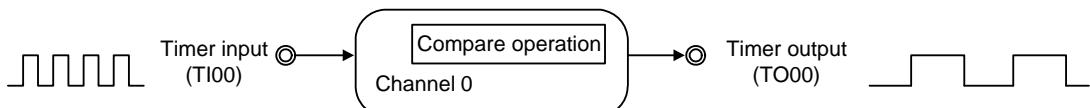
(3) External event counter

Each timer of a unit can be used as an event counter that generates an interrupt when the number of the valid edges of a signal input to the timer input pin (TI<sub>Mn</sub>) has reached a specific value.



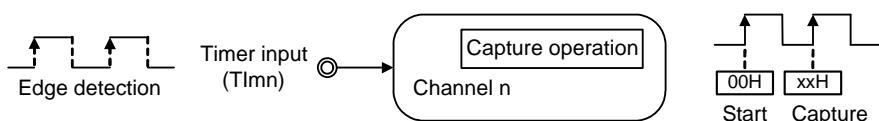
(4) Divider function (channel 0 only)

A clock input from a timer input pin (TI<sub>00</sub>) is divided and output from an output pin (TO<sub>00</sub>).



(5) Input pulse interval measurement

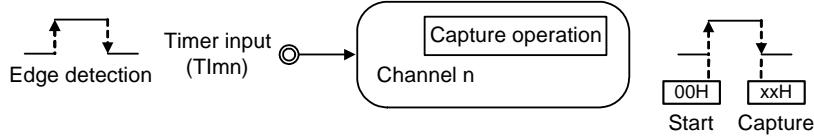
Counting is started by the valid edge of a pulse signal input to a timer input pin (TI<sub>Mn</sub>). The count value of the timer is captured at the valid edge of the next pulse. In this way, the interval of the input pulse can be measured.



(Remarks are listed on the next page.)

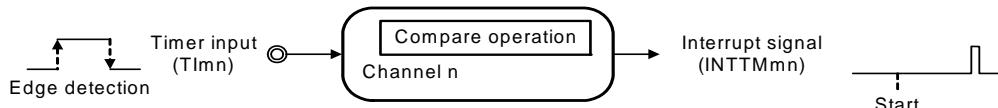
(6) Measurement of high-/low-level width of input signal

Counting is started by a single edge of the signal input to the timer input pin (TImn), and the count value is captured at the other edge. In this way, the high-level or low-level width of the input signal can be measured.



(7) Delay counter

Counting is started at the valid edge of the signal input to the timer input pin (TImn), and an interrupt is generated after any delay period.



**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

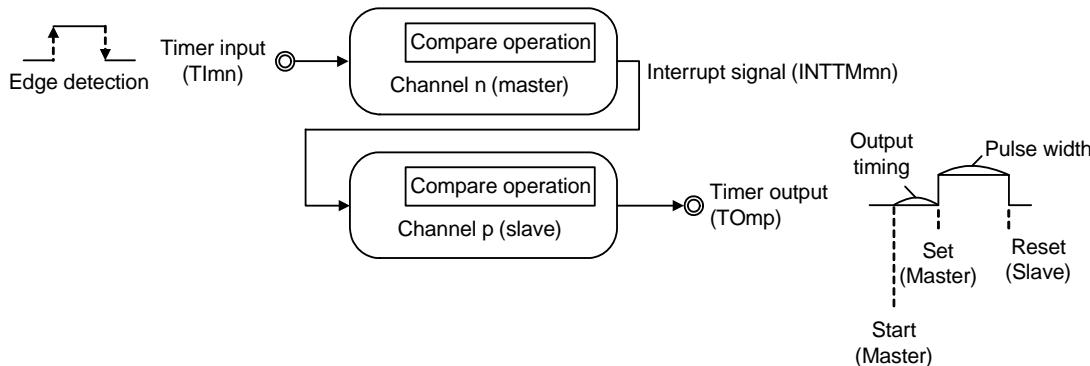
**Remark 2.** The presence or absence of timer I/O pins of channel 0 to 3 depends on the product. See **Table 6 - 2 Timer I/O Pins provided in Each Product** for details.

### 6.1.2 Simultaneous channel operation function

By using the combination of a master channel (a reference timer mainly controlling the cycle) and slave channels (timers operating according to the master channel), channels can be used for the following purposes.

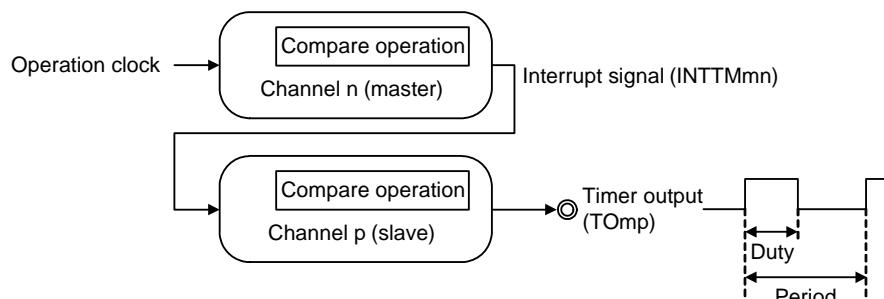
(1) One-shot pulse output

Two channels are used as a set to generate a one-shot pulse with a specified output timing and a specified pulse width.



(2) PWM (Pulse Width Modulation) output

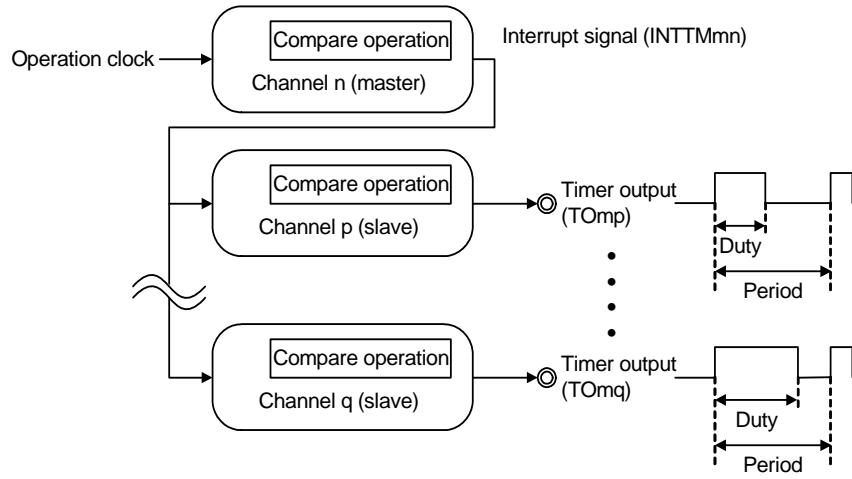
Two channels are used as a set to generate a pulse with a specified period and a specified duty factor.



(Caution and Remark are listed on the next page.)

## (3) Multiple PWM (Pulse Width Modulation) output

By extending the PWM function and using one master channel and two or more slave channels, up to three types of PWM signals that have a specific period and a specified duty factor can be generated.



**Caution** For details about the rules of simultaneous channel operation function, see 6.4.1 Basic rules of simultaneous channel operation function.

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ),  
p, q: Slave channel number ( $n < p < q \leq 3$ )

### 6.1.3 8-bit timer operation function (channels 1 and 3 only)

The 8-bit timer operation function makes it possible to use a 16-bit timer channel in a configuration consisting of two 8-bit timer channels. This function can only be used for channels 1 and 3.

**Caution** There are several rules for using 8-bit timer operation function.

For details, see 6.4.2 Basic rules of 8-bit timer operation function (channels 1 and 3 only).

### 6.1.4 LIN-bus supporting function (channel 3 of unit 0 only)

Timer array unit is used to check whether signals received in LIN-bus communication match the LIN-bus communication format.

(1) Detection of wakeup signal

The timer starts counting at the falling edge of a signal input to the serial data input pin (RxD0) of UART0 and the count value of the timer is captured at the rising edge. In this way, a low-level width can be measured. If the low-level width is greater than a specific value, it is recognized as a wakeup signal.

(2) Detection of break field

The timer starts counting at the falling edge of a signal input to the serial data input pin (RxD0) of UART0 after a wakeup signal is detected, and the count value of the timer is captured at the rising edge. In this way, a low-level width is measured. If the low-level width is greater than a specific value, it is recognized as a break field.

(3) Measurement of pulse width of sync field

After a break field is detected, the low-level width and high-level width of the signal input to the serial data input pin (RxD0) of UART0 are measured. From the bit interval of the sync field measured in this way, a baud rate is calculated.

**Remark** For details about setting up the operations used to implement the LIN-bus, see **6.3.13 Input switch control register (ISC)** and **6.8.5 Operation as input signal high-/low-level width measurement**.

## 6.2 Configuration of Timer Array Unit

Timer array unit includes the following hardware.

**Table 6 - 1 Configuration of Timer Array Unit**

Item	Configuration
Timer/counter	Timer count register mn (TCRmn)
Register	Timer data register mn (TDRmn)
Timer input	TI00 to TI03, TI10 to TI13 Note 1, RxD0 pin (for LIN-bus)
Timer output	TO00 to TO03, TO10 to TO13 pins Note 1, output controller
Control registers	<ul style="list-style-type: none"> <li>&lt;Registers of unit setting block&gt;           <ul style="list-style-type: none"> <li>• Peripheral enable register 0 (PER0)</li> <li>• Timer clock select register m (TPSm)</li> <li>• Timer channel enable status register m (TEM)</li> <li>• Timer channel start register m (TSm)</li> <li>• Timer channel stop register m (TTm)</li> <li>• Timer input select register 0 (TIS0)</li> <li>• Timer output enable register m (TOEm)</li> <li>• Timer output register m (TOm)</li> <li>• Timer output level register m (TOLm)</li> <li>• Timer output mode register m (TOMm)</li> </ul> </li>   <li>&lt;Registers of each channel&gt;           <ul style="list-style-type: none"> <li>• Timer mode register mn (TMRmn)</li> <li>• Timer status register mn (TSRmn)</li> <li>• Input switch control register (ISC)</li> <li>• Noise filter enable registers 1, 2 (NFEN1, NFEN2)</li> <li>• Port mode control register (PMCxx) Note 2</li> <li>• Port mode register (PMxx) Note 2</li> <li>• Port register (Pxx) Note 2</li> </ul> </li> </ul>

**Note 1.** The presence or absence of timer I/O pins of channel 0 to 3 depends on the product. See **Table 6 - 2 Timer I/O Pins provided in Each Product** for details.

**Note 2.** The port mode control register (PMCxx), port mode registers (PMxx) and port registers (Pxx) to be set differ depending on the product. For details, see **4.5 Register Settings When Using Alternate Function**.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

The presence or absence of timer I/O pins in each timer array unit channel depends on the product.

**Table 6 - 2 Timer I/O Pins provided in Each Product**

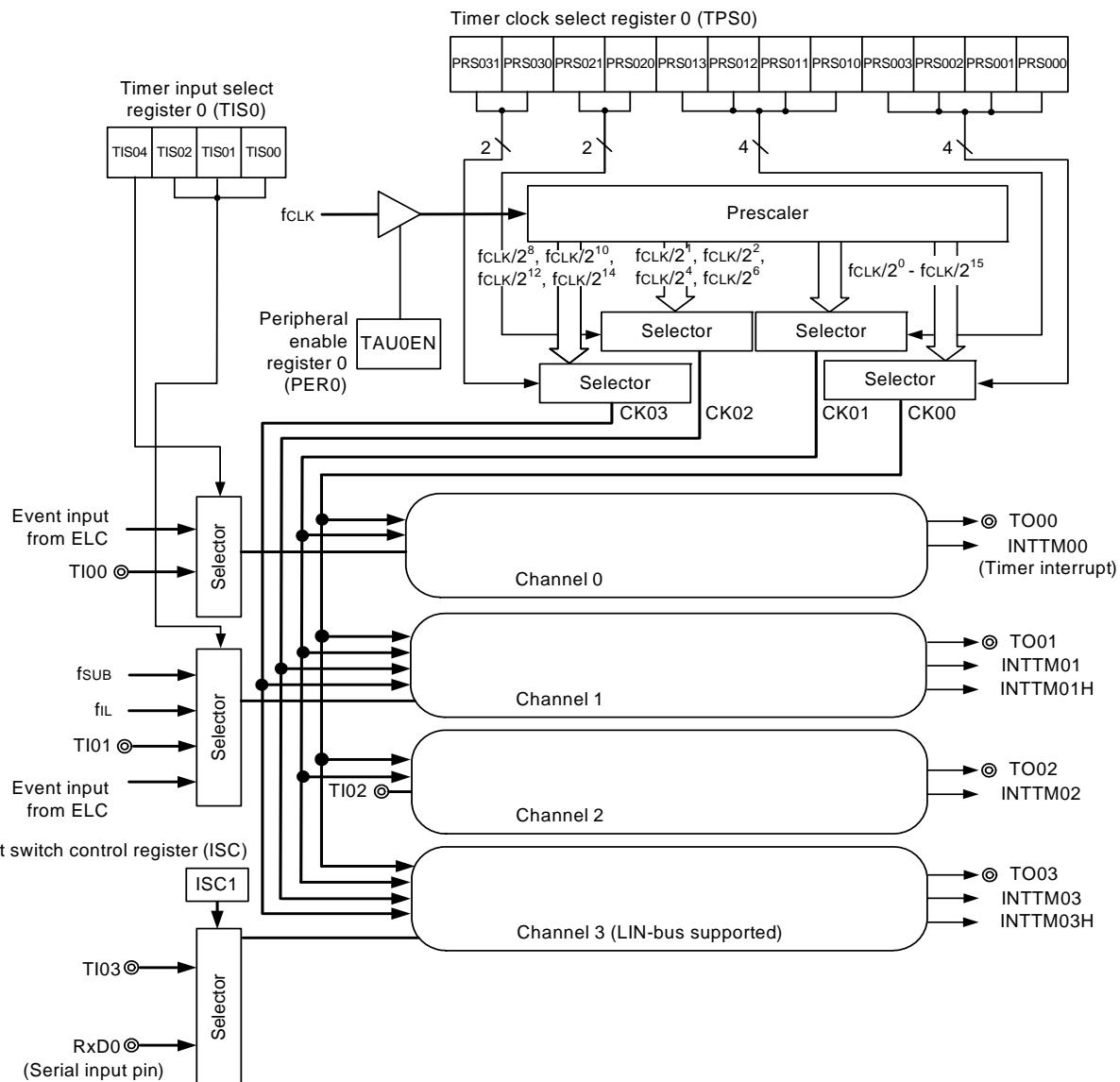
Timer array unit channels		I/O Pins of Each Product		
		100-pin	80-pin	30, 32, 36, 40, 44, 48, 52, 64-pin
Unit 0	Channel 0			TI00, TO00
	Channel 1			TI01/TO01
	Channel 2			TI02/TO02
	Channel 3			TI03/TO03
Unit 1	Channel 0	TI10/TO10	x	
	Channel 1	TI11/TO11	x	
	Channel 2	TI12/TO12	x	
	Channel 3	TI13/TO13	x	

**Remark 1.** When timer input and timer output are shared by the same pin, either only timer input or only timer output can be used.

**Remark 2.** x: The channel is not available.

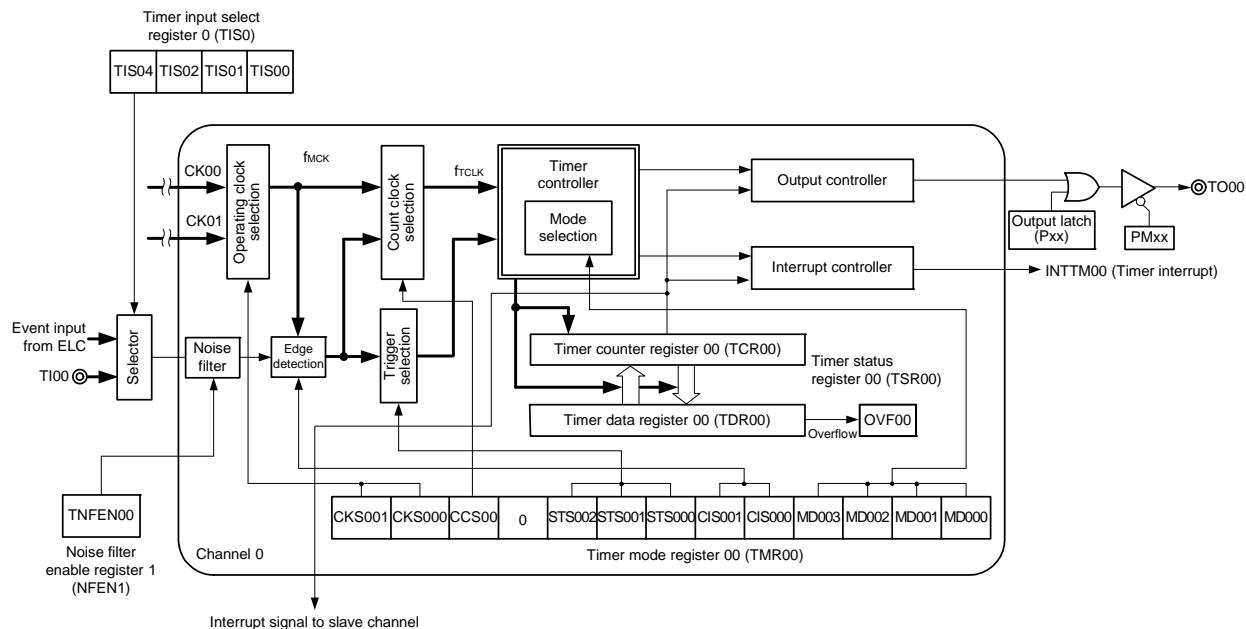
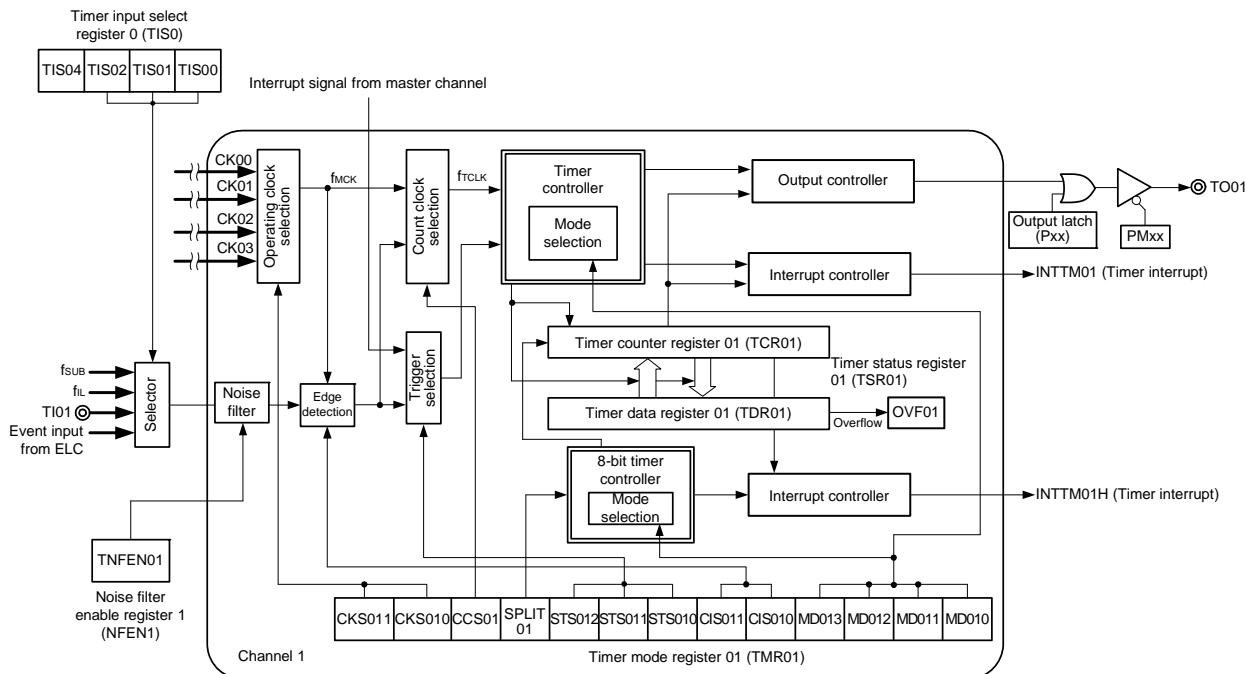
Tables 6 - 1 and 6 - 2 show the block diagrams of the timer array unit.

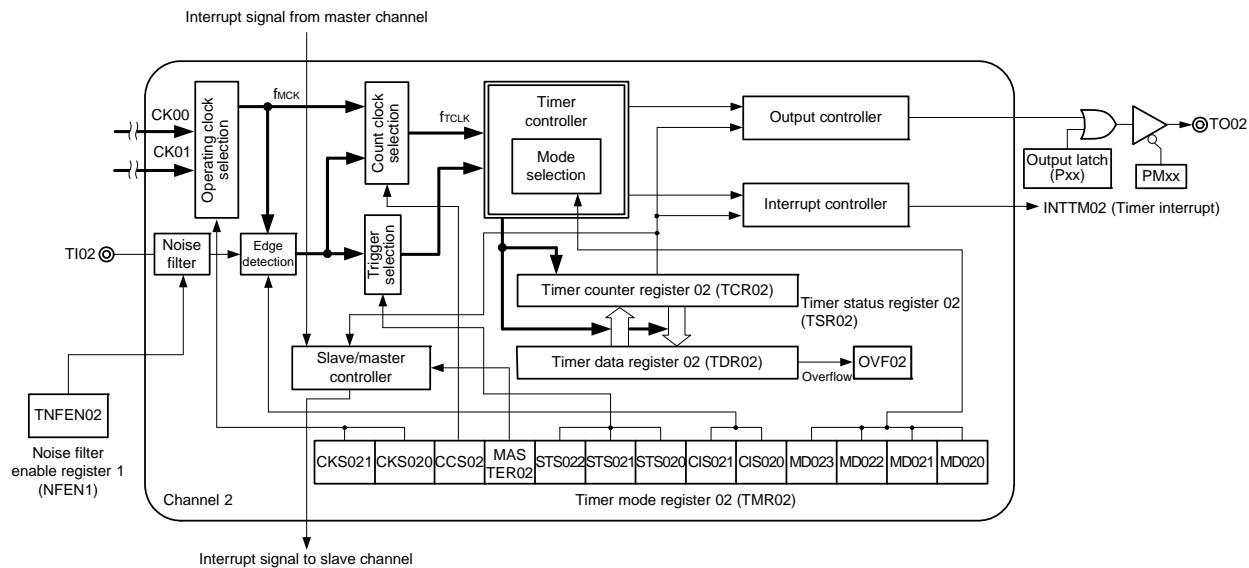
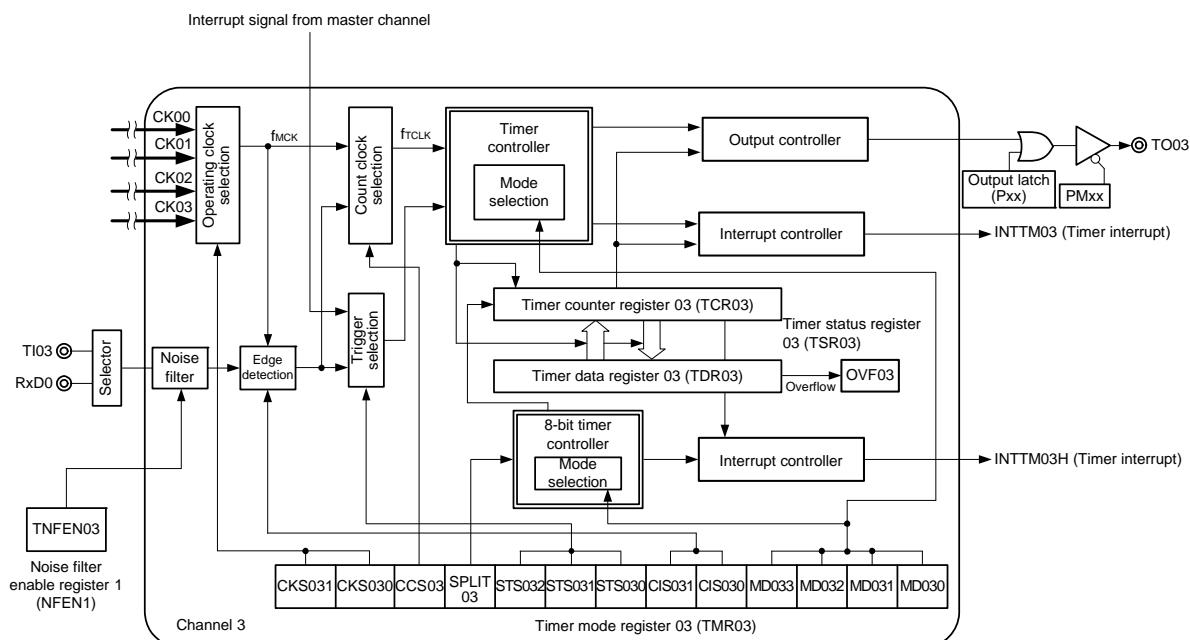
Figure 6 - 1 Entire Configuration of Timer Array Unit 0



**Remark**

- fSUB:** Subsystem clock frequency
- fil:** Low-speed on-chip oscillator clock frequency

**Figure 6 - 2 Internal Block Diagram of Channel 0 of Timer Array Unit 0****Figure 6 - 3 Internal Block Diagram of Channel 1 of Timer Array Unit 0**

**Figure 6 - 4 Internal Block Diagram of Channel 2 of Timer Array Unit 0****Figure 6 - 5 Internal Block Diagram of Channel 3 of Timer Array Unit 0**

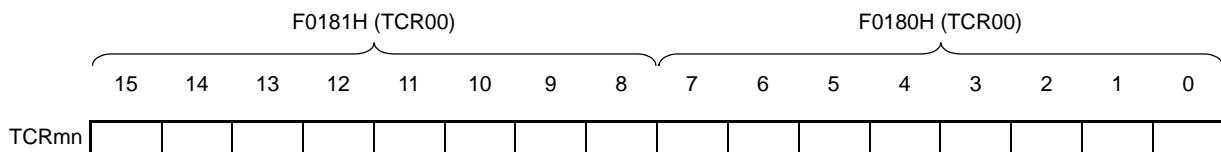
### 6.2.1 Timer count register mn (TCRmn)

The TCRmn register is a 16-bit read-only register and is used to count clocks.

The value of this counter is incremented or decremented in synchronization with the rising edge of a count clock. Whether the counter is incremented or decremented depends on the operation mode that is selected by the MDmn3 to MDmn0 bits of timer mode register mn (TMRmn) (refer to **6.3.3 Timer mode register mn (TMRmn)**).

**Figure 6 - 6 Format of Timer count register mn (TCRmn)**

Address: F0180H, F0181H (TCR00) to F0186H, F0187H (TCR03)  
F01C0H, F01C1H (TCR10) to F01C6H, F01C7H (TCR13)  
After reset: FFFFH R



**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

The count value can be read by reading timer count register mn (TCRmn).

The count value is set to FFFFH in the following cases.

- When the reset signal is generated
- When the TAUMEN bit of peripheral enable register 0 (PER0) is cleared
- When counting of the slave channel has been completed in the PWM output mode
- When counting of the slave channel has been completed in the delay count mode
- When counting of the master/slave channel has been completed in the one-shot pulse output mode
- When counting of the slave channel has been completed in the multiple PWM output mode

The count value is cleared to 0000H in the following cases.

- When the start trigger is input in the capture mode
- When capturing has been completed in the capture mode

**Caution** The count value is not captured to timer data register mn (TDRmn) even when the TCRmn register is read.

The TCRmn register read value differs as follows according to operation mode changes and the operating status.

**Table 6 - 3 Timer Count Register mn (TCRmn) Read Value in Various Operation Modes**

Operation Mode	Count Mode	Timer count register mn (TCRmn) Read Value Note			
		Value if the operation mode was changed after releasing reset	Value if the Operation was restarted after count operation paused (TTmn = 1)	Value if the operation mode was changed after count operation paused (TTmn = 1)	Value when waiting for a start trigger after one count
Interval timer mode	Count down	FFFFH	Value if stop	Undefined	—
Capture mode	Count up	0000H	Value if stop	Undefined	—
Event counter mode	Count down	FFFFH	Value if stop	Undefined	—
One-count mode	Count down	FFFFH	Value if stop	Undefined	FFFFH
Capture & one-count mode	Count up	0000H	Value if stop	Undefined	Capture value of TDRmn register + 1

**Note** This indicates the value read from the TCRmn register when channel n has stopped operating as a timer (TEmn = 0) and has been enabled to operate as a counter (TSmn = 1). The read value is held in the TCRmn register until the count operation starts.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.2.2 Timer data register mn (TDRmn)

This is a 16-bit register from which a capture function and a compare function can be selected.

The capture or compare function can be switched by selecting an operation mode by using the MDmn3 to MDmn0 bits of timer mode register mn (TMRmn).

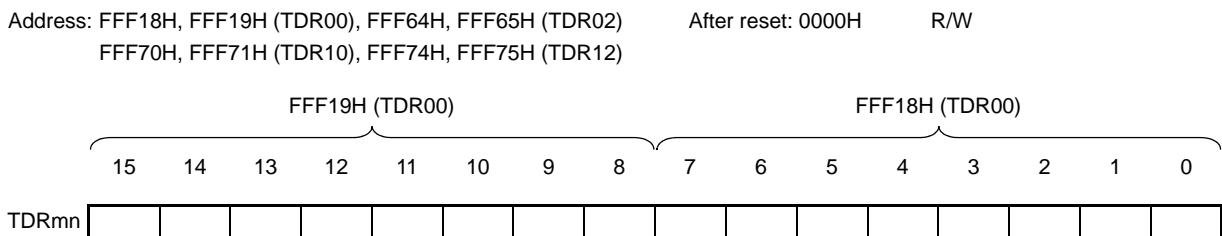
The value of the TDRmn register can be changed at any time.

This register can be read or written in 16-bit units.

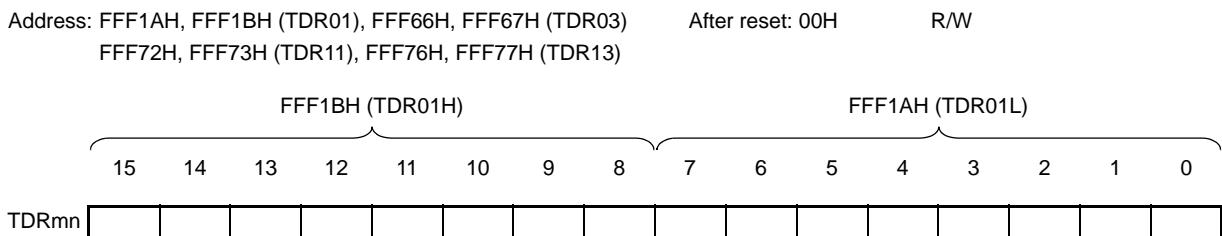
In addition, for the TDRm1 and TDRm3 registers, while in the 8-bit timer mode (when the SPLIT bits of timer mode registers m1 and m3 (TMRm1, TMRm3) are 1), it is possible to read and write the data in 8-bit units, with TDRm1H and TDRm3H used as the higher 8 bits, and TDRm1L and TDRm3L used as the lower 8 bits.

Reset signal generation clears this register to 0000H.

**Figure 6 - 7 Format of Timer data register mn (TDRmn) (n = 0, 2)**



**Figure 6 - 8 Format of Timer data register mn (TDRmn) (n = 1, 3)**



(i) When timer data register mn (TDRmn) is used as compare register

Counting down is started from the value set to the TDRmn register. When the count value reaches 0000H, an interrupt signal (INTTMmn) is generated. The TDRmn register holds its value until it is rewritten.

**Caution** The TDRmn register does not perform a capture operation even if a capture trigger is input, when it is set to the compare function.

(ii) When timer data register mn (TDRmn) is used as capture register

The count value of timer count register mn (TCRmn) is captured to the TDRmn register when the capture trigger is input.

A valid edge of the TI<sub>mn</sub> pin can be selected as the capture trigger. This selection is made by timer mode register mn (TMRmn).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3 Registers Controlling Timer Array Unit

Timer array unit is controlled by the following registers.

- Peripheral enable register 0 (PER0)
- Timer clock select register m (TPSm)
- Timer mode register mn (TMRmn)
- Timer status register mn (TSRmn)
- Timer channel enable status register m (TEM)
- Timer channel start register m (TSm)
- Timer channel stop register m (TTm)
- Timer input select register 0 (TIS0)
- Timer output enable register m (TOEm)
- Timer output register m (TOm)
- Timer output level register m (TOLm)
- Timer output mode register m (TOMm)
- Input switch control register (ISC)
- Noise filter enable registers 1, 2 (NFEN1, NFEN2)
- Port mode control register (PMCxx)
- Port mode register (PMxx)
- Port register (Pxx)

**Caution** Which registers and bits are included depends on the product. Be sure to set bits that are not mounted to their initial values.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.1 Peripheral enable register 0 (PER0)

This register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the timer array unit 0 is used, be sure to set bit 0 (TAU0EN) of this register to 1.

When the timer array unit 1 is used, be sure to set bit 1 (TAU1EN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 6 - 9 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

PER0	RTCEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
------	-------	--------------	-------	---------	--------	--------	-------------	--------

TAU1EN	Control of timer array unit 1 input clock
0	Stops supply of input clock. <ul style="list-style-type: none"> <li>• SFR used by the timer array unit 1 cannot be written.</li> <li>• The timer array unit 1 is in the reset status.</li> </ul>
1	Supplies input clock. <ul style="list-style-type: none"> <li>• SFR used by the timer array unit 1 can be read/written.</li> </ul>

TAU0EN	Control of timer array 0 unit input clock
0	Stops supply of input clock. <ul style="list-style-type: none"> <li>• SFR used by the timer array unit 0 cannot be written.</li> <li>• The timer array unit 0 is in the reset status.</li> </ul>
1	Supplies input clock. <ul style="list-style-type: none"> <li>• SFR used by the timer array unit 0 can be read/written.</li> </ul>

**Note**      80 and 100-pin products only.

**Caution 1.** When setting the timer array unit, be sure to set the following registers first while the TAUMEN bit is set to 1. If TAUMEN = 0, the values of the registers which control the timer array unit are cleared to their initial values and writing to them is ignored (except for the timer input select register 0 (TIS0), input switch control register (ISC), noise filter enable registers 1, 2 (NFEN1, NFEN2), port mode control registers 0, 1, 10, 12, 14 (PMC0, PMC1, PMC10, PMC12, PMC14), port mode registers 0, 1, 3, 6 (PM0, PM1, PM3, PM6), and port registers 0, 1, 3, 6 (P0, P1, P3, P6)).

- Timer status register mn (TSRmn)
- Timer channel enable status register m (TEm)
- Timer channel start register m (TSm)
- Timer channel stop register m (TTm)
- Timer output enable register m (TOEm)
- Timer output register m (TOM)
- Timer output level register m (TOLm)
- Timer output mode register m (TOMm)

**Caution 2.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

### 6.3.2 Timer clock select register m (TPSm)

The TPSm register is a 16-bit register that is used to select two types or four types of operation clocks (CKm0, CKm1, CKm2, CKm3) that are commonly supplied to each channel. CKm0 is selected by using bits 3 to 0 of the TPSm register, and CKm1 is selected by using bits 7 to 4 of the TPSm register. In addition, only for channels 1 and 3, CKm2 and CKm3 can be also selected. CKm2 is selected by using bits 9 and 8 of the TPSm register, and CKm3 is selected by using bits 13 and 12 of the TPSm register.

Rewriting of the TPSm register during timer operation is possible only in the following cases.

If the PRSm00 to PRSm03 bits can be rewritten ( $n = 0$  to 3):

All channels for which CKm0 is selected as the operation clock ( $CKSmn1, CKSmn0 = 0, 0$ ) are stopped ( $TEmn = 0$ ).

If the PRSm10 to PRSm13 bits can be rewritten ( $n = 0$  to 3):

All channels for which CKm2 is selected as the operation clock ( $CKSmn1, CKSmn0 = 0, 1$ ) are stopped ( $TEmn = 0$ ).

If the PRSm20 and PRSm21 bits can be rewritten ( $n = 1, 3$ ):

All channels for which CKm1 is selected as the operation clock ( $CKSmn1, CKSmn0 = 1, 0$ ) are stopped ( $TEmn = 0$ ).

If the PRSm30 and PRSm31 bits can be rewritten ( $n = 1, 3$ ):

All channels for which CKm3 is selected as the operation clock ( $CKSmn1, CKSmn0 = 1, 1$ ) are stopped ( $TEmn = 0$ ).

The TPSm register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Figure 6 - 10 Format of Timer clock select register m (TPSm) (1/2)**

Address: F01B6H, F01B7H (TPS0), F01F6H, F01F7H (TPS1) After reset: 0000H R/W

Symbol 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

TPSm	0	0	PRSm 31	PRSm 30	0	0	PRSm 21	PRSm 20	PRSm 13	PRSm 12	PRSm 11	PRSm 10	PRSm 03	PRSm 02	PRSm 01	PRSm 00
------	---	---	------------	------------	---	---	------------	------------	------------	------------	------------	------------	------------	------------	------------	------------

PRS mk3	PRS mk2	PRS mk1	PRS mk0	Selection of operation clock (CKmk) Note (k = 0, 1)					
				fCLK = 2 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 20 MHz	fCLK = 32 MHz	
0	0	0	0	fCLK	2 MHz	4 MHz	8 MHz	20 MHz	32 MHz
0	0	0	1	fCLK/2	1 MHz	2 MHz	4 MHz	10 MHz	16 MHz
0	0	1	0	fCLK/2 <sup>2</sup>	500 kHz	1 MHz	2 MHz	5 MHz	8 MHz
0	0	1	1	fCLK/2 <sup>3</sup>	250 kHz	500 kHz	1 MHz	2.5 MHz	4 MHz
0	1	0	0	fCLK/2 <sup>4</sup>	125 kHz	250 kHz	500 kHz	1.25 MHz	2 MHz
0	1	0	1	fCLK/2 <sup>5</sup>	62.5 kHz	125 kHz	250 kHz	625 kHz	1 MHz
0	1	1	0	fCLK/2 <sup>6</sup>	31.3 kHz	62.5 kHz	125 kHz	313 kHz	500 kHz
0	1	1	1	fCLK/2 <sup>7</sup>	15.6 kHz	31.3 kHz	62.5 kHz	156 kHz	250 kHz
1	0	0	0	fCLK/2 <sup>8</sup>	7.81 kHz	15.6 kHz	31.3 kHz	78.1 kHz	125 kHz
1	0	0	1	fCLK/2 <sup>9</sup>	3.91 kHz	7.81 kHz	15.6 kHz	39.1 kHz	62.5 kHz
1	0	1	0	fCLK/2 <sup>10</sup>	1.95 kHz	3.91 kHz	7.81 kHz	19.5 kHz	31.25 kHz
1	0	1	1	fCLK/2 <sup>11</sup>	977 Hz	1.95 kHz	3.91 kHz	9.77 kHz	15.6 kHz
1	1	0	0	fCLK/2 <sup>12</sup>	488 Hz	977 Hz	1.95 kHz	4.88 kHz	7.81 kHz
1	1	0	1	fCLK/2 <sup>13</sup>	244 Hz	488 Hz	977 Hz	2.44 kHz	3.91 kHz
1	1	1	0	fCLK/2 <sup>14</sup>	122 Hz	244 Hz	488 Hz	1.22 kHz	1.95 kHz
1	1	1	1	fCLK/2 <sup>15</sup>	61.0 Hz	122 Hz	244 Hz	610 Hz	977 Hz

**Note** When changing the clock selected for fCLK (by changing the system clock control register (CKC) value), stop timer array unit (TTm = 000FH).

The timer array unit must also be stopped if the operating clock (fmck) or the valid edge of the signal input from the TImn pin is selected.

**Caution 1.** Be sure to clear bits 15, 14, 11, 10 to “0”.

**Caution 2.** If fCLK (undivided) is selected as the operation clock (CKmk) and TDRnm is set to 0000H (n = 0 or 1, m = 0 to 3), interrupt requests output from timer array units cannot be used.

**Remark 1.** fCLK: CPU/peripheral hardware clock frequency

**Remark 2.** Waveform of the clock to be selected in the TPSm register which becomes high level for one period of fCLK from its rising edge (m = 1 to 15). For details, see **6.5.1 Count clock (fTCLK)**.

**Figure 6 - 11 Format of Timer clock select register m (TPSm) (2/2)**

Address: F01B6H, F01B7H (TPS0), F01F6H, F01F7H (TPS1) After reset: 0000H R/W

Symbol 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

TPSm	0	0	PRSm 31	PRSm 30	0	0	PRSm 21	PRSm 20	PRSm 13	PRSm 12	PRSm 11	PRSm 10	PRSm 03	PRSm 02	PRSm 01	PRSm 00
------	---	---	------------	------------	---	---	------------	------------	------------	------------	------------	------------	------------	------------	------------	------------

PRS m21	PRS m20	Selection of operation clock (CKm2) Note					
		fCLK = 2 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 20 MHz	fCLK = 32 MHz	
0	0	fCLK/2	1 MHz	2 MHz	4 MHz	10 MHz	16 MHz
0	1	fCLK/2 <sup>2</sup>	500 kHz	1 MHz	2 MHz	5 MHz	8 MHz
1	0	fCLK/2 <sup>4</sup>	125 kHz	250 kHz	500 kHz	1.25 MHz	2 MHz
1	1	fCLK/2 <sup>6</sup>	31.3 kHz	62.5 kHz	125 kHz	313 kHz	500 kHz

PRS m31	PRS m30	Selection of operation clock (CKm3) Note					
		fCLK = 2 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 20 MHz	fCLK = 32 MHz	
0	0	fCLK/2 <sup>8</sup>	7.81 kHz	15.6 kHz	31.3 kHz	78.1 kHz	125 kHz
0	1	fCLK/2 <sup>10</sup>	1.95 kHz	3.91 kHz	7.81 kHz	19.5 kHz	31.3 kHz
1	0	fCLK/2 <sup>12</sup>	488 Hz	977 Hz	1.95 kHz	4.88 kHz	7.81 kHz
1	1	fCLK/2 <sup>14</sup>	122 Hz	244 Hz	488 Hz	1.22 kHz	1.95 kHz

**Note** When changing the clock selected for fCLK (by changing the system clock control register (CKC) value), stop timer array unit (TTm = 000FH).

The timer array unit must also be stopped if the operating clock (fmck) or the valid edge of the signal input from the TImn pin is selected.

**Caution** Be sure to clear bits 15, 14, 11, 10 to “0”.

By using channels 1 and 3 in the 8-bit timer mode and specifying CKm2 or CKm3 as the operation clock, the interval times shown in Table 6 - 4 can be achieved by using the interval timer function.

**Table 6 - 4 Interval Times Available for Operation Clock CKSm2 or CKSm3**

Clock		Interval time Note (fCLK = 32 MHz)			
		10 µs	100 µs	1 ms	10 ms
CKm2	fCLK/2	✓	—	—	—
	fCLK/2 <sup>2</sup>	✓	—	—	—
	fCLK/2 <sup>4</sup>	✓	✓	—	—
	fCLK/2 <sup>6</sup>	✓	✓	—	—
CKm3	fCLK/2 <sup>8</sup>	—	✓	✓	—
	fCLK/2 <sup>10</sup>	—	✓	✓	—
	fCLK/2 <sup>12</sup>	—	—	✓	✓
	fCLK/2 <sup>14</sup>	—	—	✓	✓

**Note** The margin is within 5%.

**Remark 1.** fCLK: CPU/peripheral hardware clock frequency

**Remark 2.** For details of a signal of fCLK/2<sup>i</sup> selected with the TPSm register, see **6.5.1 Count clock (ftCLK)**.

### 6.3.3 Timer mode register mn (TMRmn)

The TMRmn register sets an operation mode of channel n. This register is used to select the operation clock (fMCK), select the count clock, select the master/slave, select the 16 or 8-bit timer (only for channels 1 and 3), specify the start trigger and capture trigger, select the valid edge of the timer input, and specify the operation mode (interval, capture, event counter, one-count, or capture and one-count).

Rewriting the TMRmn register is prohibited when the register is in operation (when TEmn = 1). However, bits 7 and 6 (CISmn1, CISmn0) can be rewritten even while the register is operating with some functions (when TEmn = 1) (for details, see **6.8 Independent Channel Operation Function of Timer Array Unit** and **6.9 Simultaneous Channel Operation Function of Timer Array Unit**).

The TMRmn register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Caution** The bits mounted depend on the channels in the bit 11 of TMRmn register.

**TMRm2: MASTERmn bit (n = 2)**

**TMRm1, TMRm3: SPLITmn bit (n = 1, 3)**

**TMRm0: Fixed to 0**

**Figure 6 - 12 Format of Timer mode register mn (TMRmn) (1/4)**

Address: F0190H, F0191H (TMR00) to F0196H, F0197H (TMR03), After reset: 0000H R/W  
F01D0H, F01D1H (TMR10) to F01D6H, F01D7H (TMR13)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 2)	CKSm n1	CKSm n0	0	CCSm n	MAST ERmn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 1, 3)	CKSm n1	CKSm n0	0	CCSm n	SPLIT mn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 0)	CKSm n1	CKSm n0	0	CCSm n	0 Note 1	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

CKS mn1	CKS mn0	Selection of operation clock (fmck) of channel n
0	0	Operation clock CKm0 set by timer clock select register m (TPSm)
0	1	Operation clock CKm2 set by timer clock select register m (TPSm)
1	0	Operation clock CKm1 set by timer clock select register m (TPSm)
1	1	Operation clock CKm3 set by timer clock select register m (TPSm)
Operation clock (fmck) is used by the edge detector. A count clock (fCLK) and a sampling clock are generated depending on the setting of the CCSmn bit.		
The operation clocks CKm2 and CKm3 can only be selected for channels 1 and 3.		

CCSmn	Selection of count clock (fCLK) of channel n
0	Operation clock (fmck) specified by the CKSmn0 and CKSmn1 bits
1	Valid edge of input signal input from the Tlmn pin When using unit 0: In channel 0, Valid edge of input signal selected by TIS0 In channel 1, Valid edge of input signal selected by TIS0 In channel 3, Valid edge of input signal selected by ISC

Count clock (fCLK) is used for the counter, output controller, and interrupt controller.

**Note 1.** Bit 11 is fixed at 0 of read only, write is ignored.

**Caution 1. Be sure to clear bits 13, 5, and 4 to “0”.**

**Caution 2. The timer array unit must be stopped (TTm = 00FFH) if the clock selected for fCLK is changed (by changing the value of the system clock control register (CKC)), even if the operating clock specified by using the CKSmn0 and CKSmn1 bits (fmck) or the valid edge of the signal input from the Tlmn pin is selected as the count clock (fCLK).**

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 13 Format of Timer mode register mn (TMRmn) (2/4)**

Address: F0190H, F0191H (TMR00) to F0196H, F0197H (TMR03), After reset: 0000H R/W  
F01D0H, F01D1H (TMR10) to F01D6H, F01D7H (TMR13)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 2)	CKSm n1	CKSm n0	0	CCSm n	MAST ERmn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 1, 3)	CKSm n1	CKSm n0	0	CCSm n	SPLIT mn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 0)	CKSm n1	CKSm n0	0	CCSm n	0 Note 1	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

(Bit 11 of TMRmn (n = 2))

MASTERmn	Selection between using channel n independently or simultaneously with another channel (as a slave or master)
0	Operates in independent channel operation function or as slave channel in simultaneous channel operation function.
1	Operates as master channel in simultaneous channel operation function.
Only the channel 2 can be set as a master channel (MASTERmn = 1). Be sure to use channel 0 is fixed to 0 (regardless of the bit setting, channel 0 operates as master, because it is the highest channel). Clear the MASTERmn bit to 0 for a channel that is used with the independent channel operation function.	

(Bit 11 of TMRmn (n = 1, 3))

SPLITmn	Selection of 8 or 16-bit timer operation for channels 1 and 3
0	Operates as 16-bit timer. (Operates in independent channel operation function or as slave channel in simultaneous channel operation function.)
1	Operates as 8-bit timer.

STS mn2	STS mn1	STS mn0	Setting of start trigger or capture trigger of channel n
0	0	0	Only software trigger start is valid (other trigger sources are unselected).
0	0	1	Valid edge of the TImn pin input is used as both the start trigger and capture trigger.
0	1	0	Both the edges of the TImn pin input are used as a start trigger and a capture trigger.
1	0	0	Interrupt signal of the master channel is used (when the channel is used as a slave channel with the simultaneous channel operation function).
Other than above		Setting prohibited	

**Note 1.** Bit 11 is fixed at 0 of read only, write is ignored.**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 14 Format of Timer mode register mn (TMRmn) (3/4)**

Address: F0190H, F0191H (TMR00) to F0196H, F0197H (TMR03), After reset: 0000H R/W  
 F01D0H, F01D1H (TMR10) to F01D6H, F01D7H (TMR13)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 2)	CKSm n1	CKSm n0	0	CCSm n	MAST ERmn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 1, 3)	CKSm n1	CKSm n0	0	CCSm n	SPLIT mn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 0)	CKSm n1	CKSm n0	0	CCSm n	0 Note 1	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

CIS mn1	CIS mn0	Selection of TImn pin input valid edge
0	0	Falling edge
0	1	Rising edge
1	0	Both edges (when low-level width is measured) Start trigger: Falling edge, Capture trigger: Rising edge
1	1	Both edges (when high-level width is measured) Start trigger: Rising edge, Capture trigger: Falling edge

If both the edges are specified when the value of the STSmn2 to STSmn0 bits is other than 010B, set the CISmn1 to CISmn0 bits to 10B.

**Note 1.** Bit 11 is fixed at 0 of read only, write is ignored.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 15 Format of Timer mode register mn (TMRmn) (4/4)**

Address: F0190H, F0191H (TMR00) to F0196H, F0197H (TMR03), After reset: 0000H R/W  
F01D0H, F01D1H (TMR10) to F01D6H, F01D7H (TMR13)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 2)	CKSm n1	CKSm n0	0	CCSm n	MAST ERmn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 1, 3)	CKSm n1	CKSm n0	0	CCSm n	SPLIT mn	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn (n = 0)	CKSm n1	CKSm n0	0	CCSm n	0 Note 1	STS <sub>m</sub> n2	STS <sub>m</sub> n1	STS <sub>m</sub> n0	CISmn 1	CISmn 0	0	0	MDmn 3	MDmn 2	MDmn 1	MDmn 0

MD mn3	MD mn2	MD mn1	Operation mode of channel n	Corresponding function	Count operation of TCR
0	0	0	Interval timer mode	Interval timer / Square wave output / Divider function / PWM output (master)	Counting down
0	1	0	Capture mode	Input pulse interval measurement	Counting up
0	1	1	Event counter mode	External event counter	Counting down
1	0	0	One-count mode	Delay counter / One-shot pulse output / PWM output (slave)	Counting down
1	1	0	Capture & one-count mode	Measurement of high-/low-level width of input signal	Counting up
Other than above		Setting prohibited			

The operation of each mode varies depending on MDmn0 bit (see table below).

Operation mode (Value set by the MDmn3 to MDmn1 bits (see table above))	MDm n0	Setting of starting counting and interrupt
• Interval timer mode (0, 0, 0) • Capture mode (0, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
	1	Timer interrupt is generated when counting is started (timer output also changes).
• Event counter mode (0, 1, 1)	0	Timer interrupt is not generated when counting is started (timer output does not change, either).
	1	Start trigger is invalid during counting operation. At that time, interrupt is not generated.
• One-count mode Note 2 (1, 0, 0)	0	Start trigger is valid during counting operation Note 3. At that time, interrupt is not generated.
	1	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, interrupt is not generated.
• Capture & one-count mode (1, 1, 0)	0	Timer interrupt is not generated when counting is started (timer output does not change, either). Start trigger is invalid during counting operation. At that time, interrupt is not generated.

**Note 1.** Bit 11 is fixed at 0 of read only, write is ignored.

**Note 2.** In one-count mode, interrupt output (INTTMmn) when starting a count operation and TOmn output are not controlled.

**Note 3.** If the start trigger (TSmn = 1) is issued during operation, the counter is initialized, and recounting is started (does not occur the interrupt request).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.4 Timer status register mn (TSRmn)

The TSRmn register indicates the overflow status of the counter of channel n.

The TSRmn register is valid only in the capture mode ( $MDmn3$  to  $MDmn1 = 010B$ ) and capture & one-count mode ( $MDmn3$  to  $MDmn1 = 110B$ ). See **Table 6 - 5** for the operation of the OVF bit in each operation mode and set/clear conditions.

The TSRmn register can be read by a 16-bit memory manipulation instruction.

The lower 8 bits of the TSRmn register can be set with an 8-bit memory manipulation instruction with TSRmnL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 16 Format of Timer status register mn (TSRmn)**

Address:	F01A0H, F01A1H (TSR00) to F01A6H, F01A7H (TSR03)	After reset:	0000H	R												
	F01E0H, F01E1H (TSR10) to F01E6H, F01E7H (TSR13)															
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TSRmn	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OVF
OVF	Counter overflow status of channel n															
0	Overflow does not occur.															
1	Overflow occurs.															
When OVF = 1, this flag is cleared (OVF = 0) when the next value is captured without overflow.																

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ )

**Table 6 - 5 OVF Bit Operation and Set/Clear Conditions in Each Operation Mode**

Timer operation mode	OVF bit	Set/clear conditions
• Capture mode	clear	When no overflow has occurred upon capturing
	set	When an overflow has occurred upon capturing
• Interval timer mode • Event counter mode • One-count mode	clear	—
	set	(Use prohibited)

**Remark** The OVF bit does not change immediately after the counter has overflowed, but changes upon the subsequent capture.

### 6.3.5 Timer channel enable status register m (TEm)

The TE<sub>m</sub> register is used to enable or stop the timer operation of each channel.

Each bit of the TE<sub>m</sub> register corresponds to each bit of the timer channel start register m (TS<sub>m</sub>) and the timer channel stop register m (TT<sub>m</sub>). When a bit of the TS<sub>m</sub> register is set to 1, the corresponding bit of this register is set to 1. When a bit of the TT<sub>m</sub> register is set to 1, the corresponding bit of this register is cleared to 0.

The TE<sub>m</sub> register can be read by a 16-bit memory manipulation instruction.

The lower 8 bits of the TE<sub>m</sub> register can be set with a 1-bit or 8-bit memory manipulation instruction with TE<sub>m</sub>L.

Reset signal generation clears this register to 0000H.

**Figure 6 - 17 Format of Timer channel enable status register m (TE<sub>m</sub>)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TE <sub>m</sub>	0	0	0	0	TEH <sub>m</sub> 3	0	TEH <sub>m</sub> 1	0	0	0	0	0	TE <sub>m</sub> 3	TE <sub>m</sub> 2	TE <sub>m</sub> 1	TE <sub>m</sub> 0	
TEH m3	Indication of whether operation of the higher 8-bit timer is enabled or stopped when channel 3 is in the 8-bit timer mode																
0	Operation is stopped.																
1	Operation is enabled.																
TEH m1	Indication of whether operation of the higher 8-bit timer is enabled or stopped when channel 1 is in the 8-bit timer mode																
0	Operation is stopped.																
1	Operation is enabled.																
TE <sub>m</sub> n	Indication of operation enable/stop status of channel n																
0	Operation is stopped.																
1	Operation is enabled.																
This bit displays whether operation of the lower 8-bit timer for TE <sub>m</sub> 1 and TE <sub>m</sub> 3 is enabled or stopped when channel 1 or 3 is in the 8-bit timer mode.																	

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.6 Timer channel start register m (TSm)

The TSm register is a trigger register that is used to initialize timer count register mn (TCRmn) and start the counting operation of each channel.

When a bit of this register is set to 1, the corresponding bit of timer channel enable status register m (TEm) is set to 1. The TSmn, TSHm1, TSHm3 bits are immediately cleared when operation is enabled (TEmn, TEHm1, TEHm3 = 1), because they are trigger bits.

The TSm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TSm register can be set with a 1-bit or 8-bit memory manipulation instruction with TSmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 18 Format of Timer channel start register m (TSm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address: F01B2H, F01B3H (TS0) to F01F2H, F01F3H (TS1)	After reset: 0000H	R/W
TSm	0	0	0	0	TSHm 3	0	TSHm 1	0	0	0	0	0	0	TSm3	TSm2	TSm1	TSm0		
<hr/>																			
TSH m3																			
Trigger to enable operation (start operation) of the higher 8-bit timer when channel 3 is in the 8-bit timer mode																			
0																			
1																			
The TEHm3 bit is set to 1 and the count operation becomes enabled. The TCRm3 register count operation start in the interval timer mode in the count operation enabled state (see <b>Table 6 - 6 in 6.5.2 Start timing of counter</b> ).																			
<hr/>																			
TSH m1																			
Trigger to enable operation (start operation) of the higher 8-bit timer when channel 1 is in the 8-bit timer mode																			
0																			
1																			
The TEHm1 bit is set to 1 and the count operation becomes enabled. The TCRm1 register count operation start in the interval timer mode in the count operation enabled state (see <b>Table 6 - 6 in 6.5.2 Start timing of counter</b> ).																			
<hr/>																			
TSm n																			
Operation enable (start) trigger of channel n																			
0																			
1																			
The TEmn bit is set to 1 and the count operation becomes enabled. The TCRmn register count operation start in the count operation enabled state varies depending on each operation mode (see <b>Table 6 - 6 in 6.5.2 Start timing of counter</b> ). This bit is the trigger to enable operation (start operation) of the lower 8-bit timer for TSm1 and TSm3 when channel 1 or 3 is in the 8-bit timer mode.																			

(Cautions and Remarks are listed on the next page.)

**Caution 1. Be sure to clear bits 15 to 12, 10, 8 to 4 to “0”**

**Caution 2. When switching from a function that does not use TImn pin input to one that does, the following wait period is required from when timer mode register mn (TMRmn) is set until the TSmn (TSHm1, TSHm3) bit is set to 1.**

**When the TImn pin noise filter is enabled (TNFENmn = 1): Four cycles of the operation clock (fMCK)**

**When the TImn pin noise filter is disabled (TNFENmn = 0): Two cycles of the operation clock (fMCK)**

**Remark 1.** When the TS<sub>m</sub> register is read, 0 is always read.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.7 Timer channel stop register m (TTm)

The TTm register is a trigger register that is used to stop the counting operation of each channel.

When a bit of this register is set to 1, the corresponding bit of timer channel enable status register m (TEm) is cleared to 0. The TTmn, TTHm1, TTHm3 bits are immediately cleared when operation is stopped (TEmn, TEHm1, TEHm3 = 0), because they are trigger bits.

The TTm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TTm register can be set with a 1-bit or 8-bit memory manipulation instruction with TTmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 19 Format of Timer channel stop register m (TTm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Address: F01B4H, F01B5H (TT0), F01F4H, F01F5H (TT1)	After reset: 0000H	R/W
TTm	0	0	0	0	TTHm3	0	TTHm1	0	0	0	0	0	0	TTm3	TTm2	TTm1	TTm0		
<hr/>																			
TTHm3																			
Trigger to stop operation of the higher 8-bit timer when channel 3 is in the 8-bit timer mode																			
0																			
1																			
TTHm1																			
Trigger to stop operation of the higher 8-bit timer when channel 1 is in the 8-bit timer mode																			
0																			
1																			
TTm n																			
Operation stop trigger of channel n																			
0																			
1																			

**Caution** Be sure to clear bits 15 to 12, 10, 8 to 4 of the TTm register to "0".

**Remark 1.** When the TTm register is read, 0 is always read.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.8 Timer input select register 0 (TIS0)

The TIS0 register is used to select the channels 0 and 1 of unit 0 timer input.

The TIS0 register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 6 - 20 Format of Timer input select register 0 (TIS0)**

Address: F0074H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
TIS0	0	0	0	TIS04	0	TIS02	TIS01	TIS00
TIS04	Selection of timer input used with channel 0							
0	Input signal of timer input pin (TI00)							
1	Event input signal from ELC							
TIS02	TIS01	TIS00	Selection of timer input used with channel 1					
0	0	0	Input signal of timer input pin (TI01)					
0	0	1	Event input signal from ELC					
0	1	0	Input signal of timer input pin (TI01)					
0	1	1						
1	0	0	Low-speed on-chip oscillator clock (f <sub>IL</sub> )					
1	0	1	Subsystem clock (f <sub>SUB</sub> )					
Other than above			Setting prohibited					

**Caution 1.** At least  $1/f_{MCK} + 10$  ns is necessary as the high-level and low-level widths of the timer input to be selected. Thus, the TIS02 bit cannot be set to 1 when f<sub>SUB</sub> is selected as f<sub>CLK</sub> (CSS in CKC register = 1).

**Caution 2.** When selecting an event input signal from the ELC using timer input select register 0 (TIS0), select f<sub>CLK</sub> using timer clock select register 0 (TPS0).

### 6.3.9 Timer output enable register m (TOEm)

The TOEm register is used to enable or disable timer output of each channel.

Channel n for which timer output has been enabled becomes unable to rewrite the value of the TOmn bit of timer output register m (TOm) described later by software, and the value reflecting the setting of the timer output function through the count operation is output from the timer output pin (TOmn).

The TOEm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TOEm register can be set with a 1-bit or 8-bit memory manipulation instruction with TOEmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 21 Format of Timer output enable register m (TOEm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TOEm	0	0	0	0	0	0	0	0	0	0	0	0	TOEm 3	TOEm 2	TOEm 1	TOEm 0	
<b>TOE mn</b>																	
0	Timer output is disabled. Timer operation is not applied to the TOmn bit and the output is fixed. Writing to the TOmn bit is enabled and the level set in the TOmn bit is output from the TOmn pin.																
1	Timer output is enabled. Timer operation is applied to the TOmn bit and an output waveform is generated. Writing to the TOmn bit is ignored.																

**Caution** Be sure to clear bits 15 to 4 to “0”.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.10 Timer output register m (TOm)

The TOm register is a buffer register of timer output of each channel.

The value of each bit in this register is output from the timer output pin (TOmn) of each channel.

The TOmn bit of this register can be rewritten by software only when timer output is disabled (TOEmn = 0).

When timer output is enabled (TOEmn = 1), rewriting this register by software is ignored, and the value is changed only by the timer operation.

To use the TI00, TO00, TI01/TO01, TI02/TO02, TI03/TO03, TI10/TO10 to TI13/TO13 pins as a port function pin, set the corresponding TOmn bit to "0".

The TOm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TOm register can be set with an 8-bit memory manipulation instruction with TOmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 22 Format of Timer output register m (TOm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TOm	0	0	0	0	0	0	0	0	0	0	0	0	0	TOm3	TOm2	TOm1	TOm0
<hr/>																	
TOm n	Timer output of channel n																
0	Timer output value is "0".																
1	Timer output value is "1".																

**Caution** Be sure to clear bits 15 to 4 to "0".

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.11 Timer output level register m (TOLm)

The TOLm register is a register that controls the timer output level of each channel.

The setting of the inverted output of channel n by this register is reflected at the timing of set or reset of the timer output signal while the timer output is enabled ( $TOEmn = 1$ ) in the Slave channel output mode ( $TOMmn = 1$ ). In the master channel output mode ( $TOMmn = 0$ ), this register setting is invalid.

The TOLm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TOLm register can be set with an 8-bit memory manipulation instruction with TOLmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 23 Format of Timer output level register m (TOLm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TOLm	0	0	0	0	0	0	0	0	0	0	0	0	TOLm 3	TOLm 2	TOLm 1	0	
<b>TOL mn</b>																	
0	Control of timer output level of channel n																
1	Positive logic output (active-high)																
<b>Caution Be sure to clear bits 15 to 4, and 0 to “0”.</b>																	

**Remark 1.** If the value of this register is rewritten during timer operation, the timer output logic is inverted when the timer output signal changes next, instead of immediately after the register value is rewritten.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.3.12 Timer output mode register m (TOMm)

The TOMm register is used to control the timer output mode of each channel.

When a channel is used for the independent channel operation function, set the corresponding bit of the channel to be used to 0.

When a channel is used for the simultaneous channel operation function (PWM output, one-shot pulse output, or multiple PWM output), set the corresponding bit of the master channel to 0 and the corresponding bit of the slave channel to 1.

The setting of each channel n by this register is reflected at the timing when the timer output signal is set or reset while the timer output is enabled ( $TOEmn = 1$ ).

The TOMm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the TOMm register can be set with an 8-bit memory manipulation instruction with TOMmL.

Reset signal generation clears this register to 0000H.

**Figure 6 - 24 Format of Timer output mode register m (TOMm)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOMm	0	0	0	0	0	0	0	0	0	0	0	0	TOMm 3	TOMm 2	TOMm 1	0
TOM mn																
Control of timer output mode of channel n																
0 Master channel output mode (to produce toggle output by timer interrupt request signal (INTTMmn))																
1 Slave channel output mode (output is set by the timer interrupt request signal (INTTMmn) of the master channel, and reset by the timer interrupt request signal (INTTM0p) of the slave channel)																

**Caution** Be sure to clear bits 15 to 4, and 0 to “0”.

**Remark** m: Unit number (m = 0, 1)  
n: Channel number  
n = 0 to 3 (n = 0, 2 for master channel)  
p: Slave channel number  
n = 0, p = 1, 2, 3  
n = 2, p = 3  
(For details of the relation between the master channel and slave channel, refer to **6.4.1 Basic rules of simultaneous channel operation function.**)

### 6.3.13 Input switch control register (ISC)

The ISC1 and ISC0 bits of the ISC register are used to implement LIN-bus communication operation by using channel 3 in association with the serial array unit. When the ISC1 bit is set to 1, the input signal of the serial data input pin (RxD0) is selected as a timer input signal.

For details about setting the SSIE00 bit, see **17.3.15 Input switch control register (ISC)**.

The ISC register can be set by a 1-bit or 8-bit memory manipulation instruction.

**Figure 6 - 25 Format of Input switch control register (ISC)**

Address: F0073H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ISC	SSIE00	0	0	0	0	0	ISC1	ISC0
SSIE00		Setting SSI00 pin input when CSI00 communication and slave mode are applied						
0		SSI00 pin input is invalid.						
1		SSI00 pin input is valid.						
ISC1		Switching channel 3 input of timer array unit 0						
0		Uses the input signal of the TI03 pin as a timer input (normal operation).						
1		Input signal of the RxD0 pin is used as timer input (detects the wakeup signal and measures the low width of the break field and the pulse width of the sync field).						
ISC0		Switching external interrupt (INTP0) input						
0		Uses the input signal of the INTP0 pin as an external interrupt (normal operation).						
1		Uses the input signal of the RxD0 pin as an external interrupt (wakeup signal detection).						

**Caution** Be sure to clear bits 6 to 2 to “0”.

**Remark** When the LIN-bus communication function is used, select the input signal of the RxD0 pin by setting ISC1 to 1.

### 6.3.14 Noise filter enable registers 1, 2 (NFEN1, NFEN2)

The NFEN1, NFEN2 registers are used to set whether the noise filter can be used for the timer input signal to each channel.

Enable the noise filter by setting the corresponding bits to 1 on the pins in need of noise removal. When the noise filter is enabled, after synchronization with the operating clock (fMCK) for the target channel, whether the signal keeps the same value for two clock cycles is detected.

When the noise filter is OFF, only synchronization is performed with the operation clock of target channel (fMCK).  
Note.

The NFEN1, NFEN2 registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Note** For details, see **6.5.1 (2) When valid edge of input signal via the TImn pin is selected (CCSmn = 1)**, **6.5.2 Start timing of counter**, and **6.7 Timer Input (TImn) Control**.

**Figure 6 - 26 Format of Noise filter enable register 1, 2 (NFEN1, NFEN2) (1/2)**

Address: F0071H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
NFEN1	0	0	0	0	TNFEN03	TNFEN02	TNFEN01	TNFEN00

Address: F0072H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
NFEN2	0	0	0	0	TNFEN13	TNFEN12	TNFEN11	TNFEN10

TNFEN03	Enable/disable using noise filter of TI03 pin or Rx0D pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN02	Enable/disable using noise filter of TI02 pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN01	Enable/disable using noise filter of TI01 pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN00	Enable/disable using noise filter of TI00 pin input signal
0	Noise filter OFF
1	Noise filter ON

**Note**      The applicable pin can be switched by setting the ISC1 bit of the ISC register.

ISC1 = 0: Whether or not to use the noise filter of the TI03 pin can be selected.

ISC1 = 1: Whether or not to use the noise filter of the Rx0D pin can be selected.

**Remark**      The presence or absence of timer I/O pins of channel 0 to 3 depends on the product. See **Table 6 - 2 Timer I/O Pins provided in Each Product** for details.

**Figure 6 - 27 Format of Noise filter enable register 1, 2 (NFEN1, NFEN2) (2/2)**

Address: F0071H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
NFEN1	0	0	0	0	TNFEN03	TNFEN02	TNFEN01	TNFEN00

Address: F0072H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
NFEN2	0	0	0	0	TNFEN13	TNFEN12	TNFEN11	TNFEN10

TNFEN13	Enable/disable using noise filter of TI13 pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN12	Enable/disable using noise filter of TI12 pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN11	Enable/disable using noise filter of TI11 pin input signal
0	Noise filter OFF
1	Noise filter ON

TNFEN10	Enable/disable using noise filter of TI10 pin input signal
0	Noise filter OFF
1	Noise filter ON

**Remark** The presence or absence of timer I/O pins of channel 0 to 3 depends on the product. See **Table 6 - 2 Timer I/O Pins provided in Each Product** for details.

### 6.3.15 Registers controlling port functions of pins to be used for timer I/O

Using port pins for the timer array unit functions requires setting of the registers that control the port functions multiplexed on the target pins (port mode register (PMxx), port register (Pxx), and port mode control register (PMCxx)). For details, see **4.3.1 Port mode registers (PMxx)**, **4.3.2 Port registers (Pxx)**, and **4.3.6 Port mode control registers (PMCxx)**.

The port mode register (PMxx), port register (Pxx), and port mode control register (PMCxx) to be set depend on the product. For details, see **4.5 Register Settings When Using Alternate Function**.

When using the ports (such as P00/TI00 and P01/TO00) to be shared with the timer output pin for timer output, set the port mode control register (PMCxx) bit, port mode register (PMxx) bit and port register (Pxx) bit corresponding to each port to 0.

Example: When using P01/TO00 for timer output

Set the PMC01 bit of port mode control register 0 to 0.

Set the PM01 bit of port mode register 0 to 0.

Set the P01 bit of port register 0 to 0.

When using the ports (such as P00/TI00) to be shared with the timer input pin for timer input, set the port mode register (PMxx) bit corresponding to each port to 1. And set the port mode control register (PMCxx) bit corresponding to each port to 0. At this time, the port register (Pxx) bit may be 0 or 1.

Example: When using P00/TI00 for timer input

Set the PMC00 bit of port mode control register 0 to 0.

Set the PM00 bit of port mode register 0 to 1.

Set the P00 bit of port register 0 to 0 or 1.

## 6.4 Basic Rules of Timer Array Unit

### 6.4.1 Basic rules of simultaneous channel operation function

When simultaneously using multiple channels, namely, a combination of a master channel (a reference timer mainly counting the cycle) and slave channels (timers operating according to the master channel), the following rules apply.

- (1) Only an even channel (channel 0, 2) can be set as a master channel.
- (2) Any channel, except channel 0, can be set as a slave channel.
- (3) The slave channel must be lower than the master channel.

Example: If channel 0 is set as a master channel, channel 1 or those that follow (channels 1, 2, 3) can be set as a slave channel.

- (4) Two or more slave channels can be set for one master channel.
- (5) When two or more master channels are to be used, slave channels with a master channel between them may not be set.

Example: If channels 0 and 2 are set as master channels, channel 1 can be set as the slave channel of master channel 0. Channel 3 cannot be set as the slave channel of master channel 0.

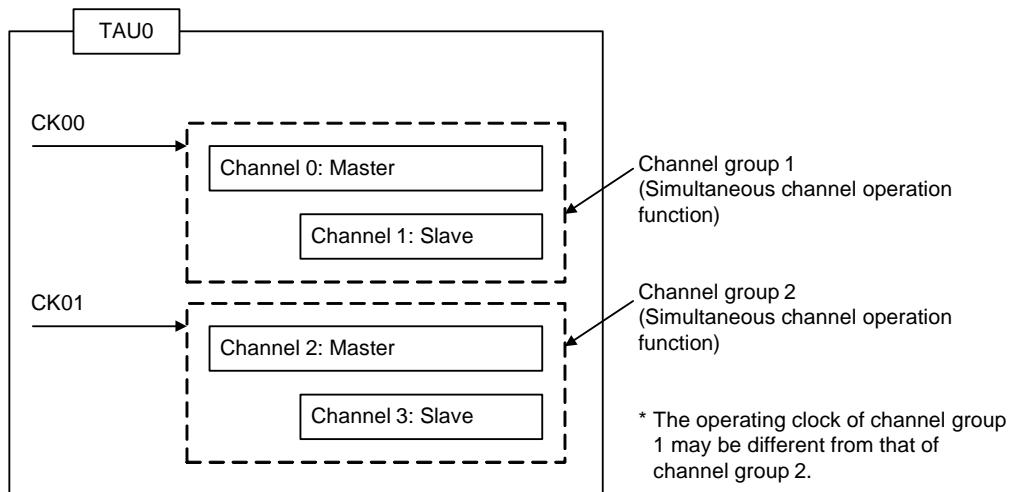
- (6) The operating clock for a slave channel in combination with a master channel must be the same as that of the master channel. The CKSmn0, CKSmn1 bits (bit 15, 14 of timer mode register mn (TMRmn)) of the slave channel that operates in combination with the master channel must be the same value as that of the master channel.
- (7) A master channel can transmit INTTMmn (interrupt), start software trigger, and count clock to the lower channels.
- (8) A slave channel can use INTTMmn (interrupt), a start software trigger, or the count clock of the master channel as a source clock, but cannot transmit its own INTTMmn (interrupt), start software trigger, or count clock to channels with lower channel numbers.
- (9) A master channel cannot use INTTMmn (interrupt), a start software trigger, or the count clock from the other higher master channel as a source clock.
- (10) To simultaneously start channels that operate in combination, the channel start trigger bit (TSmn) of the channels in combination must be set at the same time.
- (11) During the counting operation, a TSmn bit of a master channel or TSmn bits of all channels which are operating simultaneously can be set. It cannot be applied to TSmn bits of slave channels alone.
- (12) To stop the channels in combination simultaneously, the channel stop trigger bit (TTmn) of the channels in combination must be set at the same time.
- (13) CKm2/CKm3 cannot be selected while channels are operating simultaneously, because the operating clocks of master channels and slave channels have to be synchronized.
- (14) Timer mode register m0 (TMRm0) has no master bit (it is fixed as "0"). However, as channel 0 is the highest channel, it can be used as a master channel during simultaneous operation.

The rules of the simultaneous channel operation function are applied in a channel group (a master channel and slave channels forming one simultaneous channel operation function).

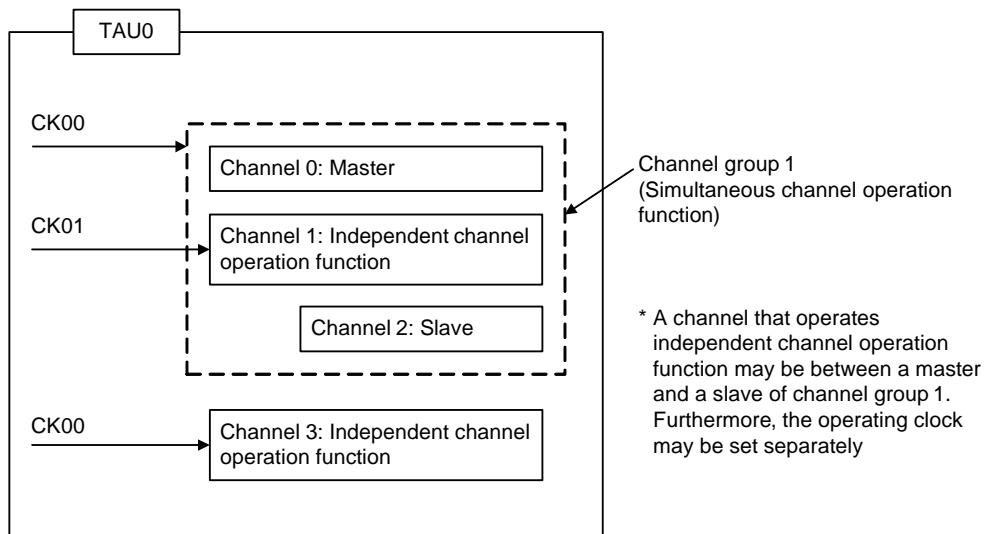
If two or more channel groups that do not operate in combination are specified, the basic rules of the simultaneous channel operation function in **6.4.1 Basic rules of simultaneous channel operation function** do not apply to the channel groups.

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

Example 1



Example 2



### 6.4.2 Basic rules of 8-bit timer operation function (channels 1 and 3 only)

The 8-bit timer operation function makes it possible to use a 16-bit timer channel in a configuration consisting of two 8-bit timer channels.

This function can only be used for channels 1 and 3, and there are several rules for using it.

The basic rules for this function are as follows:

- (1) The 8-bit timer operation function applies only to channels 1 and 3.
- (2) When using 8-bit timers, set the SPLIT bit of timer mode register mn (TMRmn) to 1.
- (3) The higher 8 bits can be operated as the interval timer function.
- (4) At the start of operation, the higher 8 bits output INTTMm1H/INTTMm3H (an interrupt) (which is the same operation performed when MDmn0 is set to 1).
- (5) The operation clock of the higher 8 bits is selected according to the CKSmn1 and CKSmn0 bits of the lower-bit TMRmn register.
- (6) For the higher 8 bits, the TSHm1/TSHm3 bit is manipulated to start channel operation and the TTHm1/TTHm3 bit is manipulated to stop channel operation. The channel status can be checked using the TEHm1/TEHm3 bit.
- (7) The lower 8 bits operate according to the TMRmn register settings. The following three functions support operation of the lower 8 bits:
  - Interval timer function
  - External event counter function
  - Delay count function
- (8) For the lower 8 bits, the TSm1/TSm3 bit is manipulated to start channel operation and the TTm1/TTm3 bit is manipulated to stop channel operation. The channel status can be checked using the TEM1/TEM3 bit.
- (9) During 16-bit operation, manipulating the TSHm1, TSHm3, TTHm1, and TTHm3 bits is invalid. The TSm1, TSm3, TTm1, and TTm3 bits are manipulated to operate channels 1 and 3. The TEHm3 and TEHm1 bits are not changed.
- (10) For the 8-bit timer function, the simultaneous operation functions (one-shot pulse, PWM, and multiple PWM) cannot be used.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 1, 3)

## 6.5 Operation of Counter

### 6.5.1 Count clock (fTCLK)

The count clock (fTCLK) of the timer array unit can be selected between following by CCSmn bit of timer mode register mn (TMRmn).

- Operation clock (fmck) specified by the CKSmn0 and CKSmn1 bits
- Valid edge of input signal input from the TImn pin

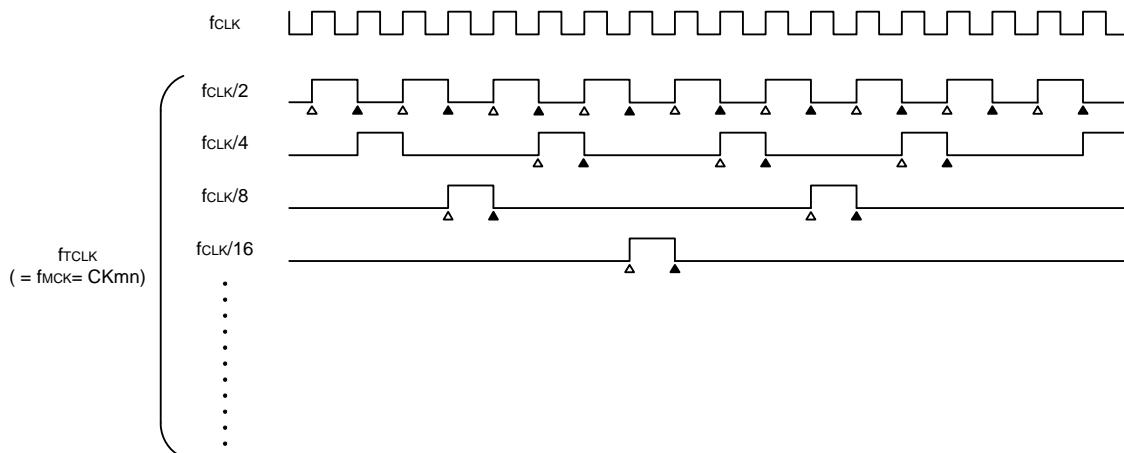
Because the timer array unit is designed to operate in synchronization with fCLK, the timings of the count clock (fTCLK) are shown below.

- (1) When operation clock (fmck) specified by the CKSmn0 and CKSmn1 bits is selected (CCSmn = 0)

The count clock (fTCLK) is between fCLK to fCLK /2<sup>15</sup> by setting of timer clock select register m (TPSm). When a divided fCLK is selected, however, the clock selected in TPSmn register, but a signal which becomes high level for one period of fCLK from its rising edge. When a fCLK is selected, fixed to high level.

Counting of timer count register mn (TCRmn) delayed by one period of fCLK from rising edge of the count clock, because of synchronization with fCLK. But, this is described as “counting at rising edge of the count clock”, as a matter of convenience.

**Figure 6 - 28 Timing of fCLK and count clock (fTCLK) (When CCSmn = 0)**



**Remark 1.** △ : Rising edge of the count clock

▲ : Synchronization, increment/decrement of counter

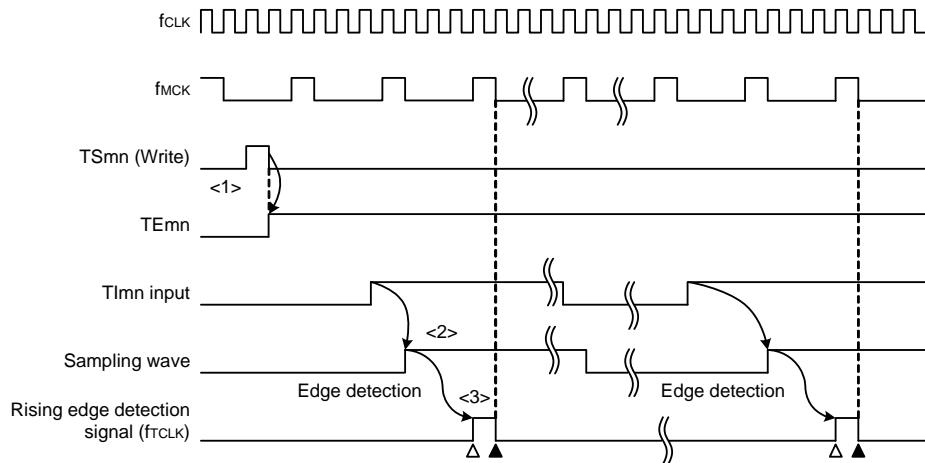
**Remark 2.** fCLK: CPU/peripheral hardware clock

- (2) When valid edge of input signal via the TImn pin is selected ( $CCSmn = 1$ )

The count clock ( $f_{TCLK}$ ) becomes the signal that detects valid edge of input signal via the TImn pin and synchronizes next rising fmck. The count clock ( $f_{TCLK}$ ) is delayed for 1 to 2 period of fmck from the input signal via the TImn pin (when a noise filter is used, the delay becomes 3 to 4 clock).

Counting of timer count register mn (TCRmn) delayed by one period of fCLK from rising edge of the count clock, because of synchronization with fCLK. But, this is described as “counting at valid edge of input signal via the TImn pin”, as a matter of convenience.

**Figure 6 - 29 Timing of fCLK and count clock (f<sub>TCLK</sub>) (When CCSmn = 1, noise filter unused)**



<1> Setting TSmn bit to 1 enables the timer to be started and to become wait state for valid edge of input signal via the TImn pin.

<2> The rise of input signal via the TImn pin is sampled by fmck.

<3> The edge is detected by the rising of the sampled signal and the detection signal (count clock) is output.

**Remark 1.**  $\Delta$  : Rising edge of the count clock

$\blacktriangle$  : Synchronization, increment/decrement of counter

**Remark 2.** fCLK: CPU/peripheral hardware clock

fmck: Operation clock of channel n

**Remark 3.** The waveform of the input signal via TImn pin of the input pulse interval measurement, the measurement of high/low width of input signal, and the delay counter, and the one-shot pulse output are the same as that shown in Figure 6 - 29.

### 6.5.2 Start timing of counter

Timer count register mn (TCRmn) becomes enabled to operation by setting of TSmn bit of timer channel start register m (TSm).

Operations from count operation enabled state to timer count Register mn (TCRmn) count start is shown in Table 6 - 6.

**Table 6 - 6 Operations from Count Operation Enabled State to Timer count Register mn (TCRmn) Count Start**

Timer operation mode	Operation when TSmn = 1 is set
• Interval timer mode	No operation is carried out from start trigger detection (TSmn=1) until count clock generation. The first count clock loads the value of the TDRmn register to the TCRmn register and the subsequent count clock performs count down operation (see <b>6.5.3 (1) Operation of interval timer mode</b> ).
• Event counter mode	Writing 1 to the TSmn bit loads the value of the TDRmn register to the TCRmn register. If detect edge of TImn input, the subsequent count clock performs count down operation (see <b>6.5.3 (2) Operation of event counter mode</b> ).
• Capture mode	No operation is carried out from start trigger detection until count clock generation. The first count clock loads 0000H to the TCRmn register and the subsequent count clock performs count up operation (see <b>6.5.3 (3) Operation of capture mode (input pulse interval measurement)</b> ).
• One-count mode	The waiting-for-start-trigger state is entered by writing 1 to the TSmn bit while the timer is stopped (TEmn = 0). No operation is carried out from start trigger detection until count clock generation. The first count clock loads the value of the TDRmn register to the TCRmn register and the subsequent count clock performs count down operation (see <b>6.5.3 (4) Operation of one-count mode</b> ).
• Capture & one-count mode	The waiting-for-start-trigger state is entered by writing 1 to the TSmn bit while the timer is stopped (TEmn = 0). No operation is carried out from start trigger detection until count clock generation. The first count clock loads 0000H to the TCRmn register and the subsequent count clock performs count up operation (see <b>6.5.3 (5) Operation of capture &amp; one-count mode (high-level width measurement)</b> ).

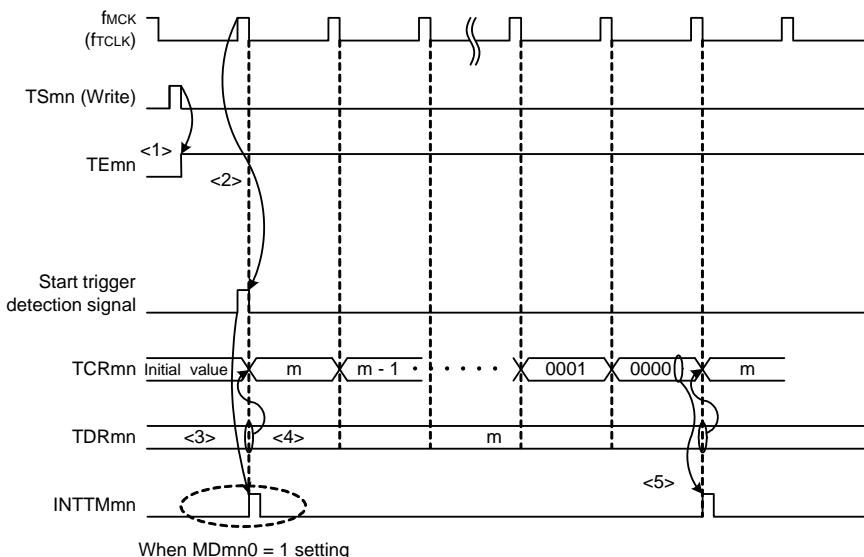
### 6.5.3 Operation of counter

Here, the counter operation in each mode is explained.

#### (1) Operation of interval timer mode

- <1> Operation is enabled ( $TE_{mn} = 1$ ) by writing 1 to the  $TS_{mn}$  bit. Timer count register  $mn$  ( $TCR_{mn}$ ) holds the initial value until count clock generation.
- <2> A start trigger is generated at the first count clock after operation is enabled.
- <3> When the  $MD_{mn0}$  bit is set to 1,  $INTT{M}_{mn}$  is generated by the start trigger.
- <4> By the first count clock after the operation enable, the value of timer data register  $mn$  ( $TDR_{mn}$ ) is loaded to the  $TCR_{mn}$  register and counting starts in the interval timer mode.
- <5> When the  $TCR_{mn}$  register counts down and its count value is  $0000H$ ,  $INTT{M}_{mn}$  is generated and the value of timer data register  $mn$  ( $TDR_{mn}$ ) is loaded to the  $TCR_{mn}$  register and counting keeps on.

**Figure 6 - 30 Operation Timing (In Interval Timer Mode)**

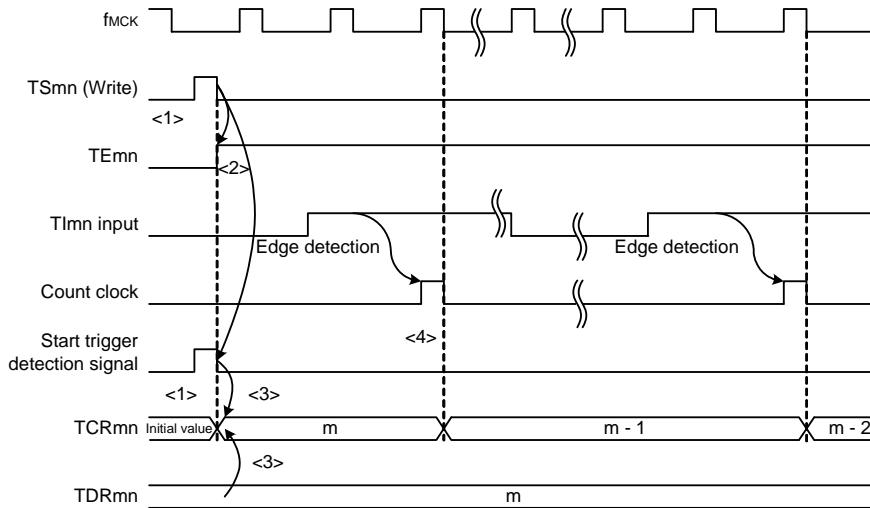


**Caution** In the first cycle operation of count clock after writing the  $TS_{mn}$  bit, an error at a maximum of one clock is generated since count start delays until count clock has been generated. When the information on count start timing is necessary, an interrupt can be generated at count start by setting  $MD_{mn0} = 1$ .

**Remark** fMCK, the start trigger detection signal, and  $INTT{M}_{mn}$  become active between one clock in synchronization with fCLK.

## (2) Operation of event counter mode

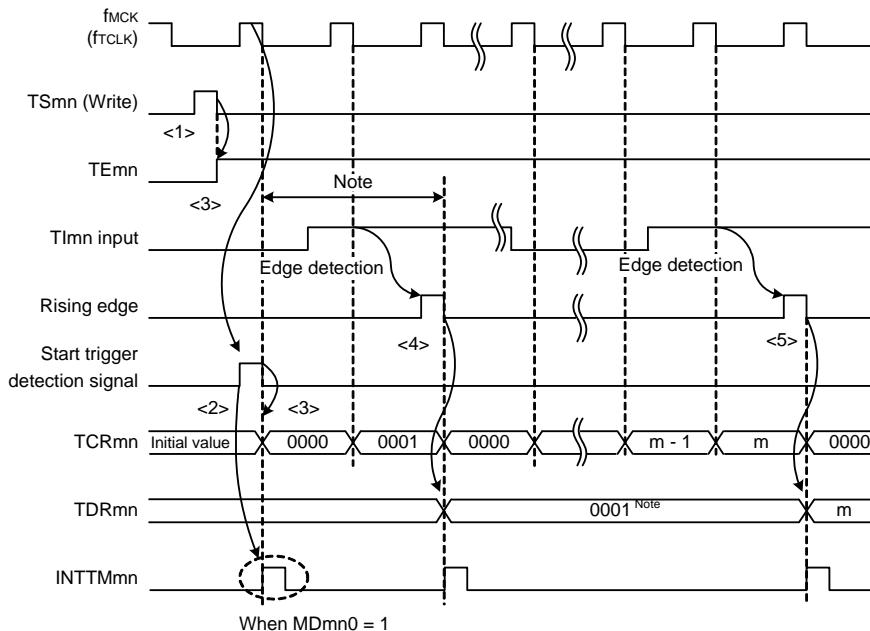
- <1> Timer count register mn (TCRmn) holds its initial value while operation is stopped (TEmn = 0).
- <2> Operation is enabled (TEmn = 1) by writing 1 to the TSmn bit.
- <3> As soon as 1 has been written to the TSmn bit and 1 has been set to the TEmn bit, the value of timer data register mn (TDRmn) is loaded to the TCRmn register to start counting.
- <4> After that, the TCRmn register value is counted down according to the count clock of the valid edge of the TImn input.

**Figure 6 - 31 Operation Timing (In Event Counter Mode)**

**Remark** Figure 6 - 31 shows the timing when the noise filter is not used. By making the noise filter on-state, the edge detection becomes 2 fmck cycles (it sums up to 3 to 4 cycles) later than the normal cycle of TImn input. The error per one period occurs be the asynchronous between the period of the TImn input and that of the count clock (fmck).

- (3) Operation of capture mode (input pulse interval measurement)
- <1> Operation is enabled ( $TE_{mn} = 1$ ) by writing 1 to the  $TS_{mn}$  bit.
  - <2> Timer count register  $mn$  ( $TCR_{mn}$ ) holds the initial value until count clock generation.
  - <3> A start trigger is generated at the first count clock after operation is enabled. And the value of  $0000H$  is loaded to the  $TCR_{mn}$  register and counting starts in the capture mode. (When the  $MD_{mn0}$  bit is set to 1,  $INTT_{Mmn}$  is generated by the start trigger.)
  - <4> On detection of the valid edge of the  $TImn$  input, the value of the  $TCR_{mn}$  register is captured to timer data register  $mn$  ( $TDR_{mn}$ ) and  $INTT_{Mmn}$  is generated. However, this capture value is no meaning. The  $TCR_{mn}$  register keeps on counting from  $0000H$ .
  - <5> On next detection of the valid edge of the  $TImn$  input, the value of the  $TCR_{mn}$  register is captured to timer data register  $mn$  ( $TDR_{mn}$ ) and  $INTT_{Mmn}$  is generated.

**Figure 6 - 32 Operation Timing (In Capture Mode: Input Pulse Interval Measurement)**



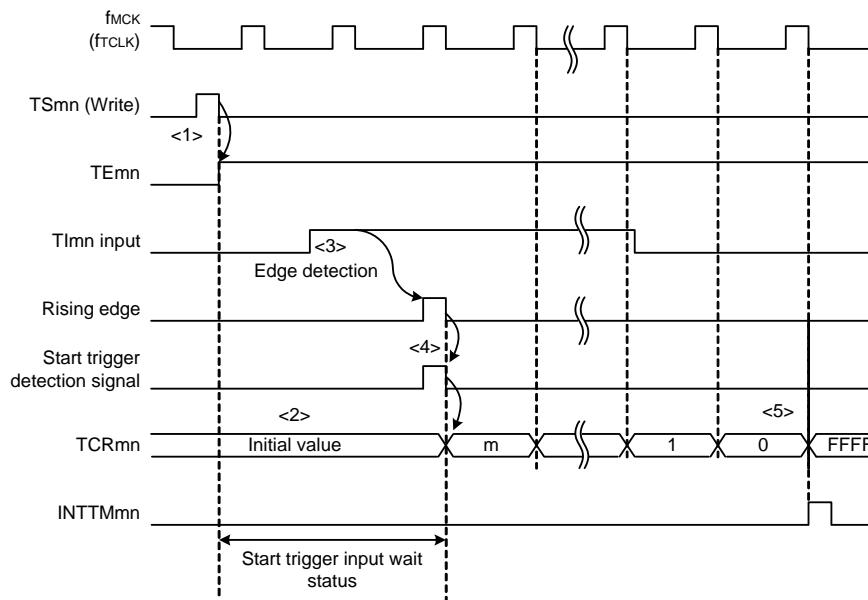
**Note** If a clock has been input to  $TImn$  (the trigger exists) when capturing starts, counting starts when a trigger is detected, even if no edge is detected. Therefore, the first captured value (<4>) does not determine a pulse interval (in the above figure, 0001 just indicates two clock cycles but does not determine the pulse interval) and so the user can ignore it.

**Caution** In the first cycle operation of count clock after writing the  $TS_{mn}$  bit, an error at a maximum of one clock is generated since count start delays until count clock has been generated. When the information on count start timing is necessary, an interrupt can be generated at count start by setting  $MD_{mn0} = 1$ .

**Remark** Figure 6 - 32 shows the timing when the noise filter is not used. By making the noise filter on-state, the edge detection becomes 2 fmck cycles (it sums up to 3 to 4 cycles) later than the normal cycle of  $TImn$  input. The error per one period occurs be the asynchronous between the period of the  $TImn$  input and that of the count clock (fmck).

## (4) Operation of one-count mode

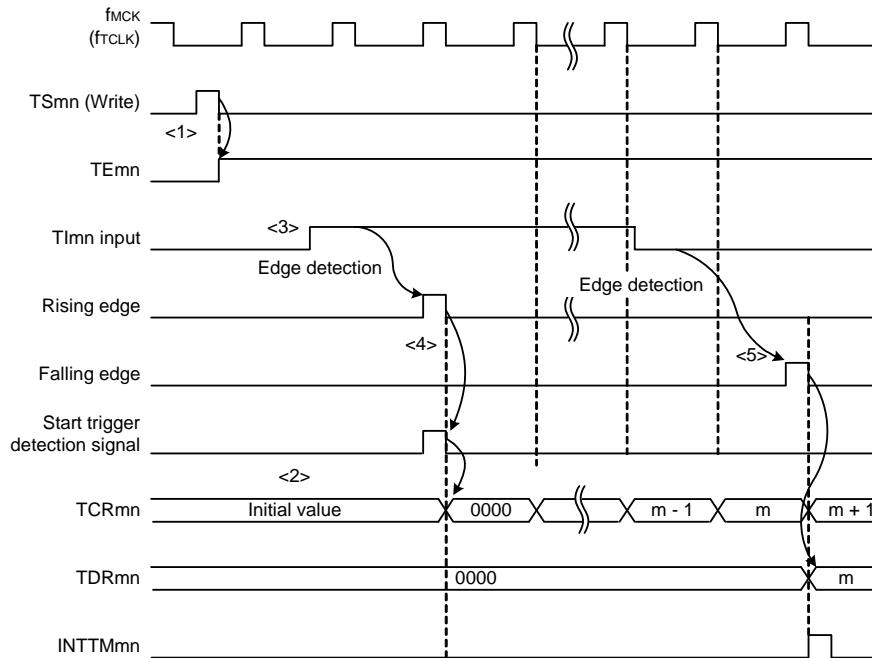
- <1> Operation is enabled ( $TE_{mn} = 1$ ) by writing 1 to the  $TS_{mn}$  bit.
- <2> Timer count register  $mn$  ( $TCR_{mn}$ ) holds the initial value until start trigger generation.
- <3> Rising edge of the  $TImn$  input is detected.
- <4> On start trigger detection, the value of timer data register  $mn$  ( $TDR_{mn}$ ) is loaded to the  $TCR_{mn}$  register and count starts.
- <5> When the  $TCR_{mn}$  register counts down and its count value is  $0000H$ ,  $INTT_{Mmn}$  is generated and the value of the  $TCR_{mn}$  register becomes  $FFFFH$  and counting stops.

**Figure 6 - 33 Operation Timing (In One-count Mode)**

**Remark** Figure 6 - 33 shows the timing when the noise filter is not used. By making the noise filter on-state, the edge detection becomes 2 fmck cycles (it sums up to 3 to 4 cycles) later than the normal cycle of  $TImn$  input. The error per one period occurs be the asynchronous between the period of the  $TImn$  input and that of the count clock (fmck).

- (5) Operation of capture & one-count mode (high-level width measurement)
- <1> Operation is enabled (TEmn = 1) by writing 1 to the TSmn bit of timer channel start register m (TSm).
  - <2> Timer count register mn (TCRmn) holds the initial value until start trigger generation.
  - <3> Rising edge of the TImn input is detected.
  - <4> On start trigger detection, the value of 0000H is loaded to the TCRmn register and count starts.
  - <5> On detection of the falling edge of the TImn input, the value of the TCRmn register is captured to timer data register mn (TDRmn) and INTTMmn is generated.

**Figure 6 - 34 Operation Timing (In Capture & One-count Mode: High-level Width Measurement)**

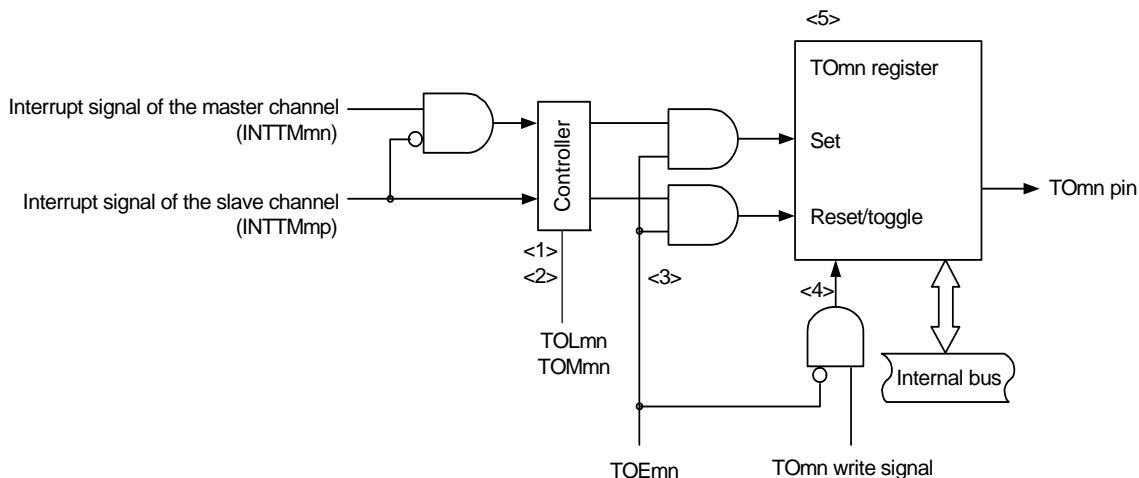


**Remark** Figure 6 - 34 shows the timing when the noise filter is not used. By making the noise filter on-state, the edge detection becomes 2 fmck cycles (it sums up to 3 to 4 cycles) later than the normal cycle of TImn input. The error per one period occurs be the asynchronous between the period of the TImn input and that of the count clock (fmck).

## 6.6 Channel Output (TOmn pin) Control

### 6.6.1 TOmn pin output circuit configuration

Figure 6 - 35 Output Circuit Configuration



The following describes the TOmn pin output circuit.

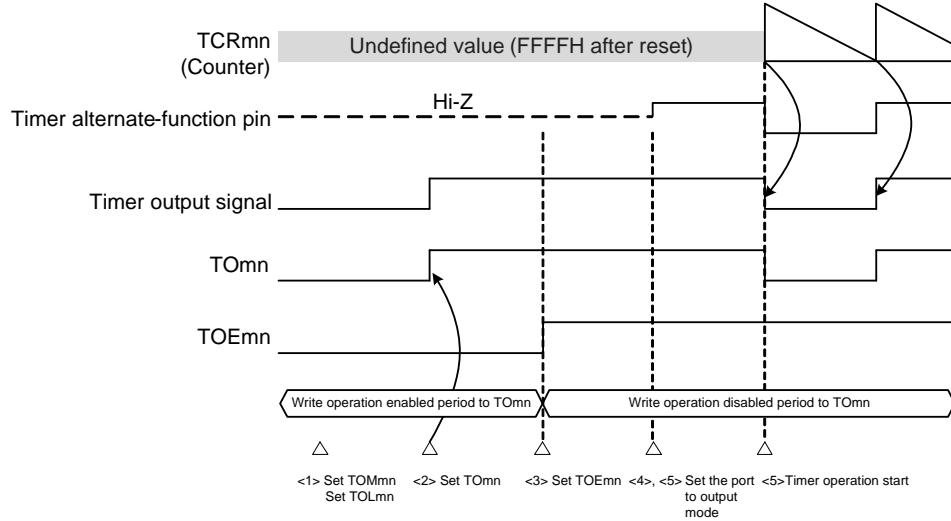
- <1> When  $\text{TOMmn} = 0$  (master channel output mode), the set value of timer output level register m ( $\text{TOLm}$ ) is ignored and only  $\text{INTTM0p}$  (slave channel timer interrupt) is transmitted to timer output register m ( $\text{TOm}$ ).
- <2> When  $\text{TOMmn} = 1$  (slave channel output mode), both  $\text{INTTMmn}$  (master channel timer interrupt) and  $\text{INTTM0p}$  (slave channel timer interrupt) are transmitted to the  $\text{TOm}$  register.  
At this time, the  $\text{TOLm}$  register becomes valid and the signals are controlled as follows:
  - When  $\text{TOLmn} = 0$ : Forward operation ( $\text{INTTMmn} \rightarrow \text{set}$ ,  $\text{INTTM0p} \rightarrow \text{reset}$ )
  - When  $\text{TOLmn} = 1$ : Reverse operation ( $\text{INTTMmn} \rightarrow \text{reset}$ ,  $\text{INTTM0p} \rightarrow \text{set}$ )
 When  $\text{INTTMmn}$  and  $\text{INTTM0p}$  are simultaneously generated, (0% output of PWM),  $\text{INTTM0p}$  (reset signal) takes priority, and  $\text{INTTMmn}$  (set signal) is masked.
- <3> While timer output is enabled ( $\text{TOEmn} = 1$ ),  $\text{INTTMmn}$  (master channel timer interrupt) and  $\text{INTTM0p}$  (slave channel timer interrupt) are transmitted to the  $\text{TOm}$  register. Writing to the  $\text{TOm}$  register ( $\text{TOmn write signal}$ ) becomes invalid.
- When  $\text{TOEmn} = 1$ , the  $\text{TOmn}$  pin output never changes with signals other than interrupt signals.  
To initialize the  $\text{TOmn}$  pin output level, it is necessary to stop timer operation ( $\text{TOEmn} = 0$ ) and to write a value to the  $\text{TOm}$  register.
- <4> While timer output is disabled ( $\text{TOEmn} = 0$ ), writing to the  $\text{TOmn}$  bit to the target channel ( $\text{TOmn write signal}$ ) becomes valid. When timer output is disabled ( $\text{TOEmn} = 0$ ), neither  $\text{INTTMmn}$  (master channel timer interrupt) nor  $\text{INTTM0p}$  (slave channel timer interrupt) is transmitted to the  $\text{TOm}$  register.
- <5> The  $\text{TOm}$  register can always be read, and the  $\text{TOmn}$  pin output level can be checked.

<b>Remark</b>	m: Unit number ( $m = 0, 1$ ) n: Channel number $n = 0$ to $3$ ( $n = 0, 2$ for master channel) p: Slave channel number $n = 0$ : p = 1, 2, 3 $n = 2$ : p = 3
---------------	--

## 6.6.2 TOmn Pin Output Setting

The following figure shows the procedure and status transition of the TOmn output pin from initial setting to timer operation start.

**Figure 6 - 36 Status Transition from Timer Output Setting to Operation Start**



<1> The operation mode of timer output is set.

- TOMmn bit (0: Master channel output mode, 1: Slave channel output mode)
- TOLmn bit (0: Positive logic output, 1: Negative logic output)

<2> The timer output signal is set to the initial status by setting timer output register m (TOm).

<3> The timer output operation is enabled by writing 1 to the TOEmn bit (writing to the TOm register is disabled).

<4> The port is set to digital I/O by port mode control register (PMCxx) (see **6.3.15 Registers controlling port functions of pins to be used for timer I/O**).

<5> The port I/O setting is set to output (see **6.3.15 Registers controlling port functions of pins to be used for timer I/O**).

<6> The timer operation is enabled (TSmn = 1).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.6.3 Cautions on Channel Output Operation

- (1) Changing values set in the registers TOm, TOEm, TOLm, and TOMm during timer operation

Since the timer operations (operations of timer count register mn (TCRmn) and timer data register mn (TDRmn)) are independent of the TOmn output circuit and changing the values set in timer output register m (TOm), timer output enable register m (TOEm), and timer output level register m (TOLm) does not affect the timer operation, the values can be changed during timer operation. To output an expected waveform from the TOmn pin by timer operation, however, set the TOm, TOEm, TOLm, and TOMm registers to the values stated in the register setting example of each operation shown by 6.8 and 6.9.

When the values set to the TOEm and TOLm registers (but not the TOm register) are changed close to the occurrence of the timer interrupt (INTTMmn) of each channel, the waveform output to the TOmn pin might differ, depending on whether the values are changed immediately before or immediately after the timer interrupt (INTTMmn) occurs.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

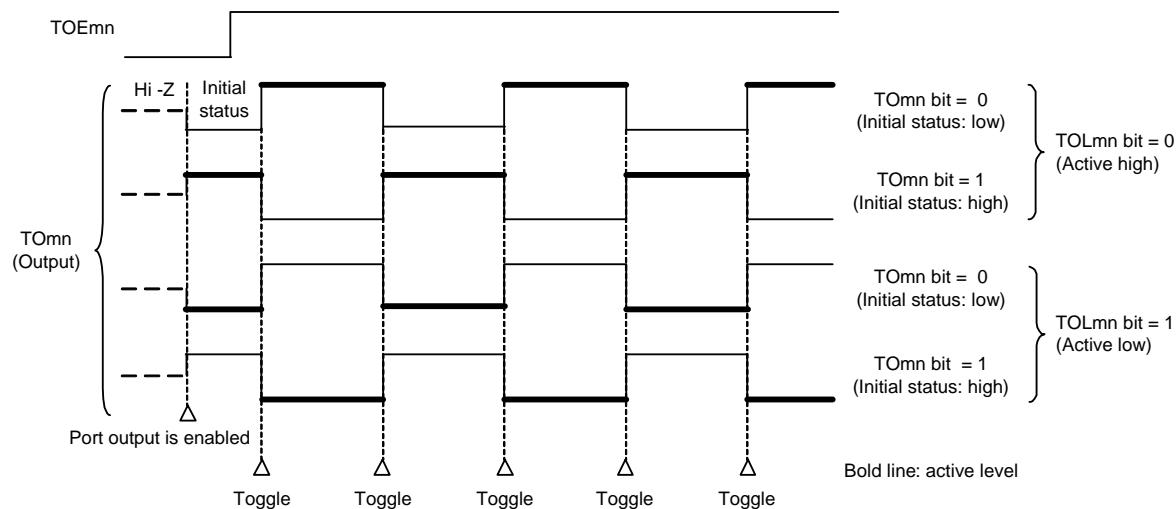
- (2) Default level of TOmn pin and output level after timer operation start

The change in the output level of the TOmn pin when timer output register m (TOm) is written while timer output is disabled ( $\text{TOEmn} = 0$ ), the initial level is changed, and then timer output is enabled ( $\text{TOEmn} = 1$ ) before port output is enabled, is shown below.

- (a) When operation starts with master channel output mode ( $\text{TOMmn} = 0$ ) setting

The setting of timer output level register m (TOLm) is invalid when master channel output mode ( $\text{TOMmn} = 0$ ). When the timer operation starts after setting the default level, the toggle signal is generated and the output level of the TOmn pin is reversed.

**Figure 6 - 37 TOmn Pin Output Status at Toggle Output ( $\text{TOMmn} = 0$ )**

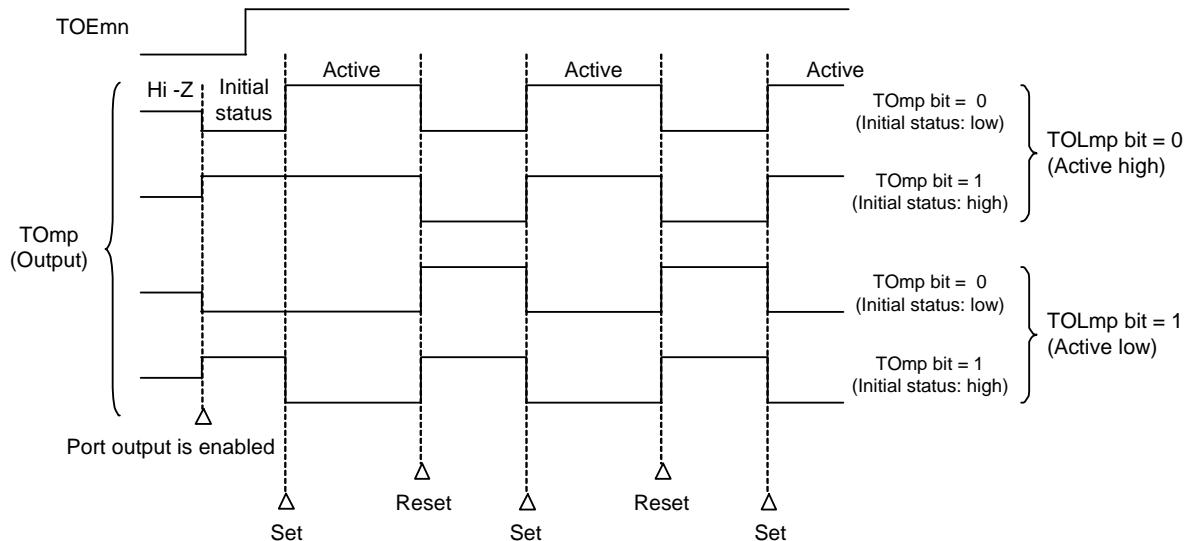


**Remark 1.** Toggle: Reverse TOmn pin output status

**Remark 2.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

- (b) When operation starts with slave channel output mode ( $\text{TO}M_{mn} = 1$ ) setting (PWM output))  
 When slave channel output mode ( $\text{TO}M_{mn} = 1$ ), the active level is determined by timer output level register m ( $\text{TO}L_m$ ) setting.

**Figure 6 - 38  $\text{TO}_{mn}$  Pin Output Status at PWM Output ( $\text{TO}M_{mn} = 1$ )**



**Remark 1.** Set: The output signal of the  $\text{TOmp}$  pin changes from inactive level to active level.

Reset: The output signal of the  $\text{TOmp}$  pin changes from active level to inactive level.

**Remark 2.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $p = 1$  to  $3$ )

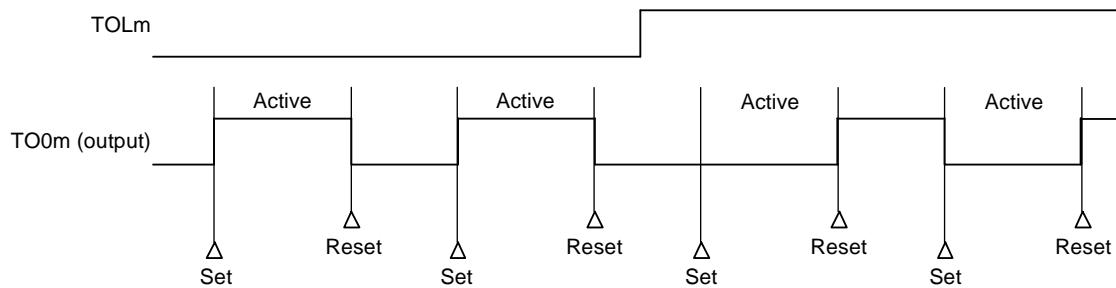
(3) Operation of TOmn pin in slave channel output mode ( $TOMmn = 1$ )

## (a) When timer output level register m (TOLm) setting has been changed during timer operation

When the TOLm register setting has been changed during timer operation, the setting becomes valid at the generation timing of the TOmn pin change condition. Rewriting the TOLm register does not change the output level of the TOmn pin.

The operation when TOMmn is set to 1 and the value of the TOLm register is changed while the timer is operating ( $TEmn = 1$ ) is shown below.

**Figure 6 - 39 Operation when TOLm Register Has Been Changed during Timer Operation**



**Remark 1.** Set: The output signal of the TOmn pin changes from inactive level to active level.

Reset: The output signal of the TOmn pin changes from active level to inactive level.

**Remark 2.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

## (b) Set/reset timing

To realize 0%/100% output at PWM output, the TOmn pin/TOMn bit set timing at master channel timer interrupt (INTTMmn) generation is delayed by 1 count clock by the slave channel.

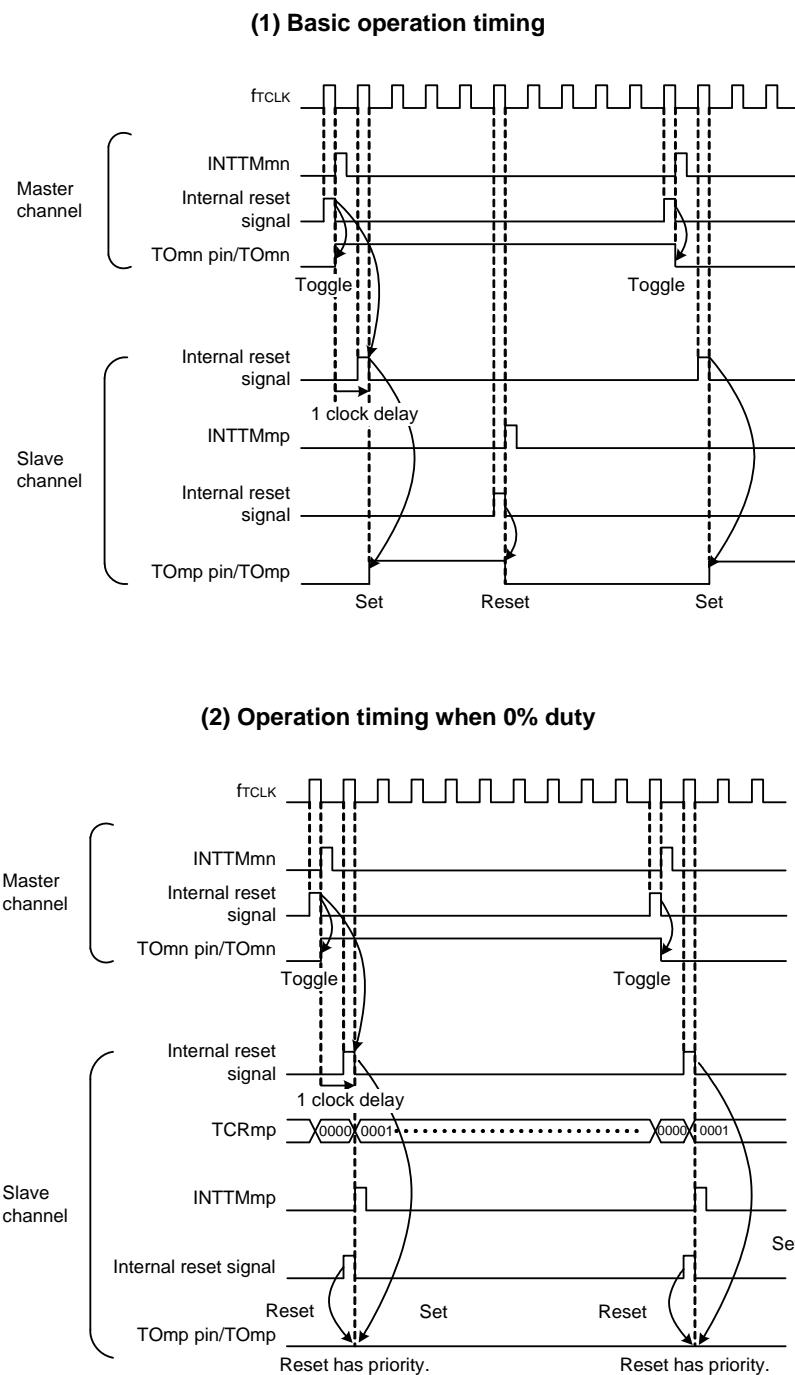
If the set condition and reset condition are generated at the same time, a higher priority is given to the latter.

Figure 6 - 40 shows the set/reset operating statuses where the master/slave channels are set as follows.

Master channel:  $TOEmn = 1$ ,  $TOMmn = 0$ ,  $TOLmn = 0$

Slave channel:  $TOEmp = 1$ ,  $TOMmp = 1$ ,  $TOLmp = 0$

Figure 6 - 40 Set/Reset Timing Operating Statuses



**Remark 1.** Internal reset signal:TOmn pin reset/toggle signal

Internal set signal: TOmn pin set signal

**Remark 2.** m: Unit number ( $m = 0, 1$ )

n: Channel number

$n = 0$  to  $3$  ( $n = 0, 2$  for master channel)

p: Slave channel number

$n = 0: p = 1, 2, 3$

$n = 2: p = 3$

### 6.6.4 Collective manipulation of TOmn bit

In timer output register m (TOm), the setting bits for all the channels are located in one register in the same way as timer channel start register m (TSm). Therefore, the TOmn bit of all the channels can be manipulated collectively.

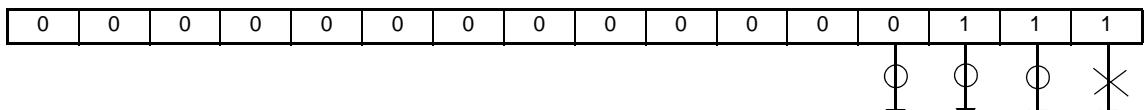
Only the desired bits can also be manipulated by enabling writing only to the TOmn bits (TOEmn = 0) that correspond to the relevant bits of the channel used to perform output (TOmn).

**Figure 6 - 41 Example of TO0n Bit Collective Manipulation**

Before writing

TO0	0	0	0	0	0	0	0	0	0	0	0	0	TO03 1	TO02 0	TO01 1	TO00 0
TOE0	0	0	0	0	0	0	0	0	0	0	0	0	TOE03 0	TOE02 0	TOE01 0	TOE00 1

Data to be written



After writing

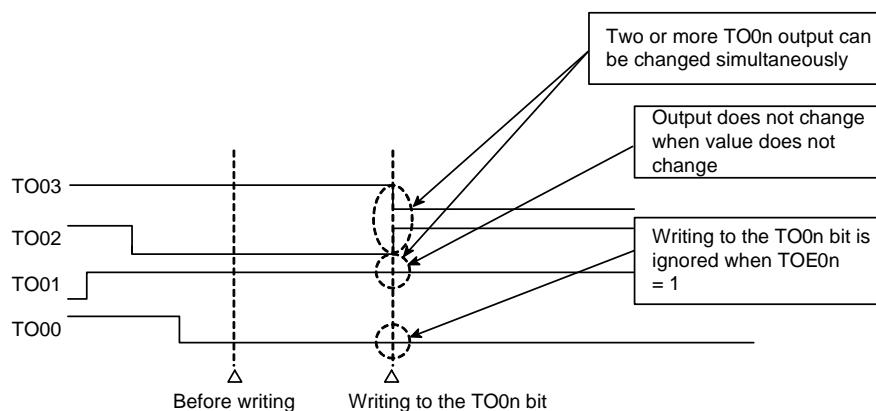
TO0	0	0	0	0	0	0	0	0	0	0	0	0	TO03 0	TO02 1	TO01 1	TO00 0
-----	---	---	---	---	---	---	---	---	---	---	---	---	-----------	-----------	-----------	-----------

Writing is done only to the TOmn bit with TOEmn = 0, and writing to the TOmn bit with TOEmn = 1 is ignored.

TOmn (channel output) to which TOEmn = 1 is set is not affected by the write operation. Even if the write operation is done to the TOmn bit, it is ignored and the output change by timer operation is normally done.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 42 TO0n Pin Statuses by Collective Manipulation of TO0n Bit**



**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.6.5 Timer Interrupt and TOmn Pin Output at Operation Start

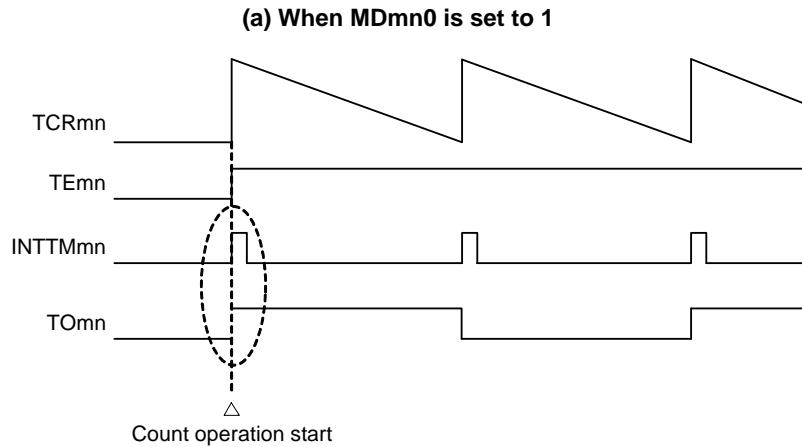
In the interval timer mode or capture mode, the MDmn0 bit in timer mode register mn (TMRmn) sets whether or not to generate a timer interrupt at count start.

When MDmn0 is set to 1, the count operation start timing can be known by the timer interrupt (INTTMmn) generation.

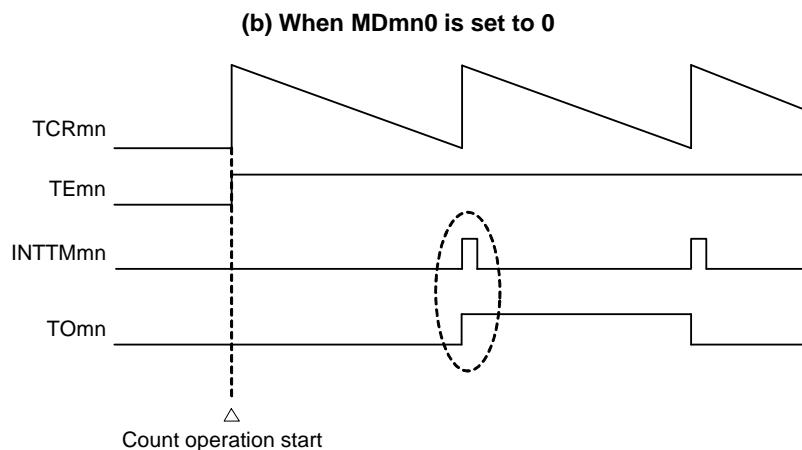
In the other modes, neither timer interrupt at count operation start nor TOmn output is controlled.

Figure 6 - 43 shows operation examples when the interval timer mode (TOEmn = 1, TOMmn = 0) is set.

**Figure 6 - 43 Operation examples of timer interrupt at count operation start and TOmn output**



When MDmn0 is set to 1, a timer interrupt (INTTMmn) is output at count operation start, and TOmn performs a toggle operation.



When MDmn0 is set to 0, a timer interrupt (INTTMmn) is not output at count operation start, and TOmn does not change either. After counting one cycle, INTTMmn is output and TOmn performs a toggle operation.

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ )

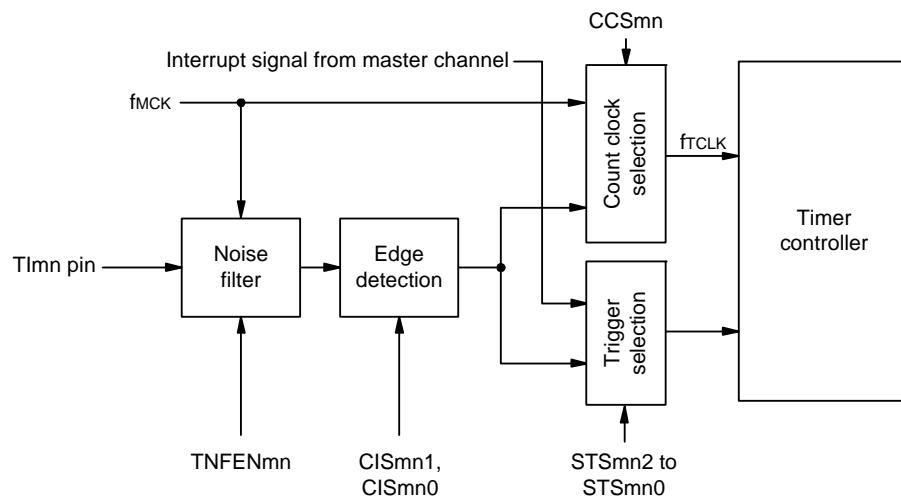
## 6.7 Timer Input (Tlmn) Control

### 6.7.1 Tlmn input circuit configuration

A signal is input from a timer input pin, goes through a noise filter and an edge detector, and is sent to a timer controller.

Enable the noise filter for the pin in need of noise removal. The following shows the configuration of the input circuit.

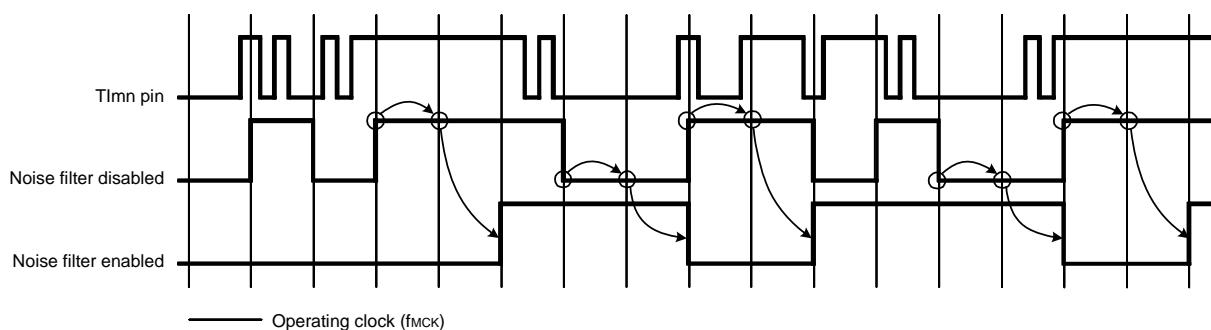
**Figure 6 - 44 Input Circuit Configuration**



### 6.7.2 Noise filter

When the noise filter is disabled, the input signal is only synchronized with the operating clock (fmck) for channel n. When the noise filter is enabled, after synchronization with the operating clock (fmck) for channel n, whether the signal keeps the same value for two clock cycles is detected. The following shows differences in waveforms output from the noise filter between when the noise filter is enabled and disabled.

**Figure 6 - 45 Sampling Waveforms through Tlmn Input Pin with Noise Filter Enabled and Disabled**



**Caution** The input waveforms to the Tlmn pin are shown to explain the operation when the noise filter is enabled or disabled. When actually inputting waveforms, input them according to the Tlmn input high-level and low-level widths listed in 34.4 or 35.4 AC Characteristics.

### 6.7.3 Cautions on channel input operation

When a timer input pin is set as unused, the operating clock is not supplied to the noise filter. Therefore, after settings are made to use the timer input pin, the following wait time is necessary before a trigger is specified to enable operation of the channel corresponding to the timer input pin.

(1) Noise filter is disabled

When bits 12 (CCSmn), 9 (STS<sub>Mn1</sub>), and 8 (STS<sub>Mn0</sub>) in the timer mode register mn (TMR<sub>Mn</sub>) are 0 and then one of them is set to 1, wait for at least two cycles of the operating clock (fMCK), and then set the operation enable trigger bit in the timer channel start register (TSm).

(2) Noise filter is enabled

When bits 12 (CCSmn), 9 (STS<sub>Mn1</sub>), and 8 (STS<sub>Mn0</sub>) in the timer mode register mn (TMR<sub>Mn</sub>) are all 0 and then one of them is set to 1, wait for at least four cycles of the operating clock (fMCK), and then set the operation enable trigger bit in the timer channel start register (TSm).

## 6.8 Independent Channel Operation Function of Timer Array Unit

### 6.8.1 Operation as interval timer/square wave output

#### (1) Interval timer

The timer array unit can be used as a reference timer that generates INTTMmn (timer interrupt) at fixed intervals.

The interrupt generation period can be calculated by the following expression.

$$\text{Generation period of INTTMmn (timer interrupt)} = \text{Period of count clock} \times (\text{Set value of TDRmn} + 1)$$

#### (2) Operation as square wave output

TOmn performs a toggle operation as soon as INTTMmn has been generated, and outputs a square wave with a duty factor of 50%.

The period and frequency for outputting a square wave from TOmn can be calculated by the following expressions.

$$\bullet \text{Period of square wave output from TOmn} = \text{Period of count clock} \times (\text{Set value of TDRmn} + 1) \times 2$$

$$\bullet \text{Frequency of square wave output from TOmn} = \text{Frequency of count clock}/(\text{Set value of TDRmn} + 1) \times 2\}$$

Timer count register mn (TCRmn) operates as a down counter in the interval timer mode.

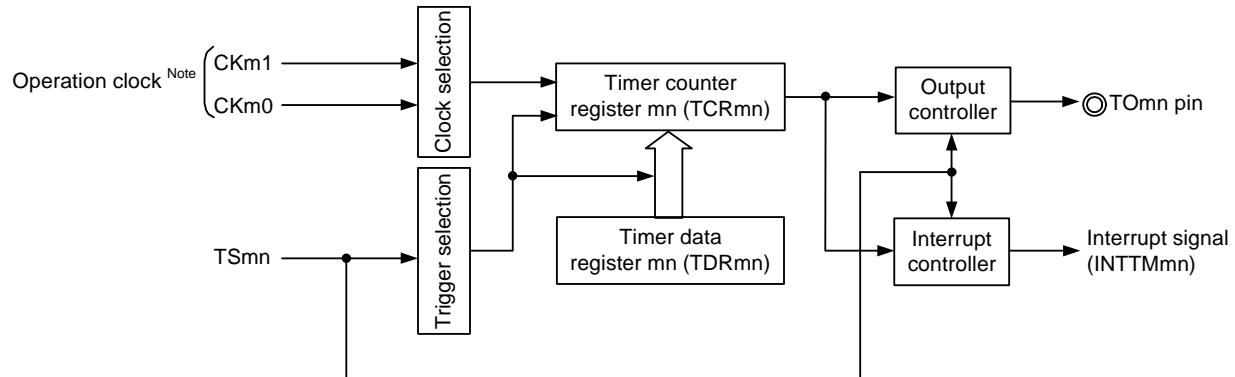
The TCRmn register loads the value of timer data register mn (TDRmn) at the first count clock after the channel start trigger bit (TSmn, TSHm1, TSHm3) of timer channel start register m (TSm) is set to 1. If the MDmn0 bit of timer mode register mn (TMRmn) is 0 at this time, INTTMmn is not output and TOmn is not toggled. If the MDmn0 bit of the TMRmn register is 1, INTTMmn is output and TOmn is toggled.

After that, the TCRmn register count down in synchronization with the count clock.

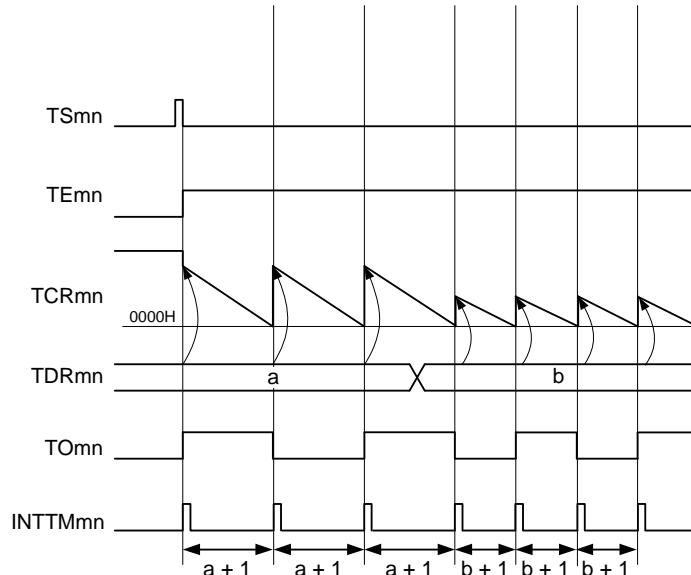
When TCRmn = 0000H, INTTMmn is output and TOmn is toggled at the next count clock. At the same time, the TCRmn register loads the value of the TDRmn register again. After that, the same operation is repeated.

The TDRmn register can be rewritten at any time. The new value of the TDRmn register becomes valid from the next period.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 46 Block Diagram of Operation as Interval Timer/Square Wave Output**

**Note** When channels 1 and 3, the clock can be selected from CKm0, CKm1, CKm2 and CKm3.

**Figure 6 - 47 Example of Basic Timing of Operation as Interval Timer/Square Wave Output (MDmn0 = 1)**

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ )

**Remark 2.** TSmn: Bit n of timer channel start register m (TSm)

TEMmn: Bit n of timer channel enable status register m (TEM)

TCRmn: Timer count register mn (TCRmn)

TDRmn: Timer data register mn (TDRmn)

TOmn: TOmn pin output signal

**Figure 6 - 48 Example of Set Contents of Registers During Operation as Interval Timer/Square Wave Output (1/2)**

## (a) Timer mode register mn (TMRmn)

TMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn1 1/0	CKSmn0 1/0	0	CCSmn 0	M/S Note 0/1	STS mn2 0	STS mn1 0	STS mn0 0	CIS mn1 0	CIS mn0 0	0	0	MD mn3 0	MD mn2 0	MD mn1 0	MD mn0 1/0

Operation mode of channel n  
000B: Interval timer

Setting of operation when counting is started  
0: Neither generates INTTMmn nor inverts timer output when counting is started.  
1: Generates INTTMmn and inverts timer output when counting is started.

Selection of TImn pin input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
000B: Selects only software start.

Setting of MASTERMn bit (channel 2)  
0: Independent channel operation function.

Setting of SPLITmn bit (channels 1, 3)  
0: 16-bit timer mode  
1: 8-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.  
01B: Selects CKm2 as operation clock of channels 1, 3 (This can only be selected using channels 1 and 3 in the 8-bit timer mode).  
11B: Selects CKm3 as operation clock of channels 1, 3 (This can only be selected using channels 1 and 3 in the 8-bit timer mode).

## (b) Timer output register m (TOm)

TOm	Bit n	0: Outputs 0 from TOmn. 1: Outputs 1 from TOmn.
	1/0	

## (c) Timer output enable register m (TOEm)

TOEm	Bit n	0: Stops the TOmn output operation by counting operation. 1: Enables the TOmn output operation by counting operation.
	1/0	

## (d) Timer output level register m (TOLm)

TOLm	Bit n	0: Cleared to 0 when TOMmn = 0 (master channel output mode)
	0	

## (e) Timer output mode register m (TOMm)

TOMm	Bit n	0: Sets master channel output mode.
	0	

**Note**    TMRm2:            MASTERMn bit

TMRm1, TMRm3: SPLITmn bit

TMRm0:            Fixed to 0

**Remark**    m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 49 Operation Procedure of Interval Timer/Square Wave Output Function (1/2)**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets timer mode register mn (TMRmn) (determines operation mode of channel).</p> <p>Sets interval (period) value to timer data register mn (TDRmn).</p> <p>To use the TOmn output</p> <ul style="list-style-type: none"> <li>Clears the TOMmn bit of timer output mode register m (TOMm) to 0 (master channel output mode).</li> <li>Clears the TOLmn bit to 0.</li> <li>Sets the TOmn bit and determines default level of the TOmn output.</li> </ul> <p>Sets the TOEmn bit to 1 and enables operation of TOmn.</p> <p>Clears the port register and port mode register to 0.</p>	<p>Channel stops operating. (Clock is supplied and some power is consumed.)</p> <p>The TOmn pin goes into Hi-Z output state.</p> <p>The TOmn default setting level is output when the port mode register is in the output mode and the port register is 0.</p> <p>TOmn does not change because channel stops operating.</p> <p>The TOmn pin outputs the TOmn set level.</p>
Operation start	<p>(Sets the TOEmn bit to 1 only if using TOmn output and resuming operation.).</p> <p>Sets the TSmn (TSHm1, TSHm3) bit to 1. The TSmn (TSHm1, TSHm3) bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn (TEHm1, TEHm3) = 1, and count operation starts.</p> <p>Value of the TDRmn register is loaded to timer count register mn (TCRmn). INTTMmn is generated and TOmn performs toggle operation if the MDmn0 bit of the TMRmn register is 1.</p>
During operation	<p>Set value of the TDRmn register can be changed.</p> <p>The TCRmn register can always be read.</p> <p>The TSRmn register is not used.</p> <p>Set values of the TOm and TOEm registers can be changed.</p> <p>Set values of the TMRmn register, TOMmn, and TOLmn bits cannot be changed.</p>	<p>Counter (TCRmn) counts down. When count value reaches 0000H, the value of the TDRmn register is loaded to the TCRmn register again and the count operation is continued. By detecting TCRmn = 0000H, INTTMmn is generated and TOmn performs toggle operation.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn (TTHm1, TTHm3) bit is set to 1. The TTmn (TTHm1, TTHm3) bit automatically returns to 0 because it is a trigger bit.</p> <p>The TOEmn bit is cleared to 0 and value is set to the TOmn bit.</p>	<p>TEmn (TEHm1, TEHm3), and count operation stops.</p> <p>The TCRmn register holds count value and stops.</p> <p>The TOmn output is not initialized but holds current status.</p> <p>The TOmn pin outputs the TOmn bit set level.</p>

(Remark is listed on the next page.)

**Figure 6 - 50 Operation Procedure of Interval Timer/Square Wave Output Function (2/2)**

	Software Operation	Hardware Status
TAU stop	<p>To hold the TOmn pin output level Clears the TOmn bit to 0 after the value to be held is set to the port register.</p> <p>When holding the TOmn pin output level is not necessary Setting not required.</p> <p>The TAUmEN bit of the PER0 register is cleared to 0.</p>	<p>The TOmn pin output level is held by port function.</p> <p>Input clock supply for timer array unit m is stopped All circuits are initialized and SFR of each channel is also initialized. (The TOmn bit is cleared to 0 and the TOmn pin is set to port mode.)</p>

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

### 6.8.2 Operation as external event counter

The timer array unit can be used as an external event counter that counts the number of times the valid input edge (external event) is detected in the TImn pin. When a specified count value is reached, the event counter generates an interrupt. The specified number of counts can be calculated by the following expression.

$$\text{Specified number of counts} = \text{Set value of TDRmn} + 1$$

Timer count register mn (TCRmn) operates as a down counter in the event counter mode.

The TCRmn register loads the value of timer data register mn (TDRmn) by setting any channel start trigger bit (TSmn, TSHm1, TSHm3) of timer channel start register m (TSm) to 1.

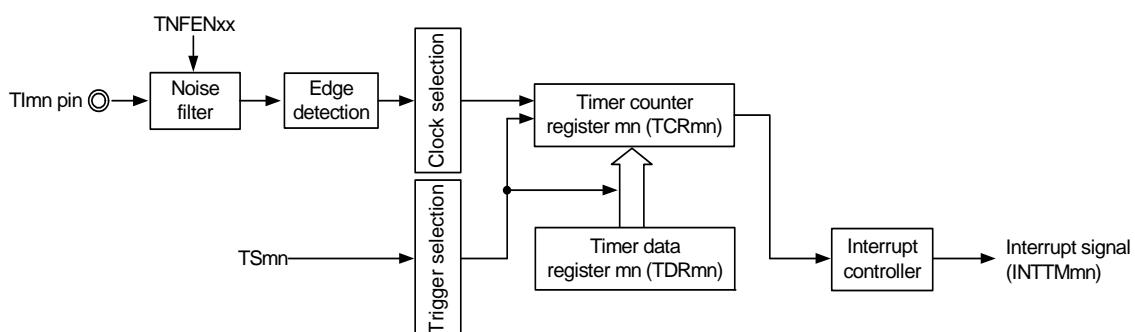
The TCRmn register counts down each time the valid input edge of the TImn pin has been detected. When TCRmn = 0000H, the TCRmn register loads the value of the TDRmn register again, and outputs INTTMmn.

After that, the above operation is repeated.

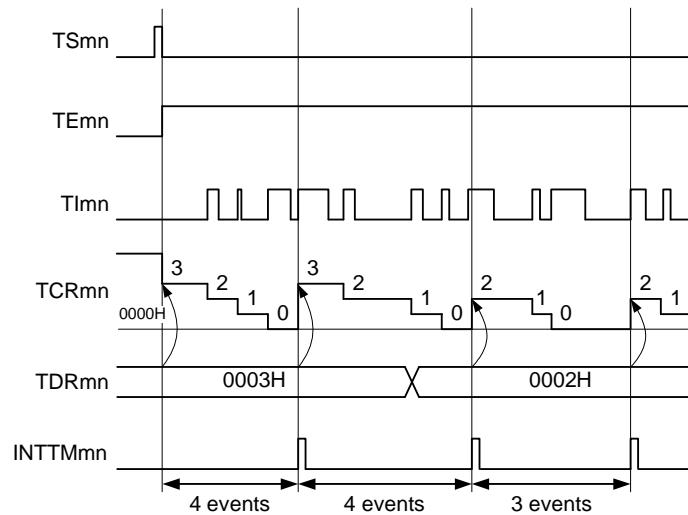
An irregular waveform that depends on external events is output from the TOmn pin. Stop the output by setting the TOEmn bit of timer output enable register m (TOEm) to 0.

The TDRmn register can be rewritten at any time. The new value of the TDRmn register becomes valid during the next count period.

**Figure 6 - 51 Block Diagram of Operation as External Event Counter**



**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

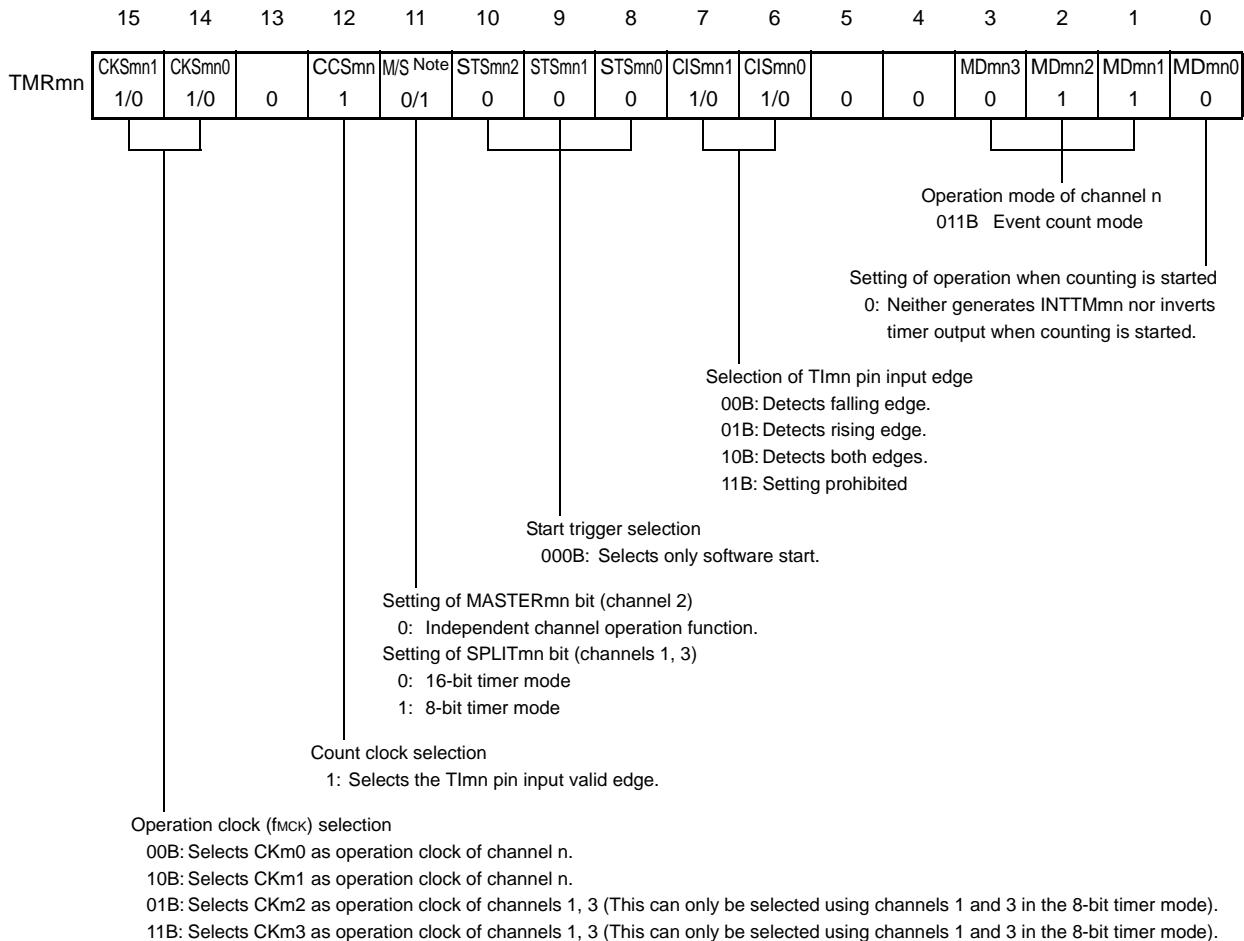
**Figure 6 - 52 Example of Basic Timing of Operation as External Event Counter**

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ )

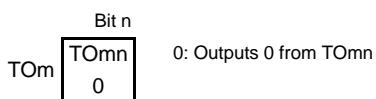
**Remark 2.** TSmn: Bit n of timer channel start register m (TSm)  
 TEmn: Bit n of timer channel enable status register m (TEm)  
 TImn: TImn pin input signal  
 TCRmn: Timer count register mn (TCRmn)  
 TDRmn: Timer data register mn (TDRmn)

**Figure 6 - 53 Example of Set Contents of Registers in External Event Counter Mode (1/2)**

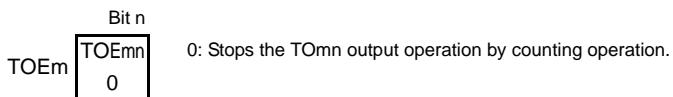
## (a) Timer mode register mn (TMRmn)



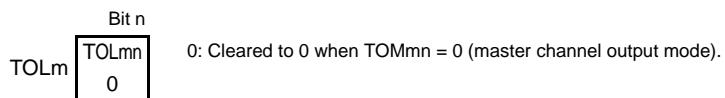
## (b) Timer output register m (TOm)



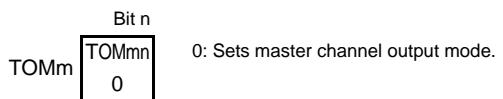
## (c) Timer output enable register m (TOEm)



## (d) Timer output level register m (TOLm)



## (e) Timer output mode register m (TOMm)



**Note**

TMRm2:	MASTERmn bit
TMRm1, TMRm3	SPLITmn bit
TMRm0:	Fixed to 0

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 54 Operation Procedure When External Event Counter Function Is Used**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on).</p> <p>Sets timer mode register mn (TMRmn) (determines operation mode of channel).</p> <p>Sets number of counts to timer data register mn (TDRmn).</p> <p>Clears the TOEmn bit of timer output enable register m (TOEm) to 0.</p>	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<p>Sets the TSmn bit to 1.</p> <p>The TSmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 1, and count operation starts.</p> <p>Value of the TDRmn register is loaded to timer count register mn (TCRmn) and detection of the TI<sub>mn</sub> pin input edge is awaited.</p>
During operation	<p>Set value of the TDRmn register can be changed.</p> <p>The TCRmn register can always be read.</p> <p>The TSRmn register is not used.</p> <p>Set values of the TMRmn register, TOMmn, TOLmn, TOmn, and TOEmn bits cannot be changed.</p>	<p>Counter (TCRmn) counts down each time input edge of the TI<sub>mn</sub> pin has been detected. When count value reaches 0000H, the value of the TDRmn register is loaded to the TCRmn register again, and the count operation is continued. By detecting TCRmn = 0000H, the INTTMmn output is generated.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn bit is set to 1.</p> <p>The TTmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 0, and count operation stops.</p> <p>The TCRmn register holds count value and stops.</p>
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0.	<p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p>

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

### 6.8.3 Operation as frequency divider (channel 0 of unit 0 only)

The timer array unit can be used as a frequency divider that divides a clock input to the TI00 pin and outputs the result from the TO00 pin.

The divided clock frequency output from TO00 can be calculated by the following expression.

- When rising edge/falling edge is selected:

$$\text{Divided clock frequency} = \text{Input clock frequency}/\{(\text{Set value of TDR00} + 1) \times 2\}$$

- When both edges are selected:

$$\text{Divided clock frequency} \approx \text{Input clock frequency}/(\text{Set value of TDR00} + 1)$$

Timer count register 00 (TCR00) operates as a down counter in the interval timer mode.

After the channel start trigger bit (TS00) of timer channel start register 0 (TS0) is set to 1, the TCR00 register loads the value of timer data register 00 (TDR00) when the TI00 valid edge is detected.

If the MD000 bit of timer mode register 00 (TMR00) is 0 at this time, INTTM00 is not output and TO00 is not toggled. If the MD000 bit of timer mode register 00 (TMR00) is 1, INTTM00 is output and TO00 is toggled.

After that, the TCR00 register counts down at the valid edge of the TI00 pin. When TCR00 = 0000H, it toggles TO00. At the same time, the TCR00 register loads the value of the TDR00 register again, and continues counting.

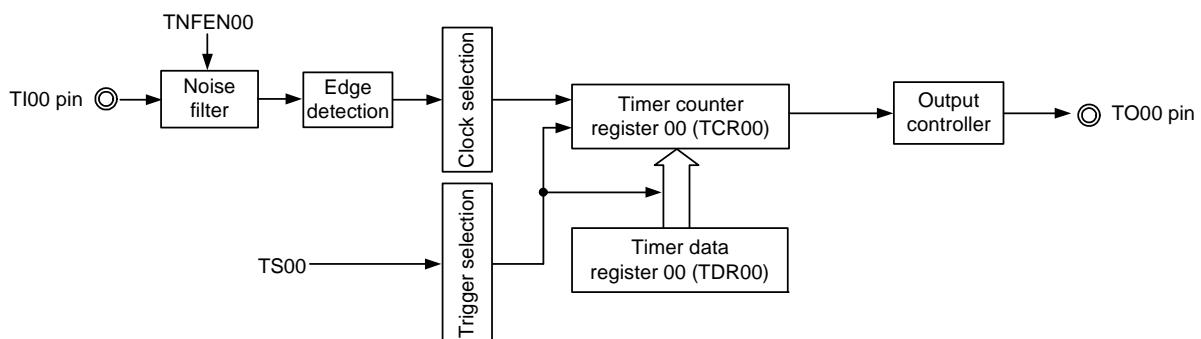
If detection of both the edges of the TI00 pin is selected, the duty factor error of the input clock affects the divided clock period of the TO00 output.

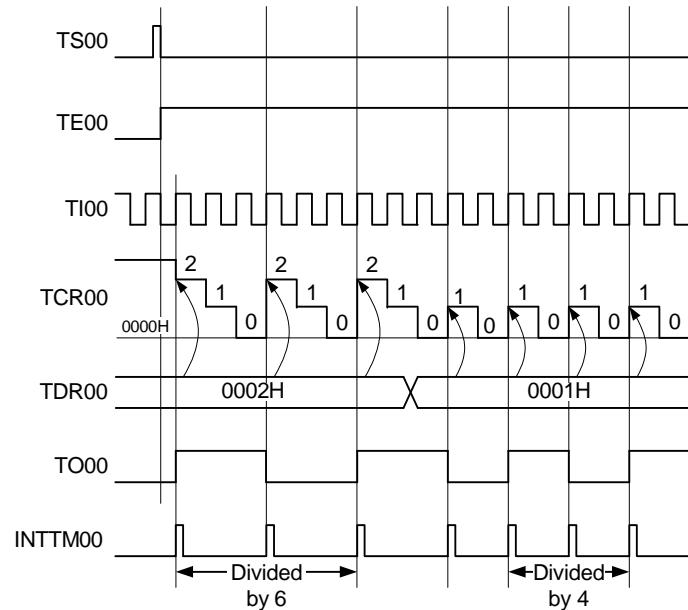
The period of the TO00 output clock includes a sampling error of one period of the operation clock.

$$\text{Clock period of TO00 output} = \text{Ideal TO00 output clock period} \pm \text{Operation clock period (error)}$$

The TDR00 register can be rewritten at any time. The new value of the TDR00 register becomes valid during the next count period.

**Figure 6 - 55 Block Diagram of Operation as Frequency Divider**



**Figure 6 - 56 Example of Basic Timing of Operation as Frequency Divider (MD000 = 1)**

**Remark**

- TS00: Bit n of timer channel start register 0 (TS0)
- TE00: Bit n of timer channel enable status register 0 (TE0)
- TI00: TI00 pin input signal
- TCR00: Timer count register 00 (TCR00)
- TDR00: Timer data register 00 (TDR00)
- TO00: TO00 pin output signal

**Figure 6 - 57 Example of Set Contents of Registers During Operation as Frequency Divider**

## (a) Timer mode register 00 (TMR00)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRm0	CKSmn1 1/0	CKSmn0 0	0	CCS00 1	0	STS002 0	STS001 0	STS000 0	CIS001 1/0	CIS000 1/0	0	0	MD003 0	MD002 0	MD001 0	MD000 1/0

Operation mode of channel 0  
000B: Interval timer

Setting of operation when counting is started  
0: Neither generates INTTM00 nor inverts timer output when counting is started.  
1: Generates INTTM00 and inverts timer output when counting is started.

Selection of TI00 pin input edge  
00B: Detects falling edge.  
01B: Detects rising edge.  
10B: Detects both edges.  
11B: Setting prohibited

Start trigger selection  
000B: Selects only software start.

Count clock selection  
1: Selects the TI00 pin input valid edge.

Operation clock (fmck) selection  
00B: Selects CK00 as operation clock of channel 0.  
10B: Selects CK01 as operation clock of channel 0.

## (b) Timer output register 0 (TO0)

Bit 0	TO0n 0	0: Outputs 0 from TO00. 1: Outputs 1 from TO00.
-------	-----------	--

## (c) Timer output enable register 0 (TOE0)

Bit n	TOEmn 0	0: Stops the TO00 output operation by counting operation. 1: Enables the TO00 output operation by counting operation.
-------	------------	--

## (d) Timer output level register 0 (TOL0)

Bit 0	TOLmn 0	0: Cleared to 0 when master channel output mode (TOM00 = 0)
-------	------------	---

## (e) Timer output mode register 0 (TOM0)

Bit 0	TOMmn 0	0: Sets master channel output mode.
-------	------------	-------------------------------------

**Figure 6 - 58 Operation Procedure When Frequency Divider Function Is Used**

	Software Operation	Hardware Status
TAU default setting	Sets the TAU0EN bit of peripheral enable register 0 (PER0) to 1. Sets timer clock select register 0 (TPS0). Determines clock frequencies of CKm0 to CKm3.	Input clock supply for timer array unit 0 is stopped (Clock supply is stopped and writing to each register is disabled.) Input clock supply for timer array unit 0 is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)
Channel default setting	Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on). Sets timer mode register 00 (TMR00) (determines operation mode of channel and selects the detection edge). Sets interval (period) value to timer data register 00 (TDR00). Clears the TOM00 bit of timer output mode register 0 (TOM0) to 0 (master channel output mode). Clears the TOL00 bit to 0. Sets the TO00 bit and determines default level of the TO00 output. Sets the TOE00 bit to 1 and enables operation of TO00. Clears the port register and port mode register to 0.	Channel stops operating. (Clock is supplied and some power is consumed.) The TO00 pin goes into Hi-Z output state. The TO00 default setting level is output when the port mode register is in output mode and the port register is 0. TO00 does not change because channel stops operating. The TO00 pin outputs the TO00 set level.
Operation start	Sets the TOE00 bit to 1 (only when operation is resumed). Sets the TS00 bit to 1. The TS00 bit automatically returns to 0 because it is a trigger bit.	TE00 = 1, and count operation starts. Value of the TDR00 register is loaded to timer count register 00 (TCR00). INTTM00 is generated and TO00 performs toggle operation if the MD000 bit of the TMR00 register is 1.
During operation	Set value of the TDR00 register can be changed. The TCR00 register can always be read. The TSR00 register is not used. Set values of the TO0 and TOE0 registers can be changed. Set values of the TMR00 register, TOM00, and TOL00 bits cannot be changed.	Counter (TCR00) counts down. When count value reaches 0000H, the value of the TDR00 register is loaded to the TCR00 register again, and the count operation is continued. By detecting TCR00 = 0000H, INTTM00 is generated and TO00 performs toggle operation. After that, the above operation is repeated.
Operation stop	The TT00 bit is set to 1. The TT00 bit automatically returns to 0 because it is a trigger bit. The TOE00 bit is cleared to 0 and value is set to the TO00 bit.	TE00 = 0, and count operation stops. The TCR00 register holds count value and stops. The TO00 output is not initialized but holds current status. The TO00 pin outputs the TO00 set level.
TAU stop	To hold the TO00 pin output level Clears the TO00 bit to 0 after the value to be held is set to the port register. When holding the TO00 pin output level is not necessary Setting not required. The TAU0EN bit of the PER0 register is cleared to 0.	The TO00 pin output level is held by port function. Input clock supply for timer array unit 0 is stopped All circuits are initialized and SFR of each channel is also initialized. (The TO00 bit is cleared to 0 and the TO00 pin is set to port mode).

### 6.8.4 Operation as input pulse interval measurement

The count value can be captured at the TImn valid edge and the interval of the pulse input to TImn can be measured. In addition, the count value can be captured by using software operation (TSmn = 1) as a capture trigger while the TEMn bit is set to 1.

The pulse interval can be calculated by the following expression.

$$\text{TImn input pulse interval} = \text{Period of count clock} \times ((10000H \times \text{TSRmn}: \text{OVF}) + (\text{Capture value of TDRmn} + 1))$$

**Caution** The TImn pin input is sampled using the operating clock selected with the CKSmn bit of timer mode register mn (TMRmn), so an error of up to one operating clock cycle occurs.

Timer count register mn (TCRmn) operates as an up counter in the capture mode.

When the channel start trigger bit (TSmn) of timer channel start register m (TSm) is set to 1, the TCRmn register counts up from 0000H in synchronization with the count clock.

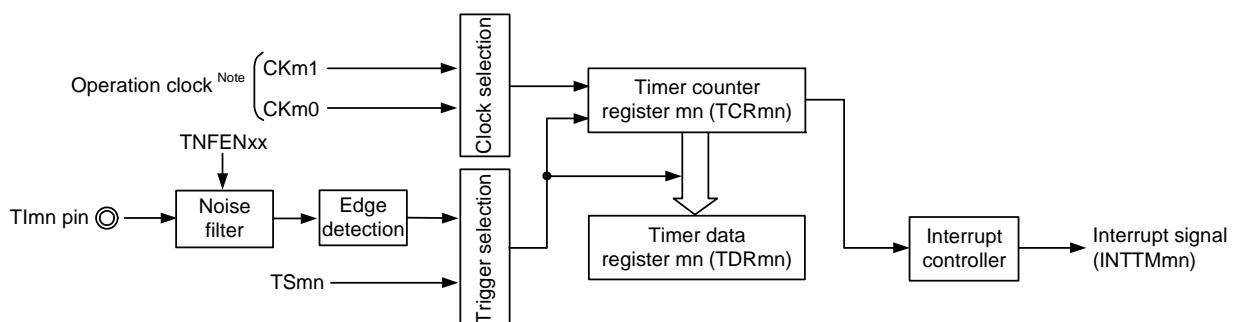
When the TImn pin input valid edge is detected, the count value of the TCRmn register is transferred (captured) to timer data register mn (TDRmn) and, at the same time, the TCRmn register is cleared to 0000H, and the INTTMmn is output. If the counter overflows at this time, the OVF bit of timer status register mn (TSRmn) is set to 1. If the counter does not overflow, the OVF bit is cleared. After that, the above operation is repeated.

As soon as the count value has been captured to the TDRmn register, the OVF bit of the TSRmn register is updated depending on whether the counter overflows during the measurement period. Therefore, the overflow status of the captured value can be checked.

If the counter reaches a full count for two or more periods, it is judged to be an overflow occurrence, and the OVF bit of the TSRmn register is set to 1. However, a normal interval value cannot be measured for the OVF bit, if two or more overflows occur.

Set the STSmn2 to STSmn0 bits of the TMRmn register to 001B to use the valid edges of TImn as a start trigger and a capture trigger.

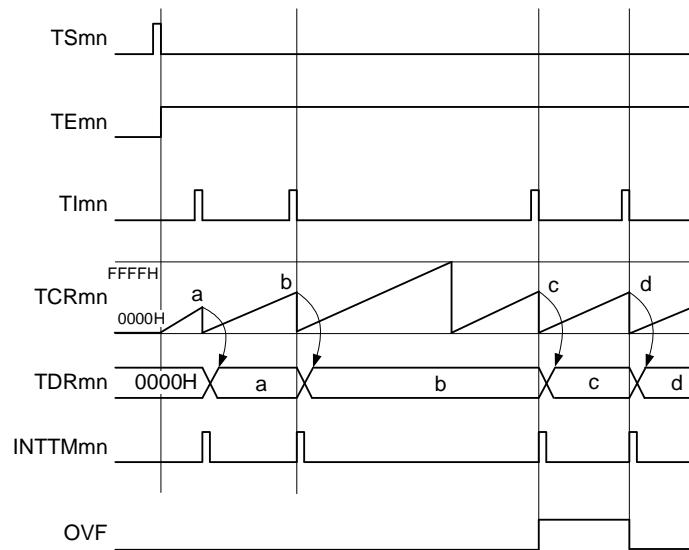
**Figure 6 - 59 Block Diagram of Operation as Input Pulse Interval Measurement**



**Note** When channels 1 and 3, the clock can be selected from CKm0, CKm1, CKm2 and CKm3.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 60 Example of Basic Timing of Operation as Input Pulse Interval Measurement (MDmn0 = 0)**



**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

**Remark 2.** TSmn: Bit n of timer channel start register m (TSm)  
 TEmn: Bit n of timer channel enable status register m (TEM)  
 TImn: TImn pin input signal  
 TCRmn: Timer count register mn (TCRmn)  
 TDRmn: Timer data register mn (TDRmn)  
 OVF: Bit 0 of timer status register mn (TSRmn)

**Figure 6 - 61 Example of Set Contents of Registers to Measure Input Pulse Interval**

## (a) Timer mode register mn (TMRmn)

TMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn1 1/0	CKSmn0 0	0	CCSmn 0	M/S Note 0	STS mn2 0	STS mn1 0	STS mn0 1	CIS mn1 1/0	CIS mn0 1/0	0	0	MD mn3 0	MD mn2 1	MD mn1 0	MD mn0 1/0

Operation mode of channel n  
010B: Capture mode

Setting of operation when counting is started  
0: Does not generate INTTMmn when counting is started.  
1: Generates INTTMmn when counting is started.

Selection of Tlmn pin input edge  
00B: Detects falling edge.  
01B: Detects rising edge.  
10B: Detects both edges.  
11B: Setting prohibited

Capture trigger selection  
001B: Selects the Tlmn pin input valid edge.

Setting of MASTERMn bit (channel 2)  
0: Independent channel operation function.

Setting of SPLITmn bit (channels 1, 3)  
0: 16-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.  
01B: Selects CKm2 as operation clock of channels 1, 3 (This can only be selected channels 1 and 3).  
11B: Selects CKm3 as operation clock of channels 1, 3 (This can only be selected channels 1 and 3).

## (b) Timer output register m (TOm)

TOm	Bit n <input type="checkbox"/> 0	0: Outputs 0 from TOmn.
-----	-------------------------------------	-------------------------

## (c) Timer output enable register m (TOEm)

TOEm	Bit n <input type="checkbox"/> 0	0: Stops TOmn output operation by counting operation.
------	-------------------------------------	---

## (d) Timer output level register m (TOLm)

TOLm	Bit n <input type="checkbox"/> 0	0: Cleared to 0 when master channel output mode (TOMmn = 0)
------	-------------------------------------	---

## (e) Timer output mode register m (TOMm)

TOMm	Bit n <input type="checkbox"/> 0	0: Sets master channel output mode.
------	-------------------------------------	-------------------------------------

**Note**    TMRm2:            MASTERMn bit

TMRm1, TMRm3: SPLITmn bit

TMRm0:            Fixed to 0

**Remark**    m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 62 Operation Procedure When Input Pulse Interval Measurement Function Is Used**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on). Sets timer mode register mn (TMRmn) (determines operation mode of channel).	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<p>Sets TSmn bit to 1.</p> <p>The TSmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 1, and count operation starts.</p> <p>Timer count register mn (TCRmn) is cleared to 0000H.</p> <p>When the MDmn0 bit of the TMRmn register is 1, INTTMmn is generated.</p>
During operation	<p>Set values of only the CISmn1 and CISmn0 bits of the TMRmn register can be changed.</p> <p>The TDRmn register can always be read.</p> <p>The TCRmn register can always be read.</p> <p>The TSRmn register can always be read.</p> <p>Set values of the TOMmn, TOLmn, TOmn, and TOEmn bits cannot be changed.</p>	<p>Counter (TCRmn) counts up from 0000H. When the valid edge of the TI<sub>mn</sub> pin input is detected or the TS<sub>mn</sub> bit is set to 1, the count value is transferred (captured) to timer data register mn (TDRmn). At the same time, the TCRmn register is cleared to 0000H, and the INTTMmn signal is generated.</p> <p>If an overflow occurs at this time, the OVF bit of timer status register mn (TSRmn) is set; if an overflow does not occur, the OVF bit is cleared.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn bit is set to 1.</p> <p>The TTmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 0, and count operation stops.</p> <p>The TCRmn register holds count value and stops.</p> <p>The OVF bit of the TSRmn register is also held.</p>
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0.	<p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p>

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

### 6.8.5 Operation as input signal high-/low-level width measurement

**Caution** When using a channel to implement the LIN-bus, set bit 1 (ISC1) of the input switch control register (ISC) to 1. In the following descriptions, read TImn as RxDO.

By starting counting at one edge of the TImn pin input and capturing the number of counts at another edge, the signal width (high-level width/low-level width) of TImn can be measured. The signal width of TImn can be calculated by the following expression.

Signal width of TImn input = Period of count clock × ((10000H × TSRmn: OVF) + (Capture value of TDRmn + 1))

**Caution** The TImn pin input is sampled using the operating clock selected with the CKSmn bit of timer mode register mn (TMRmn), so an error equivalent to one operation clock occurs.

Timer count register mn (TCRmn) operates as an up counter in the capture & one-count mode.

When the channel start trigger bit (TSmn) of timer channel start register m (TSm) is set to 1, the TEmn bit is set to 1 and the TImn pin start edge detection wait status is set.

When the TImn pin input start edge (rising edge of the TImn pin input when the high-level width is to be measured) is detected, the counter counts up from 0000H in synchronization with the count clock. When the valid capture edge (falling edge of the TImn pin input when the high-level width is to be measured) is detected later, the count value is transferred to timer data register mn (TDRmn) and, at the same time, INTTMmn is output. If the counter overflows at this time, the OVF bit of timer status register mn (TSRmn) is set to 1. If the counter does not overflow, the OVF bit is cleared. The TCRmn register stops at the value “value transferred to the TDRmn register + 1”, and the TImn pin start edge detection wait status is set. After that, the above operation is repeated.

As soon as the count value has been captured to the TDRmn register, the OVF bit of the TSRmn register is updated depending on whether the counter overflows during the measurement period. Therefore, the overflow status of the captured value can be checked.

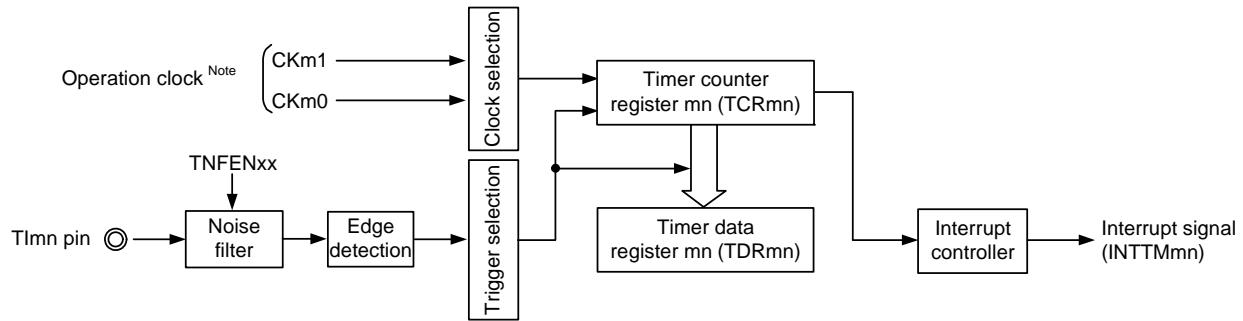
If the counter reaches a full count for two or more periods, it is judged to be an overflow occurrence, and the OVF bit of the TSRmn register is set to 1. However, a normal interval value cannot be measured for the OVF bit, if two or more overflows occur.

Whether the high-level width or low-level width of the TImn pin is to be measured can be selected by using the CISmn1 and CISmn0 bits of the TMRmn register.

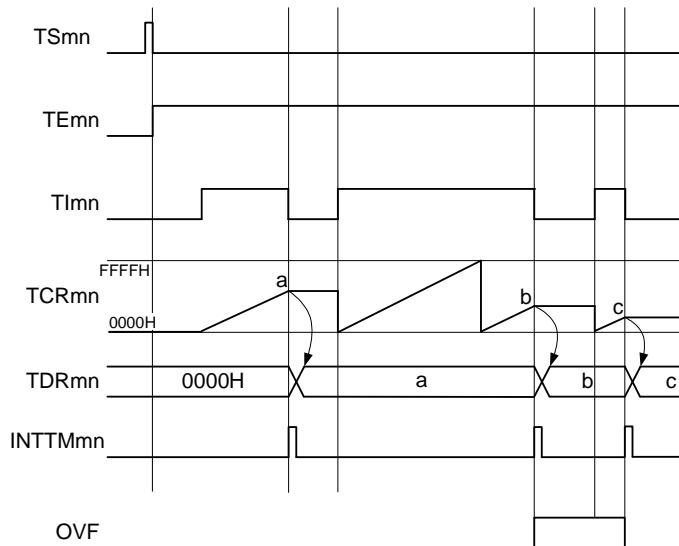
Because this function is used to measure the signal width of the TImn pin input, the TSmn bit cannot be set to 1 while the TEmn bit is 1.

CISmn1, CISmn0 of TMRmn register = 10B: Low-level width is measured.

CISmn1, CISmn0 of TMRmn register = 11B: High-level width is measured.

**Figure 6 - 63 Block Diagram of Operation as Input Signal High-/Low-Level Width Measurement**

**Note** For channels 1 and 3, the clock can be selected from CKm0, CKm1, CKm2 and CKm3.

**Figure 6 - 64 Example of Basic Timing of Operation as Input Signal High-/Low-Level Width Measurement**

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3)

**Remark 2.** TSmn: Bit n of timer channel start register m (TSm)

TEMmn: Bit n of timer channel enable status register m (TEM)

TImn: TImn pin input signal

TCRmn: Timer count register mn (TCRmn)

TDRmn: Timer data register mn (TDRmn)

OVF: Bit 0 of timer status register mn (TSRmn)

**Figure 6 - 65 Example of Set Contents of Registers to Measure Input Signal High-/Low-Level Width**

## (a) Timer mode register mn (TMRmn)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn	CKSmn1 1/0	CKSmn0 0	0	CCSmn 0	M/S Note 0	STSmn2 0	STSmn1 1	STSmn0 0	CISmn1 1	CISmn0 1/0	0	0	MDmn3 1	MDmn2 1	MDmn1 0	MDmn0 0

Operation mode of channel n  
110B: Capture & one-count

Setting of operation when counting is started  
0: Does not generate INTTMmn when counting is started.

Selection of TImn pin input edge  
10B: Both edges (to measure low-level width)  
11B: Both edges (to measure high-level width)

Start trigger selection  
010B: Selects the TImn pin input valid edge.

Setting of MASTERmn bit (channel 2)  
0: Independent channel operation function.

Setting of SPLITmn bit (channels 1, 3)  
0: 16-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.  
01B: Selects CKm2 as operation clock of channels 1, 3 (This can only be selected channels 1 and 3).  
11B: Selects CKm3 as operation clock of channels 1, 3 (This can only be selected channels 1 and 3).

## (b) Timer output register m (TOm)

TOm	Bit n TOmn 0	0: Outputs 0 from TOmn.
-----	--------------------	-------------------------

## (c) Timer output enable register m (TOEm)

TOEm	Bit n TOEmn 0	0: Stops the TOmn output operation by counting operation.
------	---------------------	---

## (d) Timer output level register m (TOLm)

TOLm	Bit n TOLmn 0	0: Cleared to 0 when master channel output mode (TOMmn = 0)
------	---------------------	---

## (e) Timer output mode register m (TOMm)

TOMm	Bit n TOMmn 0	0: Sets master channel output mode.
------	---------------------	-------------------------------------

**Note**      TMRm2:           MASTERmn bit  
              TMRm1, TMRm3: SPLITmn bit  
              TMRm0:          Fixed to 0

**Remark**    m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 66 Operation Procedure When Input Signal High-/Low-Level Width Measurement Function Is Used**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on).</p> <p>Sets timer mode register mn (TMRmn) (determines operation mode of channel).</p> <p>Clears the TOEmn bit to 0 and stops operation of TOMn.</p>	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<p>Sets the TSmn bit to 1. The TSmn bit automatically returns to 0 because it is a trigger bit.</p> <p>Detects the TImn pin input count start valid edge.</p>	<p>TEmn = 1, and the TImn pin start edge detection wait status is set.</p> <p>Clears timer count register mn (TCRmn) to 0000H and starts counting up.</p>
During operation	<p>Set value of the TDRmn register can always be read.</p> <p>The TCRmn register can always be read.</p> <p>The TSRmn register can always be read.</p> <p>Set values of the TMRmn register, TOMmn, TOLmn, TOMn, and TOEmn bits cannot be changed.</p>	<p>When the TImn pin start edge is detected, the counter (TCRmn) counts up from 0000H. If a capture edge of the TImn pin is detected, the count value is transferred to timer data register mn (TDRmn) and INTTMmn is generated.</p> <p>If an overflow occurs at this time, the OVF bit of timer status register mn (TSRmn) is set; if an overflow does not occur, the OVF bit is cleared. The TCRmn register stops the count operation until the next TImn pin start edge is detected.</p>
Operation stop	<p>The TTmn bit is set to 1. The TTmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 0, and count operation stops.</p> <p>The TCRmn register holds count value and stops.</p> <p>The OVF bit of the TSRmn register is also held.</p>
TAU stop	The TAUmEN bit of the PER0 register is cleared to 0.	<p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p>

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 6.8.6 Operation as delay counter

It is possible to start counting down when the valid edge of the TImn pin input is detected (an external event), and then generate INTTMmn (a timer interrupt) after any specified interval.

It can also generate INTTMmn (timer interrupt) at any interval by making a software set TSmn = 1 and the count down start during the period of TEmn = 1.

The interrupt generation period can be calculated by the following expression.

$$\text{Generation period of INTTMmn (timer interrupt)} = \text{Period of count clock} \times (\text{Set value of TDRmn} + 1)$$

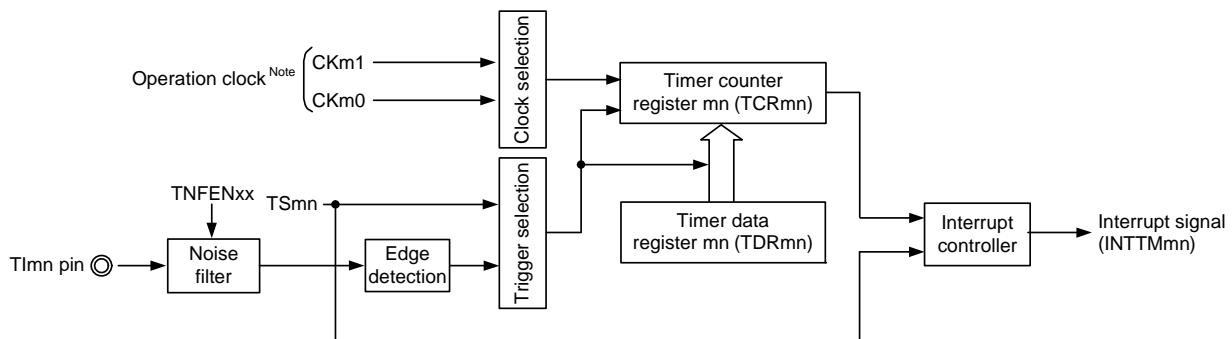
Timer count register mn (TCRmn) operates as a down counter in the one-count mode.

When the channel start trigger bit (TSmn, TSHm1, TSHm3) of timer channel start register m (TSm) is set to 1, the TEmn, TEHm1, TEHm3 bits are set to 1 and the TImn pin input valid edge detection wait status is set.

Timer count register mn (TCRmn) starts operating upon TImn pin input valid edge detection and loads the value of timer data register mn (TDRmn). The TCRmn register counts down from the value of the TDRmn register it has loaded, in synchronization with the count clock. When TCRmn = 0000H, it outputs INTTMmn and stops counting until the next TImn pin input valid edge is detected.

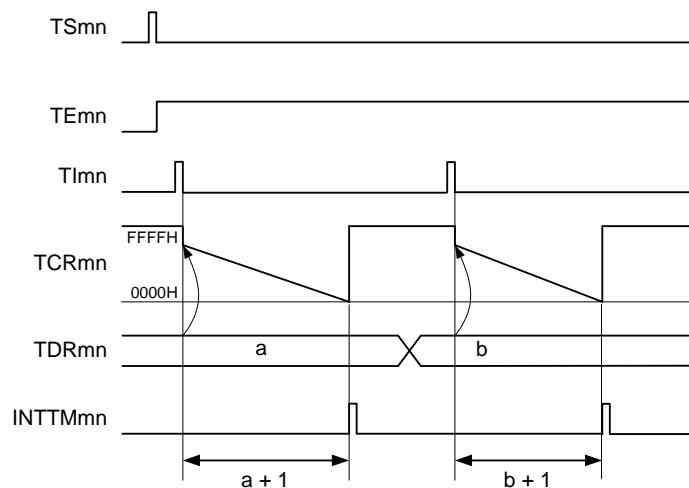
The TDRmn register can be rewritten at any time. The new value of the TDRmn register becomes valid from the next period.

**Figure 6 - 67 Block Diagram of Operation as Delay Counter**



**Note** For using channels 1 and 3, the clock can be selected from CKM0, CKM1, CKM2 and CKM3.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 68 Example of Basic Timing of Operation as Delay Counter**

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ )

**Remark 2.**   
**TSmn:** Bit n of timer channel start register m (TSm)  
**TEmn:** Bit n of timer channel enable status register m (TEM)  
**TImn:** TImn pin input signal  
**TCRmn:** Timer count register mn (TCRmn)  
**TDRmn:** Timer data register mn (TDRmn)

**Figure 6 - 69 Example of Set Contents of Registers to Delay Counter**

## (a) Timer mode register mn (TMRmn)

TMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn1 1/0	CKSmn0 1/0	0	CCSmn 0	M/S Note 0/1	STS mn2 0	STS mn1 0	STS mn0 1	CIS mn1 1/0	CIS mn0 1/0	0	0	MD mn3 1	MD mn2 0	MD mn1 0	MD mn0 1/0

Operation mode of channel n  
100B: One-count mode

Start trigger during operation  
0: Trigger input is invalid.  
1: Trigger input is valid.

Selection of Tlmn pin input edge  
00B: Detects falling edge.  
01B: Detects rising edge.  
10B: Detects both edges.  
11B: Setting prohibited

Start trigger selection  
001B: Selects the Tlmn pin input valid edge.

Setting of MASTERmn bit (channel 2)  
0: Independent channel operation function.

Setting of SPLITmn bit (channels 1, 3)  
0: 16-bit timer mode  
1: 8-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.  
01B: Selects CKm2 as operation clock of channels 1, 3 (This can only be selected using channels 1 and 3 in the 8-bit timer mode).  
11B: Selects CKm3 as operation clock of channels 1, 3 (This can only be selected using channels 1 and 3 in the 8-bit timer mode).

## (b) Timer output register m (TOm)

TOm	Bit n <input type="checkbox"/> 0	0: Outputs 0 from TOmn.
-----	-------------------------------------	-------------------------

## (c) Timer output enable register m (TOEm)

TOEm	Bit n <input type="checkbox"/> 0	0: Stops the TOmn output operation by counting operation.
------	-------------------------------------	---

## (d) Timer output level register m (TOLm)

TOLm	Bit n <input type="checkbox"/> 0	0: Cleared to 0 when master channel output mode (TOMmn = 0)
------	-------------------------------------	---

## (e) Timer output mode register m (TOMMm)

TOMMm	Bit n <input type="checkbox"/> 0	0: Sets master channel output mode.
-------	-------------------------------------	-------------------------------------

## (f)

**Note**

TMRm2:	MASTERmn bit
TMRm1, TMRm3:	SPLITmn bit
TMRm0:	Fixed to 0

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 6 - 70 Operation Procedure When Delay Counter Function Is Used**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 to CKm3.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 0 (off) or 1 (on).</p> <p>Sets timer mode register mn (TMRmn) (determines operation mode of channel).</p> <p>INTTMmn output delay is set to timer data register mn (TDRmn).</p> <p>Clears the TOEmn bit to 0 and stops operation of TOmn.</p>	Channel stops operating. (Clock is supplied and some power is consumed.)
Operation start	<p>Sets the TSmn bit to 1.</p> <p>The TSmn bit automatically returns to 0 because it is a trigger bit.</p> <p>The counter starts counting down by the next start trigger detection.</p> <ul style="list-style-type: none"> <li>• Detects the TImn pin input valid edge.</li> <li>• Sets the TSmn bit to 1 by the software.</li> </ul>	<p>TEmn = 1, and the start trigger detection (the valid edge of the TImn pin input is detected or the TSmn bit is set to 1) wait status is set.</p> <p>Value of the TDRmn register is loaded to the timer count register mn (TCRmn).</p>
During operation	<p>Set value of the TDRmn register can be changed.</p> <p>The TCRmn register can always be read.</p> <p>The TSRmn register is not used.</p>	The counter (TCRmn) counts down. When the count value of TCRmn reaches 0000H, the INTTMmn output is generated, and the count operation stops until the next start trigger detection (the valid edge of the TImn pin input is detected or the TSmn bit is set to 1).
Operation stop	<p>The TTmn bit is set to 1.</p> <p>The TTmn bit automatically returns to 0 because it is a trigger bit.</p>	<p>TEmn = 0, and count operation stops.</p> <p>The TCRmn register holds count value and stops.</p>
TAU stop	<p>The TAUmEN bit of the PER0 register is cleared to 0.</p>	<p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p>

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

## 6.9 Simultaneous Channel Operation Function of Timer Array Unit

### 6.9.1 Operation as one-shot pulse output function

By using two channels as a set, a one-shot pulse having any delay pulse width can be generated from the signal input to the TImn pin.

The delay time and pulse width can be calculated by the following expressions.

$$\text{Delay time} = \{\text{Set value of TDRmn (master)} + 2\} \times \text{Count clock period}$$

$$\text{Pulse width} = \{\text{Set value of TDRmp (slave)}\} \times \text{Count clock period}$$

The master channel operates in the one-count mode and counts the delays. Timer count register mn (TCRmn) of the master channel starts operating upon start trigger detection and loads the value of timer data register mn (TDRmn).

The TCRmn register counts down from the value of the TDRmn register it has loaded, in synchronization with the count clock. When TCRmn = 0000H, it outputs INTTMmn and stops counting until the next start trigger is detected.

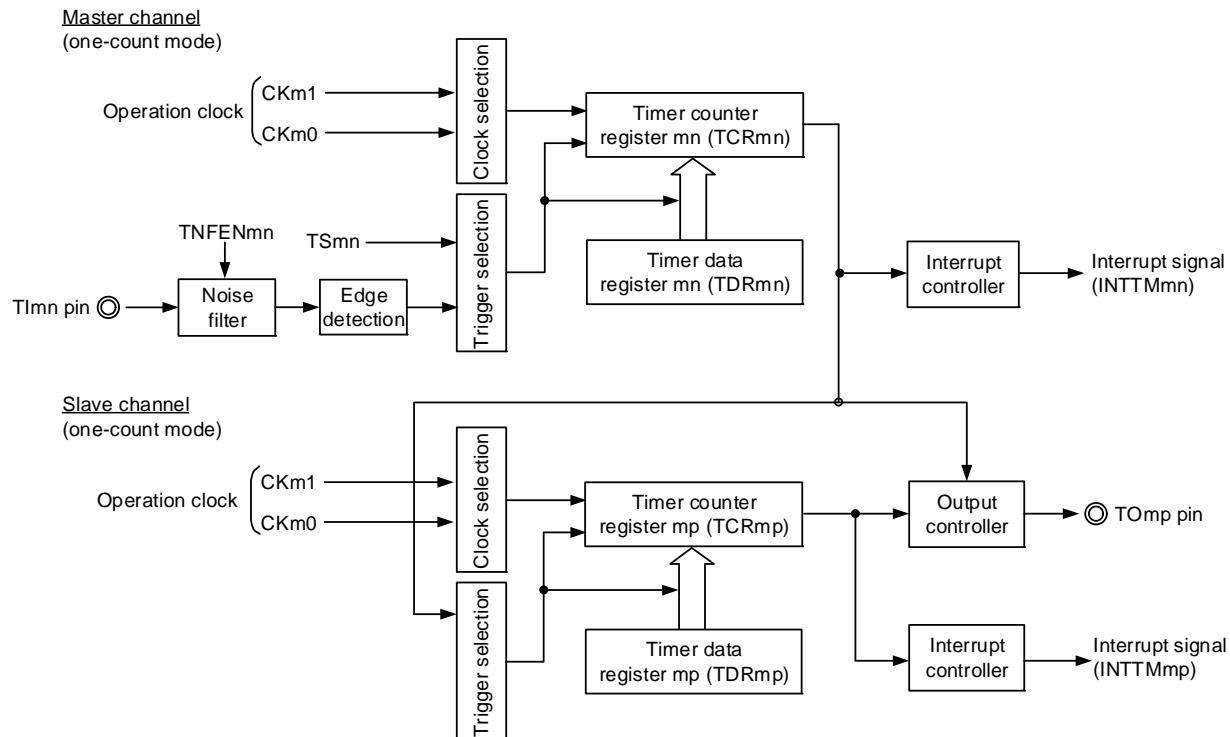
The slave channel operates in the one-count mode and counts the pulse width. The TCRmp register of the slave channel starts operation using INTTMmn of the master channel as a start trigger, and loads the value of the TDRmp register. The TCRmp register counts down from the value of The TDRmp register it has loaded, in synchronization with the count value. When count value = 0000H, it outputs INTTMmp and stops counting until the next start trigger (INTTMmn of the master channel) is detected. The output level of TOmp becomes active one count clock after generation of INTTMmn from the master channel, and inactive when TCRmp = 0000H.

Instead of using the TImn pin input, a one-shot pulse can also be output using the software operation (TSmn = 1) as a start trigger.

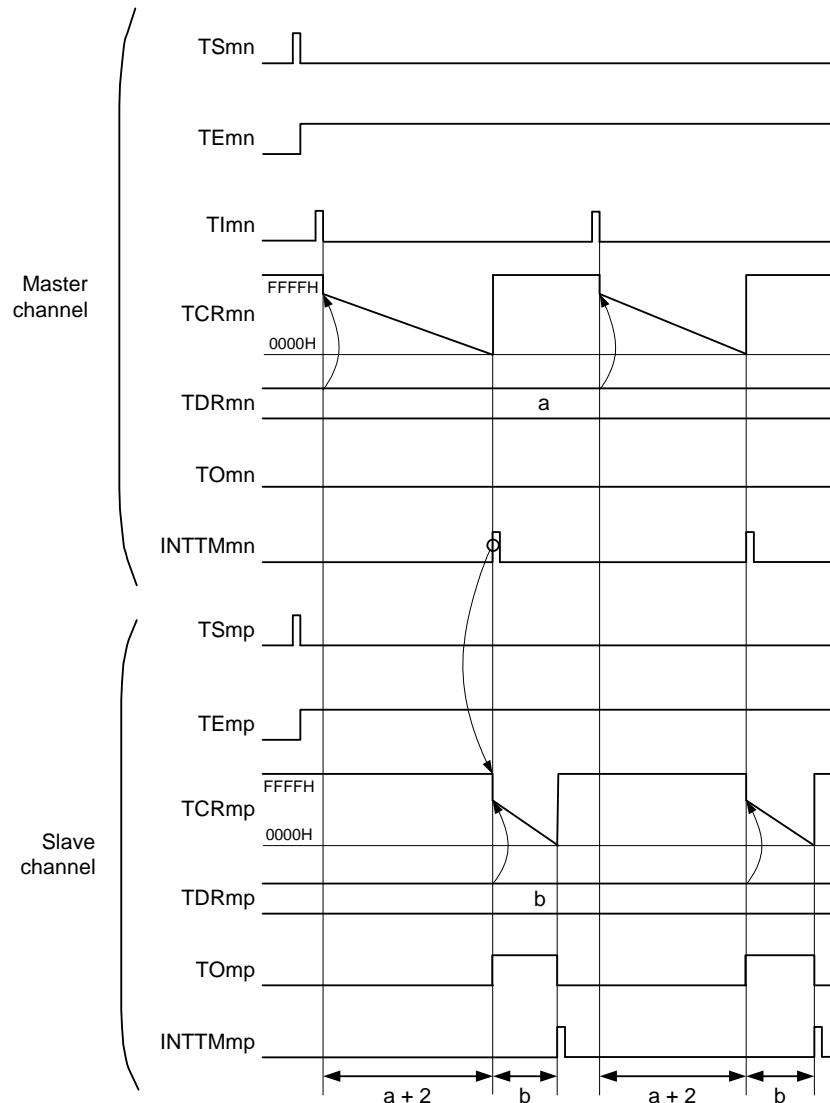
**Caution** The timing of loading of the TDRmn register of the master channel is different from that of the TDRmp register of the slave channel. If the TDRmn and TDRmp registers are rewritten during counting, therefore, an illegal waveform may be output in conflict with the timing of loading. Rewrite the TDRmn register after INTTMmn is generated and the TDRmp register after INTTMmp is generated.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)

p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

**Figure 6 - 71 Block Diagram of Operation as One-Shot Pulse Output Function**

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ )  
 p: Slave channel number ( $n = 0: p = 1, 2, 3, n = 2: p = 3$ )

**Figure 6 - 72 Example of Basic Timing of Operation as One-Shot Pulse Output Function**

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ )

p: Slave channel number ( $n = 0: p = 1, 2, 3, n = 2: p = 3$ )

**Remark 2.** TS<sub>mnn</sub>, TS<sub>mp</sub>: Bit n, p of timer channel start register m (TS<sub>m</sub>)

TE<sub>mnn</sub>, TE<sub>mp</sub>: Bit n, p of timer channel enable status register m (TE<sub>m</sub>)

TIm<sub>mnn</sub>, TIm<sub>mp</sub>: TIm<sub>mnn</sub> and TIm<sub>mp</sub> pins input signal

TCR<sub>mnn</sub>, TCR<sub>mp</sub>: Timer count registers mn, mp (TCR<sub>mnn</sub>, TCR<sub>mp</sub>)

TDR<sub>mnn</sub>, TDR<sub>mp</sub>: Timer data registers mn, mp (TDR<sub>mnn</sub>, TDR<sub>mp</sub>)

TO<sub>mnn</sub>, TO<sub>mp</sub>: TO<sub>mnn</sub> and TO<sub>mp</sub> pins output signal

**Figure 6 - 73 Example of Set Contents of Registers  
When One-Shot Pulse Output Function Is Used (Master Channel)**

(a) Timer mode register mn (TMRmn)

TMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn1 1/0	CKSmn0 0	0	CCSmn 0	MAS TERmn Note 1	STS mn2 0	STS mn1 0	STS mn0 1	CIS mn1 1/0	CIS mn0 1/0	0	0	MD mn3 1	MD mn2 0	MD mn1 0	MD mn0 0

Operation mode of channel n  
100B: One-count mode

Start trigger during operation  
0: Trigger input is invalid.

Selection of TImn pin input edge  
00B: Detects falling edge.  
01B: Detects rising edge.  
10B: Detects both edges.  
11B: Setting prohibited

Start trigger selection  
001B: Selects the TImn pin input valid edge.

Setting of the MASTERmn bit (channel 2)  
1: Master channel.

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channels n.  
10B: Selects CKm1 as operation clock of channels n.

(b) Timer output register m (TOm)

TOm	Bit n TOmn 0	0: Outputs 0 from TOmn.

(c) Timer output enable register m (TOEm)

TOEm	Bit n TOEm 0	0: Stops the TOmn output operation by counting operation.

(d) Timer output level register m (TOLm)

TOLm	Bit n TOLmn 0	0: Cleared to 0 when master channel output mode (TOMmn = 0)

(e) Timer output mode register m (TOMm)

TOMm	Bit n TOMmn 0	0: Sets master channel output mode.

**Note**      TMRm2: MASTERmn = 1  
                TMRm0: Fixed to 0

**Remark**      m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)

**Figure 6 - 74 Example of Set Contents of Registers  
When One-Shot Pulse Output Function Is Used (Slave Channel)**

(a) Timer mode register mp (TMRmp)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmp	CKSmp1 1/0	CKSmp0 0	0	CCSmp 0	M/S Note 0	STSmp2 1	STSmp1 0	STSmp0 0	CISmp1 0	CISmp0 0	0	0	MDmp3 1	MDmp2 0	MDmp1 0	MDmp0 0

Operation mode of channel p  
100B: One-count mode

Start trigger during operation  
0: Trigger input is invalid.

Selection of TI<sub>mp</sub> pin input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
100B: Selects INTTM<sub>mn</sub> of master channel.

Setting of MASTER<sub>rn</sub> bit (channel 2)  
0: Slave channel

Setting of SPLITmp bit (channels 1, 3)  
0: 16-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel p.  
10B: Selects CKm1 as operation clock of channel p.

\* Make the same setting as master channel.

(b) Timer output register m (TO<sub>m</sub>)

TO <sub>m</sub>	Bit p TO <sub>mp</sub> 1/0	0: Outputs 0 from TO <sub>mp</sub> . 1: Outputs 1 from TO <sub>mp</sub> .
-----------------	----------------------------------	--

(c) Timer output enable register m (TOEm)

TOEm	Bit p TO <sub>emp</sub> 1/0	0: Stops the TO <sub>mp</sub> output operation by counting operation. 1: Enables the TO <sub>mp</sub> output operation by counting operation.
------	-----------------------------------	--

(d) Timer output level register m (TOLm)

TOLm	Bit p TOLmp 1/0	0: Positive logic output (active-high) 1: Negative logic output (active-low)
------	-----------------------	---

(e) Timer output mode register m (TOMm)

TOMm	Bit p TOMmp 1	1: Sets the slave channel output mode.
------	---------------------	--

**Note** TMRm2: MASTERmp bit  
TMRm1, TMRm3: SPLITmp bit

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)  
p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

**Figure 6 - 75 Operation Procedure of One-Shot Pulse Output Function (1/2)**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable registers 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 and CKm1.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets the corresponding bit of the noise filter enable registers 1, 2 (NFEN1, NFEN2) to 1.</p> <p>Sets timer mode register mn, mp (TMRmn, TMRmp) of two channels to be used (determines operation mode of channels).</p> <p>An output delay is set to timer data register mn (TDRmn) of the master channel, and a pulse width is set to the TDRmp register of the slave channel.</p> <p>Sets slave channel. The TOMmp bit of timer output mode register m (TOMm) is set to 1 (slave channel output mode).</p> <p>Sets the TOLmp bit.</p> <p>Sets the TOOmp bit and determines default level of the TOOmp output.</p> <p>Sets the TOEmp bit to 1 and enables operation of TOOmp.</p> <p>Clears the port register and port mode register to 0.</p>	<p>Channel stops operating. (Clock is supplied and some power is consumed.)</p> <p>The TOOmp pin goes into Hi-Z output state.</p> <p>The TOOmp default setting level is output when the port mode register is in output mode and the port register is 0.</p> <p>TOOmp does not change because channel stops operating.</p> <p>The TOOmp pin outputs the TOOmp set level.</p>

(Note and Remark are listed on the next page.)

**Figure 6 - 76 Operation Procedure of One-Shot Pulse Output Function (2/2)**

	Software Operation	Hardware Status
Operation start	<p>Sets the TOEmp bit (slave) to 1 (only when operation is resumed).</p> <p>The TSmn (master) and TSmp (slave) bits of timer channel start register m (TSm) are set to 1 at the same time.</p> <p>The TSmn and TSmp bits automatically return to 0 because they are trigger bits.</p> <p>Count operation of the master channel is started by start trigger detection of the master channel.</p> <ul style="list-style-type: none"> <li>Detects the TImn pin input valid edge.</li> <li>Sets the TSmn bit of the master channel to 1 by software Note.</li> </ul>	<p>The TEmn and TEmp bits are set to 1 and the master channel enters the start trigger detection (the valid edge of the TImn pin input is detected or the TSmn bit of the master channel is set to 1) wait status.</p> <p>Counter stops operating.</p> <p>Master channel starts counting.</p>
During operation	<p>Set values of only the CISmn1 and CISmn0 bits of the TMRmn register can be changed.</p> <p>Set values of the TMRmp, TDRmn, TDRmp registers, TOMmn, TOMmp, TOLmn, and TOLmp bits cannot be changed.</p> <p>The TCRmn and TCRmp registers can always be read.</p> <p>The TSRmn and TSRmp registers are not used.</p> <p>Set values of the TOm and TOEm registers by slave channel can be changed.</p>	<p>Master channel loads the value of the TDRmn register to timer count register mn (TCRmn) by the start trigger detection (the valid edge of the TImn pin input is detected or the TSmn bit of the master channel is set to 1), and the counter starts counting down.</p> <p>When the count value reaches TCRmn = 0000H, the INTTMmn output is generated, and the counter stops until the next valid edge is input to the TImn pin.</p> <p>The slave channel, triggered by INTTMmn of the master channel, loads the value of the TDRmp register to the TCRmp register, and the counter starts counting down.</p> <p>The output level of TOmp becomes active one count clock after generation of INTTMmn from the master channel. It becomes inactive when TCRmp = 0000H, and the counting operation is stopped.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn (master) and TTmp (slave) bits are set to 1 at the same time.</p> <p>The TTmn (master) and TTmp (slave) bits are set to 1 at the same time.</p> <p>The TOEmp bit of slave channel is cleared to 0 and value is set to the TOmp bit.</p>	<p>TEmn, TEmp = 0, and count operation stops.</p> <p>The TCRmn and TCRmp registers hold count value and stop.</p> <p>The TOmp output is not initialized but holds current status.</p> <p>The TOmp pin outputs the TOmp set level.</p>
TAU stop	<p>To hold the TOmp pin output level</p> <p>Clears the TOmp bit to 0 after the value to be held is set to the port register.</p> <p>When holding the TOmp pin output level is not necessary</p> <p>Setting not required.</p> <p>The TAUMEN bit of the PER0 register is cleared to 0.</p>	<p>The TOmp pin output level is held by port function.</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p> <p>(The TOmp bit is cleared to 0 and the TOmp pin is set to port mode.)</p>

**Note** Do not set the TSmn bit of the slave channel to 1.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)  
p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

### 6.9.2 Operation as PWM function

Two channels can be used as a set to generate a pulse of any period and duty factor.

The period and duty factor of the output pulse can be calculated by the following expressions.

Pulse period = {Set value of TDRmn (master) + 1} × Count clock period  
 Duty factor [%] = {Set value of TDRmp (slave)}/{Set value of TDRmn (master) + 1} × 100  
 0% output: Set value of TDRmp (slave) = 0000H  
 100% output: Set value of TDRmp (slave) ≥ {Set value of TDRmn (master) + 1}

**Remark** The duty factor exceeds 100% if the set value of TDRmp (slave) > (set value of TDRmn (master) + 1), it summarizes to 100% output.

The master channel operates in the interval timer mode. If the channel start trigger bit (TSmn) of timer channel start register m (TSm) is set to 1, an interrupt (INTTMmn) is output, the value set to timer data register mn (TDRmn) is loaded to timer count register mn (TCRmn), and the counter counts down in synchronization with the count clock. When the counter reaches 0000H, INTTMmn is output, the value of the TDRmn register is loaded again to the TCRmn register, and the counter counts down. This operation is repeated until the channel stop trigger bit (TTmn) of timer channel stop register m (TTm) is set to 1.

If two channels are used to output a PWM waveform, the period until the master channel counts down to 0000H is the PWM output (TOmp) cycle.

The slave channel operates in one-count mode. By using INTTMmn from the master channel as a start trigger, the TCRmp register loads the value of the TDRmp register and the counter counts down to 0000H. When the counter reaches 0000H, it outputs INTTMmp and waits until the next start trigger (INTTMmn from the master channel) is generated.

If two channels are used to output a PWM waveform, the period until the slave channel counts down to 0000H is the PWM output (TOmp) duty.

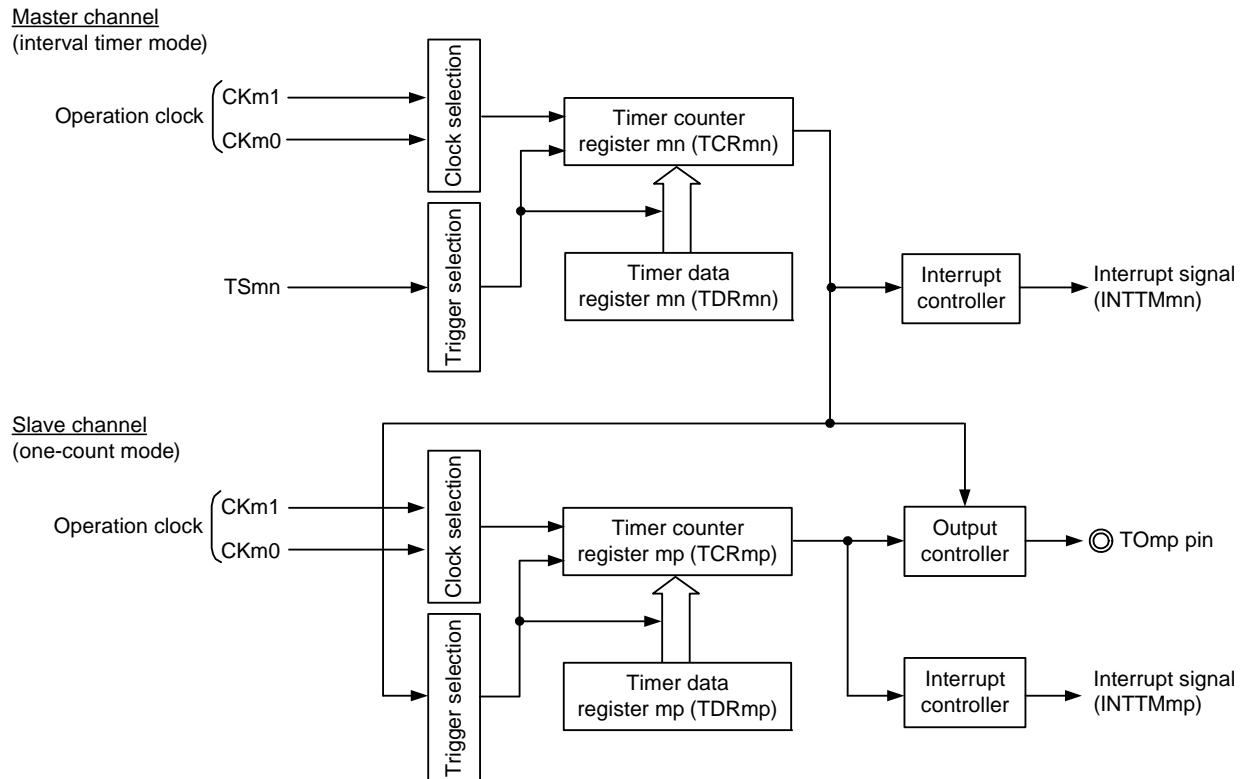
PWM output (TOmp) goes to the active level one clock after the master channel generates INTTMmn and goes to the inactive level when the TCRmp register of the slave channel becomes 0000H.

**Caution** To rewrite both timer data register mn (TDRmn) of the master channel and the TDRmp register of the slave channel, a write access is necessary two times. The timing at which the values of the TDRmn and TDRmp registers are loaded to the TCRmn and TCRmp registers is upon occurrence of INTTMmn of the master channel. Thus, when rewriting is performed split before and after occurrence of INTTMmn of the master channel, the TOmp pin cannot output the expected waveform. To rewrite both the TDRmn register of the master and the TDRmp register of the slave, therefore, be sure to rewrite both the registers immediately after INTTMmn is generated from the master channel.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)

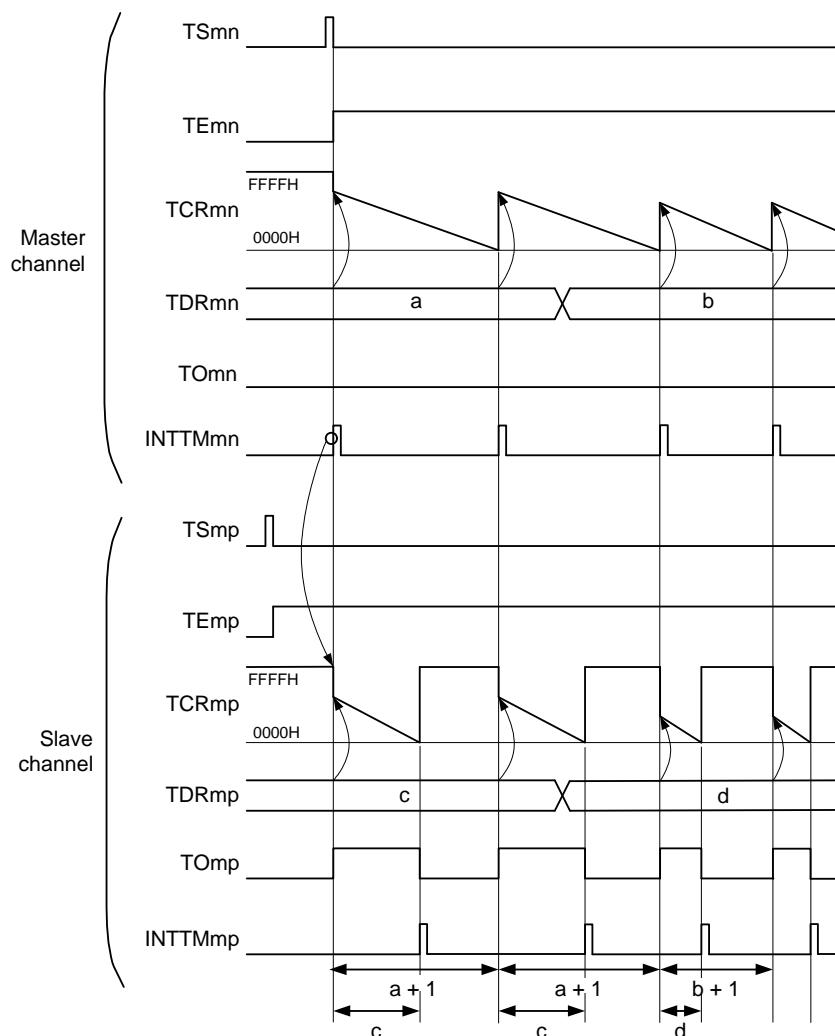
p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

Figure 6 - 77 Block Diagram of Operation as PWM Function



**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ )  
 p: Slave channel number ( $n = 0: p = 1, 2, 3, n = 2: p = 3$ )

Figure 6 - 78 Example of Basic Timing of Operation as PWM Function



**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ )

p: Slave channel number ( $n = 0: p = 1, 2, 3, n = 2: p = 3$ )

**Remark 2.** TS<sub>mn</sub>, TS<sub>mp</sub>: Bit n, p of timer channel start register m (TS<sub>m</sub>)

TE<sub>mn</sub>, TE<sub>mp</sub>: Bit n, p of timer channel enable status register m (TE<sub>m</sub>)

TCR<sub>mn</sub>, TCR<sub>mp</sub>: Timer count registers mn, mp (TCR<sub>mn</sub>, TCR<sub>mp</sub>)

TDR<sub>mn</sub>, TDR<sub>mp</sub>: Timer data registers mn, mp (TDR<sub>mn</sub>, TDR<sub>mp</sub>)

TO<sub>mn</sub>, TO<sub>mp</sub>: TO<sub>mn</sub> and TO<sub>mp</sub> pins output signal

**Figure 6 - 79 Example of Set Contents of Registers When PWM Function (Master Channel) Is Used**

## (a) Timer mode register mn (TMRmn)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmn	CKSmn1 1/0	CKSmn0 0	0	CCSmn 0	MAS TERmn Note 1	STS mn2 0	STS mn1 0	STS mn0 0	CIS mn1 0	CIS mn0 0	0	0	MDmn3 0	MDmn2 0	MDmn1 0	MDmn0 1

Operation mode of channel n  
000B: Interval timer

Setting of operation when counting is started  
1: Generates INTTmn when counting is started.

Selection of TI mn pin input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
000B: Selects only software start.

Setting of the MASTERmn bit (channel 2)  
1: Master channel.

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.

## (b) Timer output register m (TOM)

Bit n	<input type="checkbox"/> TOMn 0	0: Outputs 0 from TOMn.
-------	------------------------------------	-------------------------

## (c) Timer output enable register m (TOEm)

Bit n	<input type="checkbox"/> TOEmn 0	0: Stops the TOMn output operation by counting operation.
-------	-------------------------------------	---

## (d) Timer output level register m (TOLm)

Bit n	<input type="checkbox"/> TOLmn 0	0: Cleared to 0 when master channel output mode (TOMmn = 0)
-------	-------------------------------------	---

## (e) Timer output mode register m (TOMm)

Bit n	<input type="checkbox"/> TOMmn 0	0: Sets master channel output mode.
-------	-------------------------------------	-------------------------------------

**Note**      TMRm2: MASTERmn = 1  
                TMRm0: Fixed to 0

**Remark**    m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)

**Figure 6 - 80 Example of Set Contents of Registers When PWM Function (Slave Channel) Is Used**

## (a) Timer mode register mp (TMRmp)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMRmp	CKSmp1 1/0	CKSmp0 0	0	CCSmp 0	M/S Note 0	STSmp2 1	STSmp1 0	STSmp0 0	CISmp1 0	CISmp0 0	0	0	MDmp3 1	MDmp2 0	MDmp1 0	MDmp0 1

Operation mode of channel p  
100B: One-count mode

Start trigger during operation  
1: Trigger input is valid.

Selection of TImp pin input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
100B: Selects INTTMmn of master channel.

Setting of MASTERMn bit (channel 2)  
0: Slave channel

Setting of SPLITmp bit (channels 1, 3)  
0: 16-bit timer mode

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel p.  
10B: Selects CKm1 as operation clock of channel p.  
\* Make the same setting as master channel.

## (b) Timer output register m (TOm)

	Bit p	TOmp 1/0	0: Outputs 0 from TOmp. 1: Outputs 1 from TOmp.
TOm			

## (c) Timer output enable register m (TOEm)

	Bit p	TOEmp 1/0	0: Stops the TOmp output operation by counting operation. 1: Enables the TOmp output operation by counting operation.
TOEm			

## (d) Timer output level register m (TOLm)

	Bit p	TOLmp 1/0	0: Positive logic output (active-high) 1: Negative logic output (active-low)
TOLm			

## (e) Timer output mode register m (TOMm)

	Bit p	TOMmp 1	1: Sets the slave channel output mode.
TOMm			

**Note** TMRm2: MASTERM bit  
TMRm1, TMRm3: SPLITmp bit

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)  
p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

Figure 6 - 81 Operation Procedure When PWM Function Is Used (1/2)

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 and CKm1.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets timer mode registers mn, mp (TMRmn, TMRmp) of two channels to be used (determines operation mode of channels).</p> <p>An interval (period) value is set to timer data register mn (TDRmn) of the master channel, and a duty factor is set to the TDRmp register of the slave channel.</p> <p>Sets slave channel.</p> <p>The TOMmp bit of timer output mode register m (TOMm) is set to 1 (slave channel output mode).</p> <p>Sets the TOLmp bit.</p> <p>Sets the TOomp bit and determines default level of the TOomp output.</p> <p>Sets the TOEmp bit to 1 and enables operation of TOomp.</p> <p>Clears the port register and port mode register to 0.</p>	<p>Channel stops operating. (Clock is supplied and some power is consumed.)</p> <p>The TOomp pin goes into Hi-Z output state.</p> <p>The TOomp default setting level is output when the port mode register is in output mode and the port register is 0.</p> <p>TOomp does not change because channel stops operating.</p> <p>The TOomp pin outputs the TOomp set level.</p>

(Remark is listed on the next page.)

Figure 6 - 82 Operation Procedure When PWM Function Is Used (2/2)

	Software Operation	Hardware Status
Operation start	<p>Sets the TOEmp bit (slave) to 1 (only when operation is resumed).</p> <p>The TSmn (master) and TSmp (slave) bits of timer channel start register m (TSm) are set to 1 at the same time.</p> <p>The TSmn and TSmp bits automatically return to 0 because they are trigger bits.</p>	<p>TEmn = 1, TEmp = 1</p> <p>When the master channel starts counting, INTTMmn is generated. Triggered by this interrupt, the slave channel also starts counting.</p>
During operation	<p>Set values of the TMRmn and TMRmp registers, TOMmn, TOMmp, TOLmn, and TOLmp bits cannot be changed.</p> <p>Set values of the TDRmn and TDRmp registers can be changed after INTTMmn of the master channel is generated.</p> <p>The TCRmn and TCRmp registers can always be read. The TSRmn and TSRmp registers are not used.</p>	<p>The counter of the master channel loads the TDRmn register value to timer count register mn (TCRmn), and counts down. When the count value reaches TCRmn = 0000H, INTTMmn output is generated. At the same time, the value of the TDRmn register is loaded to the TCRmn register, and the counter starts counting down again.</p> <p>At the slave channel, the value of the TDRmp register is loaded to the TCRmp register, triggered by INTTMmn of the master channel, and the counter starts counting down. The output level of TOmp becomes active one count clock after generation of the INTTMmn output from the master channel. It becomes inactive when TCRmp = 0000H, and the counting operation is stopped.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn (master) and TTmp (slave) bits are set to 1 at the same time.</p> <p>The TTmn and TTmp bits automatically return to 0 because they are trigger bits.</p> <p>The TOEmp bit of slave channel is cleared to 0 and value is set to the TOmp bit.</p>	<p>TEmn, TEmp = 0, and count operation stops.</p> <p>The TCRmn and TCRmp registers hold count value and stop.</p> <p>The TOmp output is not initialized but holds current status.</p> <p>The TOmp pin outputs the TOmp set level.</p>
TAU stop	<p>To hold the TOmp pin output level</p> <p>Clears the TOmp bit to 0 after the value to be held is set to the port register.</p> <p>When holding the TOmp pin output level is not necessary</p> <p>Setting not required.</p> <p>The TAUMEN bit of the PER0 register is cleared to 0.</p>	<p>The TOmp pin output level is held by port function.</p> <p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p> <p>(The TOmp bit is cleared to 0 and the TOmp pin is set to port mode.)</p>

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)  
 p: Slave channel number (n = 0: p = 1, 2, 3, n = 2: p = 3)

### 6.9.3 Operation as multiple PWM output function

By extending the PWM function and using multiple slave channels, many PWM waveforms with different duty values can be output.

For example, when using two slave channels, the period and duty factor of an output pulse can be calculated by the following expressions.

$$\text{Pulse period} = \{\text{Set value of TDRmn (master)} + 1\} \times \text{Count clock period}$$

$$\text{Duty factor 1 [%]} = \{\text{Set value of TDRmp (slave 1)}\}/\{\text{Set value of TDRmn (master)} + 1\} \times 100$$

$$\text{Duty factor 2 [%]} = \{\text{Set value of TDRmq (slave 2)}\}/\{\text{Set value of TDRmn (master)} + 1\} \times 100$$

**Remark** Although the duty factor exceeds 100% if the set value of TDRmp (slave 1) > {set value of TDRmn (master) + 1} or if the {set value of TDRmq (slave 2)} > {set value of TDRmn (master) + 1}, it is summarized into 100% output.

Timer count register mn (TCRmn) of the master channel operates in the interval timer mode and counts the periods.

The TCRmp register of the slave channel 1 operates in one-count mode, counts the duty factor, and outputs a PWM waveform from the TOmp pin. The TCRmp register loads the value of timer data register mp (TDRmp), using INTTMmn of the master channel as a start trigger, and starts counting down. When TCRmp = 0000H, TCRmp outputs INTTMmp and stops counting until the next start trigger (INTTMmn of the master channel) has been input. The output level of TOmp becomes active one count clock after generation of INTTMmn from the master channel, and inactive when TCRmp = 0000H.

In the same way as the TCRmp register of the slave channel 1, the TCRmq register of the slave channel 2 operates in one-count mode, counts the duty factor, and outputs a PWM waveform from the TOmq pin. The TCRmq register loads the value of the TDRmq register, using INTTMmn of the master channel as a start trigger, and starts counting down. When TCRmq = 0000H, the TCRmq register outputs INTTMmq and stops counting until the next start trigger (INTTMmn of the master channel) has been input. The output level of TOmq becomes active one count clock after generation of INTTMmn from the master channel, and inactive when TCRmq = 0000H.

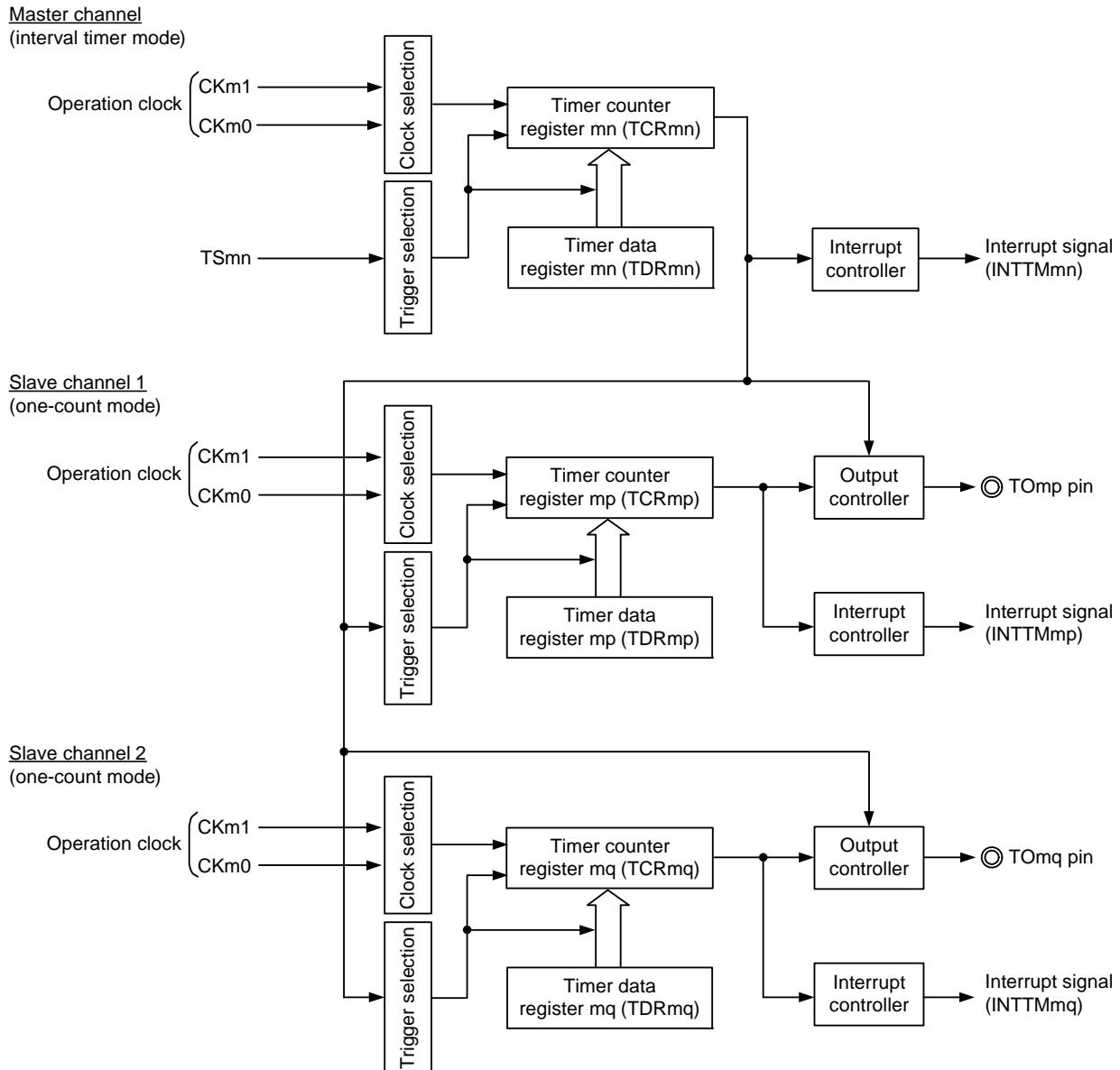
When channel 0 is used as the master channel as above, up to three types of PWM signals can be output at the same time.

**Caution** To rewrite both timer data register mn (TDRmn) of the master channel and the TDRmp register of the slave channel 1, write access is necessary at least twice. Since the values of the TDRmn and TDRmp registers are loaded to the TCRmn and TCRmp registers after INTTMmn is generated from the master channel, if rewriting is performed separately before and after generation of INTTMmn from the master channel, the TOmp pin cannot output the expected waveform. To rewrite both the TDRmn register of the master and the TDRmp register of the slave, be sure to rewrite both the registers immediately after INTTMmn is generated from the master channel (This applies also to the TDRmq register of the slave channel 2).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0)

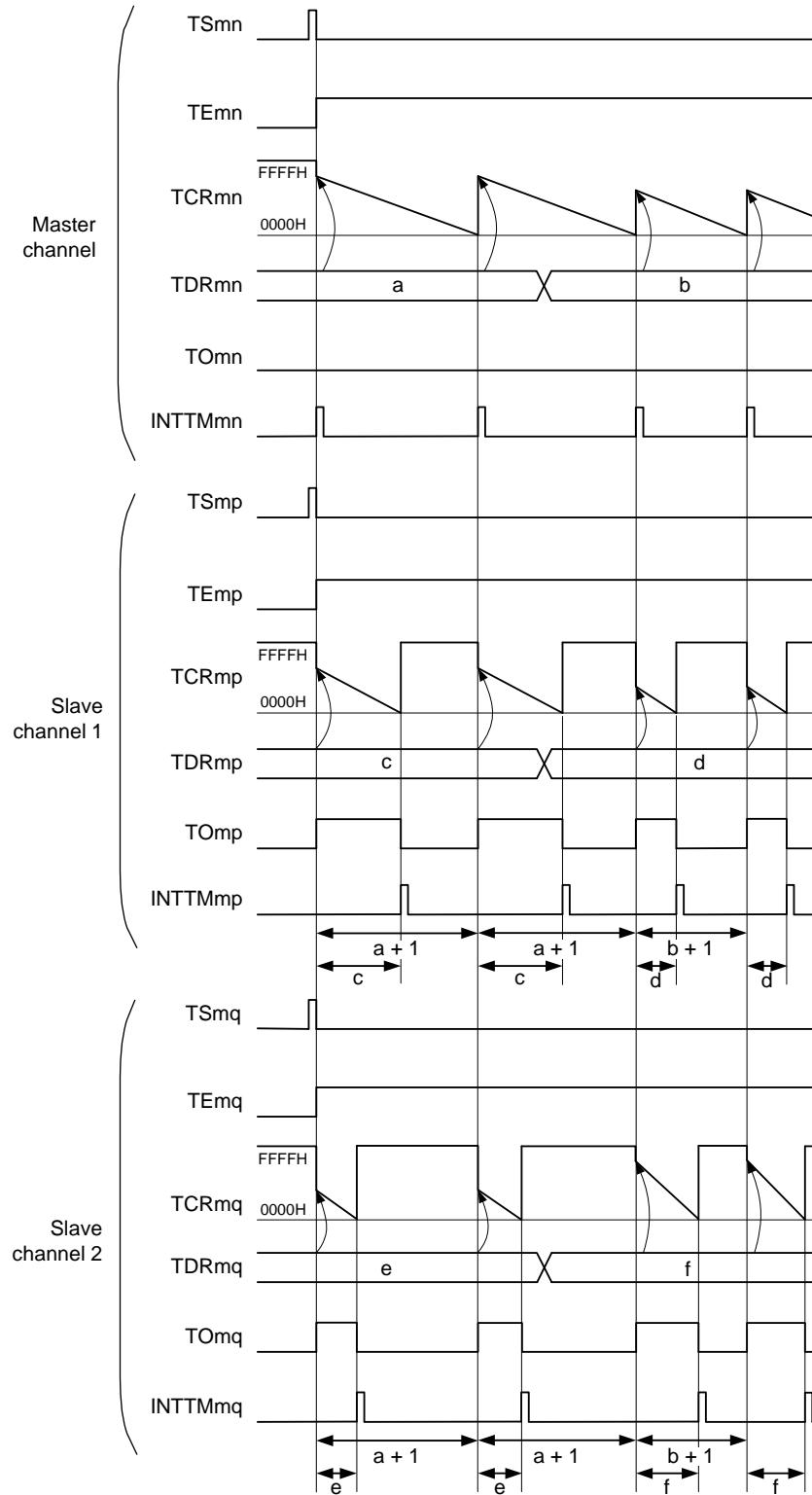
p: Slave channel number 1, q: Slave channel number 2

n < p < q ≤ 3 (Where p and q are integers greater than n)

**Figure 6 - 83 Block Diagram of Operation as Multiple PWM Output Function (output two types of PWMs)**

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$ )  
 p: Slave channel number 1, q: Slave channel number 2  
 $n < p < q \leq 3$  (Where p and q are integers greater than n)

Figure 6 - 84 Example of Basic Timing of Operation as Multiple PWM Output Function (Output two types of PWMs)



(Remarks are listed on the next page.)

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$ )

p: Slave channel number 1, q: Slave channel number 2

$n < p < q \leq 3$  (Where p and q are integers greater than n)

**Remark 2.** TSmn, TSmp, TSmq: Bit n, p, q of timer channel start register m (TSm)

TEmn, TEMp, TEMq: Bit n, p, q of timer channel enable status register m (TEM)

TCRmn, TCRmp, TCRmq: Timer count registers mn, mp, mq (TCRmn, TCRmp, TCRmq)

TDRmn, TDRmp, TDRmq: Timer data registers mn, mp, mq (TDRmn, TDRmp, TDRmq)

TOmn, TOmp, TOmq: TOmn, TOmp, and TOmq pins output signal

**Figure 6 - 85 Example of Set Contents of Registers  
When Multiple PWM Output Function (Master Channel) Is Used**

(a) Timer mode register mn (TMRmn)

TMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn1 1/0	CKSmn0 0	0	CCSmn 0	MAS TERmn Note 1	STS mn2 0	STS mn1 0	STS mn0 0	CIS mn1 0	CIS mn0 0	0	0	MD mn3 0	MD mn2 0	MD mn1 0	MD mn0 1

Operation mode of channel n  
000B: Interval timer

Setting of operation when counting is started  
1: Generates INTTMmn when counting is started.

Selection of TI mn pin input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
000B: Selects only software start.

Setting of MASTERmn bit (channel 2)  
1: Master channel.

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel n.  
10B: Selects CKm1 as operation clock of channel n.

(b) Timer output register m (TOm)

TOm	Bit n	0: Outputs 0 from TOmn.
	0	

(c) Timer output enable register m (TOEm)

TOEm	Bit n	0: Stops the TOmn output operation by counting operation.
	0	

(d) Timer output level register m (TOLm)

TOLm	Bit n	0: Cleared to 0 when master channel output mode (TOMmn = 0)
	0	

(e) Timer output mode register m (TOMm)

TOMm	Bit n	0: Sets master channel output mode.
	0	

**Note** TMRm2: MASTERmn = 1  
TMRm0: Fixed to 0

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0)

**Figure 6 - 86 Example of Set Contents of Registers  
When Multiple PWM Output Function (Slave Channel) Is Used (output two types of PWMs)**

(a) Timer mode register mp, mq (TMRmp, TMRmq)

TMRmp	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmp1 1/0	CKSmp0 0	0	CCSmp 0	M/S Note 0	STSmp2 1	STSmp1 0	STSmp0 0	CISmp1 0	CISmp0 0	0	0	MDmp3 1	MDmp2 0	MDmp1 0	MDmp0 1
TMRmq	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmq1 1/0	CKSmq0 0	0	CCSmq 0	M/S Note 0	STSmp2 1	STSmp1 0	STSmp0 0	CISmp1 0	CISmp0 0	0	0	MDmq3 1	MDmq2 0	MDmq1 0	MDmq0 1

Operation mode of channel p, q  
100B: One-count mode  
Start trigger during operation  
1: Trigger input is valid.

Selection of TI<sub>mp</sub> and TI<sub>mq</sub> pins input edge  
00B: Sets 00B because these are not used.

Start trigger selection  
100B: Selects INTTMmn of master channel.

Setting of MASTERmp and MASTERmq bits (channel 2)  
0: Slave channel

Setting of SPLITmp and SPLITmq bits (channels 1, 3)  
0: 16-bit timer mode.

Count clock selection  
0: Selects operation clock (fmck).

Operation clock (fmck) selection  
00B: Selects CKm0 as operation clock of channel p, q.  
10B: Selects CKm1 as operation clock of channel p, q.  
\* Make the same setting as master channel.

(b) Timer output register m (TOm)

TOm	Bit q Bit p	TOmp 1/0	TOmp 1/0	0: Outputs 0 from TOmp or TOmq. 1: Outputs 1 from TOmp or TOmq.
-----	-------------	-------------	-------------	--

(c) Timer output enable register m (TOEm)

TOEm	Bit q Bit p	TOEmq 1/0	TOEmp 1/0	0: Stops the TOmp or TOmq output operation by counting operation. 1: Enables the TOmp or TOmq output operation by counting operation.
------	-------------	--------------	--------------	--

(d) Timer output level register m (TOLm)

TOLm	Bit q Bit p	TOLmq 1/0	TOLmp 1/0	0: Positive logic output (active-high) 1: Negative logic output (active-low)
------	-------------	--------------	--------------	---

(e) Timer output mode register m (TOMm)

TOMm	Bit q Bit p	TOMmq 1	TOMmp 1	1: Sets the slave channel output mode.
------	-------------	------------	------------	--

**Note**      TMRm2: MASTERmp, MASTERmq bit  
                TMRm1, TMRm3: SPLITmp, SPLITmq bit

**Remark**    m: Unit number (m = 0, 1), n: Channel number (n = 0)  
                p: Slave channel number 1, q: Slave channel number 2  
                n < p < q ≤ 3 (Where p and q are integers greater than n)

**Figure 6 - 87 Operation Procedure When Multiple PWM Output Function Is Used (output two types of PWMs) (1/2)**

	Software Operation	Hardware Status
TAU default setting	<p>Sets the TAUmEN bit of peripheral enable register 0 (PER0) to 1.</p> <p>Sets timer clock select register m (TPSm). Determines clock frequencies of CKm0 and CKm1.</p>	<p>Input clock supply for timer array unit m is stopped (Clock supply is stopped and writing to each register is disabled.)</p> <p>Input clock supply for timer array unit m is supplied. Each channel stops operating. (Clock supply is started and writing to each register is enabled.)</p>
Channel default setting	<p>Sets timer mode registers mn, mp, mq (TMRmn, TMRmp, TMRmq) of each channel to be used (determines operation mode of channels).</p> <p>An interval (period) value is set to timer data register mn (TDRmn) of the master channel, and a duty factor is set to the TDRmp and TDRmq registers of the slave channels.</p> <p>Sets slave channels.</p> <p>The TOMmp and TOMmq bits of timer output mode register m (TOMm) are set to 1 (slave channel output mode).</p> <p>Clears the TOLmp and TOLmq bits to 0.</p> <p>Sets the TOomp and TOomq bits and determines default level of the TOomp and TOomq outputs.</p> <p>Sets the TOEmp and TOEmq bits to 1 and enables operation of TOomp and TOomq.</p> <p>Clears the port register and port mode register to 0.</p>	<p>Channel stops operating. (Clock is supplied and some power is consumed.)</p> <p>The TOomp and TOomq pins go into Hi-Z output state.</p> <p>The TOomp and TOomq default setting levels are output when the port mode register is in output mode and the port register is 0.</p> <p>TOomp and TOomq do not change because channels stop operating.</p> <p>The TOomp and TOomq pins output the TOomp and TOomq set levels.</p>

(Remark is listed on the next page.)

**Figure 6 - 88 Operation Procedure When Multiple PWM Output Function Is Used (output two types of PWMs) (2/2)**

	Software Operation	Hardware Status
Operation start	<p>(Sets the TOEmp and TOEmq (slave) bits to 1 only when resuming operation.)</p> <p>The TSmn bit (master), and TSmp and TSmq (slave) bits of timer channel start register m (TSm) are set to 1 at the same time.</p> <p>The TSmn, TSmp, and TSmq bits automatically return to 0 because they are trigger bits.</p>	<p>TEMn = 1, TEmp, TEmq = 1</p> <p>When the master channel starts counting, INTTMmn is generated. Triggered by this interrupt, the slave channel also starts counting.</p>
During operation	<p>Set values of the TMRmn, TMRmp, TMRmq registers, TOMmn, TOMmp, TOMmq, TOLmn, TOLmp, and TOLmq bits cannot be changed.</p> <p>Set values of the TDRmn, TDRmp, and TDRmq registers can be changed after INTTMmn of the master channel is generated.</p> <p>The TCRmn, TCRmp, and TCRmq registers can always be read.</p> <p>The TSRmn, TSRmp, and TSRmq registers are not used.</p>	<p>The counter of the master channel loads the TDRmn register value to timer count register mn (TCRmn) and counts down. When the count value reaches TCRmn = 0000H, INTTMmn output is generated. At the same time, the value of the TDRmn register is loaded to the TCRmn register, and the counter starts counting down again.</p> <p>At the slave channel 1, the values of the TDRmp register are transferred to the TCRmp register, triggered by INTTMmn of the master channel, and the counter starts counting down. The output levels of TOmp become active one count clock after generation of the INTTMmn output from the master channel. It becomes inactive when TCRmp = 0000H, and the counting operation is stopped.</p> <p>At the slave channel 2, the values of the TDRmq register are transferred to TCRmq register, triggered by INTTMmn of the master channel, and the counter starts counting down. The output levels of TOMq become active one count clock after generation of the INTTMmn output from the master channel. It becomes inactive when TCRmq = 0000H, and the counting operation is stopped.</p> <p>After that, the above operation is repeated.</p>
Operation stop	<p>The TTmn bit (master), TTmp, and TTmq (slave) bits are set to 1 at the same time.</p> <p>The TTmn, TTmp, and TTmq bits automatically return to 0 because they are trigger bits.</p> <p>The TOEmp and TOEmq bits of slave channels are cleared to 0 and value is set to the TOmp and TOMq bits.</p>	<p>TEMn, TEmp, TEmq = 0, and count operation stops.</p> <p>The TCRmn, TCRmp, and TCRmq registers hold count value and stop.</p> <p>The TOmp and TOMq output are not initialized but hold current status.</p> <p>The TOmp and TOMq pins output the TOmp and TOMq set levels.</p>
TAU stop	<p>To hold the TOmp and TOMq pin output levels</p> <p>Clears the TOmp and TOMq bits to 0 after the value to be held is set to the port register.</p> <p>When holding the TOmp and TOMq pin output levels are not necessary</p> <p>Setting not required</p> <p>The TAUMEN bit of the PER0 register is cleared to 0.</p>	<p>The TOmp and TOMq pin output levels are held by port function.</p> <p>Input clock supply for timer array unit m is stopped</p> <p>All circuits are initialized and SFR of each channel is also initialized.</p> <p>(The TOmp and TOMq bits are cleared to 0 and the TOmp and TOMq pins are set to port mode.)</p>

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$ )

p: Slave channel number, q: Slave channel number

$n < p < q \leq 3$  (Where p and q are integer greater than n)

## 6.10 Cautions When Using Timer Array Unit

### 6.10.1 Cautions When Using Timer output

Depends on products, a pin is assigned a timer output and other alternate functions. In this case, outputs of the other alternate functions must be set in initial status.

For details, see [4.5 Register Settings When Using Alternate Function](#).

## CHAPTER 7 TIMER RJ

### 7.1 Functions of Timer RJ

Timer RJ is a 16-bit timer that can be used for pulse output, external pulse width or period measurement, and counting external events.

This 16-bit timer consists of a reload register and a down counter. The reload register and the down counter are allocated to the same address, and they can be accessed by accessing the TRJ0 register.

Table 7 - 1 lists the Timer RJ Specifications. Figure 7 - 1 shows the Timer RJ Block Diagram.

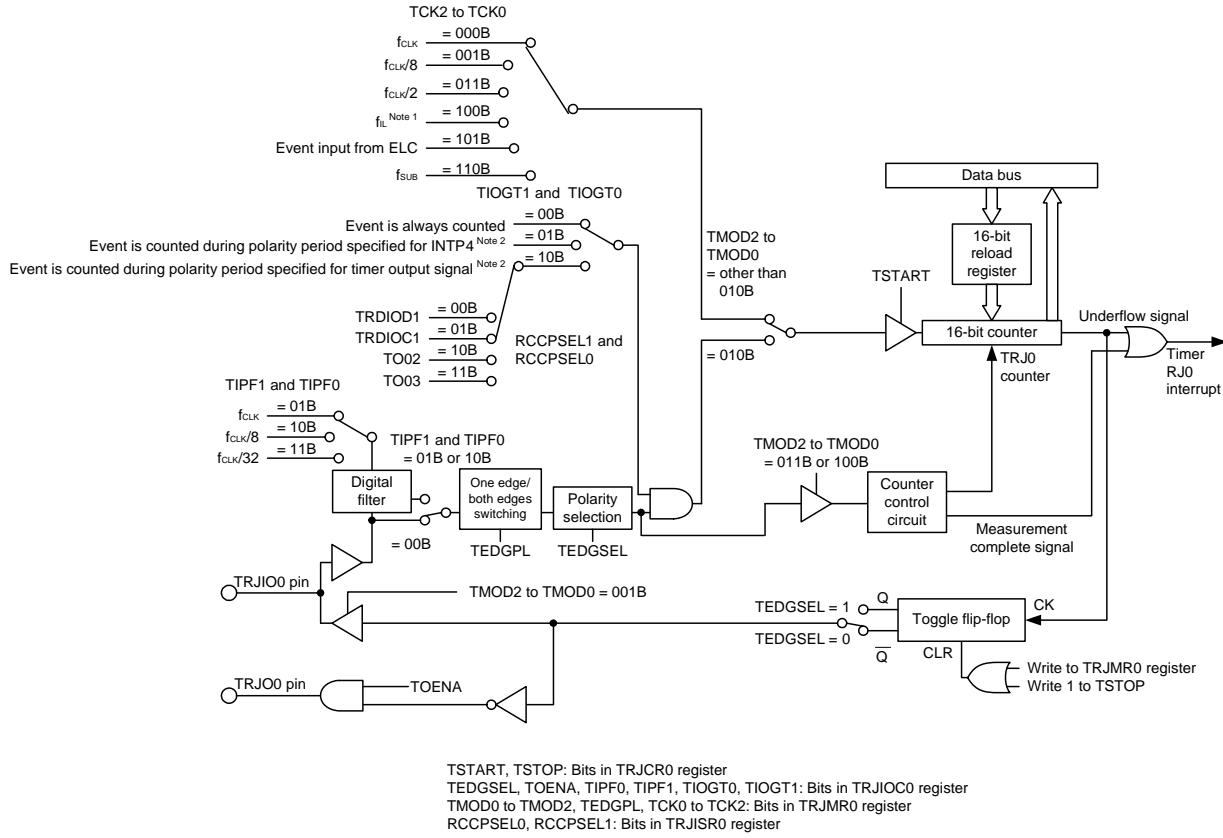
**Table 7 - 1 Timer RJ Specifications**

Item		Description
Operating modes	Timer mode	The count source is counted.
	Pulse output mode	The count source is counted and the output is inverted at each underflow of the timer.
	Event counter mode	An external event is counted. Operation is possible in STOP mode.
	Pulse width measurement mode	An external pulse width is measured.
	Pulse period measurement mode	An external pulse period is measured.
Count source (Operating clock)		fCLK, fCLK/2, fCLK/8, fIL, fSUB, or event input from the event link controller (ELC) selectable
Interrupt		<ul style="list-style-type: none"> <li>• When the counter underflows.</li> <li>• When the measurement of the active width of the external input (TRJIO0) is completed in pulse width measurement mode.</li> <li>• When the set edge of the external input (TRJIO0) is input in pulse period measurement mode.</li> </ul>
Selectable functions		<ul style="list-style-type: none"> <li>• Coordination with the event link controller (ELC). Event input from the ELC is selectable as a count source.</li> </ul>

## 7.2 Configuration of Timer RJ

Figure 7 - 1 shows the Timer RJ Block Diagram and Table 7 - 2 lists the Timer RJ Pin Configuration.

**Figure 7 - 1 Timer RJ Block Diagram**



- Note 1.** When selecting  $f_{IL}$  as the count source, set the WUTMMCK0 bit in the subsystem clock supply mode control register (OSMC) to 1. However,  $f_{IL}$  cannot be selected as the count source for timer RJ when  $f_{SUB}$  is selected as the count source for the real-time clock or the 12-bit interval timer.
- Note 2.** The polarity can be selected by the RCCPSEL2 bit in the TRJSR0 register.

**Table 7 - 2 Timer RJ Pin Configuration**

Pin Name	I/O	Function
INTP4	Input	Event counter mode control for timer RJ
TRJIO0 Note	Input/output	External event input and pulse output for timer RJ
TRJO0 Note	Output	Pulse output for timer RJ

- Note** The assignment of the TRJIO0 pin is selected by bits PIOR12 and PIOR13 in the PIOR1 register. The assignment of the TRJO0 pin is selected by bits PIOR10 and PIOR11 in the PIOR1 register. Refer to **CHAPTER 4 PORT FUNCTIONS** for details.

### 7.3 Registers Controlling Timer RJ

Table 7 - 3 lists the Registers Controlling Timer RJ.

**Table 7 - 3 Registers Controlling Timer RJ**

Register Name	Symbol
Peripheral I/O redirection register 1	PIOR1
Peripheral enable register 1	PER1
Subsystem clock supply mode control register	OSMC
Timer RJ counter register 0 Note	TRJ0
Timer RJ control register 0	TRJCR0
Timer RJ I/O control register 0	TRJIOC0
Timer RJ mode register 0	TRJMR0
Timer RJ event pin select register 0	TRJISR0
Port register 0	P0
Port register 3	P3
Port register 4	P4
Port register 5	P5
Port mode register 0	PM0
Port mode register 3	PM3
Port mode register 4	PM4
Port mode register 5	PM5

**Note** When the TRJ0 register is accessed, the CPU does not proceed to the next instruction processing but enters the wait state for CPU processing. For this reason, if this wait state occurs, the number of instruction execution clocks is increased by the number of wait clocks. The number of wait clocks for access to the TRJ0 register is one clock for both writing and reading.

### 7.3.1 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to the hardware that is not used is also stopped so as to decrease the power consumption and noise.

To use Timer RJ, be sure to set bit 0 (TRJ0EN) to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 2 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN
TRJ0EN	Control of timer RJ0 input clock supply							
0	Stops input clock supply. • SFR used by timer RJ0 cannot be written. • Timer RJ0 is in the reset status.							
1	Enables input clock supply. • SFR used by timer RJ0 can be read and written.							

**Note**      Only for products with 96 KB or more code flash memory.

**Caution 1. When setting timer RJ, be sure to set the TRJ0EN bit to 1 first. If TRJ0EN = 0, writing to a control register of timer RJ is ignored, and all read values are default values (except for port mode registers 0, 3, 4, 5 (PM0, PM3, PM4, PM5), and port registers 0, 3, 4, 5 (P0, P3, P4, P5)).**

**Caution 2. Be sure to set the following bits to 0:**

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 7.3.2 Subsystem clock supply mode control register (OSMC)

The WUTMMCK0 bit can be used to select the timer RJ operation clock.

In addition, by stopping clock functions that are unnecessary, the RTCLPC bit can be used to reduce power consumption. For details about setting the RTCLPC bit, see **CHAPTER 5 CLOCK GENERATOR**.

The OSMC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 3 Format of Subsystem clock supply mode control register (OSMC)**

Address: F00F3H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

WUTMMCK0	Selection of operation clock (fRTC) for real-time clock, 12-bit interval timer, and timer RJ
0	<p>Subsystem clock (fSUB)</p> <ul style="list-style-type: none"> <li>The subsystem clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>The low-speed on-chip oscillator cannot be selected as the count source for timer RJ.</li> </ul>
1	<p>Low-speed on-chip oscillator clock (fIL)</p> <ul style="list-style-type: none"> <li>The low-speed on-chip oscillator clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>Either the low-speed on-chip oscillator or the subsystem clock can be selected as the count source for timer RJ.</li> </ul>

### 7.3.3 Timer RJ counter register 0 (TRJ0)

TRJ0 is a 16-bit register. The write value is written to the reload register and the read value is read from the counter.

The states of the reload register and the counter are changed depending on the TSTART bit in the TRJCR0 register. For details, see **7.4.1 Reload Register and Counter Rewrite Operation**.

The TRJ0 register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to FFFFH.

**Figure 7 - 4 Format of Timer RJ counter register 0 (TRJ0)**

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRJ0																
—	Function															Setting Range
Bits 15 to 0	16-bit counter Notes 1, 2															0000H to FFFFH

**Note 1.** When 1 is written to the TSTOP bit in the TRJCR0 register, the 16-bit counter is forcibly stopped and set to FFFFH.

**Note 2.** When the setting of bits TCK2 to TCK0 in the TRJMR0 register is other than 001B (fCLK/8) or 011B (fCLK/2), if the TRJ0 register is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. However, the TRJ00 and TRJIO0 output is toggled.

When the TRJ0 register is set to 0000H in event counter mode, regardless of the value of bits TCK2 to TCK0, a request signal to the DTC and the ELC is generated only once immediately after the count starts.

In addition, the TRJ00 output is toggled even during a period other than the specified count period.

When the TRJ0 register is set to 0000H or a higher value, a request signal is generated each time TRJ underflows.

**Caution** When the TRJ0 register is accessed, the CPU does not proceed to the next instruction processing but enters the wait state for CPU processing. For this reason, if this wait state occurs, the number of instruction execution clocks is increased by the number of wait clocks. The number of wait clocks for access to the TRJ0 register is one clock for both writing and reading.

### 7.3.4 Timer RJ control register 0 (TRJCR0)

The TRJCR0 register starts or stops count operation and indicates the status of timer RJ.

The TRJCR0 register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 5 Format of Timer RJ control register 0 (TRJCR0)**

Address: F0240H After Reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
TRJCR0	0	0	TUNDF	TEDGF	0	TSTOP	TCSTF	TSTART

TUNDF	Timer RJ underflow flag
0	No underflow
1	Underflow
[Condition for setting to 0]	
• When 0 is written to this bit by a program.	
[Condition for setting to 1]	
• When the counter underflows.	

TEDGF	Active edge judgement flag
0	No active edge received
1	Active edge received
[Condition for setting to 0]	
• When 0 is written to this bit by a program.	
[Conditions for setting to 1]	
• When the measurement of the active width of the external input (TRJIO) is completed in pulse width measurement mode.	
• The set edge of the external input (TRJIO) is input in pulse period measurement mode.	

TSTOP	Timer RJ count forced stop Note 1
When 1 is written to this bit, the count is forcibly stopped. The read value is 0.	

TCSTF	Timer RJ count status flag Note 2
0	Count stops
1	Count in progress
[Conditions for setting to 0]	
• When 0 is written to the TSTART bit (the TCSTF bit is set to 0 in synchronization with the count source).	
• When 1 is written to the TSTOP bit.	
[Condition for setting to 1]	
• When 1 is written to the TSTART bit (the TCSTF bit is set to 1 in synchronization with the count source).	

TSTART	Timer RJ count start Note 2
0	Count stops
1	Count starts
Count operation is started by writing 1 to the TSTART bit and stopped by writing 0. When the TSTART bit is set to 1 (count starts), the TCSTF bit is set to 1 (count in progress) in synchronization with the count source. Also, after 0 is written to the TSTART bit, the TCSTF bit is set to 0 (count stops) in synchronization with the count source. For details, see 7.5.1 Count Operation Start and Stop Control.	

**Note 1.** When 1 (count is forcibly stopped) is written to the TSTOP bit, bits TSTART and TCSTF are initialized at the same time. The pulse output level is also initialized.

**Note 2.** For notes on using bits TSTART and TCSTF, see 7.5.1 Count Operation Start and Stop Control.

### 7.3.5 Timer RJ I/O control register 0 (TRJIOC0)

The TRJIOC0 register sets the input/output of timer RJ.

The TRJIOC0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 6 Format of Timer RJ I/O control register 0 (TRJIOC0)**

Address: F0241H After Reset: 00H R/W

Symbol 7 6 5 4 3 2 1 0

TRJIOC0	TIOGT1	TIOGT0	TIPF1	TIPF0	0	TOENA	0	TEDGSEL
---------	--------	--------	-------	-------	---	-------	---	---------

TIOGT1	TIOGT0	TRJIO count control Notes 1, 2
0	0	Event is always counted
0	1	Event is counted during polarity period specified for INTP4
1	0	Event is counted during polarity period specified for timer output signal
Other than above		Setting prohibited

TIPF1	TIPF0	TRJIO input filter select
0	0	No filter
0	1	Filter sampled at fCLK
1	0	Filter sampled at fCLK/8
1	1	Filter sampled at fCLK/32

These bits are used to specify the sampling frequency of the filter for the TRJIO input. If the input to the TRJIO0 pin is sampled and the value matches three successive times, that value is taken as the input value.

TOENA	TRJO output enable
0	TRJO output disabled (port)
1	TRJO output enabled

TEDGSEL	I/O polarity switch
Function varies depending on the operating mode (see <b>Tables 7 - 4</b> and <b>7 - 5</b> ).	

**Note 1.** When INTP4 or the timer output signal is used, the polarity to count an event can be selected by the RCCPSEL2 bit in the TRJSR0 register.

**Note 2.** Bits TIOGT0 and TIOGT1 are enabled only in event counter mode.

**Table 7 - 4 TRJIO I/O Edge and Polarity Switching**

Operating Mode	Function
Timer mode	Not used (I/O port)
Pulse output mode	0: Output is started at high (Initialization level: High) 1: Output is started at low (Initialization level: Low)
Event counter mode	0: Count at rising edge 1: Count at falling edge
Pulse width measurement mode	0: Low-level width is measured 1: High-level width is measured
Pulse period measurement mode	0: Measure from one rising edge to the next rising edge 1: Measure from one falling edge to the next falling edge

**Table 7 - 5 TRJO Output Polarity Switching**

Operating Mode	Function
All modes	0: Output is started at low (Initialization level: Low) 1: Output is started at high (Initialization level: High)

### 7.3.6 Timer RJ mode register 0 (TRJMR0)

The TRJMR0 register sets the operating mode of timer RJ.

The TRJMR0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 7 Format of Timer RJ mode register 0 (TRJMR0)**

Address: F0242H      After Reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

TRJMR0	0	TCK2	TCK1	TCK0	TEDGPL	TMOD2	TMOD1	TMOD0
--------	---	------	------	------	--------	-------	-------	-------

TCK2	TCK1	TCK0	Timer RJ count source select Notes 1, 2
0	0	0	fCLK
0	0	1	fCLK/8
0	1	1	fCLK/2
1	0	0	fIL Note 4
1	0	1	Event input from ELC
1	1	0	fSUB
Other than above			Setting prohibited

TEDGPL	TRJIO edge polarity select Note 5
0	One edge
1	Both edges

TMOD2	TMOD1	TMOD0	Timer RJ operating mode select Note 3
0	0	0	Timer mode
0	0	1	Pulse output mode
0	1	0	Event counter mode
0	1	1	Pulse width measurement mode
1	0	0	Pulse period measurement mode
Other than above			Setting prohibited

**Note 1.** When event counter mode is selected, the external input (TRJIO) is selected as the count source regardless of the setting of bits TCK0 to TCK2.

**Note 2.** Do not switch count sources during count operation. Count sources should be switched when both the TSTART and TCSTF bits in the TRJCR0 register are set to 0 (count stops).

**Note 3.** The operating mode can be changed only when the count is stopped while both the bits TSTART and TCSTF in the TRJCR0 register are set to 0 (count stops). Do not change the operating mode during count operation.

**Note 4.** When selecting fIL as the count source, set the WUTMMCK0 bit in the subsystem clock supply mode control register (OSMC) to 1.

However, fIL cannot be selected as the count source for timer RJ when fsub is selected as the count source for the real-time clock or the 12-bit interval timer.

**Note 5.** The TEDGPL bit is enabled only in event counter mode.

**Note 6.** Write access to the TRJMR0 register initializes the output from pins TRJO0 and TRJIO0 of timer RJ.

For details on the output level at initialization, refer to the description of **Figure 7 - 6 Format of Timer RJ I/O control register 0 (TRJIOC0)**.

### 7.3.7 Timer RJ event pin select register 0 (TRJISR0)

The TRJISR0 register selects the timer for controlling the event count period and sets the polarity in event counter mode.

The TRJISR0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 7 - 8 Format of Timer RJ event pin select register 0 (TRJISR0)**

Address: F0243H      After Reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

TRJISR0	0	0	0	0	0	RCCPSEL2 Note	RCCPSEL1 Note	RCCPSEL0 Note
---------	---	---	---	---	---	------------------	------------------	------------------

RCCPSEL2 Note	Timer output signal and INTP4 polarity selection		
0	An event is counted during the low-level period		
1	An event is counted during the high-level period		

RCCPSEL1 Note	RCCPSEL0 Note	Timer output signal selection
0	0	TRDIOD1
0	1	TRDI0C1
1	0	TO02
1	1	TO03

**Note** Bits RCCPSEL0 to RCCPSEL2 are enabled only in event counter mode.

### 7.3.8 Port mode registers 0, 3, 4, 5 (PM0, PM3, PM4, PM5)

These registers set input/output of ports 0, 3, 4, 5 in 1-bit units.

When using the ports (P01/TRJIO0, P30/TRJIO0, etc.) to be shared with the timer output pin for timer output, set the port mode register (PMxx) bit and port register (Pxx) bit corresponding to each port to 0.

Example: When using P01/TRJIO0 for timer output

- Set the PM01 bit of port mode register 0 to 0.
- Set the P01 bit of port register 0 to 0.

When using the ports (P01/TRJIO0, etc.) to be shared with the timer input pin for timer input, set the port mode register (PMxx) bit corresponding to each port to 1. At this time, the port register (Pxx) bit may be 0 or 1.

Example: When using P01/TRJIO0 for timer input

- Set the PM01 bit of port mode register 0 to 1.
- Set the P01 bit of port register 0 to 0 or 1.

The PM0, PM3, PM4, PM5 registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Figure 7 - 9 Format of Port Mode Registers 0, 3, 4, 5 (PM0, PM3, PM4, PM5) (100-pin products)**

Address: FFF20H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	PM04	PM03	PM02	PM01	PM00

Address: FFF23H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM3	1	1	1	1	1	1	PM31	PM30

Address: FFF24H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM4	PM47	PM46	PM45	PM44	PM43	PM42	PM41	PM40

Address: FFF25H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50

PMmn	Pmn pin I/O mode selection (m = 0, 3, 4, 5; n = 0 to 7)
0	Output mode (output buffer on)
1	Input mode (output buffer off)

**Remark** The figure shown above presents the format of port mode registers 0, 3, 4, and 5 of the 100-pin products. The format of the port mode register of other products, see **Tables 4 - 7 to 4 - 13 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product.**

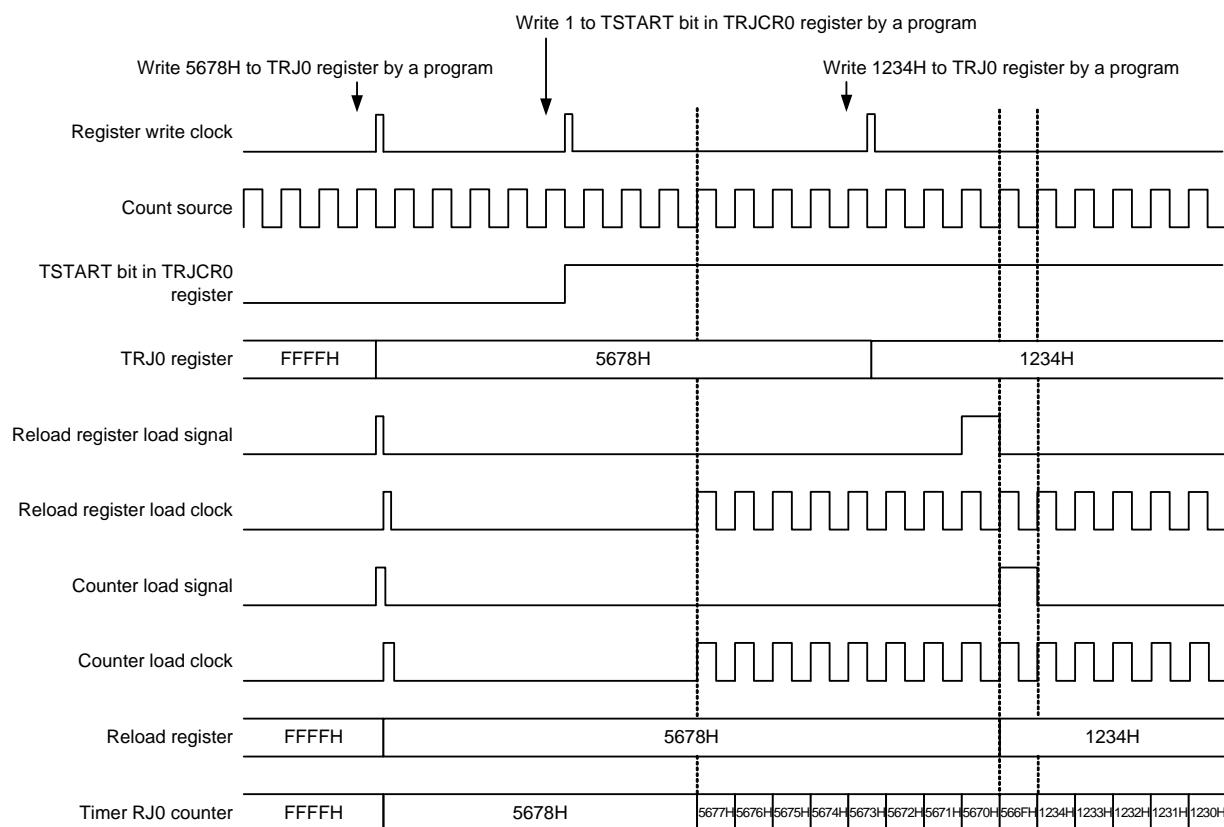
## 7.4 Timer RJ Operation

### 7.4.1 Reload Register and Counter Rewrite Operation

Regardless of the operating mode, the timing of the rewrite operation to the reload register and the counter differs depending on the value in the TSTART bit in the TRJCR0 register. When the TSTART bit is 0 (count stops), the count value is directly written to the reload register and the counter. When the TSTART bit is 1 (count starts), the value is written to the reload register in synchronization with the count source, and then to the counter in synchronization with the next count source.

Figure 7 - 10 shows the Timing of Rewrite Operation with TSTART Bit Value.

**Figure 7 - 10 Timing of Rewrite Operation with TSTART Bit Value**



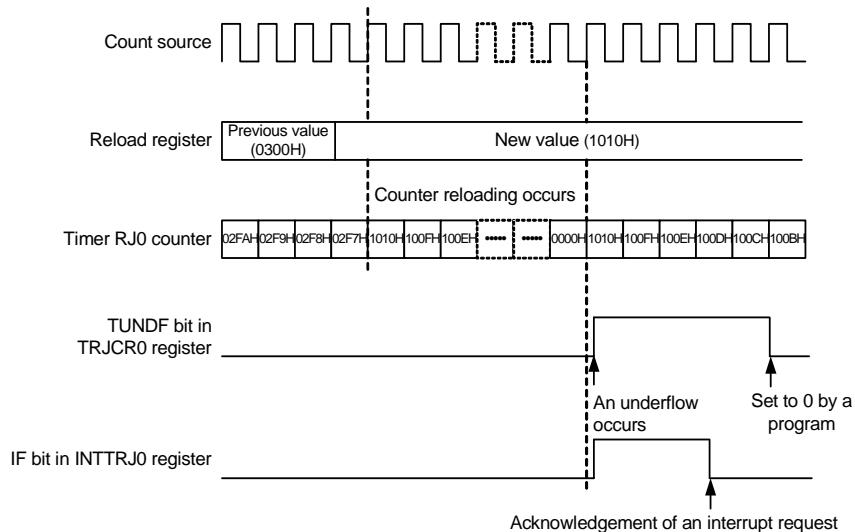
### 7.4.2 Timer Mode

In this mode, the counter is decremented by the count source selected by bits TCK0 to TCK2 in the TRJMR0 register.

In timer mode, the count value is decremented by 1 each time the count source is input. When the count value reaches 0000H and the next count source is input, an underflow occurs and an interrupt request is generated.

Figure 7 - 11 shows the Operation Example in Timer Mode.

**Figure 7 - 11 Operation Example in Timer Mode**



### 7.4.3 Pulse Output Mode

In this mode, the counter is decremented by the count source selected by bits TCK0 to TCK2 in the TRJMR0 register, and the output level of pins TRJIO and TRJO pin is inverted each time an underflow occurs.

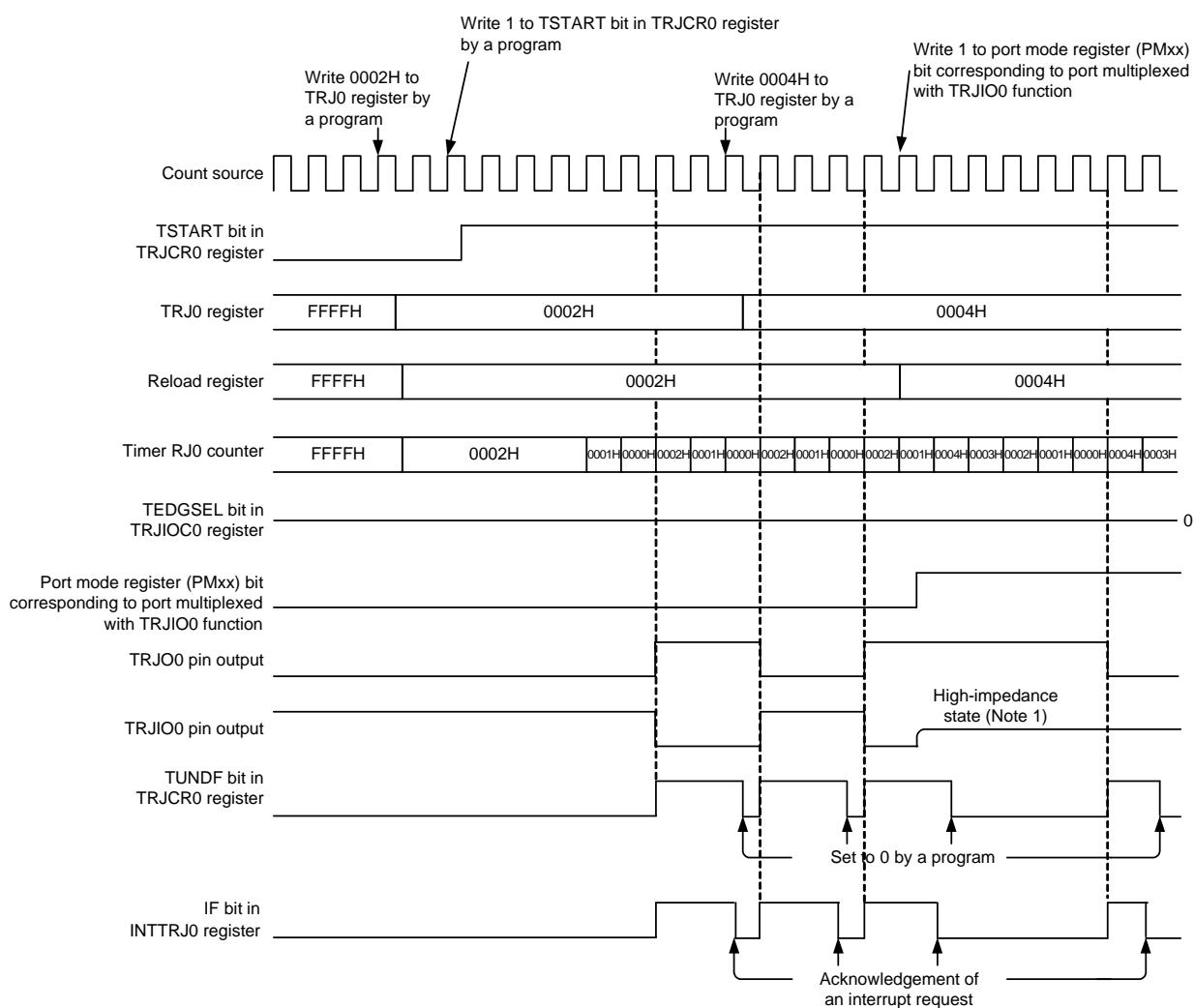
In pulse output mode, the count value is decremented by 1 each time the count source is input. When the count value reaches 0000H and the next count source is input, an underflow occurs and an interrupt request is generated.

In addition, a pulse can be output from pins TRJIO0 and TRJO0. The output level is inverted each time an underflow occurs. The pulse output from the TRJO0 pin can be stopped by the TOENA bit in the TRJIOC0 register.

Also, the output level can be selected by the TEDGSEL bit in the TRJIOC0 register.

Figure 7 - 12 shows the Operation Example in Pulse Output Mode.

**Figure 7 - 12 Operation Example in Pulse Output Mode**



### 7.4.4 Event Counter Mode

In this mode, the counter is decremented by an external event signal (count source) input to the TRJIO0 pin. Various periods for counting events can be set by bits TIOGT0 and TIOGT1 in the TRJIOC0 register and the TRJISR0 register. In addition, the filter function for the TRJIO0 input can be specified by bits TIPF0 and TIPF1 in the TRJIOC0 register.

Also, the output from the TRJIO0 pin can be toggled even in event counter mode.

When event counter mode is used, see **7.5.5 Procedure for Setting Pins TRJIO0 and TRJIO0**.

Figure 7 - 13 shows the Operation Example 1 in Event Counter Mode.

**Figure 7 - 13 Operation Example 1 in Event Counter Mode**

The diagram illustrates the operation of the Timer RJ0 in Event Counter Mode. It shows the timing sequence for entering event counter mode, setting up the timer, and handling interrupt requests.

- Registers and their initial values:**
  - Bits TMOD2 to TMOD0 in TRJMRO register: 010B
  - Control bit in TRJIOC0 register: 00H
  - TSTART bit in TRJCR0 register: (Set at the start of the event input)
  - TRJIO0 pin event input: (Pulse train starting at the rising edge of TSTART)
  - Timer RJ0 counter: FFFFH, FFFEH, FFFDH, 0000H, FFFFH, FFFEH
  - TUNDF bit in TRJCR0 register: (Set to 1 when the counter reaches 0000H)
  - IF bit in INTTRJ0 register: (Set to 1 when the counter reaches 0000H)
- Events and State Changes:**
  - "Event counter mode is entered" (arrow pointing to TMOD bits)
  - "Event is counted at rising edge" (arrow pointing to Control bit)
  - "Event input is started" (arrow pointing to TRJIO0 event input)
  - "Event input is completed" (arrow pointing to TRJIO0 event input)
  - "Counter initial value is set" (arrow pointing to Timer counter)
  - "Set to 0 by a program" (arrow pointing to Timer counter)
  - "Acknowledgement of an interrupt request" (arrow pointing to IF bit)

R01UH0186EJ0331 Rev. 3.31  
Nov 2, 2020

**RENESAS**

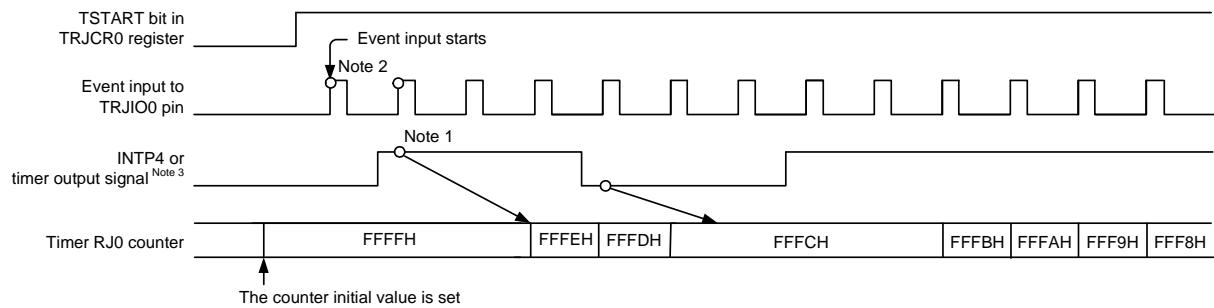
Page 408 of 1347

Figure 7 - 14 shows an operation example for counting during the specified period in event counter mode (bits TIOGT1 and TIOGT0 in the TRJIO0 register are set to 01B or 10B).

**Figure 7 - 14 Operation Example 2 in Event Counter Mode**

Timing example when the setting of operating mode is as follows:

TRJMRO register: TMOD2, 1, 0 = 010B (event counter mode)  
 TRJIOC0 register: TIOGT1, 0 = 01B (event is counted during specified period for external interrupt pin)  
 TIPF1, 0 = 00B (no filter)  
 TEDGSEL = 0 (count at rising edge)  
 TRJSR0 register: RCCPSEL2 = 1 (high-level period is counted)



The following notes apply only when bits TIOGT1 and TIOGT0 in the TRJIOC0 register are 01B or 10B for the setting of operating mode in event count mode.

- Note 1.** To control synchronization, there is a delay of two cycles of the count source until count operation is affected.
- Note 2.** Count operation may be performed for two cycles of the count source immediately after the count is started, depending on the previous state before the count is stopped.  
 To disable the count for two cycles immediately after the count is started, write 1 to the TSTOP bit in the TRJCR0 register to initialize the internal circuit, and then make operation settings before starting count operation.
- Note 3.** For the timer output signal selected by the RCCPSEL1 and RCCPSEL0 bits in the TRJSR0 register, the pin assigned to the timer output function cannot be used as the output of any multiplexed function other than the timer.

### 7.4.5 Pulse Width Measurement Mode

In this mode, the pulse width of an external signal input to the TRJIO0 pin is measured.

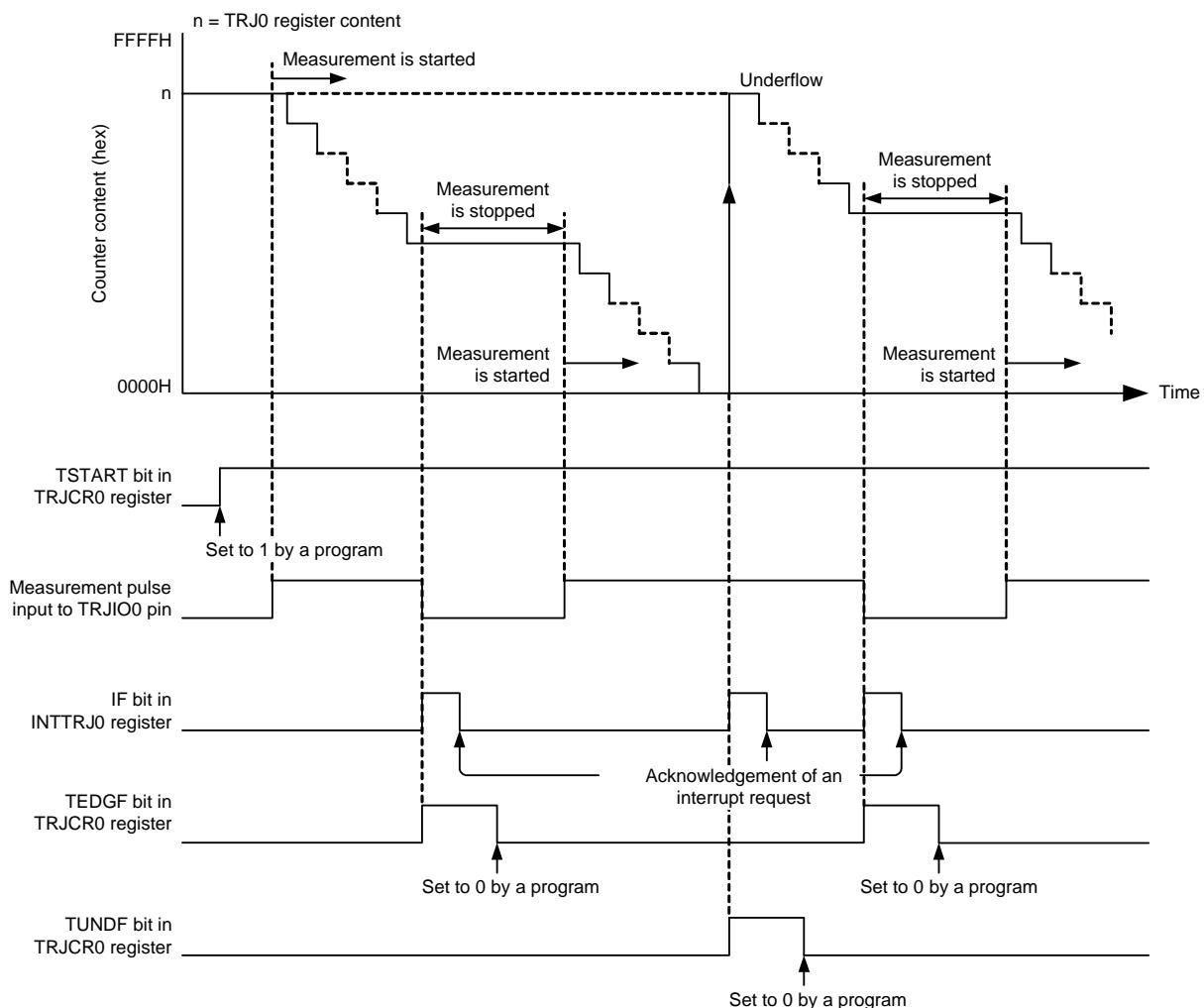
When the level specified by the TEDGSEL bit in the TRJIOC0 register is input to the TRJIO0 pin, the decrement is started with the selected count source. When the specified level on the TRJIO0 pin ends, the counter is stopped, the TEDGF bit in the TRJCR0 register is set to 1 (active edge received), and an interrupt request is generated. The measurement of pulse width data is performed by reading the count value while the counter is stopped. Also, when the counter underflows during measurement, the TUNDF bit in the TRJCR0 register is set to 1 (underflow) and an interrupt request is generated.

Figure 7 - 15 shows the Operation Example in Pulse Width Measurement Mode.

When accessing bits TEDGF and TUNDF in the TRJCR0 register, see **7.5.2 Access to Flags (Bits TEDGF and TUNDF in TRJCR0 Register)**.

**Figure 7 - 15 Operation Example in Pulse Width Measurement Mode**

This example applies when the high-level width of the measurement pulse is measured (TEDGSEL bit in TRJIOC0 register = 1)



### 7.4.6 Pulse Period Measurement Mode

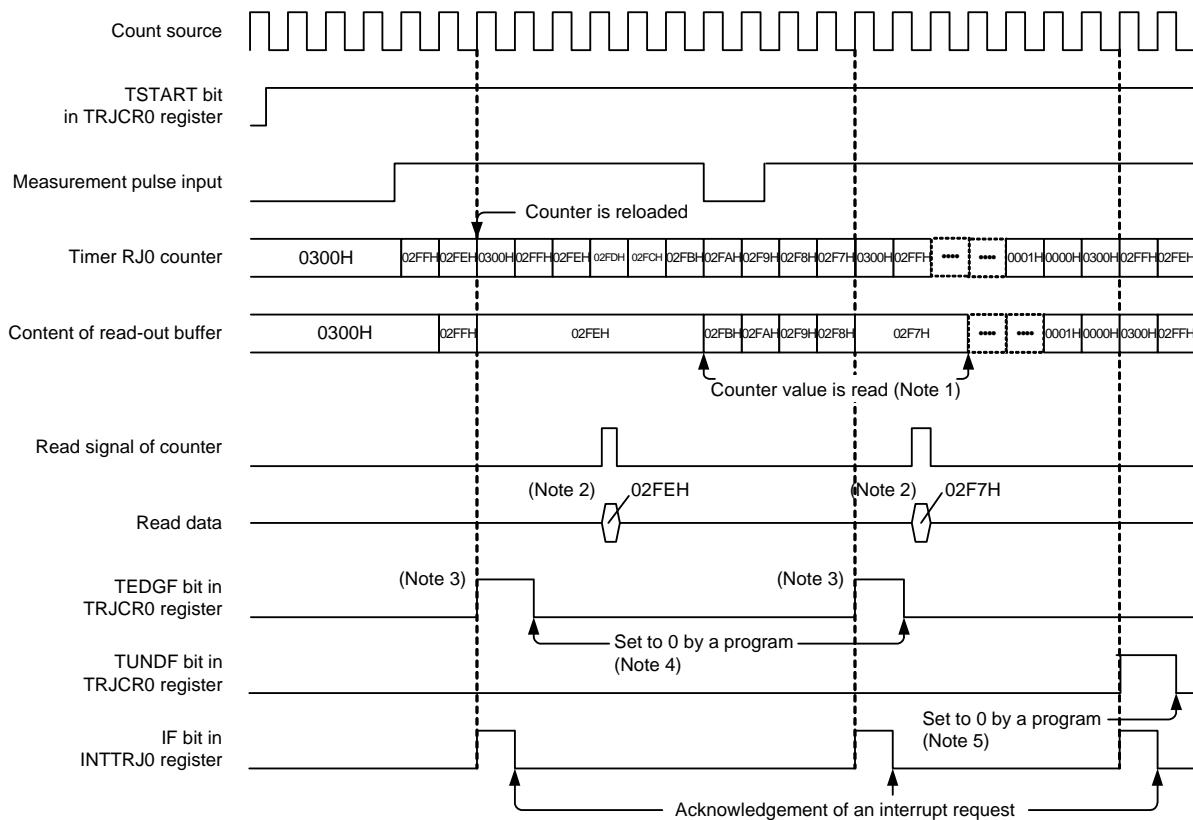
In this mode, the pulse period of an external signal input to the TRJIO0 pin is measured.

The counter is decremented by the count source selected by bits TCK0 to TCK2 in the TRJMR0 register. When a pulse with the period specified by the TEDGSEL bit in the TRJIOC0 register is input to the TRJIO0 pin, the count value is transferred to the read-out buffer at the rising edge of the count source. The value in the reload register is loaded to the counter at the next rising edge. Simultaneously, the TEDGF bit in the TRJCR0 register is set to 1 (active edge received) and an interrupt request is generated. The read-out buffer (TRJ0 register) is read at this time and the difference from the reload value is the period data of the input pulse. The period data is retained until the read-out buffer is read. When the counter underflows, the TUNDF bit in the TRJCR0 register is set to 1 (underflow) and an interrupt request is generated.

Figure 7 - 16 shows the Operation Example in Pulse Period Measurement Mode.

Only input pulses with a period longer than twice the period of the count source. Also, the low-level and high-level widths must be both longer than the period of the count source. If a pulse period shorter than these conditions is input, the input may be ignored

**Figure 7 - 16 Operation Example in Pulse Period Measurement Mode**



This example applies when the initial value of the TRJ0 register is set to 0300H, the TEDGSEL bit in the TRJIOC0 register is set to 0, and the period from one rising edge to the next edge of the measurement pulse is measured.

- Note 1.** Reading from the TRJ0 register must be performed during the period from when the TEDGF bit is set to 1 (active edge received) until the next active edge is input. The content of the read-out buffer is retained until the TRJ0 register is read. If it is not read before the active edge is input, the measurement result of the previous period is retained.
- Note 2.** When the TRJ0 register is read in pulse period measurement mode, the content of the read-out buffer is read.
- Note 3.** When the active edge of the measurement pulse is input and then the set edge of an external pulse is input, the TEDGF bit in the TRJCR0 register is set to 1 (active edge received).
- Note 4.** To set to 0 by a program, write 0 to the TEDGF bit in the TRJCR0 register using an 8-bit memory manipulation instruction.
- Note 5.** To set to 0 by a program, write 0 to the TUNDF bit in the TRJCR0 register using an 8-bit memory manipulation instruction.

### 7.4.7 Coordination with Event Link Controller (ELC)

Through coordination with the ELC, event input from the ELC can be set to be the count source. Bits TCK0 to TCK2 in the TRJMR0 register count at the rising edge of event input from the ELC. However, ELC input does not function in event counter mode.

The ELC setting procedure is shown below:

- Procedure for starting operation
  - (1) Set the event output destination select register (ELSELRn) for the ELC.
  - (2) Set the operating mode for the event generation source.
  - (3) Set the mode for timer RJ.
  - (4) Start the count operation of timer RJ.
  - (5) Start the operation of the event generation source.
- Procedure for stopping operation
  - (1) Stop the operation of the event generation source.
  - (2) Stop the count operation of timer RJ.
  - (3) Set the event output destination select register (ELSELRn) for the ELC to 0.

### 7.4.8 Output Settings for Each Mode

Tables 7 - 6 and 7 - 7 list the states of pins TRJO0 and TRJIO0 in each mode.

**Table 7 - 6 TRJO0 Pin Setting**

Operating Mode	TRJIOC0 Register		TRJO0 Pin Output
	TOENA Bit	TEDGSEL Bit	
All modes	1	1	Inverted output
		0	Normal output
	0	0 or 1	Output disabled

**Table 7 - 7 TRJIO0 Pin Setting**

Operating Mode	TRJIOC0 Register		TRJIO0 Pin I/O
	PMXX Bit Note	TEDGSEL Bit	
Timer mode	0 or 1	0 or 1	Input (Not used)
Pulse output mode	1	0 or 1	Output disabled (Hi-z output)
	0	1	Normal output
		0	Inverted output
Event counter mode	1	0 or 1	Input
Pulse width measurement mode			
Pulse period measurement mode			

**Note** The port mode register (PMxx) bit corresponding to port multiplexed with TRJIO0 function.

## 7.5 Cautions for Timer RJ

### 7.5.1 Count Operation Start and Stop Control

- When event count mode is set or the count source is set to other than the ELC

After 1 (count starts) is written to the TSTART bit in the TRJCR0 register while the count is stopped, the TCSTF bit in the TRJCR0 register remains 0 (count stops) for three cycles of the count source. Do not access the registers associated with timer RJ Note other than the TCSTF bit until this bit is set to 1 (count in progress).

After 0 (count stops) is written to the TSTART bit during a count operation, the TCSTF bit remains 1 for three cycles of the count source. When the TCSTF bit is set to 0, the count is stopped. Do not access the registers associated with timer RJ Note other than the TCSTF bit until this bit is set to 0.

Clear the interrupt register before changing the TATART bit from 0 to 1. Refer to **CHAPTER 21 INTERRUPT FUNCTIONS** for details.

**Note** Registers associated with timer RJ: TRJ0, TRJCR0, TRJIOC0, TRJMR0, and TRJISR0

- When event count mode is set or the count source is set to the ELC

After 1 (count starts) is written to the TSTART bit in the TRJCR0 register while the count is stopped, the TCSTF bit in the TRJCR0 register remains 0 (count stops) for two cycles of the CPU clock. Do not access the registers associated with timer RJ Note other than the TCSTF bit until this bit is set to 1 (count in progress).

After 0 (count stops) is written to the TSTART bit during a count operation, the TCSTF bit remains 1 for two cycles of the CPU clock. When the TCSTF bit is set to 0, the count is stopped. Do not access the registers associated with timer RJ Note other than the TCSTF bit until this bit is set to 0.

Clear the interrupt register before changing the TATART bit from 0 to 1. Refer to **CHAPTER 21 INTERRUPT FUNCTIONS** for details.

**Note** Registers associated with timer RJ: TRJ0, TRJCR0, TRJIOC0, TRJMR0, and TRJISR0

### 7.5.2 Access to Flags (Bits TEDGF and TUNDF in TRJCR0 Register)

Bits TEDGF and TUNDF in the TRJCR0 register are set to 0 by writing 0 by a program, but writing 1 to these bits has no effect. If a read-modify-write instruction is used to set the TRJCR0 register, bits TEDGF and TUNDF may be erroneously set to 0 depending on the timing, even when the TEDGF bit is set to 1 (active edge received) and the TUNDF bit is set to 1 (underflow) during execution of the instruction. Use an 8-bit memory manipulation instruction to access to the TRJCR0 register.

### 7.5.3 Access to Counter Register

When bits TSTART and TCSTF in the TRJCR0 register are both 1 (count starts), allow at least three cycles of the count source clock between writes when writing to the TRJ0 register successively.

### 7.5.4 When Changing Mode

The registers associated with timer RJ operating mode (TRJIOC0, TRJMR0, and TRJISR0) can be changed only when the count is stopped with both the TSTART and TCSTF bits set to 0 (count stops). Do not change these registers during count operation.

When the registers associated with timer RJ operating mode are changed, the values of bits TSTART and TCSTF are undefined. Write 0 (no active edge received) to the TEDGF bit and 0 (no underflow) to the TUNDF bit before starting the count.

### 7.5.5 Procedure for Setting Pins TRJO0 and TRJIO0

After a reset, the I/O ports multiplexed with pins TRJO0 and TRJIO0 function as input ports.

To output from pins TRJO0 and TRJIO0, use the following setting procedure:

Changing procedure

- (1) Set the mode.
- (2) Set the initial value/output enabled.
- (3) Set the port register bits corresponding to pins TRJO0 and TRJIO0 to 0.
- (4) Set the port mode register bits corresponding to pins TRJO0 and TRJIO0 to output mode.  
(Output is started from pins TRJO0 and TRJIO0)
- (5) Start the count (TSTART in TRJCR0 register = 1).

To input from the TRJIO0 pin, use the following setting procedure:

- (1) Set the mode.
- (2) Set the initial value/edge selected.
- (3) Set the port mode register bit corresponding to TRJIO0 pin to input mode.  
(Input is started from the TRJIO0 pin)
- (4) Start the count (TSTART in TRJMR0 register = 1).
- (5) Wait until the TCSTF bit in the TRJCR0 register is set to 1 (count in progress).  
(In event counter mode only)
- (6) Input an external event from the TRJIO0 pin.
- (7) The processing on completion of the first measurement is invalid (the measured value is valid for the second and subsequent times). (In pulse width measurement mode and pulse period measurement mode only)

### 7.5.6 When Timer RJ is not Used

When timer RJ is not used, set bits TMOD2 to TMOD0 in the TRJMR0 register to 000B (timer mode) and set the TOENA bit in the TRJIOC0 register to 0 (TRJO output disabled).

### 7.5.7 When Timer RJ Operating Clock is Stopped

Supplying or stopping the timer RJ clock can be controlled by the TRJ0EN bit in the PER1 register. Note that the following SFRs cannot be accessed while the timer RJ clock is stopped. Make sure the timer RJ clock is supplied before accessing any of these registers.

Registers TRJ0, TRJCR0, TRJMR0, TRJIOC0, and TRJISR0.

### 7.5.8 Procedure for Setting STOP Mode (Event Counter Mode)

To perform event counter mode operation during STOP mode, first supply the timer RJ clock and then use the following procedure to enter STOP mode.

Setting procedure

- (1) Set the operating mode.
- (2) Start the count ( $TSTART = 1$ ,  $TCSTF = 1$ ).
- (3) Stop supplying the timer RJ clock.

To stop event counter mode operation during STOP mode, use the following procedure to stop operation.

- (1) Supply the timer RJ clock.
- (2) Stop the count ( $TSTART = 0$ ,  $TCSTF = 0$ )

### 7.5.9 Functional Restriction in STOP Mode (Event Counter Mode Only)

When event counter mode operation is performed during STOP mode, the digital filter function cannot be used.

### 7.5.10 When Count is Forcibly Stopped by TSTOP Bit

After the counter is forcibly stopped by the TSTOP bit in the TRJCR0 register, do not access the following SFRs for one cycle of the count source.

Registers TRJ0, TRJCR0, and TRJMR0

### 7.5.11 Digital Filter

When the digital filter is used, do not start timer operation for five cycles of the digital filter clock after setting bits TIPF1 and TIPF0.

Also, do not start timer operation for five cycles of the digital filter clock when the TEDGSEL bit in the TRJIOC register is changed while the digital filter is used.

### 7.5.12 When Selecting f<sub>L</sub> as Count Source

When selecting f<sub>L</sub> as the count source, set the WUTMMCK0 bit in the subsystem clock supply mode control register (OSMC) to 1. However, f<sub>L</sub> cannot be selected as the count source for timer RJ when f<sub>SUB</sub> is selected as the count source for the real-time clock or the 12-bit interval timer.

## CHAPTER 8 TIMER RD

### 8.1 Functions of Timer RD

Timer RD has four modes:

- Timer mode
  - Input capture function Transfer the counter value to a register with an external signal as the trigger
  - Output compare function Detect register value matches with a counter (Pin output can be changed at detection)
  - PWM function Output pulse of any width continuously

The following three modes use the PWM function.

- Reset synchronous PWM mode Output three-phase waveforms (6) without sawtooth wave modulation and dead time
- Complementary PWM mode Output three-phase waveforms (6) with triangular wave modulation and dead time
- PWM3 mode Output PWM waveforms (2) with a fixed period

The timer mode input capture function, output compare function, and PWM function are equivalent in timer RD0 and timer RD1, and these functions can be selected individually for each pin. Also, a combination of these functions can be used in timer RD0 and timer RD1.

In reset synchronous PWM mode, complementary PWM mode, and PWM3 mode, a waveform is output with a combination of counters and registers in timer RD0 and timer RD1. Pin functions depend on the mode.

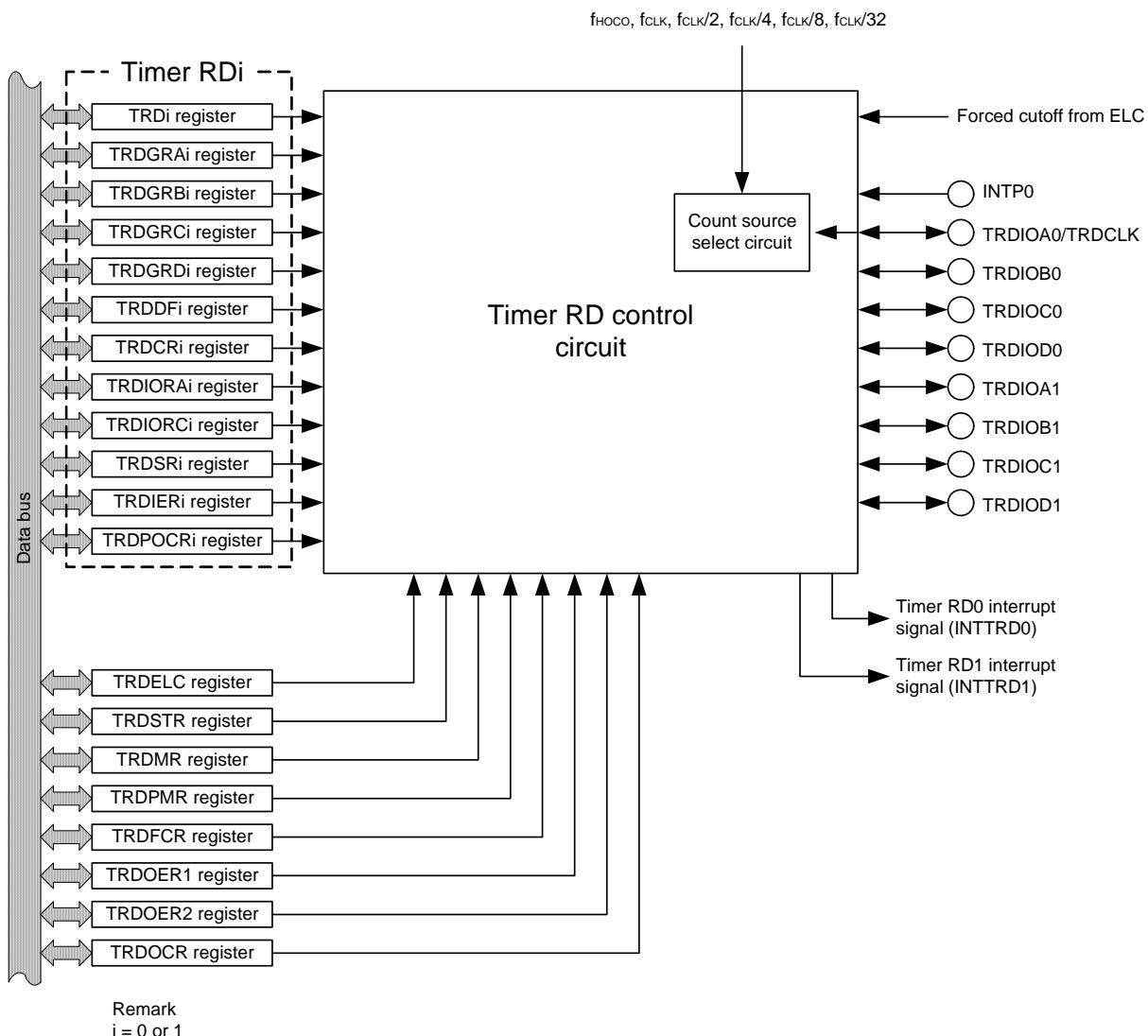
Timer RD has four I/O pins.

The operating clock for timer RD is fCLK or fHOCO.

## 8.2 Configuration of Timer RD

Figure 8 - 1 shows the Timer RD Block Diagram and Table 8 - 1 lists the Timer RD Pin Configuration.

**Figure 8 - 1 Timer RD Block Diagram**



**Table 8 - 1 Timer RD Pin Configuration**

Pin Name	Alternate Port Name	I/O	Function
TRDIOA0/TRDCLK	P17	Input/Output	Function varies depending on the mode. Refer to descriptions of individual modes for details.
TRDIOB0	P15	Input/Output	
TRDIOC0	P16	Input/Output	
TRDIOD0	P14	Input/Output	
TRDIOA1	P13	Input/Output	
TRDIOB1	P12	Input/Output	
TRDIOC1	P11	Input/Output	
TRDIOD1	P10	Input/Output	

### 8.3 Registers Controlling Timer RD

Table 8 - 2 lists the Registers Controlling Timer RD.

**Table 8 - 2 Registers Controlling Timer RD**

Register Name	Symbol
Peripheral enable register 1	PER1
Timer RD ELC register	TRDELC
Timer RD start register	TRDSTR
Timer RD mode register	TRDMR
Timer RD PWM function select register	TRDPMR
Timer RD function control register	TRDFCR
Timer RD output master enable register 1	TRDOER1
Timer RD output master enable register 2	TRDOER2
Timer RD output control register	TRDOCR
Timer RD digital filter function select register 0	TRDDF0
Timer RD digital filter function select register 1	TRDDF1
Timer RD control register 0	TRDCR0
Timer RD I/O control register A0	TRDIORA0
Timer RD I/O control register C0	TRDIORC0
Timer RD status register 0	TRDSR0
Timer RD interrupt enable register 0	TRDIER0
Timer RD PWM function output level control register 0	TRDPOCR0
Timer RD counter 0	TRD0
Timer RD general register A0	TRDGRA0
Timer RD general register B0	TRDGRB0
Timer RD general register C0	TRDGRC0
Timer RD general register D0	TRDGRD0
Timer RD control register 1	TRDCR1
Timer RD I/O control register A1	TRDIORA1
Timer RD I/O control register C1	TRDIORC1
Timer RD status register 1	TRDSR1
Timer RD interrupt enable register 1	TRDIER1
Timer RD PWM function output level control register 1	TRDPOCR1
Timer RD counter 1	TRD1
Timer RD general register A1	TRDGRA1
Timer RD general register B1	TRDGRB1
Timer RD general register C1	TRDGRC1
Timer RD general register D1	TRDGRD1
Port register 1	P1
Port mode register 1	PM1

### 8.3.1 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to the hardware that is not used is also stopped so as to decrease the power consumption and noise.

To use timer RD, be sure to set bit 4 (TRD0EN) to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 8 - 2 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After Reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN
TRD0EN	Control of timer RD input clock supply							
0	Stops input clock supply. • SFR used by timer RD cannot be written. • Timer RD is in the reset status.							
1	Enables input clock supply. • SFR used by timer RD can be read and written.							

**Note** Only for products with 96 KB or more code flash memory.

**Caution 1. When setting timer RD, be sure to set the TRD0EN bit to 1 first. If TRD0EN = 0, writing to a control register of timer RD is ignored, and all read values are default values (except for port mode register 1 (PM1), and port register 1 (P1)).**

**Caution 2. Be sure to set the following bits to 0:**

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

**Caution 3. When selecting fHOCO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.**

### 8.3.2 Timer RD ELC register (TRDELC)

Figure 8 - 3 Format of Timer RD ELC register (TRDELC)

Symbol	7	6	5	4	3	2	1	0
TRDELC	0	0	ELCOBE1	ELCICE1	0	0	ELCOBE0	ELCICE0
ELCOBE1	ELC event input 1 enable for timer RD pulse output forced cutoff							
0	Forced cutoff is disabled							
1	Forced cutoff is enabled							
ELCICE1	ELC event input 1 select for timer RD input capture D1							
0	TRDIOD1 input capture is selected							
1	Event input 1 from the event link controller (ELC) is selected							
ELCOBE0	ELC event input 0 enable for timer RD pulse output forced cutoff							
0	Forced cutoff is disabled							
1	Forced cutoff is enabled							
ELCICE0	ELC event input 0 select for timer RD input capture D0							
0	TRDIOD0 input capture is selected							
1	Event input 0 from the event link controller (ELC) is selected							

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.3 Timer RD start register (TRDSTR)

The TRDSTR register can be set by an 8-bit memory manipulation instruction. See 8.7.1 (1) TRDSTR Register.

**Figure 8 - 4 Format of Timer RD start register (TRDSTR)**

Symbol	7	6	5	4	3	2	1	0
TRDSTR	0	0	0	0	CSEL1	CSEL0	TSTART1	TSTART0
<b>CSEL1</b>								
TRD1 count operation select Note 2								
0 Count stops at compare match with TRDGRA1 register								
1 Count continues after compare match with TRDGRA1 register Note 3								
<b>CSEL0</b>								
TRD0 count operation select								
0 Count stops at compare match with TRDGRA0 register								
1 Count continues after compare match with TRDGRA0 register Note 3								
<b>TSTART1</b>								
TRD1 count start flag Notes 4, 5								
0 Count stops								
1 Count starts								
<b>TSTART0</b>								
TRD0 count start flag Notes 6, 7								
0 Count stops								
1 Count starts								

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** Do not use in PWM3 mode.

**Note 3.** Set to 1 for the input capture function.

**Note 4.** Write 0 to the TSTART1 bit while the CSEL1 bit is set to 1.

**Note 5.** When the CSEL1 bit is 0 and a compare match signal (TRDIOA1) is generated, this flag is set to 0 (count stops).

**Note 6.** Write 0 to the TSTART0 bit while the CSEL0 bit is set to 1.

**Note 7.** When the CSEL0 bit is 0 and a compare match signal (TRDIOA0) is generated, this flag is set to 0 (count stops).

### 8.3.4 Timer RD mode register (TRDMR)

**Figure 8 - 5 Format of Timer RD mode register (TRDMR)**

Address: F0264H		After Reset: 00H Note 1		R/W												
Symbol	<7>	<6>	<5>	<4>	3	2	1	<0>								
TRDMR	TRDBFD1	TRDBFC1	TRDBFD0	TRDBFC0	0	0	0	TRDSYNC								
TRDBFD1	TRDGRD1 register function select Note 2															
0	General register															
1	Buffer register for TRDGRB1 register															
TRDBFC1	TRDGRC1 register function select Note 2															
0	General register															
1	Buffer register for TRDGRA1 register															
TRDBFD0	TRDGRD0 register function select Note 2															
0	General register															
1	Buffer register for TRDGRB0 register															
TRDBFC0	TRDGRC0 register function select Notes 2, 3															
0	General register															
1	Buffer register for TRDGRA0 register															
TRDSYNC	Timer RD synchronous Note 4															
0	TRD0 and TRD1 operate independently															
1	TRD0 and TRD1 operate synchronously															

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** In the output compare function, if 0 (TRDGRji register output pin is changed) is selected for the IOj3 (j = C or D) bit in the TRDIORCi (i = 0 or 1) register, set the TRDBFji bit in the TRDMR register to 0.
- Note 3.** Set to 0 (general register) in complementary PWM mode.
- Note 4.** Set to 0 (TRD0 and TRD1 operate independently) in reset synchronous PWM mode, complementary PWM mode, and PWM3 mode.

### 8.3.5 Timer RD PWM function select register (TRDPMR)

**Figure 8 - 6 Format of Timer RD PWM function select register (TRDPMR) [Timer Mode]**

Symbol	7	<6>	<5>	<4>	3	<2>	<1>	<0>
TRDPMR	0	TRDPWMD1	TRDPWMC1	TRDPWMB1	0	TRDPWMD0	TRDPWMC0	TRDPWMB0
TRDPWMD1	PWM function of TRDIOD1 select							
0	Input capture function or output compare function							
1	PWM function							
TRDPWMC1	PWM function of TRDIOC1 select							
0	Input capture function or output compare function							
1	PWM function							
TRDPWMB1	PWM function of TRDIOB1 select							
0	Input capture function or output compare function							
1	PWM function							
TRDPWMD0	PWM function of TRDIOD0 select							
0	Input capture function or output compare function							
1	PWM function							
TRDPWMC0	PWM function of TRDIOC0 select							
0	Input capture function or output compare function							
1	PWM function							
TRDPWMB0	PWM function of TRDIOB0 select							
0	Input capture function or output compare function							
1	PWM function							

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.6 Timer RD function control register (TRDFCR)

Figure 8 - 7 Format of Timer RD function control register (TRDFCR)

Symbol	7	6	5	4	3	2	1	0	
TRDFCR	PWM3	STCLK	0	0	OLS1	OLS0	CMD1	CMD0	
PWM3		PWM3 mode select Note 2							
<ul style="list-style-type: none"> <li>In the timer mode, set to 1 (other than PWM3 mode).</li> <li>In PWM3 mode, set to 0 (PWM3 mode).</li> <li>Disabled in reset synchronous and complementary PWM modes.</li> </ul>									
STCLK		External clock input select							
<ul style="list-style-type: none"> <li>In the timer mode, the reset synchronous PWM mode, and the complementary PWM mode,</li> <li>0: External clock input disabled</li> <li>1: External clock input enabled</li> <li>In PWM3 mode, set to 0 (external clock input disabled).</li> </ul>									
OLS1		Counter-phase output level select (in reset synchronous PWM mode or complementary PWM mode)							
<ul style="list-style-type: none"> <li>In reset synchronous and complementary PWM modes,</li> <li>0: High initial output and low active level</li> <li>1: Low initial output and high active level</li> <li>Disabled in timer and PWM3 modes.</li> </ul>									
OLS0		Phase output level select (in reset synchronous PWM mode or complementary PWM mode)							
<ul style="list-style-type: none"> <li>In reset synchronous and complementary PWM modes,</li> <li>0: High initial output and low active level</li> <li>1: Low initial output and high active level</li> <li>Disabled in timer and PWM3 modes.</li> </ul>									
CMD1		CMD0	Combination mode select Notes 3, 4						
<ul style="list-style-type: none"> <li>In timer and PWM3 modes, set to 00B (timer mode or PWM3 mode).</li> <li>In reset synchronous PWM mode, set to 01B (reset synchronous PWM mode).</li> <li>In complementary PWM mode,</li> </ul>									
CMD1    CMD0 1      0: Complementary PWM mode (transfer from the buffer register to the general register when TRD1 underflows) 1      1: Complementary PWM mode (transfer from the buffer register to the general register at compare match between registers TRD0 and TRDGRA0)									
Other than the above: Setting prohibited.									

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** When bits CMD1 and CMD0 are set to 00B (timer mode or PWM3 mode), the setting of the PWM3 bit is enabled.
- Note 3.** Set bits CMD0 and CMD1 when both the TSTART0 and TSTART1 bits in the TRDSTR register are set to 0 (count stops).
- Note 4.** When bits CMD1 and CMD0 are set to 01B, 10B, or 11B, the MCU enters reset synchronous PWM mode or complementary PWM mode regardless of the settings of the TRDPMR register.

### 8.3.7 Timer RD output master enable register 1 (TRDOER1)

**Figure 8 - 8 Format of Timer RD output master enable register 1 (TRDOER1)  
[Output Compare Function, PWM Function, Reset Synchronous PWM Mode,  
Complementary PWM Mode, and PWM3 Mode]**

Symbol	7	6	5	4	3	2	1	0
TRDOER1	ED1	EC1	EB1	EA1	ED0	EC0	EB0	EA0
ED1	TRDIOD1 output disable Note 2							
0	Output enabled							
1	Output disabled (TRDIOD1 pin functions as an I/O port.)							
EC1	TRDIOC1 output disable Note 2							
0	Output enabled							
1	Output disabled (TRDIOC1 pin functions as an I/O port.)							
EB1	TRDIOB1 output disable Note 2							
0	Output enabled							
1	Output disabled (TRDIOB1 pin functions as an I/O port.)							
EA1	TRDIOA1 output disable Notes 2, 3							
0	Output enabled							
1	Output disabled (TRDIOA1 pin functions as an I/O port)							
ED0	TRDIOD0 output disable Note 2							
0	Output enabled							
1	Output disabled (TRDIOD0 pin functions as an I/O port.)							
EC0	TRDIOC0 output disable Note 2							
0	Output enabled							
1	Output disabled (TRDIOC0 pin functions as an I/O port.)							
EB0	TRDIOB0 output disable							
0	Output enabled							
1	Output disabled (TRDIOB0 pin functions as an I/O port.)							
EA0	TRDIOA0 output disable Notes 3, 4							
0	Output enabled							
1	Output disabled (TRDIOA0 pin functions as an I/O port)							

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRDOEN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** Set to 1 in PWM3 mode.
- Note 3.** Set to 1 in PWM function.
- Note 4.** Set to 1 in reset synchronous PWM mode and complementary PWM mode.

### 8.3.8 Timer RD output master enable register 2 (TRDOER2)

**Figure 8 - 9 Format of Timer RD output master enable register 2 (TRDOER2)  
[PWM Function, Reset Synchronous PWM Mode, Complementary PWM Mode, and PWM3 Mode]**

Address: F0268H		After Reset: 00H Note 1						R/W
Symbol	<7>	6	5	4	3	2	1	<0>
TRDOER2	TRDPTO	0	0	0	0	0	0	TRDSHUTS
TRDPTO		INTP0 pin of pulse output forced cutoff signal input enabled Note 2						
0		Pulse output forced cutoff input disabled						
1		Pulse output forced cutoff input enabled (The TRDSHUTS bit is set to 1 when a low level is applied to the INTP0 pin.)						
TRDSHUTS		Forced cutoff flag						
0		Not forcibly cut off						
1		Forcibly cut off						
This bit is set to 1 when the pulse is forcibly cut off by an INTP0 pin or ELC input event. This bit is not automatically cleared. To stop the forced cutoff of the pulse, write 0 to this bit while the count is stopped (TSTARTi = 0). The pulse is also forcibly cut off when 1 is written to the TRDSHUTS bit in an enabled mode.								

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** See 8.4.4 Pulse Output Forced Cutoff.

### 8.3.9 Timer RD output control register (TRDOCR)

Write to the TRDOCR register when bits TSTART0 and TSTART1 in the TRDSTR register are both 0 (count stops).

**Figure 8 - 10 Format of Timer RD output control register (TRDOCR) [Output Compare Function]**

Address: F0269H After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
TRDOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
TOD1	TRDIOD1 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOC1	TRDIOC1 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOB1	TRDIOB1 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOA1	TRDIOA1 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOD0	TRDIOD0 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOC0	TRDIOC0 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOB0	TRDIOB0 initial output level select Note 2							
0	Low initial output							
1	High initial output							
TOA0	TRDIOA0 initial output level select Note 2							
0	Low initial output							
1	High initial output							

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If the pin function is set for waveform output, the initial output level is output when the TRDOCR register is set.

**Figure 8 - 11 Format of Timer RD output control register (TRDOCR) [PWM Function]**

Address: F0269H After Reset: 00H Note 1 R/W

Symbol 7 6 5 4 3 2 1 0

TRDOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
--------	------	------	------	------	------	------	------	------

TOD1	TRDIOD1 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOC1	TRDIOC1 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOB1	TRDIOB1 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOA1	TRDIOA1 initial output level select Note 2
Set to 0.	

TOD0	TRDIOD0 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOC0	TRDIOC0 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOB0	TRDIOB0 initial output level select Note 2
0	Initial output is not active level
1	Initial output is active level

TOA0	TRDIOA0 initial output level select Note 2
Set to 0.	

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If the pin function is set for waveform output, the initial output level is output when the TRDOCR register is set.

**Figure 8 - 12 Format of Timer RD output control register (TRDOCR)**  
**[Reset Synchronous PWM Mode, Complementary PWM Mode]**

Symbol	7	6	5	4	3	2	1	0	
TRDOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0	
TOD1, TOC1, TOB1, TOA1, TOD0, TOB0, TOA0		<p>Setting these bits to 1 is invalid in the reset synchronous PWM mode and complementary PWM mode.          Be sure to set these bits to 0.</p> <p>In the reset synchronous PWM mode and complementary PWM mode, the setting of the OLS1 and OLS0 bits in TRDFCR determine the initial level independently of the setting in these bits.</p>							
TOC0		TRDIOC0 initial output level select <sup>Note 2</sup>							
0 Initial output L		In the reset synchronous PWM mode, the output is inverted every PWM period.							
1 Initial output H		In complementary PWM mode, the output is inverted every 1/2 PWM period.							

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If the pin function is set for waveform output, the initial output level is output when the TRDOCR register is set.

**Figure 8 - 13 Format of Timer RD output control register (TRDOCR) [PWM3 Mode]**

Symbol	7	6	5	4	3	2	1	0
TRDOCR	TOD1	TOC1	TOB1	TOA1	TOD0	TOC0	TOB0	TOA0
TOD1	TRDIOD1 initial output level select							
Disabled in PWM3 mode.								
TOC1	TRDILOC1 initial output level select							
Disabled in PWM3 mode.								
TOB1	TRDIOB1 initial output level select							
Disabled in PWM3 mode.								
TOA1	TRDIOA1 initial output level select							
Disabled in PWM3 mode.								
TOD0	TRDIOD0 initial output level select							
Disabled in PWM3 mode.								
TOC0	TRDILOC0 initial output level select							
Disabled in PWM3 mode.								
TOB0	TRDIOB0 initial output level select Note 2							
0	Low initial output, high active level, high output at TRDGRB1 compare match, and low output at TRDGRB0 compare match							
1	High initial output, low active level, low output at TRDGRB1 compare match, and high output at TRDGRB0 compare match							
TOA0	TRDIOA0 initial output level select Note 2							
0	Low initial output, high active level, high output at TRDGRA1 compare match, and low output at TRDGRA0 compare match							
1	High initial output, low active level, low output at TRDGRA1 compare match, and high output at TRDGRA0 compare match							

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If the pin function is set for waveform output, the initial output level is output when the TRDOCR register is set.

### 8.3.10 Timer RD digital filter function select register i (TRDDFi) (i = 0 or 1)

**Figure 8 - 14 Format of Timer RD digital filter function select register i (TRDDFi) (i = 0 or 1)  
[Input Capture Function]**

Address: F026AH (TRDDF0), F026BH (TRDDF1) After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

TRDDFi	DFCK1	DFCK0	PENB1	PENB0	DFD	DFC	DFB	DFA
--------	-------	-------	-------	-------	-----	-----	-----	-----

DFCK1	DFCK0	Clock select for digital filter function Note 2					
0	0	fCLK/32 Note 3					
0	1	fCLK/8 Note 3					
1	0	fCLK Note 3					
1	1	Count source (clock selected by bits TCK0 to TCK2 in the TRDCRi register)					

PENB1	PENB0	TRDIOB pin pulse forced cutoff control					
0	0	Set to 00B.					

DFD	TRDIODi pin digital filter function select						
0	Digital filter function disabled						
1	Digital filter function enabled						

When the digital filter is enabled, edge detection is performed after up to five cycles of the digital filter sampling clock.

DFC	TRDIOCi pin digital filter function select						
0	Digital filter function disabled						
1	Digital filter function enabled						

When the digital filter is enabled, edge detection is performed after up to five cycles of the digital filter sampling clock.

DFB	TRDIOBi pin digital filter function select						
0	Digital filter function disabled						
1	Digital filter function enabled						

When the digital filter is enabled, edge detection is performed after up to five cycles of the digital filter sampling clock.

DFA	TRDIOAi pin digital filter function select						
0	Digital filter function disabled						
1	Digital filter function enabled						

When the digital filter is enabled, edge detection is performed after up to five cycles of the digital filter sampling clock.

(Notes are listed on the next page.)

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** Set bits DFCK0 and DFCK1 before starting count operation.
- Note 3.** When FRQSEL4 = 1 in the user option byte (000C2H), fCLK/32, fCLK/8, and fCLK are set to fHO/32, fHO/8, and fHO, respectively.

**Figure 8 - 15 Format of Timer RD digital filter function select register i (TRDDFi) (i = 0 or 1)  
[PWM Function, Reset Synchronous PWM Mode, Complementary PWM Mode, and PWM3 Mode]**

Symbol	7	6	5	4	3	2	1	0	After Reset: 00H Note R/W
TRDDFi	DFCK1	DFCK0	PENB1	PENB0	DFD	DFC	DFB	DFA	
TRDIOA pin pulse forced cutoff control									
	DFCK1	DFCK0							
	0	0	Forced cutoff disabled						
	0	1	High-impedance output						
	1	0	Low output						
	1	1	High output						
Set these bits to 00B (forced cutoff disabled) if the corresponding pin is not used as a timer RD output port in these modes. Also, set these bits while the count is stopped.									
TRDIOB pin pulse forced cutoff control									
	PENB1	PENB0							
	0	0	Forced cutoff disabled						
	0	1	High-impedance output						
	1	0	Low output						
	1	1	High output						
Set these bits to 00B (forced cutoff disabled) if the corresponding pin is not used as a timer RD output port in these modes. Also, set these bits while the count is stopped.									
TRDIOC pin pulse forced cutoff control									
	DFD	DFC							
	0	0	Forced cutoff disabled						
	0	1	High-impedance output						
	1	0	Low output						
	1	1	High output						
Set these bits to 00B (forced cutoff disabled) if the corresponding pin is not used as a timer RD output port in these modes. Also, set these bits while the count is stopped.									
TRDIOD pin pulse forced cutoff control									
	DFB	DFA							
	0	0	Forced cutoff disabled						
	0	1	High-impedance output						
	1	0	Low output						
	1	1	High output						
Set these bits to 00B (forced cutoff disabled) if the corresponding pin is not used as a timer RD output port in these modes. Also, set these bits while the count is stopped.									

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.11 Timer RD control register i (TRDCR<sub>i</sub>) ( $i = 0$ or $1$ )

The TRDCR1 register is not used in reset synchronous PWM mode or PWM3 mode.

**Figure 8 - 16 Format of Timer RD control register i (TRDCR<sub>i</sub>) ( $i = 0$  or  $1$ )  
[Input Capture Function and Output Compare Function]**

Address: F0270H (TRDCR0), F0280H (TRDCR1) After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0						
TRDCR <sub>i</sub>	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TCK2	TCK1	TCK0						
CCLR2 CCLR1 CCLR0 TRDi counter clear select														
0	0	0	Clear disabled (free-running operation)											
0	0	1	Clear by input capture/compare match with TRDGRA <sub>i</sub>											
0	1	0	Clear by input capture/compare match with TRDGRB <sub>i</sub>											
0	1	1	Synchronous clear (clear simultaneously with other timer RD <sub>i</sub> counter) Note 2											
1	0	1	Clear by input capture/compare match with TRDGRC <sub>i</sub>											
1	1	0	Clear by input capture/compare match with TRDGRD <sub>i</sub>											
Other than above			Setting prohibited											
CKEG1 CKEG0 External clock edge select Note 3														
0	0	Count at the rising edge												
0	1	Count at the falling edge												
1	0	Count at both edges												
Other than above		Setting prohibited												
TCK2 TCK1 TCK0 Count source select														
0	0	0	fCLK, fHO <sub>O</sub> Note 4											
0	0	1	fCLK/2 Note 5											
0	1	0	fCLK/4 Note 5											
0	1	1	fCLK/8 Note 5											
1	0	0	fCLK/32 Note 5											
1	0	1	TRDCLK input Note 6											
Other than above			Setting prohibited											

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** Enabled when the TRDSYNC bit in the TRDMR register is 1 (TRD0 and TRD1 operate synchronously).
- Note 3.** Valid when bits TCK2 to TCK0 are set to 101B (TRDCLK input) and the STCLK bit is set to 1 (external clock input enabled).
- Note 4.** fCLK is selected when FRQSEL4 = 0 and fHO<sub>O</sub> is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO<sub>O</sub> as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.
- Note 5.** Do not set this value when FRQSEL4 = 1 in the user option byte (000C2H).
- Note 6.** Valid when the STCLK bit in the TRDFCR register is set to 1 (external clock input enabled).

**Figure 8 - 17 Format of Timer RD control register i (TRDCR*i*) (*i* = 0 or 1) [PWM Mode]**

Address: F0270H (TRDCR0), F0280H (TRDCR1) After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
TRDCR <i>i</i>	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TCK2	TCK1	TCK0
CCLR2 CCLR1 CCLR0 TRDi counter clear select								
Set to 001B (TRDi register is cleared at compare match with TRDGRA <i>i</i> register).								
CKEG1	CKEG0	External clock edge select Note 2						
0	0	Count at the rising edge						
0	1	Count at the falling edge						
1	0	Count at both edges						
Other than above		Setting prohibited						
TCK2	TCK1	TCK0	Count source select					
0	0	0	fCLK, fHO CO Note 3					
0	0	1	fCLK/2 Note 4					
0	1	0	fCLK/4 Note 4					
0	1	1	fCLK/8 Note 4					
1	0	0	fCLK/32 Note 4					
1	0	1	TRDCLK input Note 5					
Other than above			Setting prohibited					

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** Valid when bits TCK2 to TCK0 are set to 101B (TRDCLK input) and the STCLK bit is set to 1 (external clock input enabled).

**Note 3.** fCLK is selected when FRQSEL4 = 0 and fHO CO is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Note 4.** Do not set this value when FRQSEL4 = 1 in the user option byte (000C2H).

**Note 5.** Valid when the STCLK bit in the TRDFCR register is set to 1 (external clock input enabled).

**Figure 8 - 18 Format of Timer RD control register 0 (TRDCR0) [Reset Synchronous PWM Mode]**

Address: F0270H After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

TRDCR0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TCK2	TCK1	TCK0
--------	-------	-------	-------	-------	-------	------	------	------

CCLR2	CCLR1	CCLR0	TRD0 counter clear select					
Set to 001B (TRD0 register is cleared at compare match with TRDGRA0 register).								

CKEG1	CKEG0	External clock edge select Note 2						
0	0	Count at the rising edge						
0	1	Count at the falling edge						
1	0	Count at both edges						
Other than above		Setting prohibited						

TCK2	TCK1	TCK0	Count source select					
0	0	0	fCLK, fHO CO Note 3					
0	0	1	fCLK/2 Note 4					
0	1	0	fCLK/4 Note 4					
0	1	1	fCLK/8 Note 4					
1	0	0	fCLK/32 Note 4					
1	0	1	TRDCLK input Note 5					
Other than above			Setting prohibited					

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** Valid when bits TCK2 to TCK0 are set to 101B (TRDCLK input) and the STCLK bit is set to 1 (external clock input enabled).

**Note 3.** fCLK is selected when FRQSEL4 = 0 and fHO CO is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Note 4.** Do not set this value when FRQSEL4 = 1 in the user option byte (000C2H).

**Note 5.** Valid when the STCLK bit in the TRDFCR register is set to 1 (external clock input enabled).

**Figure 8 - 19 Format of Timer RD control register i (TRDCR*i*)(*i* = 0,1)[Complementary PWM Mode]**

Address: F0270H (TRDCR0), F0280H (TRDCR1) After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
TRDCR <i>i</i>	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TCK2	TCK1	TCK0
CCLR2 CCLR1 CCLR0 TRD0 counter clear select								
Set to 000B (clear disabled (free-running operation)).								
CKEG1	CKEG0	External clock edge select Notes 2, 3						
0	0	Count at the rising edge						
0	1	Count at the falling edge						
1	0	Count at both edges						
Other than above		Setting prohibited						
TCK2	TCK1	TCK0	Count source select					
0	0	0	fCLK, fHO CO Note 4					
0	0	1	fCLK/2 Note 5					
0	1	0	fCLK/4 Note 5					
0	1	1	fCLK/8 Note 5					
1	0	0	fCLK/32 Note 5					
1	0	1	TRDCLK input Note 6					
Other than above			Setting prohibited					

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** Valid when bits TCK2 to TCK0 are set to 101B (TRDCLK input) and the STCLK bit is set to 1 (external clock input enabled).

**Note 3.** Set the same value to bits TCK0 to TCK2, CKEG0, and CKEG1 in registers TRDCR0 and TRDCR1.

**Note 4.** fCLK is selected when FRQSEL4 = 0 and fHO CO is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Note 5.** Do not set this value when FRQSEL4 = 1 in the user option byte (000C2H).

**Note 6.** Valid when the STCLK bit in the TRDFCR register is set to 1 (external clock input enabled).

**Figure 8 - 20 Format of Timer RD control register 0 (TRDCR0) [PWM3 Mode]**

Address: F0270H After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
TRDCR0	CCLR2	CCLR1	CCLR0	CKEG1	CKEG0	TCK2	TCK1	TCK0
CCLR2 CCLR1 CCLR0 TRD0 counter clear select								
Set to 001B (TRD0 register is cleared at compare match with TRDGRA0 register).								
CKEG1 CKEG0 External clock edge select								
Disabled in PWM3 mode.								
TCK2 TCK1 TCK0 Count source select								
0 0 0 fCLK, fHO CO Note 2								
0 0 1 fCLK/2 Note 3								
0 1 0 fCLK/4 Note 3								
0 1 1 fCLK/8 Note 3								
1 0 0 fCLK/32 Note 3								
Other than above			Setting prohibited					

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** fCLK is selected when FRQSEL4 = 0 and fHO CO is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source, select fIH as fCLK before starting timer count operation.

**Note 3.** Do not set this value when FRQSEL4 = 1 in the user option byte (000C2H).

### 8.3.12 Timer RD I/O control register Ai (TRDIORA<sub>i</sub>) ( $i = 0$ or $1$ )

**Figure 8 - 21 Format of Timer RD I/O control register Ai (TRDIORA<sub>i</sub>) ( $i = 0$  or  $1$ ) [Input Capture Function]**

Symbol	7	6	5	4	3	2	1	0									
TRDIORA <sub>i</sub>	0	IOB2	IOB1	IOB0	0	IOA2	IOA1	IOA0									
IOB2	TRDGRB mode select Note 2																
Set to 1 (input capture) in the input capture function.																	
IOB1	IOB0	TRDGRB control															
0	0	Input capture to TRDGRBi at the rising edge															
0	1	Input capture to TRDGRBi at the falling edge															
1	0	Input capture to TRDGRBi at both edges															
Other than above	Setting prohibited																
IOA2	TRDGRA mode select Note 3																
Set to 1 (input capture) in the input capture function.																	
IOA1	IOA0	TRDGRA control															
0	0	Input capture to TRDGRAi at the rising edge															
0	1	Input capture to TRDGRAi at the falling edge															
1	0	Input capture to TRDGRAi at both edges															
Other than above	Setting prohibited																

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRDOEN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If 1 (buffer register for TRDGRBi register) is selected for the TRDBFD<sub>i</sub> bit in the TRDMR register, set the same value to the IOB2 bit in the TRDIORA<sub>i</sub> register and the IOC2 bit in the TRDIORCi register.

**Note 3.** If 1 (buffer register for TRDGRAi register) is selected for the TRDBFC<sub>i</sub> bit in the TRDMR register, set the same value to the IOA2 bit in the TRDIORA<sub>i</sub> register and the IOC2 bit in the TRDIORCi register.

**Figure 8 - 22 Format of Timer RD I/O control register Ai (TRDIORAi) (i = 0 or 1) [Output Compare Function]**

Address: F0271H (TRDIORA0), F0281H (TRDIORA1) After Reset: 00H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0						
TRDIORAi	0	IOB2	IOB1	IOB0	0	IOA2	IOA1	IOA0						
IOB2	TRDGRB mode select Note 2													
Set to 0 (output compare) in the output compare function.														
IOB1	IOB0	TRDGRB control												
0	0	Pin output by compare match is disabled (TRDIOBi pin functions as an I/O port)												
0	1	Low output by compare match with TRDGRBi												
1	0	High output by compare match with TRDGRBi												
1	1	Toggle output by compare match with TRDGRBi												
IOA2	TRDGRA mode select Note 3													
Set to 0 (output compare) in the output compare function.														
IOA1	IOA0	TRDGRA control												
0	0	Pin output by compare match is disabled (TRDIOAi pin functions as an I/O port)												
0	1	Low output by compare match with TRDGRAi												
1	0	High output by compare match with TRDGRAi												
1	1	Toggle output by compare match with TRDGRAi												

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If 1 (buffer register for TRDGRBi register) is selected for the TRDBFDi bit in the TRDMR register, set the same value to the IOB2 bit in the TRDIORAi register and the IOC2 bit in the TRDIORCi register.

**Note 3.** If 1 (buffer register for TRDGRAi register) is selected for the TRDBFCi bit in the TRDMR register, set the same value to the IOA2 bit in the TRDIORAi register and the IOC2 bit in the TRDIORCi register.

### 8.3.13 Timer RD I/O control register Ci (TRDIORCi) (i = 0 or 1)

**Figure 8 - 23 Format of Timer RD I/O control register Ci (TRDIORCi) [Input Capture Function]**

Address: F0272H (TRDIORC0), F0282H (TRDIORC1) After Reset: 88H Note 1 R/W

Symbol	7	6	5	4	3	2	1	0
TRDIORCi	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0
IOD3	TRDGRD register function select							
Set to 1 (general register or buffer register) in the input capture function.								
IOD2	TRDGRD mode select Note 2							
Set to 1 (input capture) in the input capture function.								
IOD1	IOD0	TRDGRD control						
0	0	Input capture to TRDGRDi at the rising edge						
0	1	Input capture to TRDGRDi at the falling edge						
1	0	Input capture to TRDGRDi at both edges						
Other than above	Setting prohibited							
IOC3	TRDGRC register function select							
Set to 1 (general register or buffer register) in the input capture function.								
IOC2	TRDGRC mode select Note 3							
Set to 1 (input capture) in the input capture function.								
IOC1	IOC0	TRDGRC control						
0	0	Input capture to TRDGRCi at the rising edge						
0	1	Input capture to TRDGRCi at the falling edge						
1	0	Input capture to TRDGRCi at both edges						
Other than above	Setting prohibited							

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** If 1 (buffer register for TRDGRBi register) is selected for the TRDBFDi bit in the TRDMR register, set the same value to the IOB2 bit in the TRDIORAi register and the IOD2 bit in the TRDIORCi register.

**Note 3.** If 1 (buffer register for TRDGRAi register) is selected for the TRDBFCi bit in the TRDMR register, set the same value to the IOA2 bit in the TRDIORAi register and the IOC2 bit in the TRDIORCi register.

**Figure 8 - 24 Format of Timer RD I/O control register Ci (TRDIORCi) (i = 0 or 1) [Output Compare Function]**

Symbol	7	6	5	4	3	2	1	0							
TRDIORCi	IOD3	IOD2	IOD1	IOD0	IOC3	IOC2	IOC1	IOC0							
IOD3	TRDGRD register function select														
0	TRDIOB output register (see 8.5.2 (2) Changing Output Pins in Registers TRDGRCi (i = 0 or 1) and TRDGRDi)														
1	General register or buffer register														
IOD2	TRDGRD mode select Note 2														
Set to 0 (output compare) in the output compare function.															
IOD1	IOD0	TRDGRD control													
0	0	Pin output by compare match is disabled													
0	1	Low output by compare match with TRDGRDi													
1	0	High output by compare match with TRDGRDi													
1	1	Toggle output by compare match with TRDGRDi													
IOC3	TRDGRC register function select														
0	TRDIOA output register (see 8.5.2 (2) Changing Output Pins in Registers TRDGRCi (i = 0 or 1) and TRDGRDi)														
1	General register or buffer register														
IOC2	TRDGRC mode select Note 3														
Set to 0 (output compare) in the output compare function.															
IOC1	IOC0	TRDGRC control													
0	0	Pin output by compare match is disabled													
0	1	Low output by compare match with TRDGRCi													
1	0	High output by compare match with TRDGRCi													
1	1	Toggle output by compare match with TRDGRCi													

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** If 1 (buffer register for TRDGRBi register) is selected for the TRDBFDi bit in the TRDMR register, set the same value to the IOB2 bit in the TRDIORAi register and the IOD2 bit in the TRDIORCi register.
- Note 3.** If 1 (buffer register for TRDGRAi register) is selected for the TRDBFCi bit in the TRDMR register, set the same value to the IOA2 bit in the TRDIORAi register and the IOC2 bit in the TRDIORCi register.

### 8.3.14 Timer RD status register 0 (TRDSR0)

**Figure 8 - 25 Format of Timer RD status register 0 (TRDSR0) [Input Capture Function]**

Symbol	7	6	5	4	3	2	1	0
TRDSR0	0	0	0	OVF	IMFD	IMFC	IMFB	IMFA
OVF	Overflow flag Note 2							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] When the TRD0 register overflows								
IMFD	Input capture/compare match flag D Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOD0 pin Note 4								
IMFC	Input capture/compare match flag C Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOC0 pin Note 4								
IMFB	Input capture/compare match flag B Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOB0 pin Note 5								
IMFA	Input capture/compare match flag A Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOA0 pin Note 5								

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** When the counter value of timer RD0 changes from FFFFH to 0000H, the overflow flag is set to 1. Also, if the counter value of timer RD0 changes from FFFFH to 0000H due to an input capture/compare match during operation according to the settings of bits CCLR0 to CCLR2 in the TRDCR0 register, the overflow flag is set to 1.

**Note 3.** The writing results are as follows:

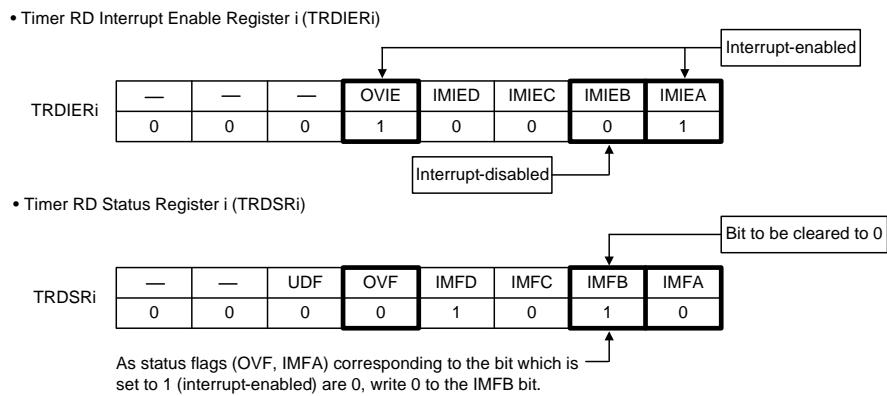
- Writing 1 has no effect.
- If the read value is 0, the bit remains unchanged even if 0 is written to it.  
(Even if the bit is changed from 0 to 1 after reading and then 0 is written to it, it remains 1.)
- If the read value is 1, writing 0 to the bit sets it to 0.

When status flags of interrupt sources (applicable status flags) of timer RD are set to 0 and their interrupts are disabled in timer RD interrupt enable register i (TRDIERi), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RD interrupt enable register i (TRDIERi) and write 0 to applicable status flags.

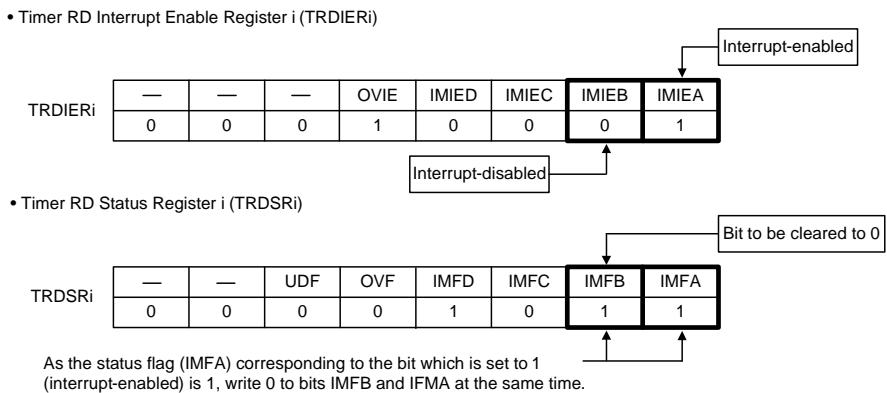
(b) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the IMFB bit to 0 when bits IMIEA and OVIE are set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



(c) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the IMFB bit to 0 when the IMIEA bit is set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



**Note 4.** Edge selected by bits IOk1 and IOk0 ( $k = C$  or  $D$ ) in the TRDIORC0 register.

Including when the TRDBFK0 bit in the TRDMR register is 1 (TRDGRk0 is buffer register).

**Note 5.** Edge selected by bits IOj1 and IOj0 ( $j = A$  or  $B$ ) in the TRDIORA0 register.

**Note 6.** When the DTC is used, bits IMFA, IMFB, IMFC, and IMF0 are set to 1 after DTC transfer is completed.

**Figure 8 - 26 Format of Timer RD status register 0 (TRDSR0)**  
**[Functions Other Than Input Capture Function]**

Symbol	7	6	5	4	3	2	1	0
TRDSR0	0	0	0	OVF	IMFD	IMFC	IMFB	IMFA
OVF	Overflow flag Note 3							
[Source for setting to 0]								
Write 0 after reading. Note 2								
[Source for setting to 1]								
When the TRD0 register overflows								
IMFD	Input capture/compare match flag D Note 5							
[Source for setting to 0]								
Write 0 after reading. Note 2								
[Source for setting to 1]								
When the values of TRD0 and TRDGRD0 match. Note 4								
IMFC	Input capture/compare match flag C Note 5							
[Source for setting to 0]								
Write 0 after reading. Note 2								
[Source for setting to 1]								
When the values of TRD0 and TRDGRC0 match. Note 4								
IMFB	Input capture/compare match flag B Note 5							
[Source for setting to 0]								
Write 0 after reading. Note 2								
[Source for setting to 1]								
When the values of TRD0 and TRDGRB0 match.								
IMFA	Input capture/compare match flag A Note 5							
[Source for setting to 0]								
Write 0 after reading. Note 2								
[Source for setting to 1]								
When the values of TRD0 and TRDGRA0 match.								

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** The writing results are as follows:

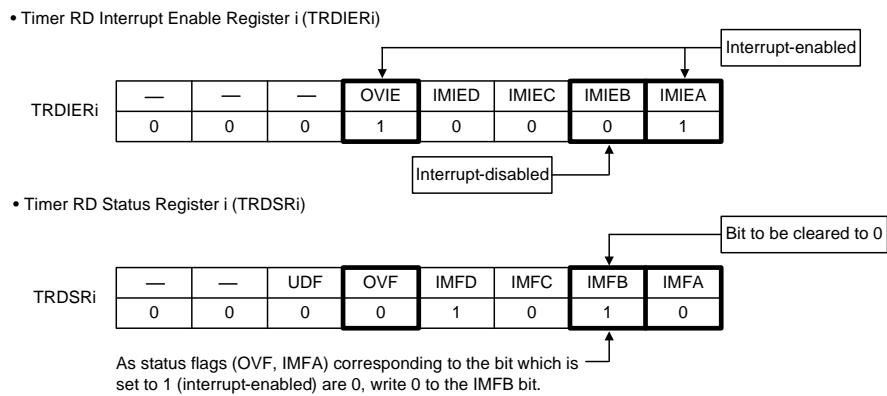
- Writing 1 has no effect.
- If the read value is 0, the bit remains unchanged even if 0 is written to it.  
(Even if the bit is changed from 0 to 1 after reading and then 0 is written to it, it remains 1.)
- If the read value is 1, writing 0 to the bit sets it to 0.

When status flags of interrupt sources (applicable status flags) of timer RD are set to 0 and their interrupts are disabled in timer RD interrupt enable register i (TRDIERi), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RD interrupt enable register i (TRDIERi) and write 0 to applicable status flags.

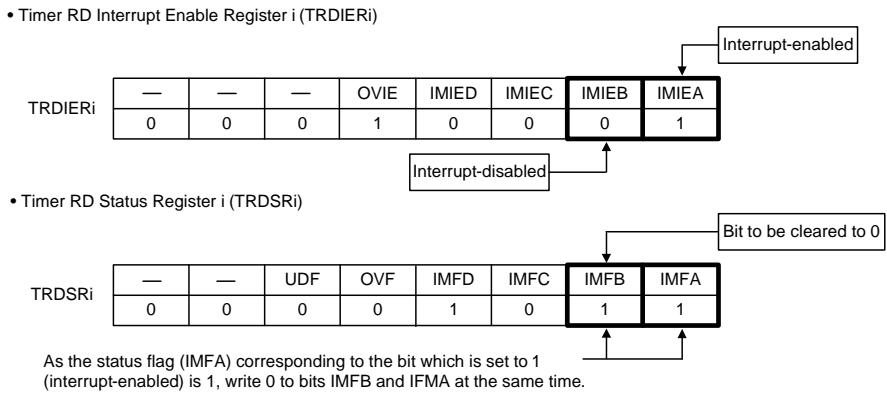
(b) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the IMFB bit to 0 when bits IMIEA and OVIE are set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



(c) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the IMFB bit to 0 when the IMIEA bit is set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



**Note 3.** When the counter value of timer RD0 changes from FFFFH to 0000H, the overflow flag is set to 1. Also, if the counter value of timer RD0 changes from FFFFH to 0000H due to an input capture/compare match during operation according to the settings of bits CCLR0 to CCLR2 in the TRDCR0 register, the overflow flag is set to 1.

**Note 4.** Including when the TRDBFK0 bit (k = C or D) in the TRDMR register is set to 1 (TRDGRK0 is buffer register).

**Note 5.** When the DTC is used, bits IMFA, IMFB, IMFC, and IMFD are set to 1 after DTC transfer is completed.

### 8.3.15 Timer RD status register 1 (TRDSR1)

Figure 8 - 27 Format of Timer RD status register 1 (TRDSR1) [Input Capture Function]

Symbol	7	6	5	4	3	2	1	0
TRDSR1	0	0	UDF	OVF	IMFD	IMFC	IMFB	IMFA
UDF	Underflow flag							
Disabled in the input capture function.								
OVF	Overflow flag Note 2							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] When the TRD1 register overflows								
IMFD	Input capture/compare match flag D Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOD1 pin Note 4								
IMFC	Input capture/compare match flag C Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOC1 pin Note 4								
IMFB	Input capture/compare match flag B Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOB1 pin Note 5								
IMFA	Input capture/compare match flag A Note 6							
[Source for setting to 0] Write 0 after reading. Note 3 [Source for setting to 1] Input edge of TRDIOA1 pin Note 5								

- Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.
- Note 2.** When the counter value of timer RD1 changes from FFFFH to 0000H, the overflow flag is set to 1. Also, if the counter value of timer RD1 changes from FFFFH to 0000H due to an input capture/compare match during operation according to the settings of bits CCLR0 to CCLR2 in the TRDCR1 register, the overflow flag is set to 1.

**Note 3.** The writing results are as follows:

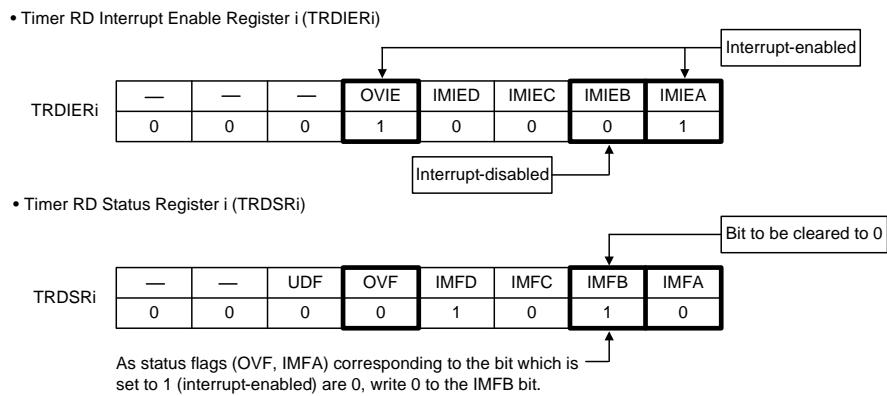
- Writing 1 has no effect.
- If the read value is 0, the bit remains unchanged even if 0 is written to it.  
(Even if the bit is changed from 0 to 1 after reading and then 0 is written to it, it remains 1.)
- If the read value is 1, writing 0 to the bit sets it to 0.

When status flags of interrupt sources (applicable status flags) of timer RD are set to 0 and their interrupts are disabled in timer RD interrupt enable register i (TRDIERi), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RD interrupt enable register i (TRDIERi) and write 0 to applicable status flags.

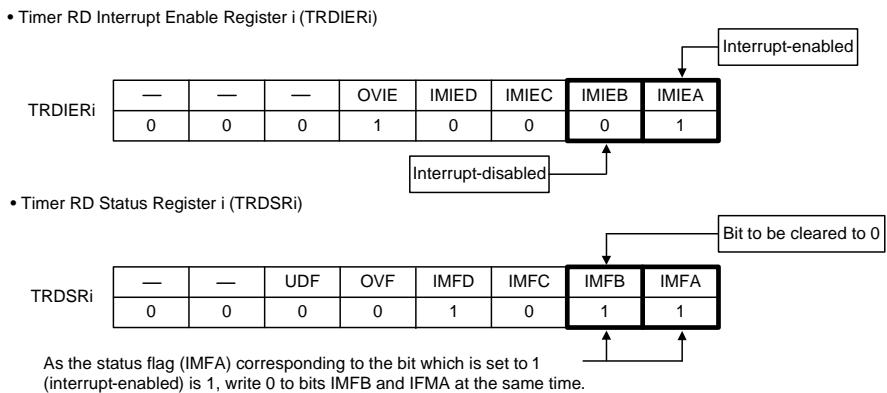
(b) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the IMFB bit to 0 when bits IMIEA and OVIE are set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



(c) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the IMFB bit to 0 when the IMIEA bit is set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



**Note 4.** Edge selected by bits IOk1 and IOk0 ( $k = C$  or  $D$ ) in the TRDIORC1 register.

Including when the TRDBFK1 bit in the TRDMR register is 1 (TRDGRk1 is buffer register).

**Note 5.** Edge selected by bits IOj1 and IOj0 ( $j = A$  or  $B$ ) in the TRDIORA1 register.

**Note 6.** When the DTC is used, bits IMFA, IMFB, IMFC, and IMFQ are set to 1 after DTC transfer is completed.

**Figure 8 - 28 Format of Timer RD status register 1 (TRDSR1) [Functions Other Than Input Capture Function]**

Address: F0283H After Reset: 00H Note 1 R/W

Symbol 7 6 5 4 3 2 1 0

TRDSR1	0	0	UDF	OVF	IMFD	IMFC	IMFB	IMFA
--------	---	---	-----	-----	------	------	------	------

UDF	Underflow flag	R/W
In complementary PWM mode [Source for setting to 0] Write 0 after reading. Note 2 [Sources for setting to 1] When TRD1 underflows. Enabled only in complementary PWM mode.		R/W
OVF	Overflow flag Note 3	R/W
[Source for setting to 0] Write 0 after reading. Note 2 [Source for setting to 1] When the TRD1 register overflows		R/W
IMFD	Input capture/compare match flag D Note 5	R/W
[Source for setting to 0] Write 0 after reading. Note 2 [Source for setting to 1] When the values of TRD1 and TRDGRD1 match. Note 4		R/W
IMFC	Input capture/compare match flag C Note 5	R/W
[Source for setting to 0] Write 0 after reading. Note 2 [Source for setting to 1] When the values of TRD1 and TRDGRC1 match. Note 4		R/W
IMFB	Input capture/compare match flag B Note 5	R/W
[Source for setting to 0] Write 0 after reading. Note 2 [Source for setting to 1] When the values of TRD1 and TRDGRB1 match.		R/W
IMFA	Input capture/compare match flag A Note 5	R/W
[Source for setting to 0] Write 0 after reading. Note 2 [Source for setting to 1] When the values of TRD1 and TRDGRA1 match.		R/W

**Note 1.** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Note 2.** The writing results are as follows:

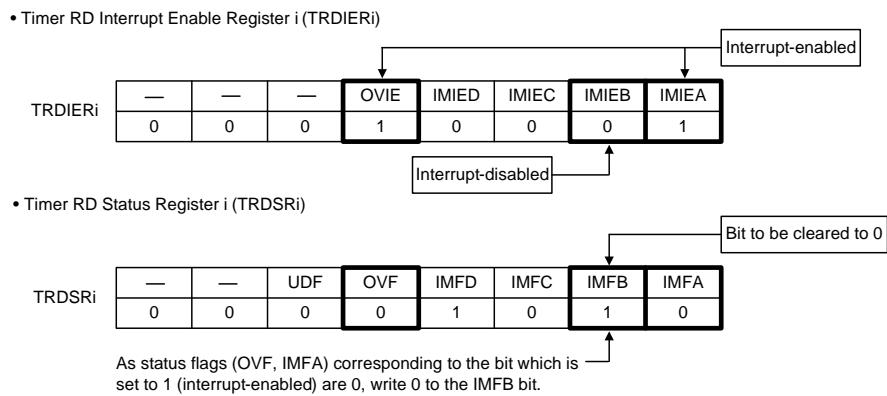
- Writing 1 has no effect.
- If the read value is 0, the bit remains unchanged even if 0 is written to it.  
(Even if the bit is changed from 0 to 1 after reading and then 0 is written to it, it remains 1.)
- If the read value is 1, writing 0 to the bit sets it to 0.

When status flags of interrupt sources (applicable status flags) of timer RD are set to 0 and their interrupts are disabled in timer RD interrupt enable register i (TRDIERi), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RD interrupt enable register i (TRDIERi) and write 0 to applicable status flags.

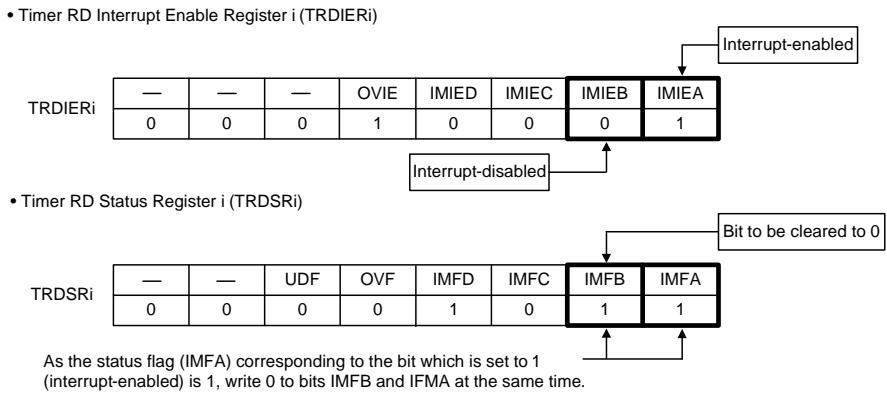
(b) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the IMFB bit to 0 when bits IMIEA and OVIE are set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



(c) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the IMFB bit to 0 when the IMIEA bit is set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



**Note 3.** When the counter value of timer RD1 changes from FFFFH to 0000H, the overflow flag is set to 1. Also, if the counter value of timer RD1 changes from FFFFH to 0000H due to an input capture/compare match during operation according to the settings of bits CCLR0 to CCLR2 in the TRDCR1 register, the overflow flag is set to 1.

**Note 4.** Including when the TRDBFK1 bit (k = C or D) in the TRDMR register is set to 1 (TRDGRK1 is buffer register).

**Note 5.** When the DTC is used, bits IMFA, IMFB, IMFC, and IMFD are set to 1 after DTC transfer is completed.

### 8.3.16 Timer RD interrupt enable register i (TRDIERi) (i = 0 or 1)

Figure 8 - 29 Format of Timer RD interrupt enable register i (TRDIERi) (i = 0 or 1)

Symbol	7	6	5	4	3	2	1	0
TRDIERi	0	0	0	OVIE	IMIED	IMIEC	IMIEB	IMIEA
OVIE	Overflow/underflow interrupt enable							
0	Interrupt (OVI) by bits OVF and UDF disabled							
1	Interrupt (OVI) by bits OVF and UDF enabled							
IMIED	Input capture/compare match interrupt enable D							
0	Interrupt (IMID) by the IMFD bit is disabled							
1	Interrupt (IMID) by the IMFD bit is enabled							
IMIEC	Input capture/compare match interrupt enable C							
0	Interrupt (IMIC) by the IMFC bit is disabled							
1	Interrupt (IMIC) by the IMFC bit is enabled							
IMIEB	Input capture/compare match interrupt enable B							
0	Interrupt (IMIB) by the IMFB bit is disabled							
1	Interrupt (IMIB) by the IMFB bit is enabled							
IMIEA	Input capture/compare match interrupt enable A							
0	Interrupt (IMIA) by the IMFA bit is disabled							
1	Interrupt (IMIA) by the IMFA bit is enabled							

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.17 Timer RD PWM function output level control register i (TRDPOCRi) (i = 0 or 1)

Settings to the TRDPOCRi register are enabled only in PWM function. When not in PWM function, they are disabled.

**Figure 8 - 30 Format of Timer RD PWM function output level control register i (TRDPOCRi) (i= 0 or 1)  
[PWM Function]**

Symbol	7	6	5	4	3	2	1	0
TRDPOCRi	0	0	0	0	0	POLD	POLC	POLB
POLD	PWM function output level control D							
0	TRDIODi output level is low active							
1	TRDIODi output level is high active							
POLC	PWM function output level control C							
0	TRDIOCi output level is low active							
1	TRDIOCi output level is high active							
POLB	PWM function output level control B							
0	TRDIOBi output level is low active							
1	TRDIOBi output level is high active							

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.18 Timer RD counter i (TRDi) (i = 0 or 1)

[Timer Mode]

Access the TRDi register in 16-bit units. Do not access it in 8-bit units.

[Reset Synchronous PWM Mode and PWM3 Mode]

Access the TRD0 register in 16-bit units. Do not access it in 8-bit units. The TRD1 register is not used in reset synchronous PWM mode and PWM3 mode.

[Complementary PWM Mode (TRD0)]

Access the TRD0 register in 16-bit units. Do not access it in 8-bit units.

[Complementary PWM Mode (TRD1)]

Access the TRD1 register in 16-bit units. Do not access it in 8-bit units.

**Figure 8 - 31 Format of Timer RD counter i (TRDi) (i = 0 or 1) [Timer Mode]**

Address: F0276H (TRD0), F0286H (TRD1)																Note	R/W
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TRDi																	
—	Function													Setting Range			
Bits 15 to 0	Count the count source. Count operation is incremented. When an overflow occurs, the OVF bit in the TRDSRi register is set to 1.													0000H to FFFFH			

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Figure 8 - 32 Format of Timer RD counter 0(TRD0) [Reset Synchronous PWM Mode and PWM3 Mode]**

Address: F0276H (TRD0)																Note	R/W
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
TRD0																	
—	Function													Setting Range			
Bits 15 to 0	Count the count source. Count operation is incremented. When an overflow occurs, the OVF bit in the TRDSR0 register is set to 1.													0000H to FFFFH			

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Figure 8 - 33 Format of Timer RD counter 0 (TRD0) [Complementary PWM Mode (TRD0)]**

Address: F0276H (TRD0) After Reset: 0000H Note R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--------	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

TRD0	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

—	Function	Setting Range
Bits 15 to 0	Dead time must be set. Count the count source. Count operation is incremented or decremented. When an overflow occurs, the OVF bit in the TRDSR0 register is set to 1.	0001H to FFFFH

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Figure 8 - 34 Format of Timer RD counter 1 (TRD1) [Complementary PWM Mode (TRD1)]**

Address: F0286H (TRD1) After Reset: 0000H Note R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--------	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

TRD1	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]	[ ]
------	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

—	Function	Setting Range
Bits 15 to 0	Set to 0000H. Count the count source. Count operation is incremented or decremented. When an underflow occurs, the UDF bit in the TRDSR1 register is set to 1.	0000H to FFFFH

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

### 8.3.19 Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi) (i = 0 or 1) [Input Capture Function]

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The following registers are disabled in the input capture function:

TRDOER1, TRDOER2, TRDOCR, TRDPOCR0, and TRDPOCR1

Set the pulse width of the input capture signal applied to the TRDIOji pin to three or more cycles of the timer RD operating clock (fCLK) when no digital filter is used (the DFj bit in the TRDDFi register is 0).

[Output Compare Function]

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The following registers are disabled in the output compare function:

TRDDF0, TRDDF1, TRDPOCR0, and TRDPOCR1

[PWM Function]

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The following registers are disabled in PWM function:

TRDDF0, TRDDF1, TRDIORA0, TRDIORC0, TRDIORA1, and TRDIORC1

[Reset Synchronous PWM Mode]

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The following registers are disabled in reset synchronous PWM mode:

TRDPMR, TRDOCR Note, TRDDF0, TRDDF1, TRDIORA0, TRDIORC0, TRDPOCR0, TRDIORA1, TRDIORC1, and TRDPOCR1

**Note** The TOC0 bit in the TRDOCR register is enabled as an initial output setting of TRDIOC0 in reset synchronous PWM mode and complementary PWM mode.

**[Complementary PWM Mode]**

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The TRDGRC0 register is not used in complementary PWM mode.

The following registers are disabled in complementary PWM mode.

TRDPMR, TRDOCR Note, TRDDF0 TRDDF1, TRDIORA0, TRDIORC0, TRDPOCR0, TRDIORA1, TRDIORC1, and TRDPOCR1

**Note** The TOC0 bit in the TRDOCR register is enabled as an initial output setting of TRDIOC0 in reset synchronous PWM mode and complementary PWM mode.

Since values cannot be written to the TRDGRB0, TRDGRA1, or TRDGRB1 register directly after count operation starts (prohibited item), use the TRDGRD0, TRDGRC1, or TRDGRD1 register as a buffer register.

However, to write data to the TRDGRD0, TRDGRC1, or TRDGRD1 register, set bits TRDBFD0, TRDBFC1, and TRDBFD1 to 0 (general register). After this, bits TRDBFD0, TRDBFC1, and TRDBFD1 may be set to 1 (buffer register).

**[PWM3 Mode]**

Access registers TRDGRAi to TRDGRDi in 16-bit units. Do not access them in 8-bit units.

The following registers are disabled in PWM3 mode:

TRDPMR, TRDDF0, TRDDF1, TRDIORA0, TRDIORC0, TRDPOCR0, TRDIORA1, TRDIORC1, and TRDPOCR1

Registers TRDGRC0, TRDGRC1, TRDGRD0, and TRDGRD1 are not used in PWM3 mode. To use them as buffer registers, set bits TRDBFC0, TRDBFC1, TRDBFD0, and TRDBFD1 to 0 (general register) and write a value to the TRDGRC0, TRDGRC1, TRDGRD0, or TRDGRD1 register. After this, bits TRDBFC0, TRDBFC1, TRDBFD0, and TRDBFD1 may be set to 1 (buffer register).

**Figure 8 - 35 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)  
(i = 0 or 1) [Input Capture Function]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFFF5EH (TRDGRD1)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRDGRAi																
TRDGRBi																
TRDGRCi																
TRDGRDi	—															
	Function															
	Bits 15 to 0 See <b>Table 8 - 3 TRDGRji Register Functions in Input Capture Function.</b>															

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 3 TRDGRji Register Functions in Input Capture Function**

Register	Setting	Register Function	Input-Capture Input Pin
TRDGRAi	—	General register. The value of the TRDi register can be read at input capture.	TRDIOAi
TRDGRBi			TRDIOBi
TRDGRCi	TRDBFCi = 0	General register. The value of the TRDi register can be read at input capture.	TRDIOCi
TRDGRDi	TRDBFDi = 0		TRDIODi
TRDGRCi	TRDBFCi = 1	Buffer register. The value of the TRDi register can be read at input capture (see <b>8.4.2 Buffer Operation</b> ).	TRDIOAi
TRDGRDi	TRDBFDi = 1		TRDIOBi

**Remark** i = 0 or 1, j = A, B, C, or D

TRDBFCi, TRDBFDi: Bits in TRDMR register

**Figure 8 - 36 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)  
(i = 0 or 1) [Output Compare Function]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFFF5EH (TRDGRD1)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRDGRAi																
TRDGRBi																
TRDGRCi	—															
TRDGRDi	—															
Bits 15 to 0	Function															

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 4 TRDGRji Register Functions in Output Compare Function**

Register	Setting		Register Function	Output-Compare Output Pin	
	TRDBFji	IOj3			
TRDGRAi	—	—	General register. Write the compare value.		
TRDGRBi					
TRDGRCi	0	1	General register. Write the compare value.		
TRDGRDi					
TRDGRCi	1	1	Buffer register. Write the next compare value (see 8.4.2 Buffer Operation).		
TRDGRDi					
TRDGRCi	0	0	TRDIOAi output control	(See 8.5.2 (2) Changing Output Pins in Registers TRDGRCi (i = 0 or 1) and TRDGRDi.)	
TRDGRDi			TRDIOBi output control		

**Caution** When the setting of bits TCK2 to TCK0 in the TRDCRi register is 000B (fCLK, fhoco) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

**Remark** i = 0 or 1, j = A, B, C, or D

TRDBFji: Bit in TRDMR register, IOj3: Bit in TRDIORCi register

**Figure 8 - 37 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)  
(i = 0 or 1) [PWM Mode]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFFF5EH (TRDGRD1)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRDGRAi																
TRDGRBi																
TRDGRCi	—															
TRDGRDi	—															
	Function															
	Bits 15 to 0 See <b>Table 8 - 5 TRDGRji Register Functions in PWM Function.</b>															

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 5 TRDGRji Register Functions in PWM Function**

Register	Setting	Register Function	PWM Output Pin
TRDGRAi	—	General register. Set the PWM period.	—
TRDGRBi	—	General register. Set the changing point of PWM output.	TRDIOBi
TRDGRCi	TRDBFCi = 0	General register. Set the changing point of PWM output.	TRDI OCI
TRDGRDi	TRDBFDi = 0		TRDI ODI
TRDGRCi	TRDBFCi = 1	Buffer register. Set the next PWM period (see <b>8.4.2 Buffer Operation</b> ).	—
TRDGRDi	TRDBFDi = 1	Buffer register. Set the changing point of the next PWM output (see <b>8.4.2 Buffer Operation</b> ).	TRDIOBi

**Caution** When the setting of bits TCK2 to TCK0 in the TRDCRi register is 000B (fCLK, fhoco) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

**Remark** i = 0 or 1, j = A, B, C, or D

TRDBFCi, TRDBFDi: Bits in TRDMR register

**Figure 8 - 38 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)**  
**(i = 0 or 1) [Reset Synchronous PWM Mode]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFFF5EH (TRDGRD1)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRDGRAi																
TRDGRBi																
TRDGRCi																
TRDGRDi	—															
	Function															
	Bits 15 to 0 See <b>Table 8 - 6 TRDGRji Register Functions in Reset Synchronous PWM Mode.</b>															

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 6 TRDGRji Register Functions in Reset Synchronous PWM Mode**

Register	Setting	Register Function	PWM Output Pin
TRDGRA0	—	General register. Set the PWM period.	(TRDIOC0, output inverted every PWM period)
TRDGRB0	—	General register. Set the changing point of PWM1 output.	TRDIOB0 TRDIOD0
TRDGRC0	TRDBFC0 = 0	(Not used in reset synchronous PWM mode.)	—
TRDGRD0	TRDBFD0 = 0		
TRDGRA1	—	General register. Set the changing point of PWM2 output.	TRDIOA1 TRDIOC1
TRDGRB1	—	General register. Set the changing point of PWM3 output.	TRDIOB1 TRDIOD1
TRDGRC1	TRDBFC1 = 0	(Not used in reset synchronous PWM mode.)	—
TRDGRD1	TRDBFD1 = 0		
TRDGRC0	TRDBFC0 = 1	Buffer register. Set the next PWM period (see <b>8.4.2 Buffer Operation</b> ).	(TRDIOC0, output inverted every PWM period)
TRDGRD0	TRDBFD0 = 1	Buffer register. Set the changing point of the next PWM1 (see <b>8.4.2 Buffer Operation</b> ).	TRDIOB0 TRDIOD0
TRDGRC1	TRDBFC1 = 1	Buffer register. Set the changing point of the next PWM2 (see <b>8.4.2 Buffer Operation</b> ).	TRDIOA1 TRDIOC1
TRDGRD1	TRDBFD1 = 1	Buffer register. Set the changing point of the next PWM3 (see <b>8.4.2 Buffer Operation</b> ).	TRDIOB1 TRDIOD1

**Caution** When the setting of bits TCK2 to TCK0 in the TRDCR0 register is 000B (fCLK, fhoco) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

**Remark** i = 0 or 1, j = A, B, C, or D

TRDBFC0, TRDBFD0, TRDBFC1, TRDBFD1: Bits in TRDMR register

**Figure 8 - 39 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)**  
**(i = 0 or 1) [Complementary PWM Mode]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFF5EH (TRDGRD1)

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 7 TRDGRji Register Functions in Complementary PWM Mode**

Register	Setting	Register Function	PWM Output Pin
TRDGRA0	—	General register. Set the PWM period at initialization. Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ FFFFh - value set in TRD0 register Do not write to this register when bits TSTART0 and TSTART1 in the TRDSTR register are set to 1 (count starts).	(TRDIOC0, output inverted every half period)
TRDGRB0	—	General register. Set the changing point of PWM1 output at initialization. Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Do not write to this register when bits TSTART0 and TSTART1 in the TRDSTR register are set to 1 (count starts).	TRDIOB0 TRDIOD0
TRDGRA1	—	General register. Set the changing point of PWM2 output at initialization. Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Do not write to this register when bits TSTART0 and TSTART1 in the TRDSTR register are set to 1 (count starts).	TRDIOA1 TRDIOC1
TRDGRB1	—	General register. Set the changing point of PWM3 output at initialization. Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Do not write to this register when bits TSTART0 and TSTART1 in the TRDSTR register are set to 1 (count starts).	TRDIOB1 TRDIOD1
TRDGRC0	—	(Not used in complementary PWM mode.)	—
TRDGRD0	TRDBFD0 = 1	Buffer register. Set the changing point of next PWM1 output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Set this register to the same value as the TRDGRB0 register for initialization.	TRDIOB0 TRDIOD0
TRDGRC1	TRDBFC1 = 1	Buffer register. Set the changing point of next PWM2 output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Set this register to the same value as the TRDGRA1 register for initialization.	TRDIOA1 TRDIOC1
TRDGRD1	TRDBFD1 = 1	Buffer register. Set the changing point of next PWM3 output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\geq$ Value set in TRD0 register (initial count value) $\leq$ Value set in TRDGRA0 register - value set in TRD0 register Set this register to the same value as the TRDGRB1 register for initialization.	TRDIOB1 TRDIOD1

**Caution** When the setting of bits TCK2 to TCK0 in the TRDCR*i* register is 000B (fCLK, fHOCO) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

**Remark** i = 0 or 1, j = A, B, C, or D  
TRDBFD0, TRDBFC1, TRDBFD1: Bits in TRDMR register

**Figure 8 - 40 Format of Timer RD general registers Ai, Bi, Ci, and Di (TRDGRAi, TRDGRBi, TRDGRCi, TRDGRDi)  
(i = 0 or 1) [PWM3 Mode]**

Address: F0278H (TRDGRA0), F027AH (TRDGRB0), After Reset: FFFFH Note R/W

FFF58H (TRDGRC0), FFFF5AH (TRDGRD0),

F0288H (TRDGRA1), F028AH (TRDGRB1),

FFF5CH (TRDGRC1), FFFF5EH (TRDGRD1)

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRDGRAi																
TRDGRBi																
TRDGRCi																
TRDGRDi	—															
	Function															
	Bits 15 to 0 See <b>Table 8 - 8 TRDGRji Register Functions in PWM3 Mode.</b>															

**Note** The value after reset is undefined when FRQSEL4 = 1 in the user option byte (000C2H) and TRD0EN = 0 in the PER1 register. If it is necessary to read the initial value, set fCLK to fIH and TRD0EN = 1 before reading.

**Table 8 - 8 TRDGRji Register Functions in PWM3 Mode**

Register	Setting	Register Function	PWM Output Pin
TRDGRA0	—	General register. Set the PWM period. Setting range: $\geq$ Value set in TRDGRA1 register	TRDIOA0
TRDGRA1		General register. Set the changing point (active level timing) of PWM output Setting range: $\leq$ Value set in TRDGRA0 register	
TRDGRB0		General register. Set the changing point (the timing for returning to initial output level) of PWM output. Setting range: $\geq$ Value set in TRDGRB1 register and $\leq$ Value set in TRDGRA0 register	TRDIOB0
TRDGRB1		General register. Set the changing point (active level timing) of PWM output Setting range: $\leq$ Value set in TRDGRB0 register	
TRDGRC0	TRDBFC0 = 0	(Not used in PWM3 mode.)	—
TRDGRC1	TRDBFC1 = 0		
TRDGRD0	TRDBFD0 = 0		
TRDGRD1	TRDBFD1 = 0		
TRDGRC0	TRDBFC0 = 1	Buffer register. Set the next PWM period (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\leq$ Value set in TRDGRC1 register	TRDIOA0
TRDGRC1	TRDBFC1 = 1	Buffer register. Set the changing point of next PWM output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\leq$ Value set in TRDGRC0 register	
TRDGRD0	TRDBFD0 = 1	Buffer register. Set the changing point of next PWM output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\geq$ Value set in TRDGRD1 register and $\leq$ Value set in TRDGRC0 register	TRDIOB0
TRDGRD1	TRDBFD1 = 1	Buffer register. Set the changing point of next PWM output (see <b>8.4.2 Buffer Operation</b> ). Setting range: $\leq$ Value set in TRDGRD0 register	

**Caution** When the setting of bits TCK2 to TCK0 in the TRDCR0 register is 000B (fCLK, fHOCo) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

**Remark** i = 0 or 1, j = A, B, C, or D

TRDBFC0, TRDBFD0, TRDBFC1, TRDBFD1: Bits in TRDMR register

### 8.3.20 Port mode register 1 (PM1)

This register sets input/output of port 1 in 1-bit units.

When using the ports (P10/TRDIOD1, P11/TRDIOC1, etc.) to be shared with the timer output pin for timer output, set the port mode register (PMxx) bit and port register (Pxx) bit corresponding to each port to 0.

Example When using P10/TRDIOD1 for timer output  
Set the PM10 bit of port mode register 1 to 0.  
Set the P10 bit of port register 1 to 0.

When using the ports (P10/TRDIOD1, P11/TRDIOC1, etc.) to be shared with the timer input pin for timer input, set the port mode register (PMxx) bit corresponding to each port to 1. At this time, the port register (Pxx) bit may be 0 or 1.

Example When using P10/TRDIOD1 for timer input  
Set the PM10 bit of port mode register 1 to 1.  
Set the P10 bit of port register 1 to 0 or 1.

The PM1 register can be set by a 1-bit or 8-bit memory manipulation instruction.  
Reset signal generation sets this register to FFH.

**Figure 8 - 41 Format of Port mode register 1 (PM1) (100-pin products)**

Address: FFF21H      After Reset: FFH      R/W

Symbol	7	6	5	4	3	2	1	0
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10
PMmn	Pmn pin I/O mode selection (m = 1; n = 0 to 7)							
0	Output mode (output buffer on)							
1	Input mode (output buffer off)							

**Remark** The figure shown above presents the format of port mode register 1 of the 100-pin products. The format of the port mode register of other products, see **Tables 4 - 7 to 4 - 13 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product.**

## 8.4 Items Common to Multiple Modes

### 8.4.1 Count Sources

The count source selection method is the same in all modes. However, the external clock cannot be selected in PWM3 mode.

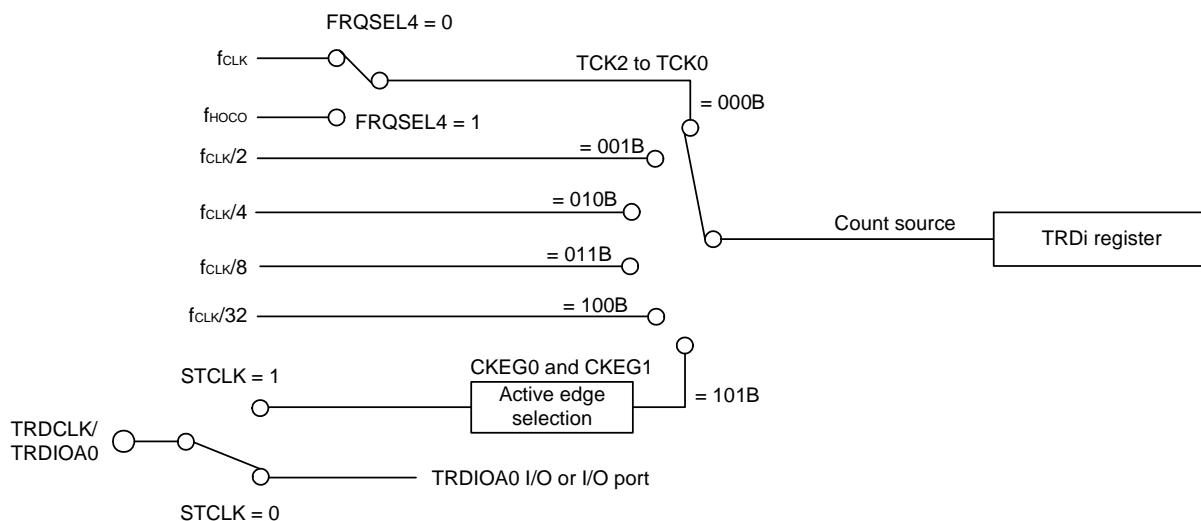
**Table 8 - 9 Count Source Selection**

Count Source	Selection
fCLK, fHO CO Note, fCLK/2, fCLK/4, fCLK/8, fCLK/32	The count source is selected by bits TCK2 to TCK0 in the TRDCR <i>i</i> register.
External signal input to TRDCLK pin	The STCLK bit in the TRDFCR register is set to 1 (external clock input enabled). Bits TCK2 to TCK0 in the TRDCR <i>i</i> register are set to 101B (count source: external clock). The active edge is selected by bits CKEG1 and CKEG0 in the TRDCR <i>i</i> register. The port mode register bit for the I/O port multiplexed with the TRDCLK pin is set to 1 (input mode).

**Remark** i = 0 or 1

**Note** fCLK is selected when FRQSEL4 = 0 and fHO CO is selected when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Figure 8 - 42 Count Source Block Diagram**



**Remark**  
i = 0 or 1

TCK0 to TCK2, CKEG0, CKEG1: Bits in TRDCR*i* register  
STCLK: Bit in TRDFCR register  
FRQSEL4: Bit in user option byte (000C2H)

Set the pulse width of the external clock applied to the TRDCLK pin to three or more cycles of the timer RD operating clock (fCLK).

### 8.4.2 Buffer Operation

The TRDGRCi register ( $i = 0$  or  $1$ ) can be used as the buffer register for the TRDGRAi register, and the TRDGRDi register can be used as the buffer register for the TRDGRBi register by means of bits TRDBFCi and TRDBFDi in the TRDMR register.

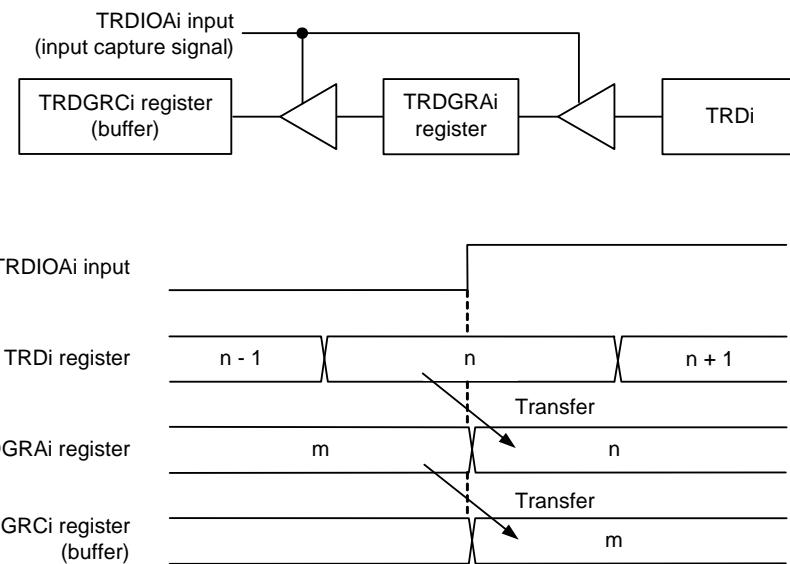
- TRDGRAi buffer register: TRDGRCi register
- TRDGRBi buffer register: TRDGRDi register

Buffer operation depends on the mode. Table 8 - 10 lists the Buffer Operation in Each Mode.

**Table 8 - 10 Buffer Operation in Each Mode**

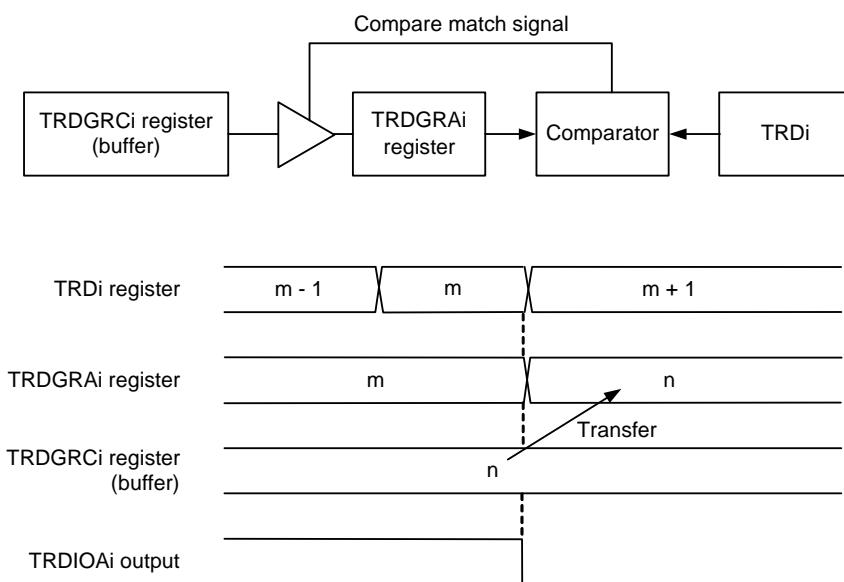
Function and Mode		Transfer Timing	Transfer Register	
Timer mode	Input capture function	TRDIOAi input signal (Input capture signal input)	Transfer content of TRDGRAi register to TRDGRCi register (buffer register)	
		TRDIOBi input signal (Input capture signal input)	Transfer content of TRDGRBi register to TRDGRDi register (buffer register)	
	Output compare function	Compare match with TRDi register and TRDGRAi register	Transfer content of TRDGRCi register (buffer register) to TRDGRAi register	
		Compare match with TRDi register and TRDGRBi register	Transfer content of TRDGRDi register (buffer register) to TRDGRBi register	
PWM function		Compare match with TRDi register and TRDGRAi register	Transfer content of TRDGRCi register (buffer register) to TRDGRAi register	
		Compare match with TRDi register and TRDGRBi register	Transfer content of TRDGRDi register (buffer register) to TRDGRBi register	
Reset synchronous PWM mode		Compare match with TRD0 register and TRDGRA0 register	Transfer content of TRDGRCi register (buffer register) to TRDGRAi register Transfer content of TRDGRDi register (buffer register) to TRDGRBi register	
Complementary PWM mode		<ul style="list-style-type: none"> <li>Underflow of TRD1 register when CMD1 and CMD0 bits in TRDFCR register are 11B</li> <li>Compare match with TRD0 register and TRDGRA0 register when CMD1 and CMD0 bits in TRDFCR register are 10B</li> </ul>	Transfer content of TRDGRC1 register (buffer register) to TRDGRA1 register Transfer content of TRDGRDi register (buffer register) to TRDGRBi register	
PWM3 mode		Compare match with TRD0 register and TRDGRA0 register	Transfer content of TRDGRCi register (buffer register) to TRDGRAi register Transfer content of TRDGRDi register (buffer register) to TRDGRBi register	

**Remark**  $i = 0$  or  $1$

**Figure 8 - 43 Buffer Operation in Input Capture Function**

Remark  
 $i = 0 \text{ or } 1$

The above diagram applies under the following conditions  
• The TRDBFCi bit in the TRDMR register is set to 1 (TRDGRCi register is buffer register for TRDGRAi register).  
• Bits IOA2 to IOA0 in the TRDIORAi register are set to 100B (input capture at the rising edge).

**Figure 8 - 44 Buffer Operation in Output Compare Function**

Remark  
 $i = 0 \text{ or } 1$

The above diagram applies under the following conditions  
• The TRDBFCi bit in the TRDMR register is set to 1 (TRDGRCi register is buffer register for TRDGRAi register).  
• Bits IOA2 to IOA0 in the TRDIORAi register are set to 001B (low output by compare match).

Perform the following for the timer mode (input capture and output compare functions).

When using the TRDGRCi ( $i = 0$  or  $1$ ) register as the buffer register for the TRDGRAi register

- Set the IOC3 bit in the TRDIORCi register to 1 (general register or buffer register).
- Set the IOC2 bit in the TRDIORCi register to the same value as the IOA2 bit in the TRDIORAi register.

When using the TRDGRDi register as the buffer register for the TRDGRBi register

- Set the IOD3 bit in the TRDIORCi register to 1 (general register or buffer register).
- Set the IOD2 bit in the TRDIORCi register to the same value as the IOB2 bit in the TRDIORAi register.

In the input capture function, when the TRDGRCi register or TRDGRDi register is used as a buffer register, the IMFC bit or IMFD bit in the TRDSRi register is set to 1 at the input edge of the TRDIOC $i$  pin or TRDIOD $i$  pin.

When also using registers TRDGRCi and TRDGRDi as buffer registers for the output compare function, PWM function, reset synchronous PWM mode, complementary PWM mode, and PWM3 mode, bits IMFC and IMFD in the TRDSRi register are set to 1 by a compare match with the TRDi register.

### 8.4.3 Synchronous Operation

The TRD1 register is synchronized with the TRD0 register

- Synchronous preset

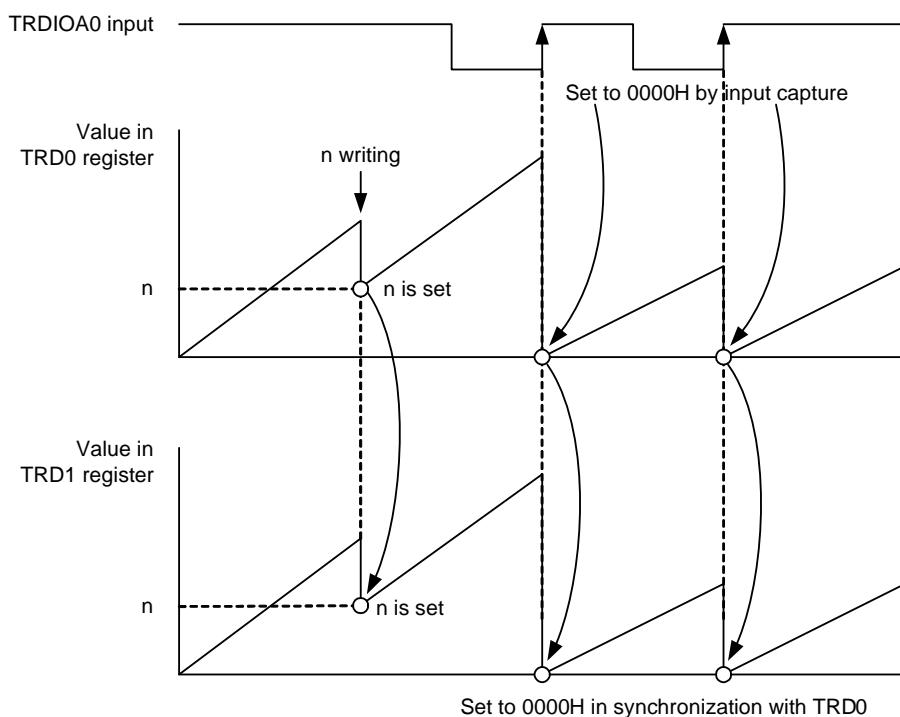
When the TRDSYNC bit in the TRDMR register is set to 1 (synchronous operation), the data is written to both the TRD0 and TRD1 registers after writing to the TRDi register.

- Synchronous clear

When the TRDSYNC bit is 1 and bits CCLR2 to CCLR0 in the TRDCR0 register are 011B (synchronous clear), the TRD0 register is set to 0000H at the same time as the TRD1 register is set to 0000H.

Also, when the TRDSYNC bit is 1 and bits CCLR2 to CCLR0 are 011B (synchronous clear), the TRD1 register is set to 0000H at the same time as the TRD0 register is set to 0000H.

**Figure 8 - 45 Synchronous Operation**



The above diagram applies under the following conditions

- The TRDSYNC bit in the TRDMR register is set to 1 (synchronous operation).
- Bits CCLR2 to CCLR0 in the TRDCR0 register are set to 001B (TRD0 is set to 0000H by input capture).
- Bits CCLR2 to CCLR0 in the TRDCR1 register are set to 011B (TRD1 is set to 0000H in synchronization with TRD0).
- Bits IOA2 to IOA0 in the TRDIOA0 register are set to 100B.
- Bits CMD1 to CMD0 in the TRDFCR register are set to 00B. } (Input capture at the rising edge of TRDIOA0 input)
- The PWM 3 bit in the TRDFCR register is set to 1.

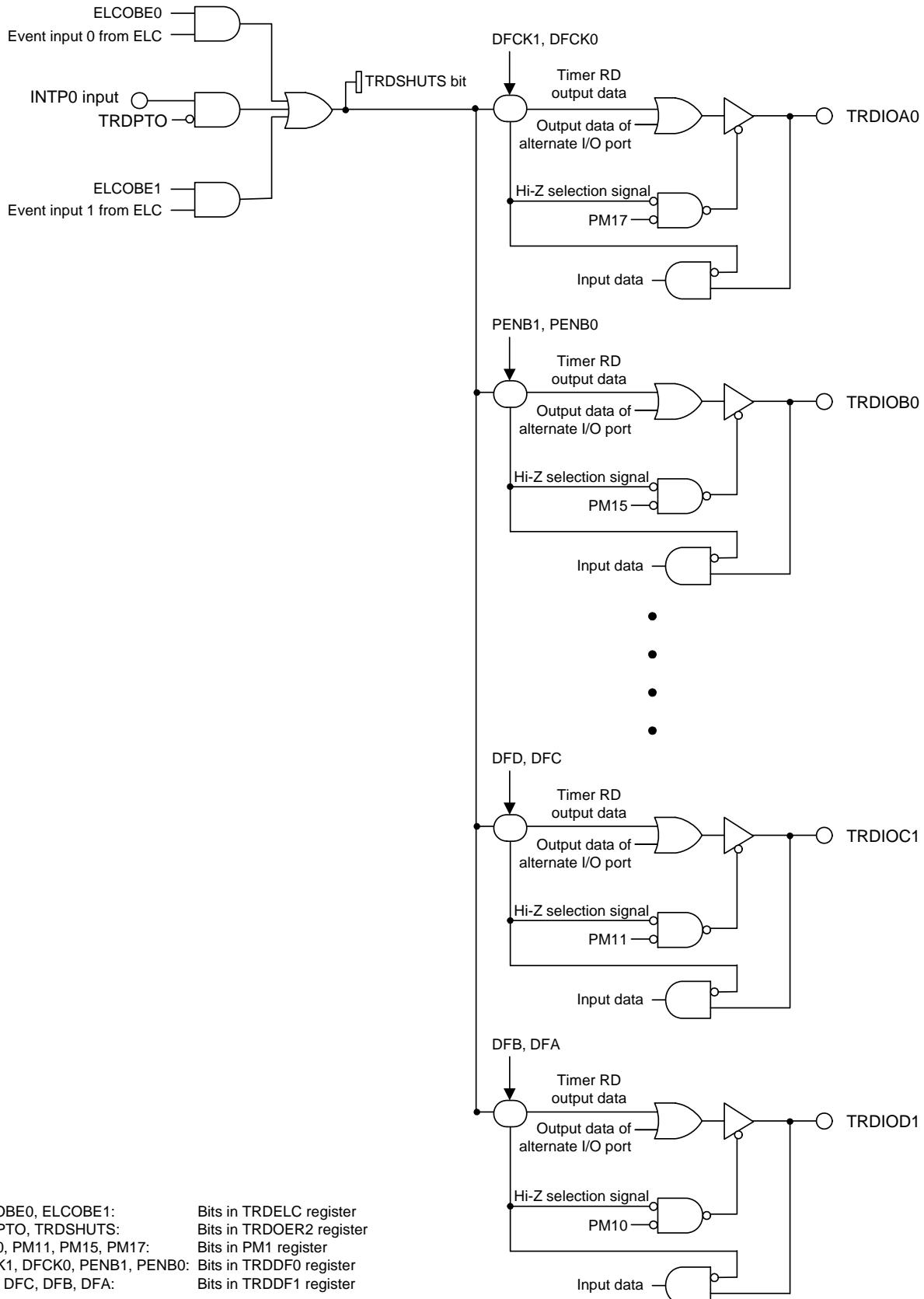
#### 8.4.4 Pulse Output Forced Cutoff

In the PWM function, reset synchronous PWM mode, complementary PWM mode, and PWM3 mode, the pulse output from the TRDIO $_{0j}$  output pin ( $i = 0$  or  $1$ ,  $j = A, B, C$ , or  $D$ ) can be cut off by the INTP0 pin input. The pins used for output in these functions or modes can function as the output pin of timer RD when the corresponding bit in the TRDOER1 register is set to 0 (timer RD output enabled). When the TRDPTO bit in the TRDOER2 register is 1 (pulse output forced cutoff signal INTP0 pin input enabled), the output pin used as a timer RD output port outputs the output value set by the DFCK1, DFCK0, PENB1, PENB0, DFD, DFC, DFB, or DFA bit in the TRDDF0 or TRDDF1 register.

Make the following settings to use this function:

- Set the pin state when the pulse output is forcibly cut off (high impedance, low output, or high output) using TRDDFi.
- Refer to **8.4.5 Event Input from Event Link Controller (ELC)** for details on pulse forced cutoff by ELC event input.
- When pulse output is forcibly cut out, the TRDSHUTS bit in the TRDOER2 register is set to 1. To suspend the forced cutoff of the pulse output, set the TRDSHUTS bit to 0 while the count is stopped ( $TSTART_i = 0$ ).
- Set the TRDPTO bit in the TRDOER2 register to 1 (pulse output forced cutoff signal INTP0 pin input enabled).

Figure 8 - 46 Pulse Output Forced Cutoff



ELCOBE0, ELCOBE1: Bits in TRDELC register  
 TRDPTO, TRDSHUTS: Bits in TRDOER2 register  
 PM10, PM11, PM15, PM17: Bits in PM1 register  
 DFCK1, DFCK0, PENB1, PENB0: Bits in TRDDF0 register  
 DFD, DFC, DFB, DFA: Bits in TRDDF1 register

### 8.4.5 Event Input from Event Link Controller (ELC)

Timer RD performs two operations by event input from the ELC.

(a) TRDIOD0/TRDIOD1 input capture

Timer RD captures the TRDIOD0/TRDIOD1 input when an event is input from the ELC. The IMFD bit in the TRDSR<sub>i</sub> register is set to 1 at this time. To use this function, select the input capture function in timer mode and set the ELCICE0 or ELCICE1 bit in the TRDELC register to 1. This function is disabled in any other modes (for the output compare function in timer mode, PWM function, reset synchronous PWM mode, complementary PWM mode, and PWM3 mode).

(b) Pulse output forced cutoff operation **Note**

The pulse output is forcibly cutoff by event input from the ELC. To use this function, select pulse output mode (PWM function, reset synchronous PWM mode, complementary PWM mode, or PWM3 mode) and set the ELCOBE0 or ELCOBE1 bit to 1. This function is disabled for the input capture function in timer mode.

**Note** The pulse output is cutoff during the low input period for forced cutoff from the INTP0 pin, but the pulse output is cutoff once by a single event input from the ELC for forced cutoff by the ELC event.

[Setting Procedure]

- (1) Set timer RD as the ELC event link destination.
- (2) Set bits ELCICE<sub>i</sub> ( $i = 0$  or  $1$ ) and ELCOBE<sub>i</sub> ( $i = 0$  or  $1$ ) to 1 in the TRDELC register.

### 8.4.6 Event Output to Event Link Controller (ELC)/Data Transfer Controller (DTC)

Table 8 - 11 lists the Timer RD Modes and Event Output to ELC/DTC.

**Table 8 - 11 Timer RD Modes and Event Output to ELC/DTC**

Used Mode	Output Source	ELC	DTC
Input capture function	TRDIOA0 edge detection set by bits IOA1 and IOA0 in the TRDIORA0 register	Available	Available
	TRDIOB0 edge detection set by bits IOB1 and IOB0 in the TRDIORA0 register	Available	Available
	TRDIOC0 edge detection set by bits IOC1 and IOC0 in the TRDIORC0 register	—	Available
	TRDIOD0 edge detection set by bits IOD1 and IOD0 in the TRDIORC0 register	—	Available
	TRDIOA1 edge detection set by bits IOA1 and IOA0 in the TRDIORA1 register	Available	Available
	TRDIOB1 edge detection set by bits IOB1 and IOB0 in the TRDIORA1 register	Available	Available
	TRDIOC1 edge detection set by bits IOC1 and IOC0 in the TRDIORC1 register	—	Available
	TRDIOD1 edge detection set by bits IOD1 and IOD0 in the TRDIORC1 register	—	Available
Output compare function, PWM function, reset synchronous PWM mode, complementary PWM mode, and PWM3 mode	Compare match between registers TRD0 and TRDGRA0	Available	Available
	Compare match between registers TRD0 and TRDGRB0	Available	Available
	Compare match between registers TRD0 and TRDGRC0	—	Available
	Compare match between registers TRD0 and TRDGRD0	—	Available
	Compare match between registers TRD1 and TRDGRA1	Available	Available
	Compare match between registers TRD1 and TRDGRB1	Available	Available
	Compare match between registers TRD1 and TRDGRC1	—	Available
	Compare match between registers TRD1 and TRDGRD1	—	Available
Complementary PWM mode	TRD1 register underflow	Available	—

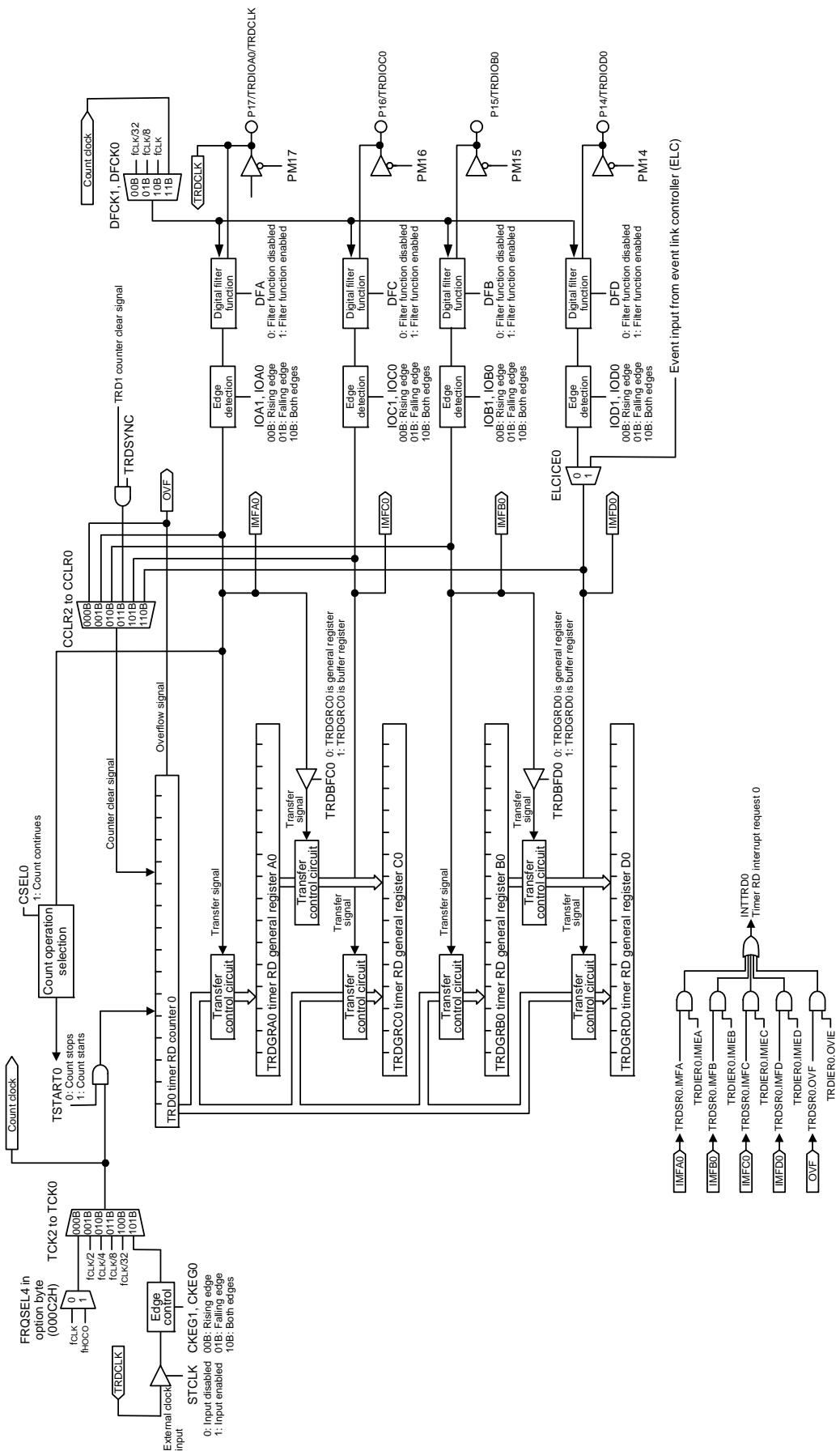
## 8.5 Timer RD Operation

### 8.5.1 Input Capture Function

The input capture function measures the external signal width and period. The content of the TRDi register (counter) is transferred to the TRDGRji register as a trigger of the TRDIOji pin ( $i = 0$  or  $1$ ,  $j = A, B, C$ , or  $D$ ) external signal (input capture). Since this function is enabled with a combination of the TRDIOji pin and TRDGRji register, the input capture function, or any other mode or function, can be selected for each individual pin.

Figure 8 - 47 shows the Block Diagram of Input Capture Function, Table 8 - 12 lists the Input Capture Function Specifications, and Figure 8 - 48 shows an Operation Example of Input Capture Function.

Figure 8 - 47 Block Diagram of Input Capture Function



**Table 8 - 12 Input Capture Function Specifications**

Item	Specification
Count sources	fHO CO Note, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRDCLK pin (active edge selected by a program)
Count operations	Increment
Count period	When bits CCLR2 to CCLR0 in the TRDCRi register are set to 000B (free-running operation). $1/fk \times 65536$ fk: Frequency of count source
Count start condition	1 (count starts) is written to the TSTARTi bit in the TRDSTR register.
Count stop condition	0 (count stops) is written to the TSTARTi bit in the TRDSTR register when the CSELi bit in the TRDSTR register is set to 1.
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Input capture (active edge of TRDIOji input)</li> <li>TRDi register overflow</li> </ul>
TRDIOA0 pin function	I/O port, input-capture input, or TRDCLK (external clock) input
TRDIOB0, TRDIOC0, TRDIOD0, TRDIOA1 to TRDIOD1 pin function	I/O port or input-capture input (selectable for each pin)
INTP0 pin function	Not used (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRDi register.
Write to timer	<ul style="list-style-type: none"> <li>When the TRDSYNC bit in the TRDMR register is 0 (timer RD0 and timer RD1 operate independently). Data can be written to the TRDi register.</li> <li>When the TRDSYNC bit in the TRDMR register is 1 (timer RD0 and timer RD1 operate synchronously). Data can be written to both the TRD0 and TRD1 registers by writing to the TRDi register.</li> </ul>
Selectable functions	<ul style="list-style-type: none"> <li>Input-capture input pin selection Either one pin or multiple pins of TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi.</li> <li>Input-capture input active edge selection Rising edge, falling edge, or both rising and falling edges</li> <li>Timing for setting the TRDi register to 0000H. At overflow or input capture</li> <li>Buffer operation (see <b>8.4.2 Buffer Operation</b>)</li> <li>Synchronous operation (see <b>8.4.3 Synchronous Operation</b>)</li> <li>Digital filter. The TRDIOji input is sampled, and when the sampled input level match three times, that level is determined.</li> <li>Input capture operation by event input from ELC.</li> </ul>

**Note** fHO CO is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHO CO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

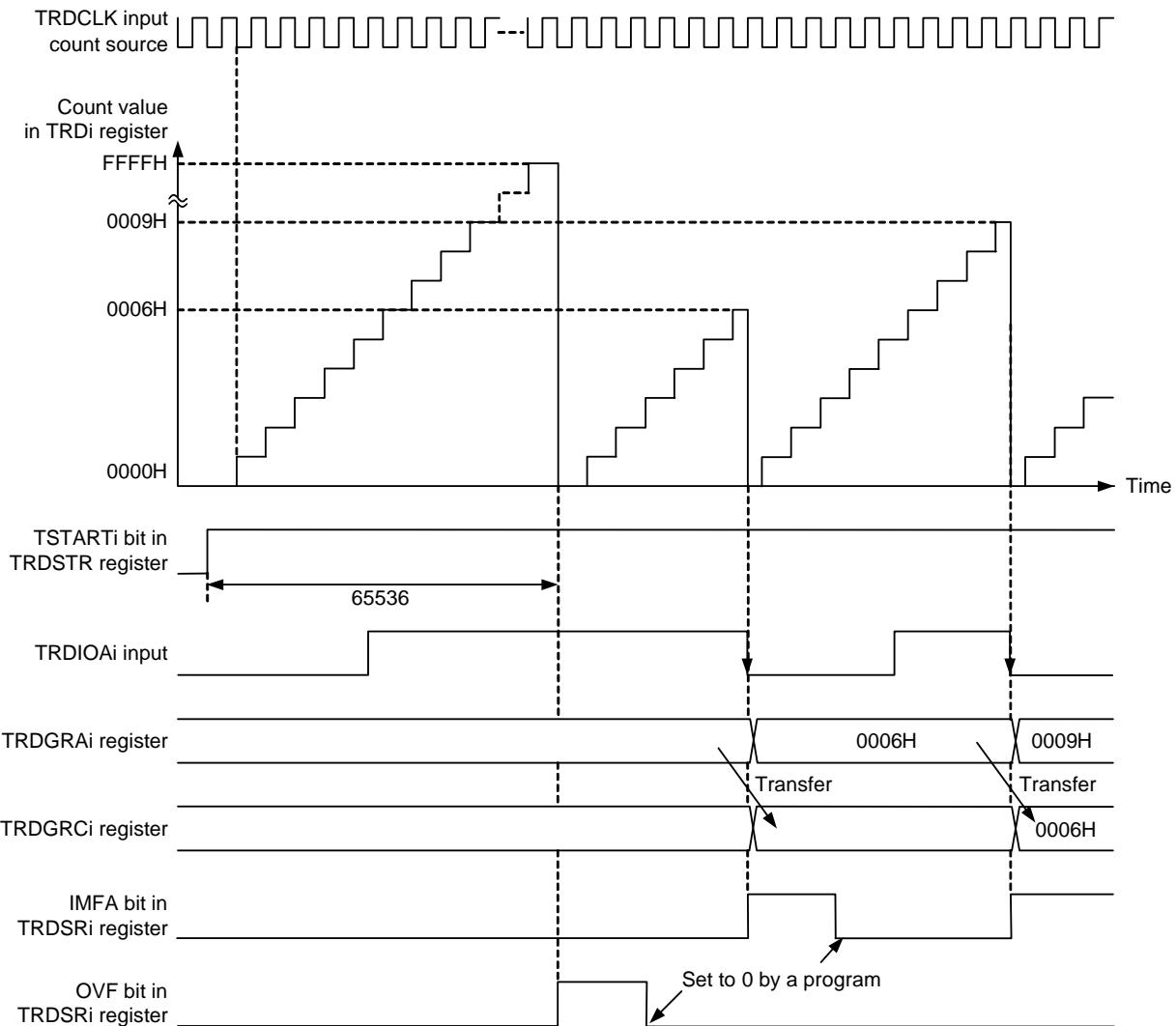
**Remark** i = 0 or 1, j = A, B, C, or D

## (1) Operation Example

By setting bits CCLR2 to CCLR0 in the TRDCR*i* register (*i* = 0 or 1), the timer RD*i* counter value is reset by an input capture/compare match. Figure 8 - 48 shows an operation example with bits CCLR2 to CCLR0 set to 001B.

If the input capture operation has been set to clear the count during operation and is performed when the timer count value is FFFFH, depending on the timing between the count source and input capture operation interrupt flags bits IMFA to IMFD and OVF in the TRDSR*i* register may be set to 1 simultaneously.

**Figure 8 - 48 Operation Example of Input Capture Function**



Remark  
*i* = 0 or 1

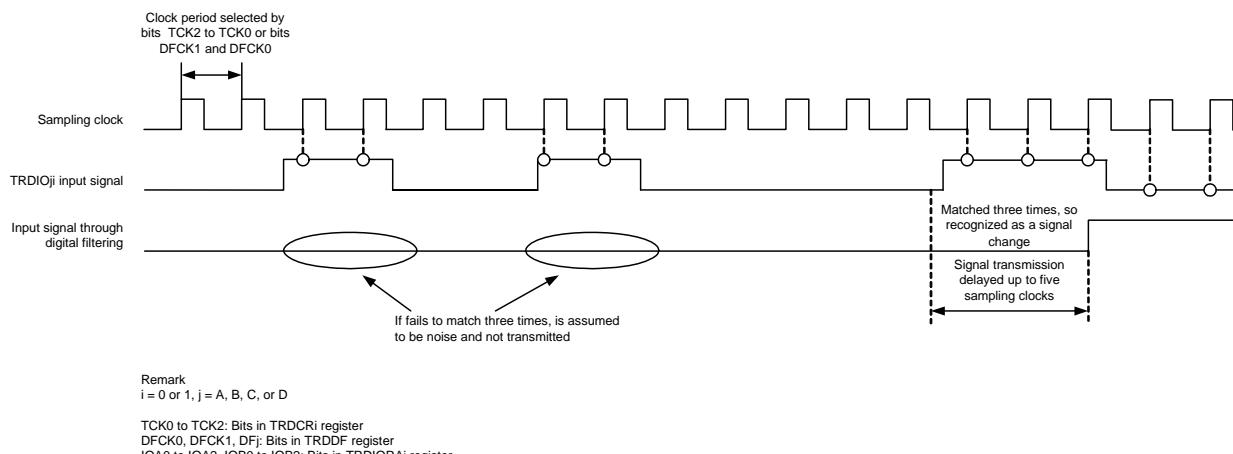
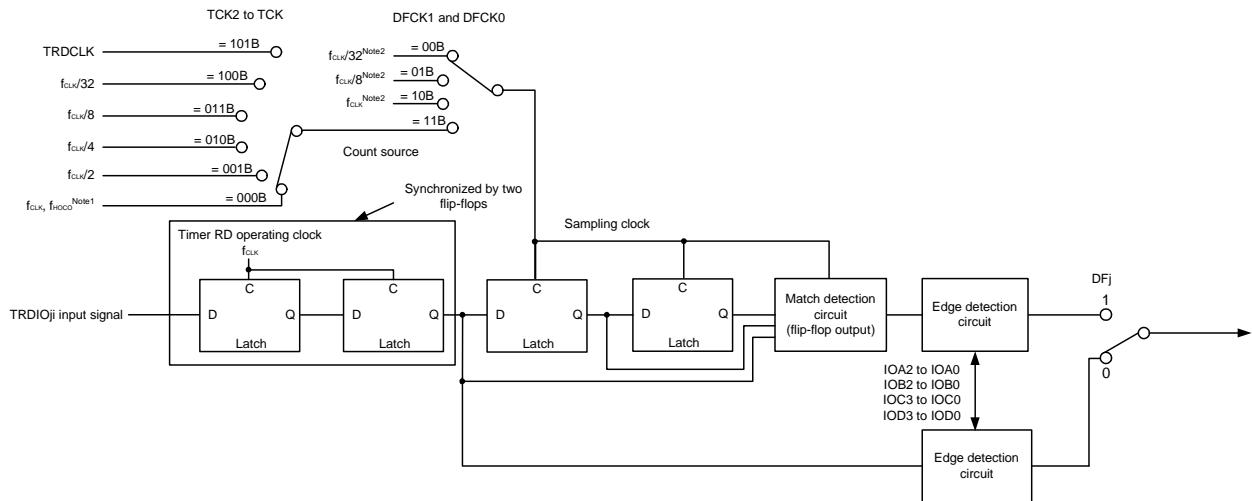
The above diagram applies under the following conditions

Bits CCLR2 to CCLR0 in the TRDCR*i* register are set to 001B (TRD*i* register is set to 0000H by TRDGRA*i* register input capture).  
Bits TCK2 to TCK0 in the TRDCR*i* register are set to 101B (TRDCLK input for the count source).  
Bits CKEG1 and CKEG0 in the TRDCR*i* register are set to 01B (count at the falling edge for the count source).  
Bits IOA2 to IOA0 in the TRDIORA*i* register are set to 101B (input capture at the falling edge of TRDIOAi input).  
The TRDBFC*i* bit in the TRDMR register is set to 1 (TRDGRC*i* register is buffer register for TRDGRA*i* register).

## (2) Digital Filter

The TRDIO<sub>ji</sub> input ( $i = 0$  or  $1$ ,  $j = A, B, C$ , or  $D$ ) is sampled, and when the sampled input level matches three times, its level is determined. Select the digital filter function and sampling clock using the TRDDFi register. Figure 8 - 49 shows the Block Diagram of Digital Filter.

Figure 8 - 49 Block Diagram of Digital Filter



**Note 1.** f<sub>CLK</sub> is selected when FRQSEL4 = 0 and f<sub>HO</sub> is selected when FRQSEL4 = 1 in the user option byte (000C2H).

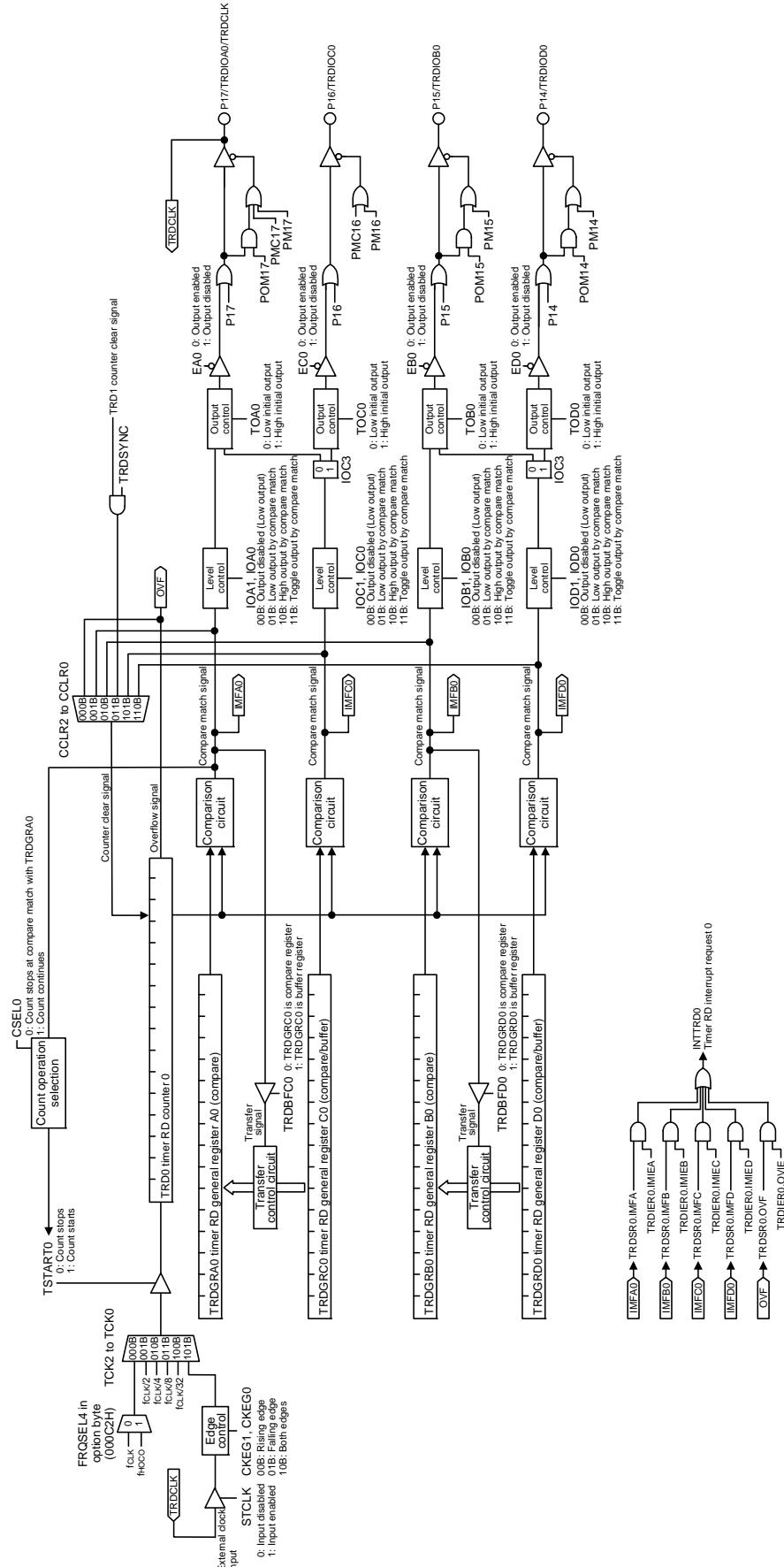
**Note 2.** When FRQSEL4 = 1 in the user option byte (000C2H), f<sub>CLK</sub>/32, f<sub>CLK</sub>/8, and f<sub>CLK</sub> are set to f<sub>HO</sub>/32, f<sub>HO</sub>/8, and f<sub>HO</sub>, respectively.

### 8.5.2 Output Compare Function

This function detects matches (compare match) between the content of the TRDGR $ji$  register ( $j = A, B, C, \text{ or } D$ ) and the content of the TRDI $i$  register (counter) ( $i = 0 \text{ or } 1$ ). When the contents match, an arbitrary level is output from the TRDIO $ji$  pin. Since this function is enabled with a combination of the TRDIO $ji$  pin and TRDGR $ji$  register, the output compare function, or any other mode or function, can be selected for each individual pin.

Figure 8 - 50 shows the Block Diagram of Output Compare Function, Table 8 - 13 lists the Output Compare Function Specifications, and Figure 8 - 51 shows an Operation Example of Output Compare Function.

**Figure 8 - 50 Block Diagram of Output Compare Function**



**Table 8 - 13 Output Compare Function Specifications**

Item	Specification
Count sources	fHOCO Note, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRDCLK pin (active edge selected by a program)
Count operations	Increment
Count period	<ul style="list-style-type: none"> <li>When bits CCLR2 to CCLR0 in the TRDCRi register are set to 000B (free-running operation). <math>1/fk \times 65536</math> fk: Frequency of count source</li> <li>When bits CCLR1 and CCLR0 in the TRDCRi register are set to 01B or 10B (TRDi register is set to 0000H at compare match with TRDGRji register). <math>1/fk \times (n + 1)</math> n: Value set in the TRDGRji register</li> </ul>
Waveform output timing	Compare match (contents of registers TRDi and TRDGRji match)
Count start condition	1 (count starts) is written to the TSTARTi bit in the TRDSTR register.
Count stop conditions	<ul style="list-style-type: none"> <li>0 (count stops) is written to the TSTARTi bit in the TRDSTR register when the CSELi bit in the TRDSTR register is set to 1. The output compare output pin holds the output level before the count stops.</li> <li>When the CSELi bit in the TRDSTR register is set to 0, the count stops at the compare match with the TRDGRAi register. The output compare output pin holds the level after output change by compare match.</li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (contents of registers TRDi and TRDGRji match)</li> <li>TRDi register overflow</li> </ul>
TRDIOA0 pin function	I/O port, output-compare output, or TRDCLK (external clock) input
TRDIOB0, TRDILOC0, TRDIOD0, TRDIOA1 to TRDIOD1 pin function	I/O port or output-compare output (selectable for each pin)
INTP0 pin function	Not used (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRDi register.
Write to timer	<ul style="list-style-type: none"> <li>When the TRDSYNC bit in the TRDMR register is set to 0 (timer RD0 and timer RD1 operate independently). Data can be written to the TRDi register.</li> <li>When the TRDSYNC bit in the TRDMR register is set to 1 (timer RD0 and timer RD1 operate synchronously). Data can be written to both the TRD0 and TRD1 registers by writing to the TRDi register.</li> </ul>
Selectable functions	<ul style="list-style-type: none"> <li>Output-compare output pin selection Either one pin or multiple pins of TRDIOAi, TRDIOBi, TRDILOCi, and TRDIODi.</li> <li>Output level selection at compare match Low output, high output, or inverted output level</li> <li>Initial output level selection The level can be set for the period from the count start to the compare match.</li> <li>Timing for setting the TRDi register to 0000H Overflow or compare match in the TRDGRAi register</li> <li>Buffer operation (see <b>8.4.2 Buffer Operation</b>)</li> <li>Synchronous operation (see <b>8.4.3 Synchronous Operation</b>)</li> <li>Changing output pins for registers TRDGRCi and TRDGRDi The TRDGRCi register can be used as output control of the TRDIOAi pin and the TRDGRDi register can be used as output control of the TRDIOBi pin.</li> <li>Timer RD can be used as the internal timer without output.</li> </ul>

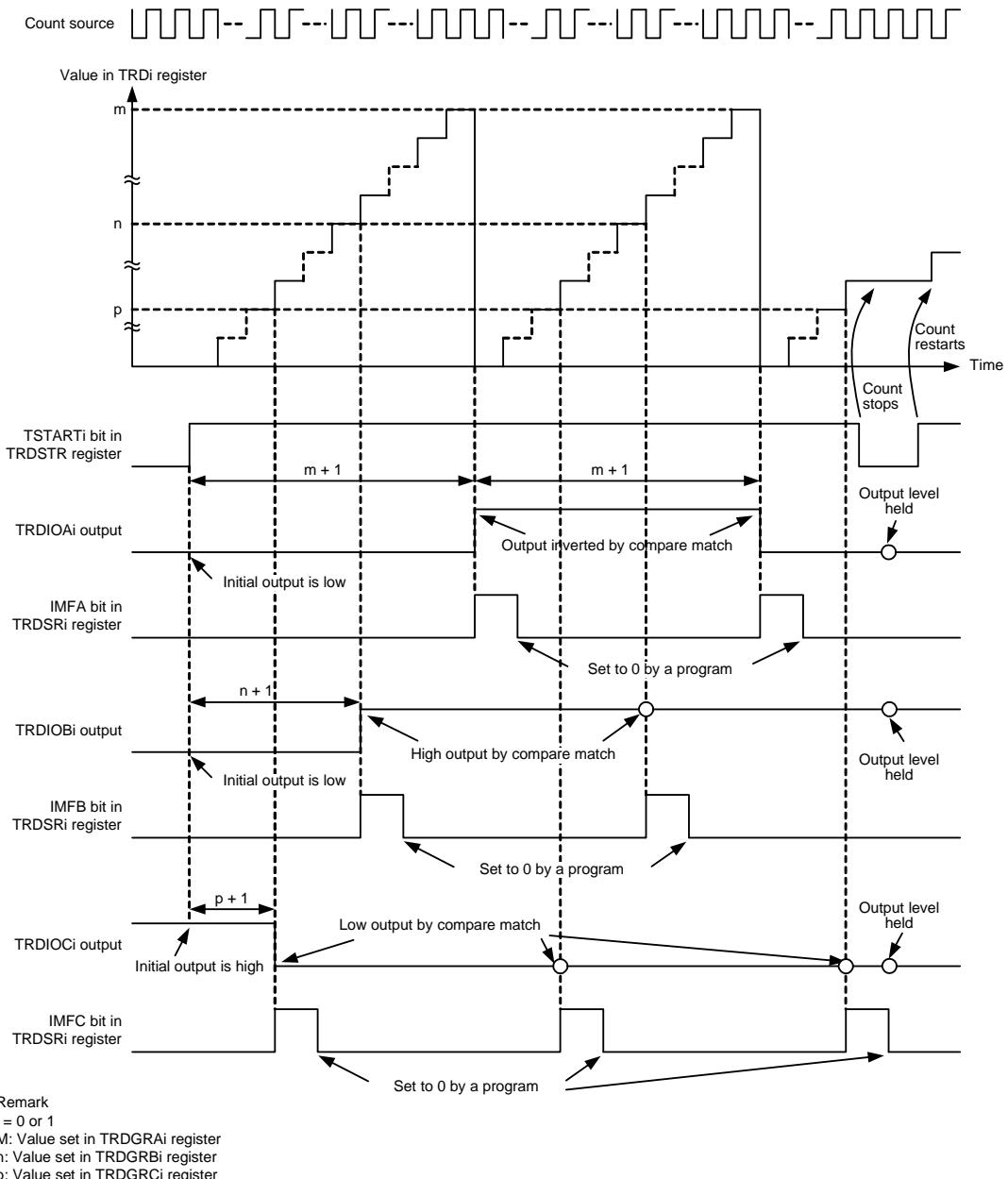
**Note** fHOCO is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHOCO as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Remark** i = 0 or 1, j = A, B, C, or D

### (1) Operation Example

By setting bits CCLR0 to CCLR2 in the TRDCR*i* register (*i* = 0 or 1), the timer RD*i* counter value is reset by an input capture/compare match. If the expected compare value is FFFFH at this time, FFFFH changes to 0000H, same as the overflow operation, and the overflow flag is set to 1.

**Figure 8 - 51 Operation Example of Output Compare Function**



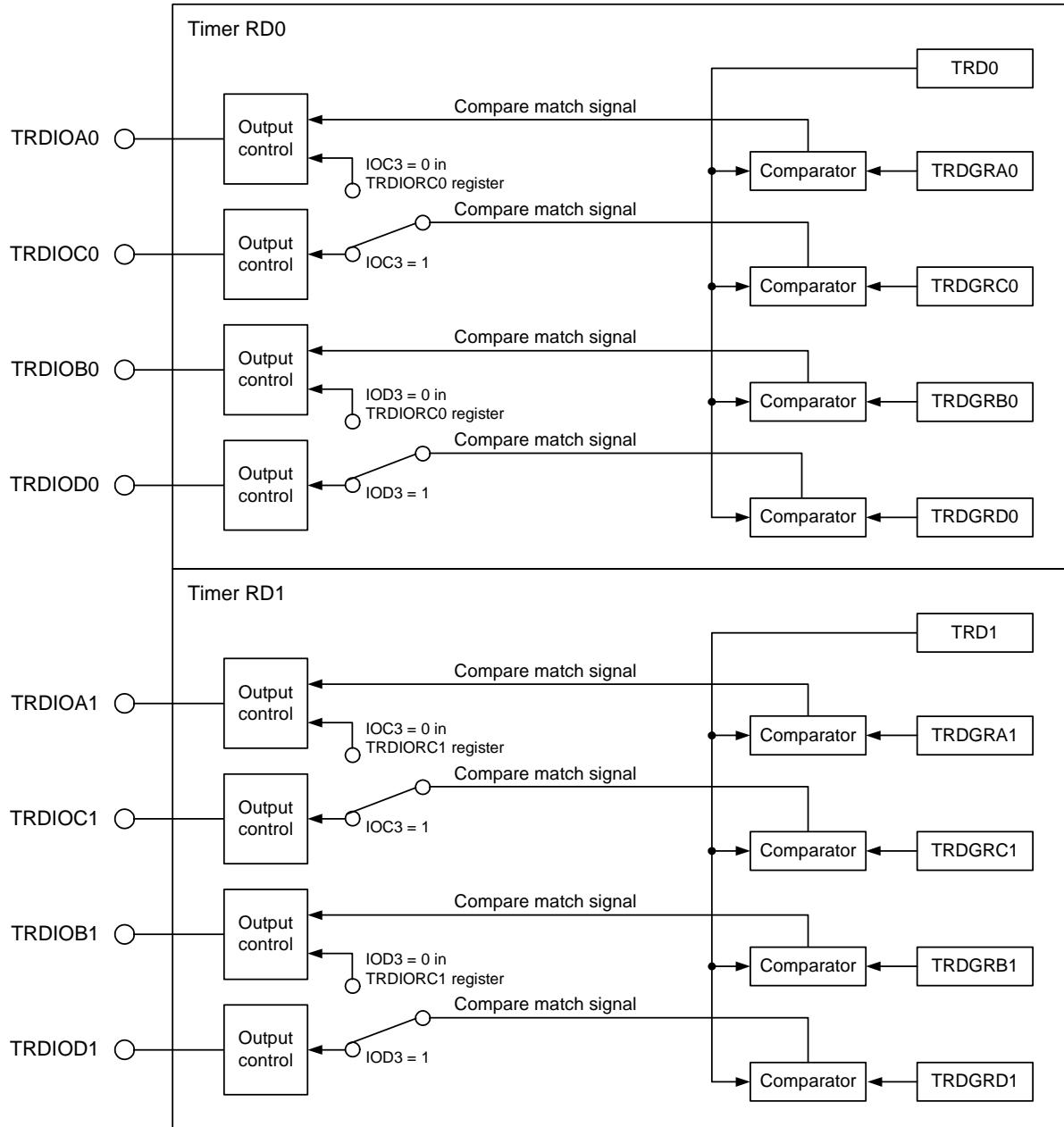
The above diagram applies under the following conditions:  
The CSELi bit in the TRDSTR register is set to 1 (TRDi is not stopped by compare match).  
Bits TRDBFC*i* and TRDBFBD*i* in the TRDMR register are set to 0 (TRDGRC*i* and TRDGRD*i* do not operate as buffers).  
Bits EA*i*, EB*i*, and EC*i* in the TRDOER1 register are set to 0 (TRDIOAi, TRDIOBi and TRDIOC*i* output enabled).  
Bits CCLR2 to CCLR0 in the TRDCR*i* register are set to 001B (TRDi is set to 0000H by compare match with TRDGRA*i*).  
Bits TOAi and TOBi in the TRDOCR register is set to 0 (initial output is low until compare match), the TOCi bit is set to 1 (initial output is high until compare match).  
Bits IOA2 to IOA0 in the TRDIORA*i* register are set to 011B (TRDIOAi output inverted at TRDGRA*i* compare match).  
Bits IOB2 to IOB0 in the TRDIORA*i* register are set to 010B (TRDIOBi high output at TRDGRBi compare match).  
Bits IOC3 to IOC0 in the TRDIORC*i* register are set to 1001B (TRDIOC*i* low output at TRDGRC*i* register compare match).  
Bits IOD3 to IOD0 in the TRDIORC*i* register are set to 1000B (TRDGRD*i* register does not control TRDIOBi pin output. Pin output by compare match is disabled).

(2) Changing Output Pins in Registers TRDGRC<sub>i</sub> (i = 0 or 1) and TRDGRD<sub>i</sub>

The TRDGRC<sub>i</sub> register can be used for output control of the TRDIOAi pin, and the TRDGRD<sub>i</sub> register can be used for output control of the TRDIObi pin. Therefore, each pin output can be controlled as follows:

- TRDIOAi output is controlled by the values in registers TRDGRA<sub>i</sub> and TRDGRC<sub>i</sub>.
- TRDIObi output is controlled by the values in registers TRDGRBi and TRDGRD<sub>i</sub>.

**Figure 8 - 52 Changing Output Pins in Registers TRDGRC<sub>i</sub> and TRDGRD<sub>i</sub>**

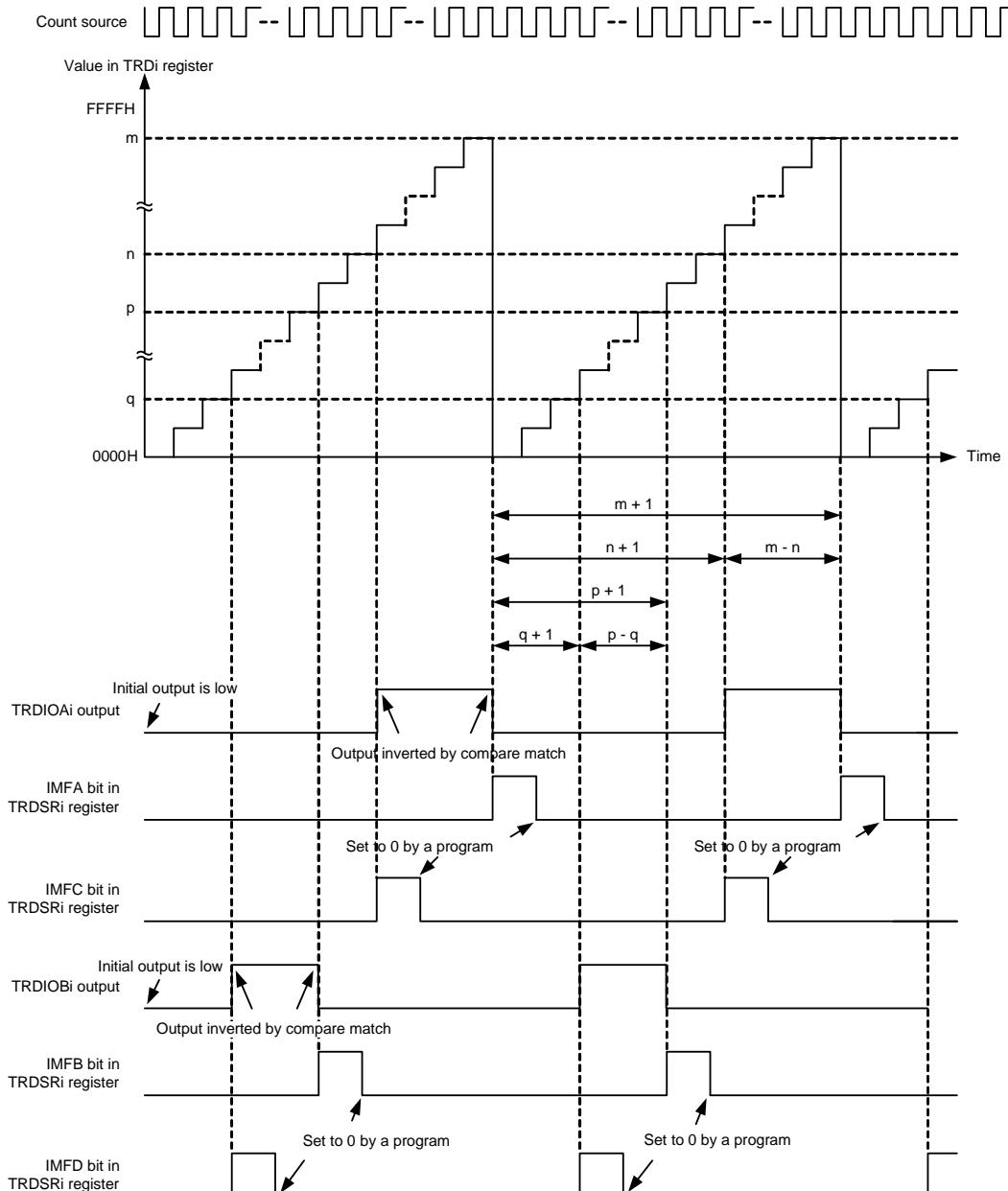


Change output pins in registers TRDGRC<sub>i</sub> and TRDGRD<sub>i</sub> as follows:

- Select 0 (TRDGRji register output pin is changed) using the IOj3 (j = C or D) bit in the TRDIORCi register.
- Set the TRDBFji bit in the TRDMR register to 0 (general register).
- Set different values in registers TRDGRC<sub>i</sub> and TRDGRA<sub>i</sub>. Also, set different values in registers TRDGRD<sub>i</sub> and TRDGRBi.

Figure 8 - 53 shows an Operation Example When TRDGRCi Register is Used for Output Control of TRDIOAi Pin and TRDGRDi Register is Used for Output Control of TRDIOBi Pin.

**Figure 8 - 53 Operation Example When TRDGRCi Register is Used for Output Control of TRDIOAi Pin and TRDGRDi Register is Used for Output Control of TRDIOBi Pin**



#### Remark

- i = 0 or 1
- m: Value set in TRDGRAi register
- n: Value set in TRDGRCi register
- p: Value set in TRDGRBi register
- q: Value set in TRDGRDi register

The above diagram applies under the following conditions :

- The CSEL1 bit in the TRDSTR register is set to 1 (TRDi register is not stopped by compare match).
- Bits TRDBFCi and TRDBFDi in the TRDMR register are set to 0 (TRDGRCi and TRDGRDi do not operate as buffers).
- Bits EAi and EBi in the TRDOER1 register are set to 0 (TRDIOAi and TRDIOBi output enabled).
- Bits CCLR2 to CCLR4 in the TRDCR register are set to 001B (TRDi is set to 0000H by compare match with TRDGRAi).
- Bits TOAi and TOBi in the TRDOCR register are set to 0 (initial output is low until compare match).
- Bits IOA2 to IOA0 in the TRDIORAi register are set to 011B (TRDIOAi output inverted at TRDGRAi compare match).
- Bits IOB2 to IOB0 in the TRDIORAi register are set to 011B (TRDIOBi output inverted at TRDGRBi compare match).
- Bits IOC3 to IOC0 in the TRDIORCi register are set to 0011B (TRDIOAi output inverted at TRDGRCi compare match).
- Bits IOD3 to IOD0 in the TRDIORCi register are set to 0011B (TRDIOBi output inverted at TRDGRDi compare match).

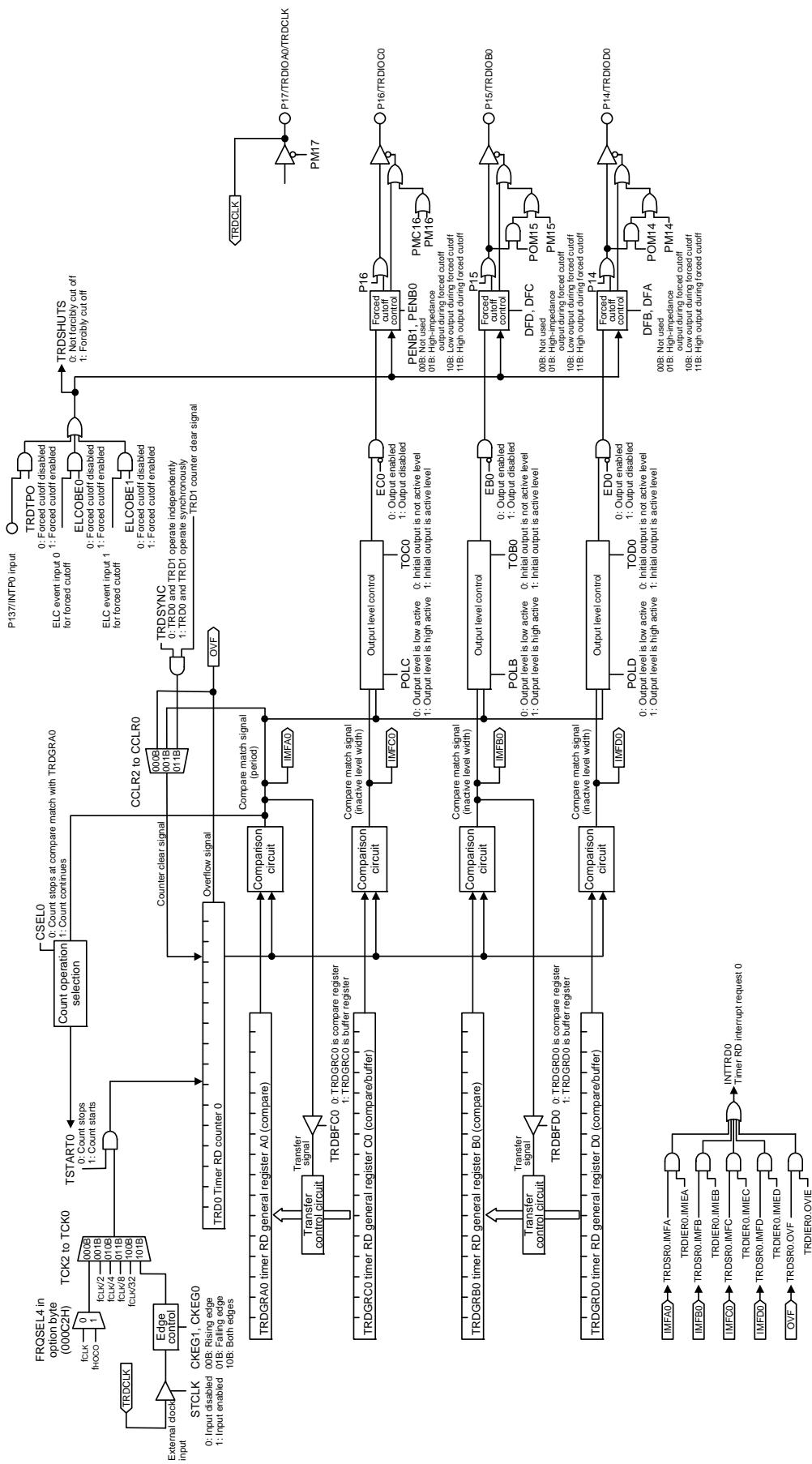
### 8.5.3 PWM Function

In PWM function, a PWM waveform is output. Up to three PWM waveforms with the same period can be output by timer RD $i$  ( $i = 0$  or  $1$ ). Also, up to six PWM waveforms with the same period can be output by synchronizing timer RD0 and timer RD1.

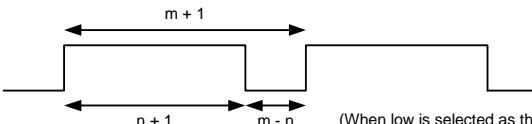
Since this mode functions by a combination of the TRDIO $_{j}i$  pin ( $i = 0$  or  $1$ ,  $j = B$ ,  $C$ , or  $D$ ) and TRDGR $_{j}i$  register, PWM function, or any other mode or function, can be selected for each individual pin. (However, since the TRDGRA $i$  register is used when using any pin for PWM function, the TRDGRA $i$  register cannot be used for other modes.)

Figure 8 - 54 shows the Block Diagram of PWM Function, Table 8 - 14 lists the PWM Mode Specifications, and Figures 8 - 55 and 8 - 56 show Operation Examples in PWM Function.

Figure 8 - 54 Block Diagram of PWM Function



**Table 8 - 14 PWM Mode Specifications**

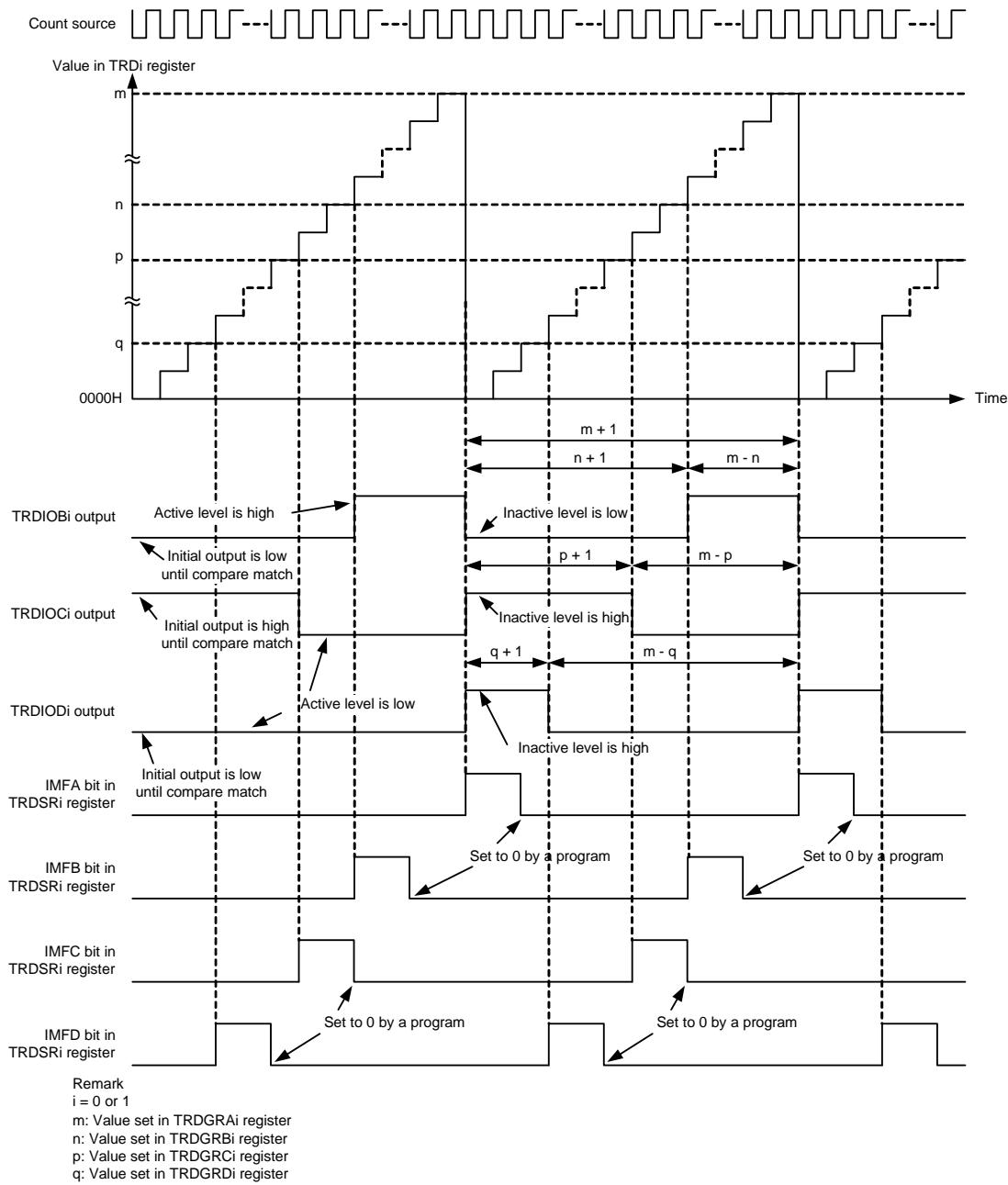
Item	Specification
Count sources	fHOOC Note, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRDCLK pin (active edge selected by a program)
Count operations	Increment
PWM waveform	<p>PWM period: <math>1/fk \times (m + 1)</math>            Active level width: <math>1/fk \times (m - n)</math>            Inactive level width: <math>1/fk \times (n + 1)</math>            fk: Frequency of count source            m: Value set in the TRDGRAi register            n: Value set in the TRDGRji register</p> 
Count start condition	1 (count starts) is written to the TSTARTi bit in the TRDSTR register.
Count stop conditions	<ul style="list-style-type: none"> <li>0 (count stops) is written to the TSTARTi bit in the TRDSTR register when the CSELi bit in the TRDSTR register is set to 1. The PWM output pin holds the output level before the count stops.</li> <li>When the CSELi bit in the TRDSTR register is set to 0, the count stops at the compare match with the TRDGRAi register. The PWM output pin holds the level after output change by compare match.</li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (content of the TRDi register matches content of the TRDGRhi register)</li> <li>TRDi register overflow</li> </ul>
TRDIOA0 pin function	I/O port or TRDCLK (external clock) input
TRDIOA1 pin function	I/O port
TRDIOB0, TRDI0C0, TRDI0D0, TRDIOB1, TRDI0C1, TRDI0D1 pin function	I/O port or pulse output (selectable for each pin)
INTP0 pin function	Pulse output forced cutoff signal input (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRDi register.
Write to timer	The value can be written to the TRDi register.
Selectable functions	<ul style="list-style-type: none"> <li>One to three PWM output pins selectable with timer RDi Either one pin or multiple pins of TRDIOBi, TRDI0Ci, and TRDI0Di.</li> <li>Active level selectable for each pin.</li> <li>Initial output level selectable for each pin.</li> <li>Synchronous operation (see <b>8.4.3 Synchronous Operation</b>)</li> <li>Buffer operation (see <b>8.4.2 Buffer Operation</b>)</li> <li>Pulse output forced cutoff signal input (see <b>8.4.4 Pulse Output Forced Cutoff</b>)</li> </ul>

**Note** fHOOC is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHOOC as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Remark** i = 0 or 1, j = B, C, or D, h = A, B, C, or D

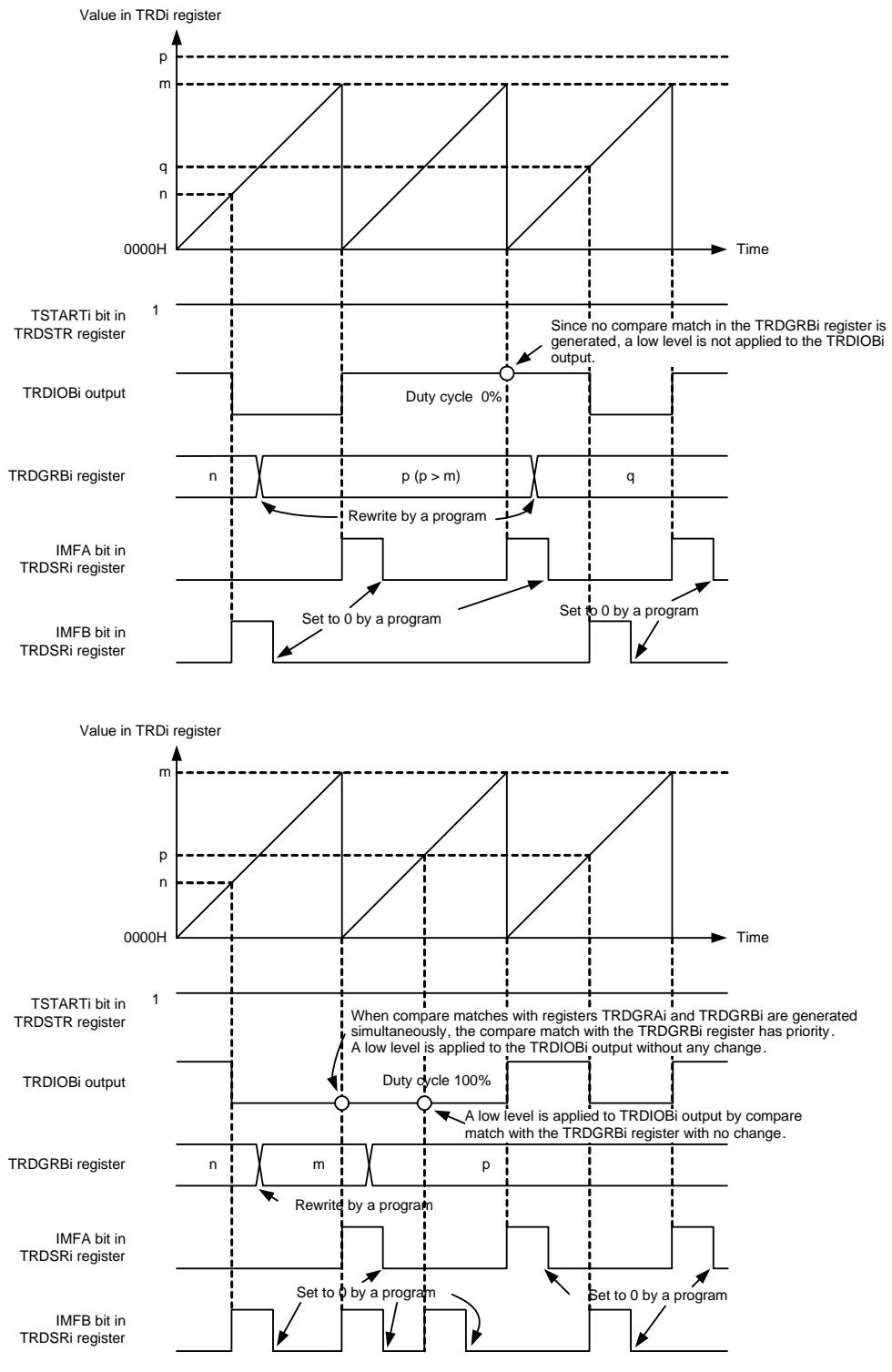
## (1) Operation Example

Figure 8 - 55 Operation Example in PWM Function



The above diagram applies under the following conditions:  
 Bits TRDBFCi and TRDBFDi in the TRDMR register are set to 0 (TRDGRCi and TRDGRDi do not operate as buffers).  
 Bits EBi, ECi, and EDi in the TRDOER1 register are set to 0 (TRDIOBi, TRDIOCi and TRDIODi output enabled).  
 Bits TOBi and TOCi in the TRDOOCR register are set to 0 (inactive level), the Todi bit is set to 1 (active level).  
 The POLB bit in the TRDPOCRi register is set to 1 (active level is high), bits POLC and POLD are set to 0 (active level is low).

Figure 8 - 56 Operation Example in PWM Function (Duty Cycle 0%, Duty Cycle 100%)

**Remark****i = 0 or 1****m: Value set in TRDGRA<sub>i</sub> register****The above diagram applies under the following conditions :****The EBI bit in the TRDOER<sub>1</sub> register is set to 0 (TRDIOBi output enabled).****The POLB bit in the TRDPOCR<sub>i</sub> register is set to 0 (active level is low).**

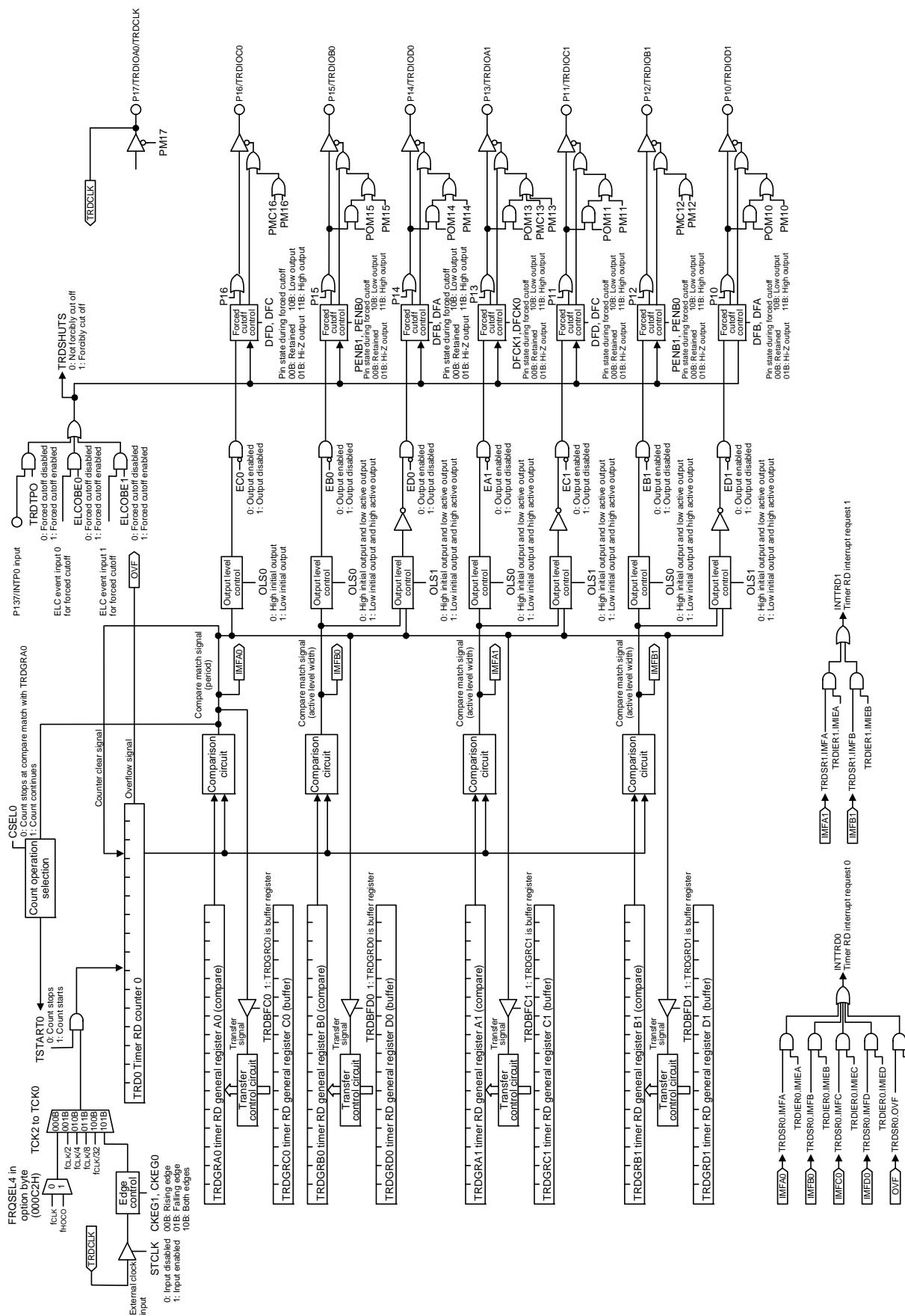
### 8.5.4 Reset Synchronous PWM Mode

In this mode, three normal-phases and three counter-phases of the PWM waveform are output with the same period (three-phase, sawtooth wave modulation, and no dead time).

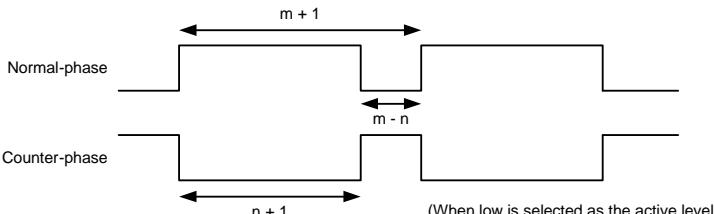
Figure 8 - 57 shows the Block Diagram of Reset Synchronous PWM Mode, Table 8 - 15 lists the Reset Synchronous PWM Mode Specifications, Figure 8 - 58 shows an Operation Example in Reset Synchronous PWM Mode.

See **Figure 8 - 56 Operation Example in PWM Function (Duty Cycle 0%, Duty Cycle 100%)** for an operation example in PWM Mode with duty cycle 0% and duty cycle 100%.

**Figure 8 - 57 Block Diagram of Reset Synchronous PWM Mode**



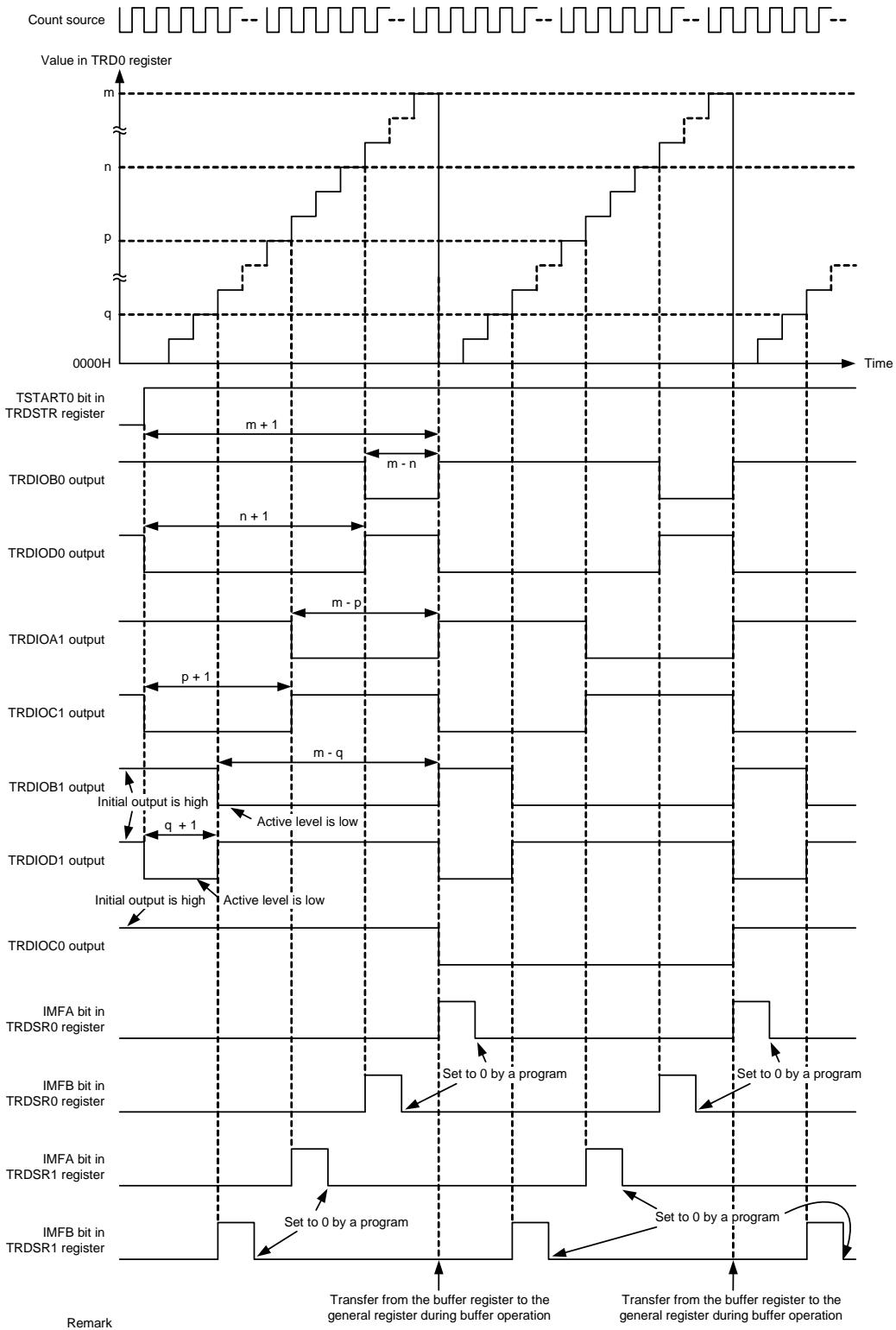
**Table 8 - 15 Reset Synchronous PWM Mode Specifications**

Item	Specification
Count sources	fHOOC Note, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRDCLK pin (active edge selected by a program)
Count operations	The TRD0 register is incremented (the TRD1 register is not used).
PWM waveform	<p>PWM period: <math>1/fk \times (m + 1)</math>            Active level of normal-phase: <math>1/fk \times (m - n)</math>            Inactive level of counter-phase: <math>1/fk \times (n + 1)</math></p> <p>fk: Frequency of count source            m: Value set in the TRDGRA0 register            n: Value set in the TRDGRB0 register (PWM1 output)            Value set in the TRDGRA1 register (PWM2 output)            Value set in the TRDGRB1 register (PWM3 output)</p>  <p>(When low is selected as the active level)</p>
Count start condition	1 (count starts) is written to the TSTART0 bit in the TRDSTR register.
Count stop conditions	<ul style="list-style-type: none"> <li>0 (count stops) is written to the TSTART0 bit when the CSEL0 bit in the TRDSTR register is set to 1.            The PWM output pin outputs the initial output level selected by bits OLS0 and OLS1 in the TRDFCR register.</li> <li>When the CSEL0 bit in the TRDSTR register is set to 0, the count stops at the compare match with the TRDGRA0 register.            The PWM output pin outputs the initial output level selected by bits OLS0 and OLS1 in the TRDFCR register.</li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (content of the TRD0 register matches content of registers TRDGRj0, TRDGRA1, and TRDGRB1)</li> <li>TRD0 register overflow</li> </ul>
TRDIOA0 pin function	I/O port or TRDCLK (external clock) input
TRDIOB0 pin function	PWM1 output normal-phase output
TRDIOD0 pin function	PWM1 output counter-phase output
TRDIOA1 pin function	PWM2 output normal-phase output
TRDIOC1 pin function	PWM2 output counter-phase output
TRDIOB1 pin function	PWM3 output normal-phase output
TRDIOD1 pin function	PWM3 output counter-phase output
TRDIOC0 pin function	Output inverted every PWM period
INTP0 pin function	Pulse output forced cutoff signal input (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRD0 register.
Write to timer	The value can be written to the TRD0 register.
Selectable functions	<ul style="list-style-type: none"> <li>The normal-phase and counter-phase active level and initial output level are selected individually.</li> <li>Buffer operation (see <b>8.4.2 Buffer Operation</b>)</li> <li>Pulse output forced cutoff signal input (see <b>8.4.4 Pulse Output Forced Cutoff</b>)</li> </ul>

**Note** fHOOC is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHOOC as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Remark** j = A, B, C, or D

## (1) Operation Example

**Figure 8 - 58 Operation Example in Reset Synchronous PWM Mode**

Remark

i = 0 or 1

m: Value set in TRDGRA0 register

n: Value set in TRDGRB0 register

p: Value set in TRDGRA1 register

q: Value set in TRDGRB1 register

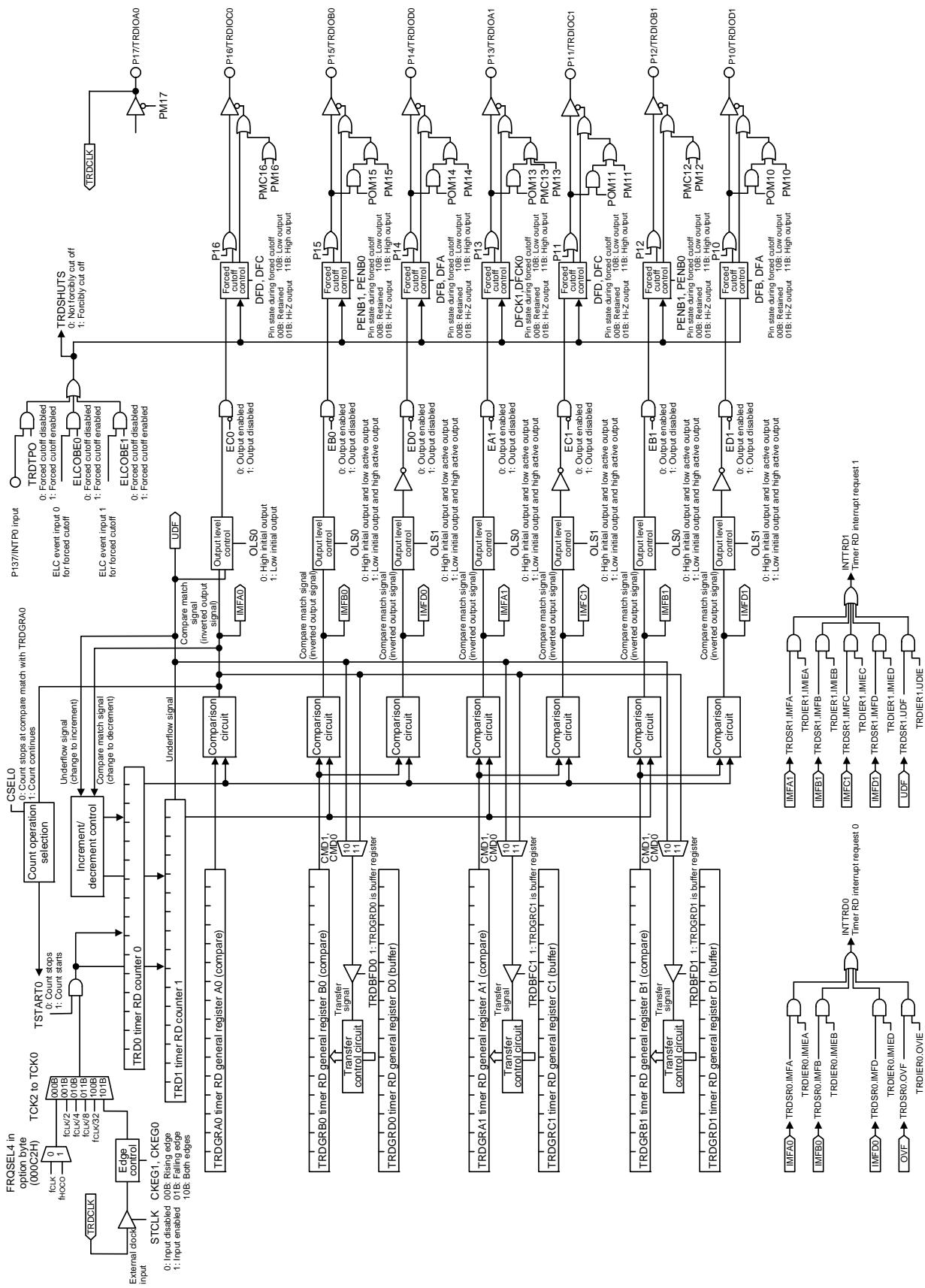
The above diagram applies under the following condition:  
Bits OLS1 and OLS0 in the TRDFCR register are set to 0 (initial output level is high, active level is low).

### 8.5.5 Complementary PWM Mode

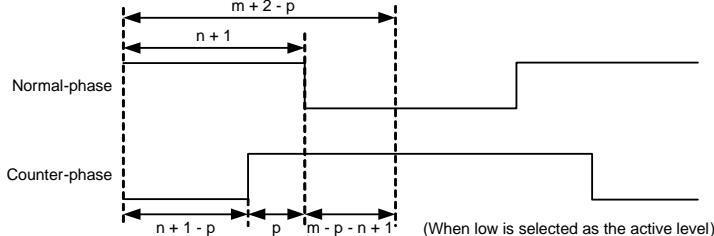
In this mode, three normal-phases and three counter-phases of the PWM waveform are output with the same period (three-phase, triangular wave modulation, and with dead time).

Figure 8 - 59 shows the Block Diagram of Complementary PWM Mode, Table 8 - 16 lists the Complementary PWM Mode Specifications, and Figure 8 - 60 shows the Output Model of Complementary PWM Mode, and Figure 8 - 61 shows an Operation Example in Complementary PWM Mode.

**Figure 8 - 59 Block Diagram of Complementary PWM Mode**



**Table 8 - 16 Complementary PWM Mode Specifications**

Item	Specification
Count sources	fHOOC Note 1, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRDCLK pin (active edge selected by a program) Set bits TCK2 to TCK0 in the TRDCR1 register to the same value (same count source) as bits TCK2 to TCK0 in the TRDCR0 register.
Count operations	Increment or decrement. Registers TRD0 and TRD1 are decremented with the compare match with registers TRD0 and TRDGRA0 during increment operation. When the TRD1 register changes from 0000H to FFFFH during decrement operation, and registers TRD0 and TRD1 are incremented.
PWM operations	PWM period: $1/fk \times (m + 2 - p) \times 2$ Note 2 Dead time: p Active level width of normal-phase: $1/fk \times (m - n - p + 1) \times 2$ Active level width of counter-phase: $1/fk \times (n + 1 - p) \times 2$ fk: Frequency of count source m: Value set in the TRDGRA0 register n: Value set in the TRDGRB0 register (PWM1 output) Value set in the TRDGRA1 register (PWM2 output) Value set in the TRDGRB1 register (PWM3 output) p: Value set in the TRD0 register  
Count start condition	1 (count starts) is written to bits TSTART0 and TSTART1 in the TRDSTR register.
Count stop condition	0 (count stops) is written to bits TSTART0 and TSTART1 in the TRDSTR register when the CSEL0 bit in the TRDSTR register is set to 1. (The PWM output pin outputs the initial output level selected by bits OLS0 and OLS1 in the TRDFCR register.)
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (content of the TRDi register matches content of the TRDGRji register)</li> <li>TRD1 register underflow</li> </ul>
TRDIOA0 pin function	I/O port or TRDCLK (external clock) input
TRDIOB0 pin function	PWM1 output normal-phase output
TRDIOD0 pin function	PWM1 output counter-phase output
TRDIOA1 pin function	PWM2 output normal-phase output
TRDIOC1 pin function	PWM2 output counter-phase output
TRDIOB1 pin function	PWM3 output normal-phase output
TRDIOD1 pin function	PWM3 output counter-phase output
TRDIOD0 pin function	Output inverted every 1/2 period of PWM
INTP0 pin function	Pulse output forced cutoff signal input (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRDi register.
Write to timer	The value can be written to the TRDi register.
Selectable functions	<ul style="list-style-type: none"> <li>Pulse output forced cutoff signal input (see <b>8.4.4 Pulse Output Forced Cutoff</b>)</li> <li>The normal-phase and counter-phase active level and initial output level are selected individually.</li> <li>Transfer timing from the buffer register selection</li> </ul>

**Note 1.** fHOOC is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHOOC as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Note 2.** After a count starts, the PWM period is fixed.

**Remark** i = 0 or 1, j = A, B, C, or D

## (1) Operation Example

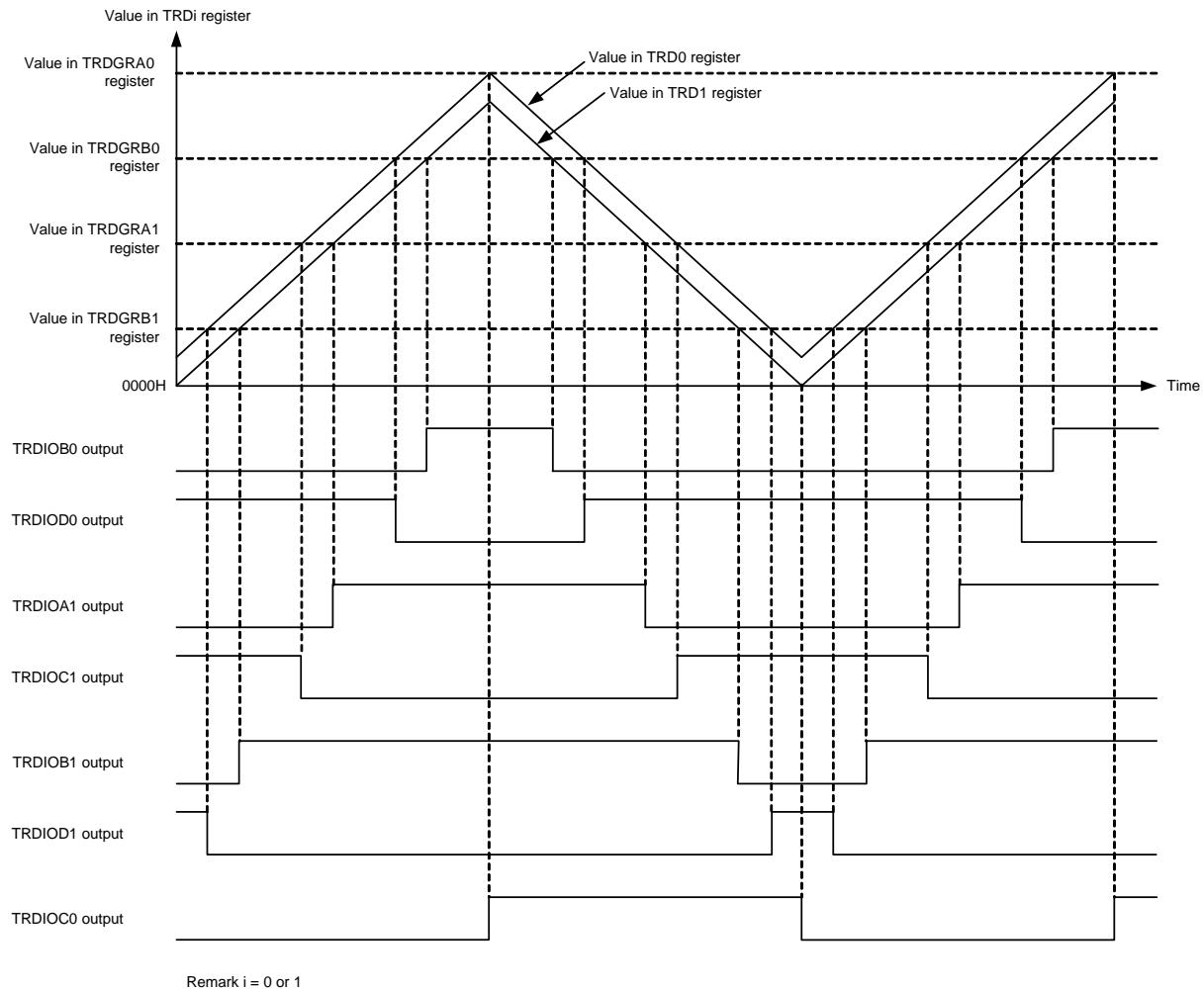
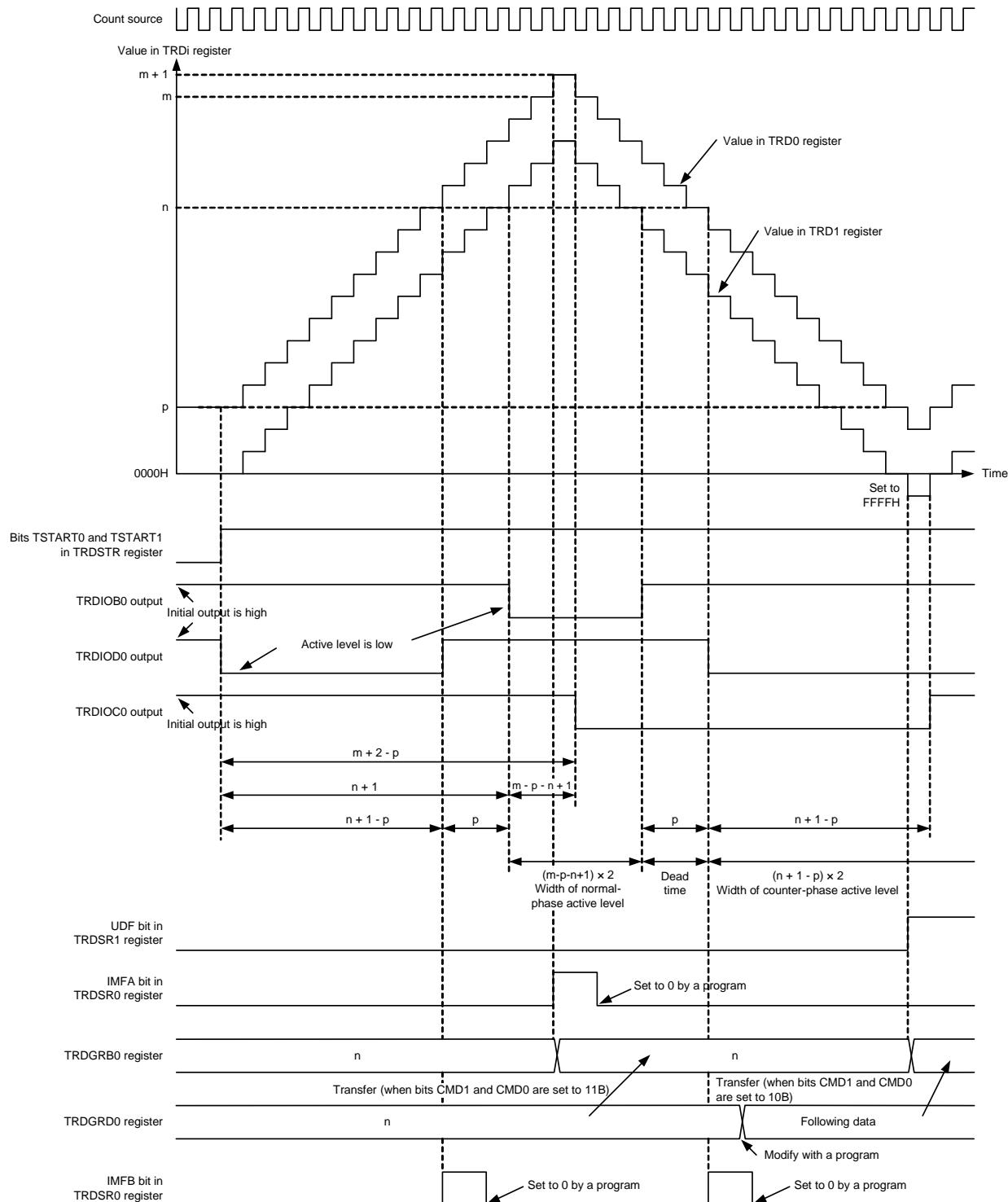
**Figure 8 - 60 Output Model of Complementary PWM Mode**

Figure 8 - 61 Operation Example in Complementary PWM Mode

**Remark**

CMD0, CMD1: Bits in TRDFCR register  
 $i = 0$  or  $1$   
 $m$ : Value set in TRDGRA0 register  
 $n$ : Value set in TRDGRB0 register  
 $p$ : Value set in TRD0 register

The above diagram applies under the following condition:  
 Bits OLS1 and OLS0 in TRDFCR are set to 0 (initial output level is high, active level is low for normal-phase and counter-phase).

(2) Transfer Timing from Buffer Register

- Transfer from the TRDGRD0, TRDGRC1, or TRDGRD1 register to the TRDGRB0, TRDGRA1, or TRDGRB1 register.

When bits CMD1 and CMD0 in the TRDFCR register are set to 10B, the content is transferred when the TRD1 register underflows.

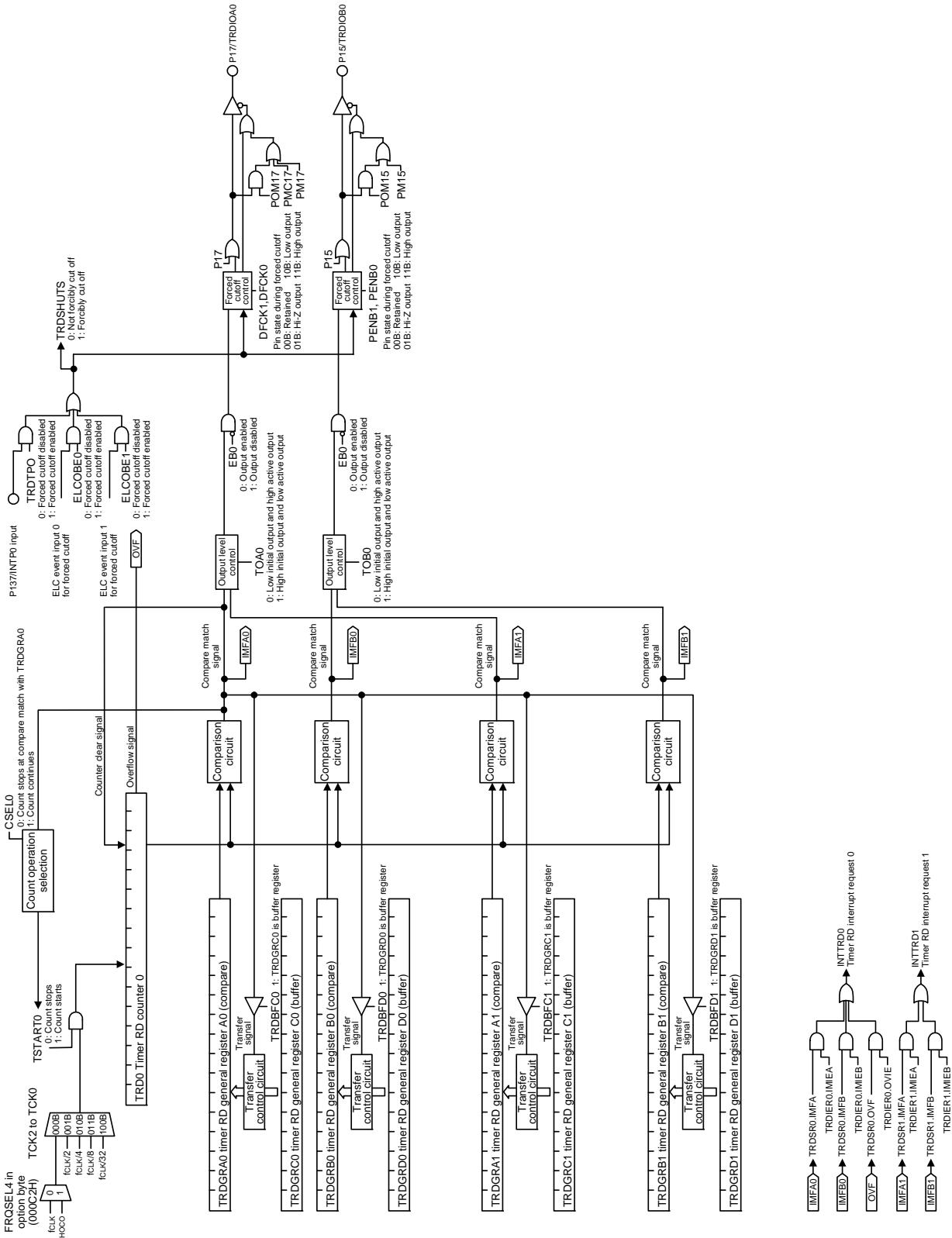
When bits CMD1 and CMD0 are set to 11B, the content is transferred at compare match between registers TRD0 and TRDGRA0.

### 8.5.6 PWM3 Mode

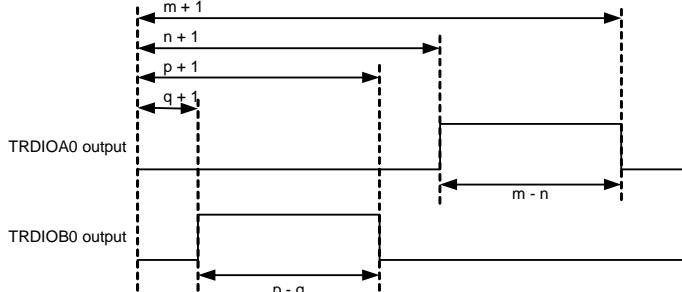
In this mode, two PWM waveforms are output with the same period.

Figure 8 - 62 shows the Block Diagram of PWM3 Mode, Table 8 - 17 lists the PWM3 Mode Specifications, and Figure 8 - 63 shows an Operation Example in PWM3 Mode.

Figure 8 - 62 Block Diagram of PWM3 Mode



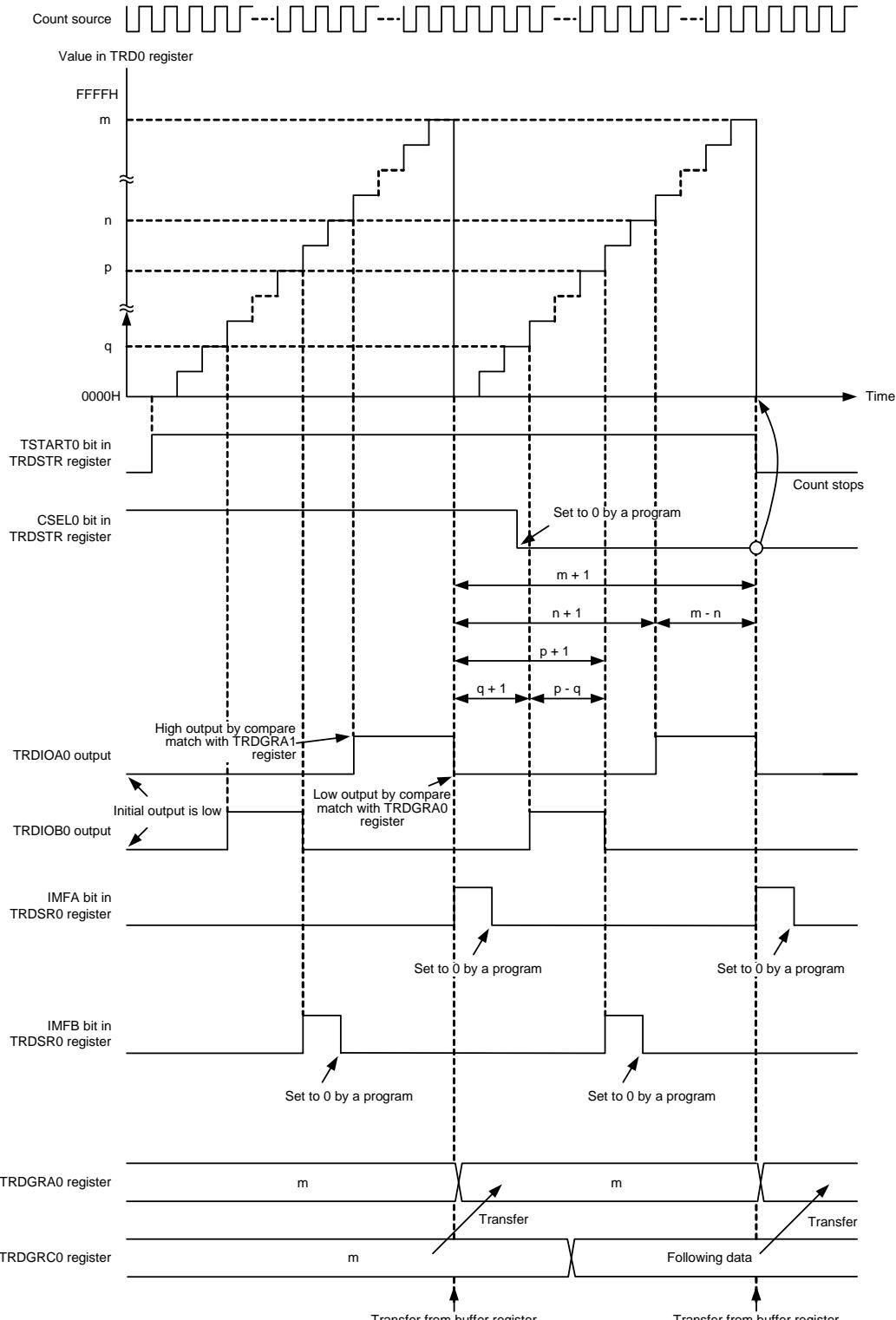
**Table 8 - 17 PWM3 Mode Specifications**

Item	Specification
Count sources	fHOOC Note, fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32
Count operations	The TRD0 register is incremented (the TRD1 register is not used).
PWM waveform	<p>PWM period: <math>1/fk \times (m + 1)</math>            Active level width of TRDIOA0 output: <math>1/fk \times (m - n)</math>            Active level width of TRDIOB0 output: <math>1/fk \times (p - q)</math></p> <p>fk: Frequency of count source            m: Value set in the TRDGRA0 register            n: Value set in the TRDGRA1 register            p: Value set in the TRDGRB0 register            q: Value set in the TRDGRB1 register</p>  <p>(When high is selected as the active level)</p>
Count start condition	1 (count starts) is written to the TSTART0 bit in the TRDSTR register.
Count stop conditions	<ul style="list-style-type: none"> <li>0 (count stops) is written to the TSTART0 bit in the TRDSTR register when the CSEL0 bit in the TRDSTR register is set to 1.            The PWM output pin holds the output level before the count stops.</li> <li>When the CSEL0 bit in the TRDSTR register is set to 0, the count stops at compare match with the TRDGRA0 register.            The PWM output pin holds the level after output change by compare match.</li> </ul>
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (content of the TRDi register matches content of the TRDGRji register)</li> <li>TRD0 register overflow</li> </ul>
TRDIOA0, TRDIOB0 pin function	PWM output
TRDIOA0, TRDIOB0, and TRDIOA1 to TRDIOB1 pin function	I/O port
INTP0 pin function	Pulse output forced cutoff signal input (input-only port or INTP0 interrupt input)
Read from timer	The count value can be read by reading the TRD0 register.
Write to timer	The value can be written to the TRD0 register.
Selectable functions	<ul style="list-style-type: none"> <li>Pulse output forced cutoff signal input (see <b>8.4.4 Pulse Output Forced Cutoff</b>)</li> <li>Active level selectable for each pin</li> <li>Buffer operation (see <b>8.4.2 Buffer Operation</b>)</li> </ul>

**Note** fHOOC is selected only when FRQSEL4 = 1 in the user option byte (000C2H). When selecting fHOOC as the count source for timer RD, set fCLK to fIH before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing fCLK to a clock other than fIH, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

**Remark** i = 0 or 1, j = A, B, C, or D

## (1) Operation Example

**Figure 8 - 63 Operation Example in PWM3 Mode**

## Remark

j = A or B

m: Value set in TRDGRA0 register

n: Value set in TRDGRA1 register

p: Value set in TRDGRB0 register

q: Value set in TRDGRB1 register

The above diagram applies under the following conditions :

- Both the TOA0 and TOB0 bits in the TRDOCR register are set to 0 (initial output is low, high output by compare match with TRDGRA1 register, low output by compare match with TRDGRA0 register).
- The TRDBFC0 bit in the TRDMR register is set to 1 (TRDRC0 register is buffer register for TRDGRA0 register).

## 8.6 Timer RD Interrupt

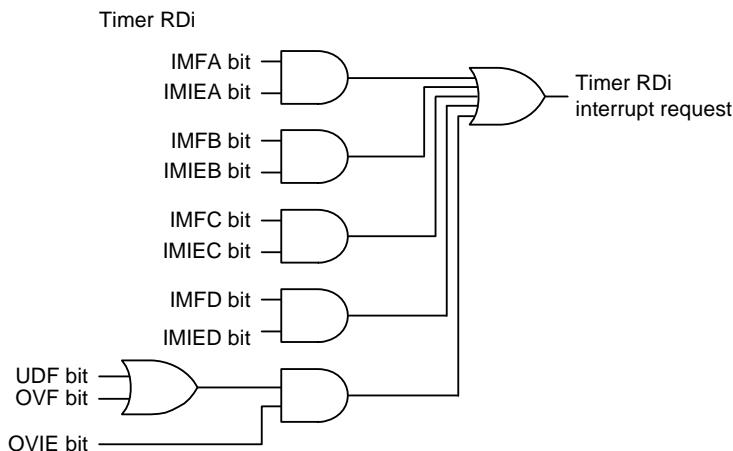
Timer RD generates the timer RD<sub>i</sub> ( $i = 0$  or  $1$ ) interrupt request from six sources for each timer RD0 and timer RD1.

Table 8 - 18 lists the Registers Associated with Timer RD Interrupt and Figure 8 - 64 shows the Timer RD Interrupt Block Diagram.

**Table 8 - 18 Registers Associated with Timer RD Interrupt**

	Timer RD Status Register	Timer RD Interrupt Enable Register	Interrupt Request Flag (Register)	Interrupt Mask Flag (Register)	Priority Specification Flag (Register)
Timer RD0	TRDSR0	TRDIER0	TRDIF0 (IF2H)	TRDMK0 (MK2H)	TRDPR00 (PR02H) TRDPR10 (PR12H)
Timer RD1	TRDSR1	TRDIER1	TRDIF1 (IF2H)	TRDMK1 (MK2H)	TRDPR01 (PR02H) TRDPR11 (PR12H)

**Figure 8 - 64 Timer RD Interrupt Block Diagram**



$i = 0$  to  $1$

IMFA, IMFB, IMFC, IMFD, OVF, UDF: TRDSR<sub>i</sub> register bit  
IMIEA, IMIEB, IMIEC, IMIED, OVIE: TRDIER<sub>i</sub> register bit

Since the interrupt source (timer RD interrupt) is generated by a combination of multiple interrupt request sources for timer RD, the following differences from other maskable interrupts except timer RG interrupt apply:

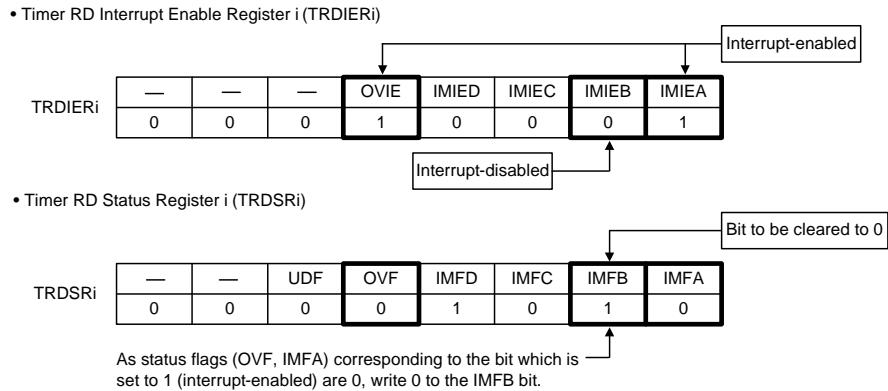
- When a bit in the TRDSR<sub>i</sub> register is 1 and the corresponding bit in the TRDIER<sub>i</sub> register is 1 (interrupt enabled), the TRDIF<sub>i</sub> bit in the IF2H register is set to 1 (interrupt requested).
- If multiple bits in the TRDIER<sub>i</sub> register are set to 1, use the TRDSR<sub>i</sub> register to determine the source of the interrupt request.
- Since the bits in the TRDSR<sub>i</sub> register are not automatically set to 0 even if the interrupt is acknowledged, set the corresponding bit to 0 in the interrupt routine.

- When status flags of interrupt sources (applicable status flags) of the timer RD are set to 0 and their interrupts are disabled in timer RD interrupt enable register i (TRDIERi), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RD interrupt enable register i (TRDIERi) and write 0 to applicable status flags.

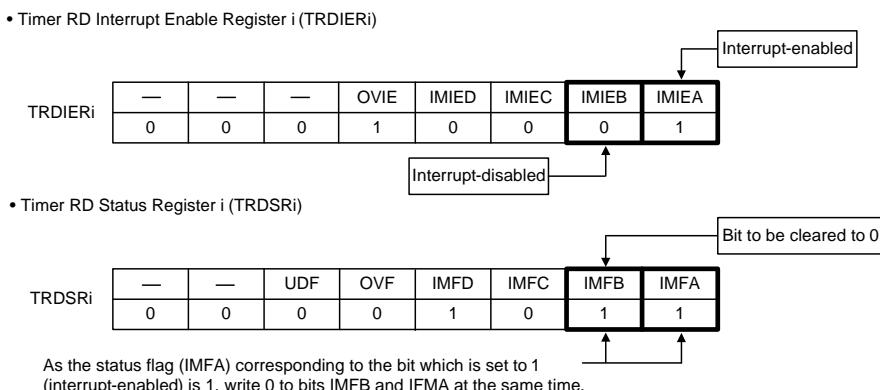
(b) When there are bits set to 1 (enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the IMFB bit to 0 when bits IMIEA and OVIE are set to 1 (interrupt-enabled) and the IMIEB bit is set to 0 (interrupt-disabled).



(c) When there are bits set to 1 (interrupt-enabled) in timer RD interrupt enable register i (TRDIERi) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the IMFB bit to 0 when the IMIEA is set to 1 (interrupt-enabled) and the IMIEB is set to 0 (interrupt-disabled).



## 8.7 Cautions for Timer RD

### 8.7.1 SFR Read/Write Access

When setting timer RD, set the TRD0EN bit in the PER1 register to 1 first. If the TRD0EN bit is 0, writes to the timer RD control registers are ignored and all the read values are the initial values (except for the port registers and the port mode registers).

The following registers must not be rewritten during count operation:

TRDELC, TRDMR, TRDPMR, TRDFCR, TRDOER1, TRDPTO bit in TRDOER2, TRDDFi, TRDCri, TRDIORAi, TRDIORCi, TRDPOCRI

#### (1) TRDSTR Register

- The TRDSTR register can be set by an 8-bit memory manipulation instruction.
- When the CSELi bit ( $i = 0$  or  $1$ ) in the TRDSTR register is set to 0 (count stops at compare match between registers TRDi and TRDGRAi), the count does not stop and the TSTARTi bit remains unchanged even if 0 (count stops) is written to the TSTARTi bit.

The TSTARTi bit is set to 0 (count stops) only by a compare match with the TRDGRAi register.

If the CSELi bit is 0 when rewriting the TRDSTR register, write 0 to the TSTARTi bit to change the CSELi bit to 1 without affecting count operation.

If 1 is written to the TSTARTi bit while the counter is stopped, count may be started.

To stop counting by a program, set the TSTARTi bit after setting the CSELi bit to 1. Even if 1 is written to the CSELi bit and 0 is written to the TSTARTi bit at the same time (using one instruction), the count cannot be stopped.

- Table 8 - 19 lists the TRDIOji ( $j = A, B, C$ , or  $D$ ) Pin Output Level When Count Stops while using the TRDIOji ( $j = A, B, C$ , or  $D$ ) pin for timer RD output.

**Table 8 - 19 TRDIOji ( $j = A, B, C$ , or  $D$ ) Pin Output Level When Count Stops**

Count Stop	TRDIOji Pin Output When Count Stops
When the CSELi bit is set to 1, write 0 to the TSTARTi bit and the count stops.	The pin holds the output level immediately before the count stops. (The pin outputs the initial output level selected by bits OLS0 and OLS1 in the TRDFCR register in timer RD complementary and reset synchronous PWM modes.)
When the CSELi bit is set to 0, the count stops at compare match with registers TRDi and TRDGRAi.	The pin holds the output level after the output changes by compare match. (The pin outputs the initial output level selected by bits OLS0 and OLS1 in the TRDFCR register in timer RD complementary and reset synchronous PWM modes.)

**Remark**  $i = 0$  or  $1$ ,  $j = A, B, C$ , or  $D$

(2) TRDDFi Register ( $i = 0$  or  $1$ )

Set bits DFCK0 and DFCK1 in the TRDDFi register before starting count operation.

(3) TRDi Register ( $i = 0$  or  $1$ )

- If the TRDi register is set to 0000H and a value is written to the TRDi register at the same timing, the value written to the register has priority.

### 8.7.2 Mode Switching

- Set the count to stopped (set bits TSTART0 and TSTART1 to 0) before switching modes during operation.
- Set bits TRDIF0 and TRDIF1 to 0 before changing bits TSTART0 and TSTART1 from 0 to 1. Refer to **CHAPTER 21 INTERRUPT FUNCTIONS** for details.

### 8.7.3 Count Source

- Switch the count source after the count stops.

[Changing procedure]

(1) Set the TSTART $i$  bit ( $i = 0$  or  $1$ ) in the TRDSTR register to 0 (count stops).

(2) Change bits TCK0 to TCK2 in the TRDCR $i$  register.

- When selecting f<sub>HO</sub>CO (64 MHz or 48 MHz) as the count source for timer RD, set f<sub>CLK</sub> to f<sub>H</sub> before setting bit 4 (TRD0EN) in peripheral enable register 1 (PER1). When changing f<sub>CLK</sub> to a clock other than f<sub>H</sub>, clear bit 4 (TRD0EN) in peripheral enable register 1 (PER1) before changing.

### 8.7.4 Input Capture Function

- Set the pulse width of the input capture signal to three or more cycles of the timer RD operating clock.
- The value of the TRDi register is transferred to the TRDGR $ji$  register two to three cycles of the timer RD operating clock (f<sub>CLK</sub>) after the input capture signal is applied to the TRDIO $ji$  pin ( $i = 0$  or  $1$ ,  $j = A, B, C$ , or  $D$ ) (when no digital filter is used).
- In input capture mode, an input capture interrupt request for the active edge of the TRDIO $ji$  input is also generated when the TRDTSTART $i$  bit in the TRDSTR register is 0 (count stops) if the edge selected by bits TRDIO $j0$  and TRDIO $j1$  in the TRDIOR $ji$  register is input to the TRDIO $ji$  pin ( $i = 0$  or  $1$ ;  $j = A, B, C$ , or  $D$ ).

### 8.7.5 Procedure for Setting Pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi (i = 0 or 1)

After a reset, the I/O ports multiplexed with pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi function as input ports.

- To output from pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi, use the following setting procedure:

Changing procedure

- (1) Set the mode and the initial value.
- (2) Enable output from pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi (TRDOER1 register).
- (3) Set the port register bits corresponding to pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi to 0.
- (4) Set the port mode register bits corresponding to pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi to output mode. (Output is started from pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi)
- (5) Start the count (set bits TSTART0 and TSTART1 to 1).

- To change the port mode register bits corresponding to pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi from output mode to input mode, use the following setting procedure:

Changing procedure

- (1) Set the port mode register bits corresponding to pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi to input mode (input is started from pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi).
  - (2) Set to the input capture function.
  - (3) Start the count (set bits TSTART0 and TSTART1 to 1).
- When switching pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi from output mode to input mode, input capture operation may be performed depending on the pin states. When the digital filter is not used, edge detection is performed after two or more cycles of the operation clock have elapsed. When the digital filter is used, edge detection is performed after up to five cycles of the sampling clock.

### 8.7.6 External clock TRDCLK

Set the pulse width of the external clock applied to the TRDCLK pin to three or more cycles of the timer RD operating clock.

### 8.7.7 Reset Synchronous PWM Mode

- When reset synchronous PWM mode is used for motor control, make sure OLS0 = OLS1.
- Set to reset synchronous PWM mode by the following procedure:

[Changing procedure]

- (1) Set the TSTART0 bit in the TRDSTR register to 0 (count stops).
- (2) Set bits CMD1 and CMD0 in the TRDFCR register to 00B (timer mode, PWM mode, and PWM3 mode).
- (3) Set bits CMD1 and CMD0 to 01B (reset synchronous PWM mode).
- (4) Set the other registers associated with timer RD again.

### 8.7.8 Complementary PWM Mode

- When complementary PWM mode is used for motor control, make sure OLS0 = OLS1.
- Change bits CMD0 and CMD1 in the TRDFCR register in the following procedure.

Changing procedure: When setting to complementary PWM mode (including re-set), or changing the transfer timing from the buffer register to the general register in complementary PWM mode.

- Set both the TSTART0 and TSTART1 bits in the TRDSTR register to 0 (count stops).
- Set bits CMD1 and CMD0 in the TRDFCR register to 00B (timer mode, PWM mode, and PWM3 mode).
- Set bits CMD1 and CMD0 to 10B or 11B (complementary PWM mode).
- Set the registers associated with other timer RD again.

Changing procedure: When stopping complementary PWM mode

- Set both the TSTART0 and TSTART1 bits in the TRDSTR register to 0 (count stops).
- Set bits CMD1 to CMD0 to 00B (timer mode, PWM mode, and PWM3 mode).

- Do not write to the TRDGRA0, TRDGRB0, TRDGRA1, or TRDGRB1 register during operation.

When changing the PWM waveform, transfer the values written to registers TRDGRD0, TRDGRC1, and TRDGRD1 to registers TRDGRB0, TRDGRA1, and TRDGRB1 using the buffer operation.

However, to write data to the TRDGRD0, TRDGRC1, or TRDGRD1 register, set bits TRDBFD0, TRDBFC1, and TRDBFD1 to 0 (general register). After this, bits TRDBFD0, TRDBFC1, and TRDBFD1 may be set to 1 (buffer register).

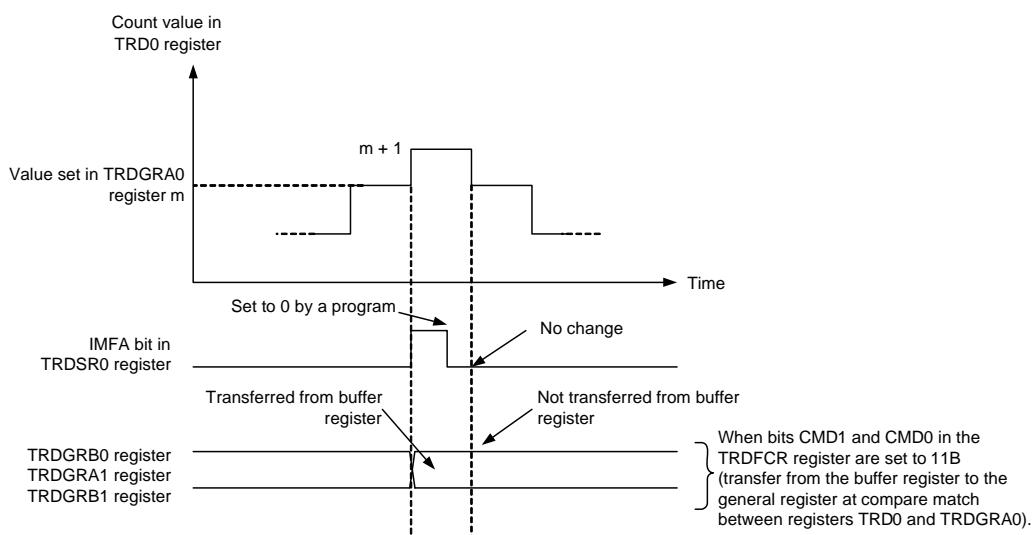
The PWM period cannot be changed.

- If the value set in the TRDGRA0 register is assumed to be m, the TRD0 register counts m - 1, m, m + 1, m, m - 1, in that order, when changing from increment to decrement operation.

When changing from m to m + 1, the IMFA bit in the TRDSR0 register is set to 1. Also, bits CMD1 and CMD0 in the TRDFCR register are set to 11B (complementary PWM mode, buffer data transferred at compare match between registers TRD0 and TRDGRA0), the content of the buffer registers (TRDGRD0, TRDGRC1, and TRDGRD1) is transferred to the general registers (TRDGRB0, TRDGRA1, and TRDGRB1).

During operation of m + 1, m, and m - 1, the IMFA bit remains unchanged and data is not transferred to registers such as the TRDGRA0 register.

**Figure 8 - 65 Operation at Compare Match between Registers TRD0 and TRDGRA0 in Complementary PWM Mode**

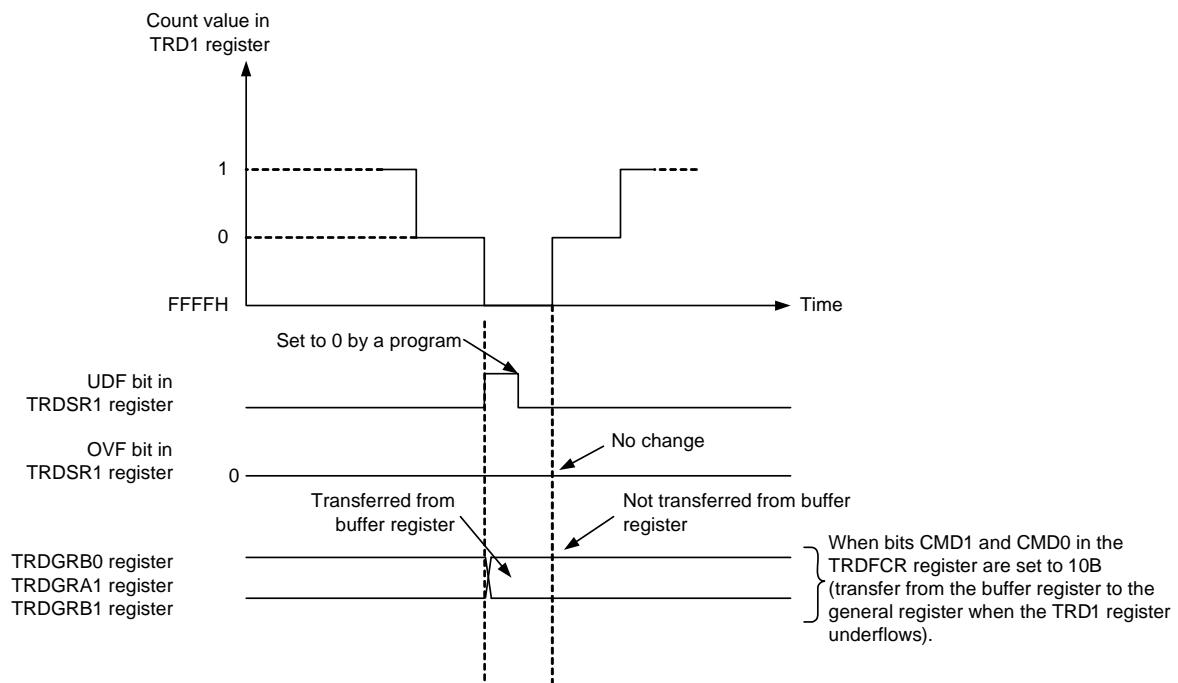


- The TRD1 register counts 1, 0, FFFFH, 0, 1, in that order, when changing from decrement to increment operation.

Counting from 1, to 0, to FFFFH causes the UDF bit in the TRDSR1 register to be set to 1. Also, when bits CMD1 and CMD0 in the TRDFCR register are set to 10B (complementary PWM mode, buffer data transferred at underflow of the TRD1 register), the content of the buffer registers (TRDGRD0, TRDGRC1, and TRDGRD1) is transferred to the general registers (TRDGRB0, TRDGRA1, and TRDGRB1).

During operation of FFFFH, 0, and 1, data is not transferred to registers such as the TRDGRB0 register. Also, at this time, the OVF bit in the TRDSR1 register remains unchanged.

**Figure 8 - 66 Operation When TRD1 Register Underflows in Complementary PWM Mode**



- The timing of data transfer from the buffer register to the general register should be selected using bits CMD0 and CMD1 in the TRDFCR register. However, regardless of the values of bits CMD0 and CMD1, transfer takes place with the following timing when duty cycle is 0% and duty cycle is 100%.

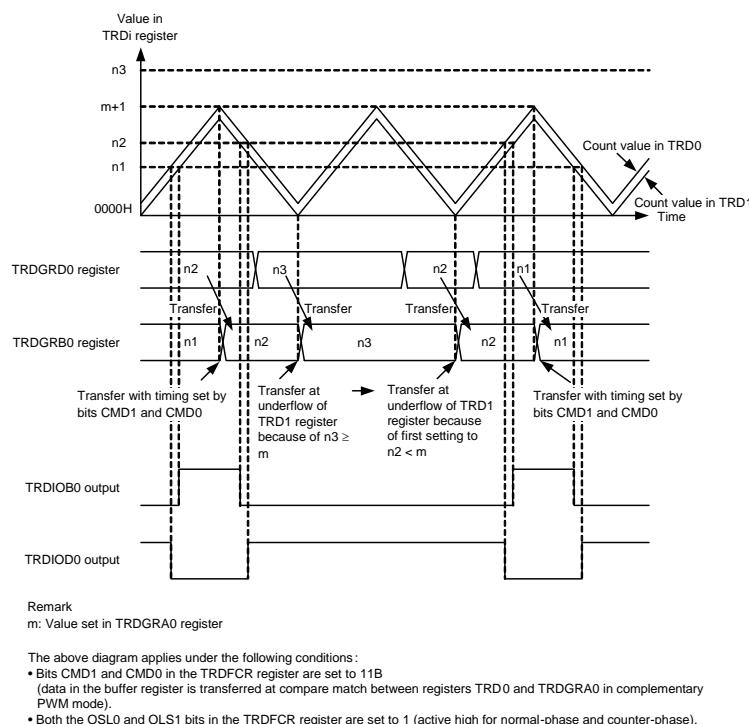
Value in buffer register  $\geq$  value in TRDGRA0 register (duty cycle is 0%):

Transfer take place at underflow of the TRD1 register.

After this, when the buffer register is set to 0001H or above and a smaller value than the value of the TRDGRA0 register, and the TRD1 register underflows for the first time after setting, the value is transferred to the general register. After that, the value is transferred with the timing selected by bits CMD1 and CMD0.

However, no waveform with duty cycle 0% can be generated while the initial value of the buffer register is FFFFH. To generate a waveform with duty cycle 0%, set the value of the buffer register  $\geq$  TRDGRA0 by writing to the buffer register.

**Figure 8 - 67 Operation When Value in Buffer Register  $\geq$  Value in TRDGRA0 Register in Complementary PWM Mode**



When a value that is larger than or equal to the value of the TRDGRA0 register is written to the buffer register, the value of the buffer register is transferred to the general register at underflow of the TRD1 counter, and the output level is fixed to normal-phase with 100% duty cycle and counter-phase with 0% duty cycle.

To cancel the fixed output level, write a value that is larger than or equal to the setting value of the TRD0 register and smaller than or equal to (TRDGRA0 setting value minus TRD0 register setting value) to the buffer register. After the value is written to the buffer register, the value of the buffer register is transferred to the general register at underflow of the TRD1 counter, and a PWM waveform is output regardless of the setting of the CMD0 bit. After a PWM waveform is output, the value of the buffer register is transferred to the general register with the timing specified by the CMD0 bit.

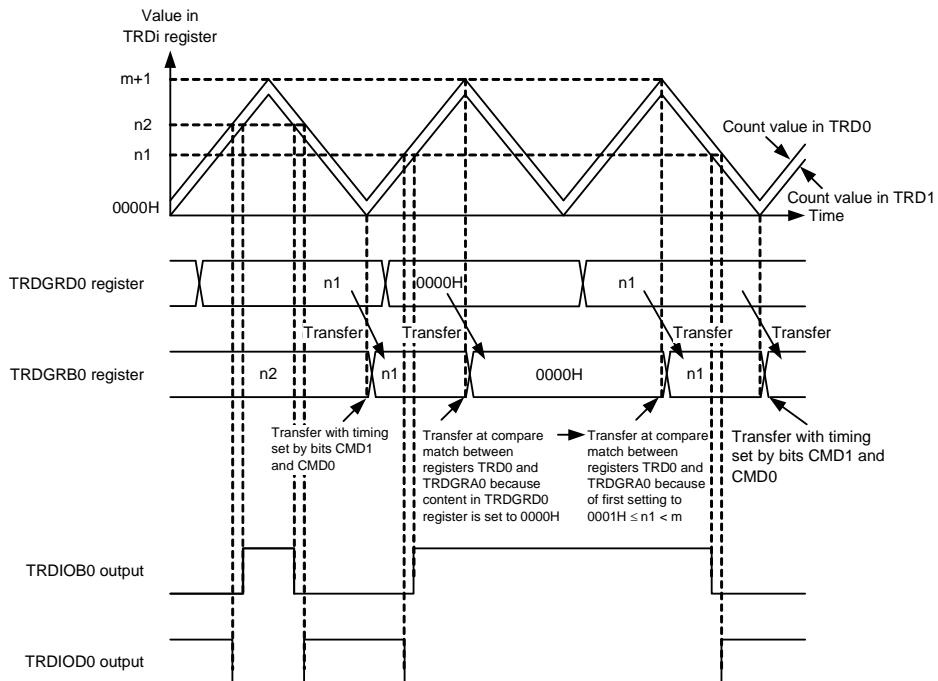
However, the initial value FFFFH of the buffer register cannot be used to set normal-phase output with 100% duty cycle and counter-phase output with 0% duty cycle. Also, while the setting is normal-phase output with 100% duty cycle and counter-phase output with 0% duty cycle, the setting cannot be directly changed to normal-phase output with 0% duty cycle and counter-phase output with 100% duty cycle.

When the value in the buffer register is set to 0000H (duty cycle is 100%):

Transfer takes place at compare match between registers TRD0 and TRDGRA0.

After this, when the buffer register is set to 0001H or above and a smaller value than the value of the TRDGRA0 register, and a compare match occurs between registers TRD0 and TRDGRA0 for the first time after setting, the value is transferred to the general register. After that, the value is transferred with the timing selected by bits CMD0 and CMD1.

**Figure 8 - 68 Operation When Value in Buffer Register is Set to 0000H in Complementary PWM Mode**



Remark

m: Value set in TRDGRA0 register

The above diagram applies under the following conditions:

- Bits CMD1 and CMD0 in the TRDFCR register are set to 10B (data in the buffer register is transferred at underflow of the TRD1 register in PWM mode).
- Both the OLS0 and OLS1 bits in the TRDFCR register are set to 1 (active high for normal-phase and counter-phase).

When 0000H is written to the buffer register, the value of the buffer register is transferred to the general register at a compare match between registers TRD0 and TRDGRA0, and the output level is fixed to normal-phase with 0% duty cycle and counter-phase with 100% duty cycle.

To cancel the fixed output level, write a value that is larger than or equal to the setting value of the TRD0 register and smaller than or equal to (TRDGRA0 setting value minus TRD0 register setting value) to the buffer register. After the value is written to the buffer register, the value of the buffer register is transferred to the general register at underflow of the TRD1 counter, and a PWM waveform is output regardless of the setting of the CMD0 bit. After a PWM waveform is output, the value of the buffer register is transferred to the general register with the timing specified by the CMD0 bit.

The setting of normal-phase output with 0% duty cycle and counter-phase output with 100% duty cycle cannot be directly changed to normal-phase output with 100% duty cycle and counter-phase output with 0% duty cycle.

## CHAPTER 9 TIMER RG

### 9.1 Functions of Timer RG

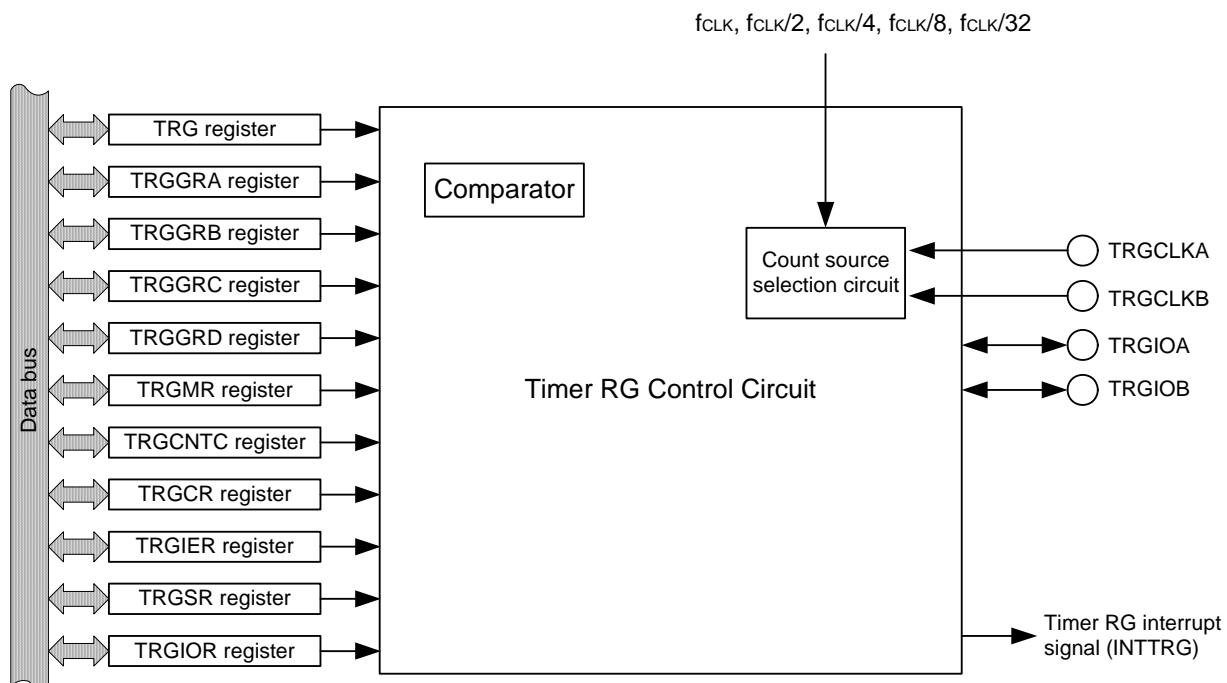
Timer RG supports the following three modes:

- Timer mode:
  - Input capture function: Count at the rising edge, falling edge, or both rising/falling edges
  - Output compare function: Low output/high output/toggle output
- PWM mode: PWM output available with any duty cycle
- Phase counting mode: Automatic measurement available for the counts of the two-phase encoder

## 9.2 Configuration of Timer RG

Figure 9 - 1 shows the Timer RG Block Diagram and Table 9 - 1 lists the Timer RG Pin Configuration.

**Figure 9 - 1 Timer RG Block Diagram**



**Table 9 - 1 Timer RG Pin Configuration**

Pin Name	Alternate Port Name	I/O	Function
TRGCLKA	P00	Input	<ul style="list-style-type: none"> <li>In phase counting mode A-phase input</li> <li>In other than phase counting mode External clock A input</li> </ul>
TRGCLKB	P01	Input	<ul style="list-style-type: none"> <li>In phase counting mode B-phase input</li> <li>In other than phase counting mode External clock B input</li> </ul>
TRGIOA	P50	Input/Output	<ul style="list-style-type: none"> <li>In timer mode (output compare function) TRGGRA output-compare output</li> <li>In timer mode (input capture function) TRGGRA input-capture input</li> <li>In PWM mode PWM output</li> </ul>
TRGIOB	P51	Input/Output	<ul style="list-style-type: none"> <li>In timer mode (output compare function) TRGGRB output-compare output</li> <li>In timer mode (input capture function) TRGGRB input-capture input</li> </ul>

### 9.3 Registers Controlling Timer RG

Table 9 - 2 lists the Registers Controlling Timer RG.

**Table 9 - 2 Registers Controlling Timer RG**

Register Name	Symbol
Peripheral enable register 1	PER1
Timer RG mode register	TRGMR
Timer RG count control register	TRGCNTC
Timer RG control register	TRGCR
Timer RG interrupt enable register	TRGIER
Timer RG status register	TRGSR
Timer RG I/O control register	TRGIOR
Timer RG counter	TRG
Timer RG general register A	TRGGRA
Timer RG general register B	TRGGRB
Timer RD general register C	TRGGRC
Timer RD general register D	TRGGRD
Port register 0	P0
Port register 5	P5
Port mode register 0	PM0
Port mode register 5	PM5

### 9.3.1 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to the hardware that is not used is also stopped so as to decrease the power consumption and noise.

When using the timer RG, be sure to set bit 6 (TRGEN) to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 9 - 2 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
--------	-----	-----	-----	-----	-----	---	---	-----

PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN
------	------------	-------	------------	--------	-------	---	---	--------

TRGEN	Control of timer RG input clock supply
0	Stops input clock supply. <ul style="list-style-type: none"> <li>• SFR used by timer RG cannot be written.</li> <li>• Timer RG is in the reset status.</li> </ul>
1	Enables input clock supply. <ul style="list-style-type: none"> <li>• SFR used by timer RG can be read and written.</li> </ul>

**Note** Only for products with 96 KB or more code flash memory.

**Caution 1. When setting timer RG, be sure to set the TRGEN bit to 1 first. If TRGEN = 0, writing to a control register of timer RG is ignored, and all read values are default values (except for port mode registers 0, 5 (PM0, PM5), and port registers 0, 5 (P0, P5)).**

**Caution 2. Be sure to set the following bits to 0:**

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 9.3.2 Timer RG mode register (TRGMR)

Figure 9 - 3 Format of Timer RG mode register (TRGMR)

Address: F0250H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

TRGMR	TRGSTART	TRGELCICE	TRGDFCK1	TRGDFCK0	TRGDFB	TRGDFA	TRGMDF	TRGPWM
-------	----------	-----------	----------	----------	--------	--------	--------	--------

TRGSTART	TRG count start							
0	Count stops, and PWM output signal (TRGIOA pin) is initialized (in PWM mode)							
1	Count starts							

TRGELCICE	ELC input capture request select Notes 1, 2							
0	External output signal B/digital filtering signal B is selected							
1	Event input (input capture) from ELC is selected							

TRGDFCK1	TRGDFCK0	Digital filter function clock select Note 1						
0	0	fCLK/32						
0	1	fCLK/8						
1	0	fCLK						
1	1	Clock selected by bits TRGTCK0 to TRGTCK2 in TRGCR register						

TRGDFB	Digital filer function select for TRGIOB pin							
0	Digital filter function not used							
1	Digital filter function used							
When the digital filter is used, edge detection is performed after up to five cycles of the sampling clock.								

TRGDFA	Digital filer function select for TRGIOA pin							
0	Digital filter function not used							
1	Digital filter function used							
When the digital filter is used, edge detection is performed after up to five cycles of the sampling clock.								

TRGMDF	Phase counting mode select															
0	Increment															
1	Phase counting mode															
When the TRGMDF bit is set to 0, the counter counts the count source set by bits TRGTCK0 to TRGTCK2 in the TRGCR register.																
When the TRGMDF bit is set to 1, the counter counts the phase of input signals from the TRGCLKj pin (j = A or B) as listed in Table 9 - 15 Increment/Decrement Conditions for TRG Register																

TRGPWM	PWM mode select							
0	Timer Mode							
1	PWM mode							

**Note 1.** Set this bit while the TRGSTART bit is 0 (count stops).

**Note 2.** To enable event input (input capture) from the ELC, set TRGIOB2 = 1 and TRGIOB1 and TRGIOB0 = 00B (rising edge) in the TRGIOR register.

### 9.3.3 Timer RG count control register (TRGCNTC)

The TRGCNTC register is used in phase counting mode. This register is used to set the count conditions for phase counting mode.

**Figure 9 - 4 Format of Timer RG count control register (TRGCNTC)**

Address: F0251H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

TRGCNTC	CNTEN7	CNTEN6	CNTEN5	CNTEN4	CNTEN3	CNTEN2	CNTEN1	CNTEN0
---------	--------	--------	--------	--------	--------	--------	--------	--------

CNTEN7	Counter enable 7							
0	Disabled							
1	Increment When TRGCLKA input is low level and at the rising edge of TRGCLKB input							

CNTEN6	Counter enable 6							
0	Disabled							
1	Increment When TRGCLKB input is high level and at the rising edge of TRGCLKA input							

CNTEN5	Counter enable 5							
0	Disabled							
1	Increment When TRGCLKA input is high level and at the falling edge of TRGCLKB input							

CNTEN4	Counter enable 4							
0	Disabled							
1	Increment When TRGCLKB input is low level and at the falling edge of TRGCLKA input							

CNTEN3	Counter enable 3							
0	Disabled							
1	Decrement When TRGCLKB input is high level and at the falling edge of TRGCLKA input							

CNTEN2	Counter enable 2							
0	Disabled							
1	Decrement When TRGCLKA input is low level and at the falling edge of TRGCLKB input							

CNTEN1	Counter enable 1							
0	Disabled							
1	Decrement When TRGCLKB input is low level and at the rising edge of TRGCLKA input							

CNTEN0	Counter enable 0							
0	Disabled							
1	Decrement When TRGCLKA input is high level and at the rising edge of TRGCLKB input							

### 9.3.4 Timer RG control register (TRGCR)

When writing to the TRGCR register, make sure the TRGSTART bit in the TRGMR register is 0 (count stops).

**Figure 9 - 5 Format of Timer RG control register (TRGCR)**

Address: F0252H After reset: 00H R/W

Symbol	7	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	---	-----	-----	-----	-----	-----	-----	-----

TRGCR	0	TRGCCLR1	TRGCCLR0	TRGCKEG1	TRGCKEG0	TRGTCK2	TRGTCK1	TRGTCK0
-------	---	----------	----------	----------	----------	---------	---------	---------

TRGCCLR1	TRGCCLR0	TRG register clear source select
0	0	Clear disabled
0	1	Clear by input capture or compare match with TRGGRA
1	0	Clear by input capture or compare match with TRGGRB
Other than above		Setting prohibited

TRGCKEG1	TRGCKEG0	External clock active edge select Notes 1, 2
0	0	Count at the rising edge
0	1	Count at the falling edge
1	0	Count at both the rising/falling edges
Other than above		Setting prohibited

TRGTCK2	TRGTCK1	TRGTCK0	Count source select Note 1
0	0	0	fCLK
0	0	1	fCLK/2
0	1	0	fCLK/4
0	1	1	fCLK/8
1	0	0	fCLK/32
1	0	1	TRGCLKA input
1	1	1	TRGCLKB input
Other than above		Setting prohibited	

**Note 1.** In phase counting mode, the settings of bits TRGTCK0 to TRGTCK2 and bits TRGCKEG0 and TRGCKEG1 are disabled and the operation of phase counting mode has priority.

**Note 2.** Bits TRGCKEG0 and TRGCKEG1 are enabled when bits TRGTCK0 to TRGTCK2 are set to an external clock (TRGCLKA or TRGCLKB). When not set to an external clock, they are disabled.

### 9.3.5 Timer RG interrupt enable register (TRGIER)

Figure 9 - 6 Format of Timer RG interrupt enable register (TRGIER)

Address: F0253H After reset: 00H R/W

Symbol	7	6	5	4	<3>	<2>	<1>	<0>
TRGIER	0	0	0	0	TRGOVIE	TRGUDIE	TRGIMIEB	TRGIMIEA
TRGOVIE	Overflow interrupt enable							
0	Interrupt by TRGOVF bit disabled							
1	Interrupt by TRGOVF bit enabled							
TRGUDIE	Underflow interrupt enable							
0	Interrupt by TRGUDF bit disabled							
1	Interrupt by TRGUDF bit enabled							
TRGIMIEB	Input-capture/compare-match interrupt enable B							
0	Interrupt by TRGIMFB bit disabled							
1	Interrupt by TRGIMFB bit enabled							
TRGIMIEA	Input-capture/compare-match interrupt enable A							
0	Interrupt by TRGIMFA bit disabled							
1	Interrupt by TRGIMFA bit enabled							

**Remark** TRGIMFA, TRGIMFB, TRGUDF, TRGOVF: Bits in TRGSR register

### 9.3.6 Timer RG status register (TRGSR)

Figure 9 - 7 Format of Timer RG status register (TRGSR)

Address: F0254H      After reset: 00H      R/W

Symbol      7      6      5      <4>      <3>      <2>      <1>      <0>

TRGSR	0	0	0	TRGDIRF	TRGOVF	TRGUDF	TRGIMFB	TRGIMFA
-------	---	---	---	---------	--------	--------	---------	---------

TRGDIRF	Count direction flag
0	TRG register is decremented
1	TRG register is incremented

TRGOVF	Overflow flag <small>Note 1</small>
[Condition for setting to 0]	
Write 0 after reading <small>Note 2</small>	
[Condition for setting to 1]	
See Table 9 - 3 Conditions for Setting Each Flag to 1.	

TRGUDF	Underflow flag
[Condition for setting to 0]	
Write 0 after reading <small>Note 2</small>	
[Condition for setting to 1]	
See Table 9 - 3 Conditions for Setting Each Flag to 1.	

TRGIMFB	Input-capture/compare-match flag B
[Condition for setting to 0]	
Write 0 after reading <small>Notes 2, 3</small>	
[Condition for setting to 1]	
See Table 9 - 3 Conditions for Setting Each Flag to 1.	

TRGIMFA	Input-capture/compare-match flag A
[Condition for setting to 0]	
Write 0 after reading <small>Notes 2, 3</small>	
[Condition for setting to 1]	
See Table 9 - 3 Conditions for Setting Each Flag to 1.	

**Note 1.** When the counter value of timer RG changes from FFFFH to 0000H, the TRGOVF bit is set to 1. Also, if the counter value of timer RG changes from FFFFH to 0000H due to an input capture/compare match during operation according to the settings of bits TRGCCLR0 and TRGCCLR1 in the TRGCR register, the TRGOVF bit is set to 1.

**Note 2.** The writing results are as follows:

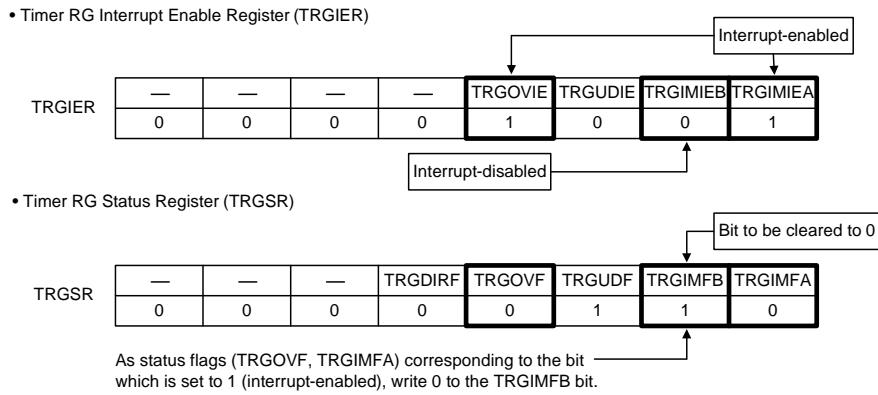
- Writing 1 has no effect.
- If the read value is 0, the bit remains unchanged even if 0 is written to it.  
(Even if the bit is changed from 0 to 1.)
- If the read value is 1, writing 0 to the bit sets it to 0.

When status flags of interrupt sources (applicable status flags) of the timer RG are set to 0 and their interrupts are disabled in the timer RG interrupt enable register (TRGIER), use either one of the following methods (a) to (c).

(a) Set 00H (all interrupts disabled) to timer RG interrupt enable register (TRGIER) and write 0 to applicable status flags.

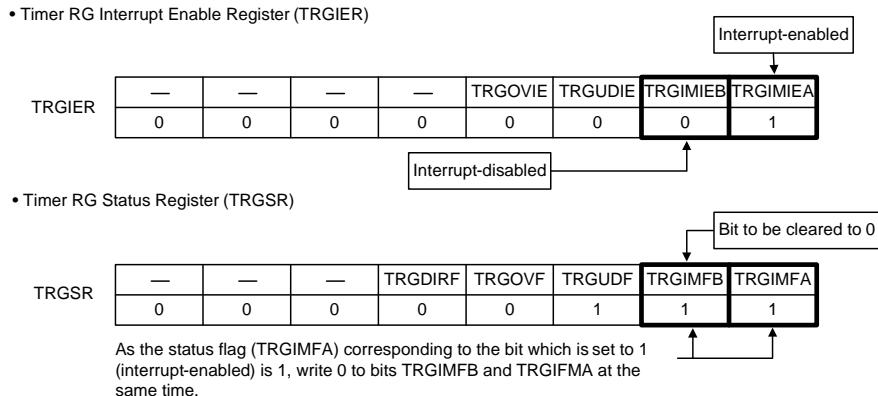
(b) When there are bits set to 1 (interrupt-enabled) in timer RG interrupt enable register (TRGIER) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the TRGIMFB bit to 0 when bits TRGIMIEA and TRGOVIE are set to 1 (interrupt-enabled) and the TRGIMIEB bit is set to 0 (interrupt-disabled)



(c) When there are bits set to 1 (interrupt-enabled) in timer RG interrupt enable register (TRGIER) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the TRGIMFB bit to 0 when the TRGIMIEA bit is set to 1 (interrupt-enabled) and the TRGIMIEB bit is set to 0 (interrupt-disabled).



**Note 3.** When the DTC is used, bits TRGIMFA and TRGIMFB are set to 1 after DTC transfer is completed.

**Table 9 - 3 Conditions for Setting Each Flag to 1**

Bit Symbol	Timer Mode Note 1		PWM Mode
	Input Capture Function	Output Compare Function	
TRGOVF	When the TRG register overflows.		
TRGUDF	When the TRG register underflows (only in phase counting mode).		
TRGIMFB	Input edge of TRGIOB pin Note 2	When the values of registers TRG and TRGGRB match.	
TRGIMFA	Input edge of TRGIOA pin Note 2	When the values of registers TRG and TRGGRA match.	

**Note 1.** Phase counting mode is the counting method of the timer RG count register. The above timer modes and PWM mode can be used by making the corresponding settings.

**Note 2.** Edge selected by bits TRGIOj0 and TRGIOj1 (j = A or B) in the TRGIOR register.

### 9.3.7 Timer RG I/O control register (TRGIOR)

Figure 9 - 8 Format of Timer RG I/O control register (TRGIOR)

Address: F0255H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

TRGIOR	TRGBUFB	TRGIOB2	TRGIOB1	TRGIOB0	TRGBUFA	TRGIOA2	TRGIOA1	TRGIOA0
--------	---------	---------	---------	---------	---------	---------	---------	---------

TRGBUFB	TRGGRD register function select						
0	Not used as buffer register for TRGGRB register						
1	Used as buffer register for TRGGRB register						

TRGIOB2	TRGGRB mode select Notes 1, 2						
0	Output compare function						
1	Input capture function						

TRGIOB1	TRGIOB0	TRGGRB control					
0	0	Pin output by compare match is disabled					
0	1	Low output					
1	0	High output					
1	1	Toggle output					
In the output compare function, output of compare match between registers TRG and TRGGRB							

TRGIOB1	TRGIOB0	TRGGRB control					
0	0	Rising edge of TRGIOB					
0	1	Falling edge of TRGIOB					
1	0	Both edges of TRGIOB					
Other than above		Setting prohibited					
In the input capture function, input capture of content of TRG register to TRGGRB register							

TRGBUFA	TRGGRC register function select						
0	Not used as buffer register for TRGGRA register						
1	Used as buffer register for TRGGRA register						

TRGIOA2	TRGGRA mode select Notes 1, 2						
0	Output compare function						
1	Input capture function						

TRGIOA1	TRGIOA0	TRGGRA control					
0	0	Pin output by compare match is disabled					
0	1	Low output					
1	0	High output					
1	1	Toggle output					
In the output compare function, output of compare match between registers TRG and TRGGRA							

		TRGGRA control
TRGIOA1	TRGIOA0	
0	0	Rising edge of TRGIOA
0	1	Falling edge of TRGIOA
1	0	Both edges of TRGIOA
Other than above		Setting prohibited
In the input capture function, input capture of content of TRG register to TRGGRA register		

**Note 1.** When the TRGIOj2 ( $j = A$  or  $B$ ) bit is 1 (input capture function), the TRGGRj register functions as an input capture register.

**Note 2.** When the TRGIOj2 ( $j = A$  or  $B$ ) bit is 0 (output compare function), the TRGGRj register functions as a compare match register. After a reset, the TRGIOj pin outputs as follows until bits TRGIOj0 and TRGIOj1 are set and the first compare match occurs.

TRGIOj1 and TRGIOj0 = 01B: High output  
10B: Low output  
11B: Low output

This TRGIOR register controls I/O pins in timer mode. It is disabled in PWM mode. It is disabled in PWM mode. Set the TRGIOR register while the count is stopped (TRGSTART in TRGMR register = 0).

### 9.3.8 Timer RG counter (TRG)

The TRG register is connected to the CPU via the internal 16-bit bus and should be always accessed in 16-bit units. This register operates incrementing and can also operate free-running, period counting, or external event counting. It can be cleared to 0000H by the compare match with the corresponding TRGGRA or TRGGRB register, or the input capture to registers TRGGRA and TRGGRB (count clear function).

When the TRG register overflows (FFFFH → 0000H), the TRGOVF flag in the TRGSR register is set to 1.

When the TRG register underflows (0000H → FFFFH), the TRGUDF flag in the TRGSR register is set to 1.

**Figure 9 - 9 Format of Timer RG counter (TRG)**

Address: F0256H      After reset: 0000H      RW

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRG																
—	Function															Setting Range
Bits 15 to 0	In phase counting mode, count operation is increment/decrement. In other modes, count operation is increment.															0000H to FFFFH

### 9.3.9 Timer RG general registers A, B, C, and D (TRGGRA, TRGGRB, TRGGRC, TRGGRD)

Registers TRGGRA and TRGGRB are 16-bit readable/writable registers with both the output compare and input capture register functions. These functions can be switched by setting the TRGIOR register.

When registers TRGGRA and TRGGRB are used as output compare registers, the values of registers TRGGRA and TRGGRB and the value of the TRG register are always compared. When their values match (compare match), bits TRGIMFA and TRGIMFB in the TRGSR register are set to 1. Compare match output can be set by the TRGIOR register.

When registers TRGGRA and TRGGRB are used as input capture registers, the value of the TRG register is stored upon detecting externally input capture signals. At this time, the TRGIMFA/TRGIMFB bit is set to 1. The detection edge of input capture signals is selected by setting the TRGIOR register.

The TRGGRC register can also be used as the buffer register for the TRGGRA register and the TRGGRD register can be used as the buffer register for the TRGGRB register, respectively. These functions can be selected by setting bits TRGBUFA and TRGBUFB in the TRGIOR register.

For example, when the TRGGRA register is set as an output compare register and the TRGGRC register is set as the buffer register for the TRGGRA register, the value of the TRGGRC register is transferred to the TRGGRA register each time compare match A occurs.

When the TRGGRA register is set as an input capture register and the TRGGRC register is set as the buffer register for the TRGGRA register, the value of the TRG register is transferred to the TRGGRA register and the value of the TRGGRA register value is transferred to the TRGGRC register each time an input capture occurs.

Registers TRGGRA, TRGGRB, TRGGRC, and TRGGRD can be read or written in 16-bit units.

**Figure 9 - 10 Format of Timer RG general registers A, B, C, and D (TRGGRA, TRGGRB, TRGGRC, TRGGRD)**

Address: F0258H (TRGGRA), F025AH (TRGGRB), FFF60H (TRGGRC), FFF62H (TRGGRD) After Reset: FFFFH RW

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--------	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

TRGGGi																
--------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

—	Function
Bits 15 to 0	Function varies depending on the mode or the function. Table 9 - 4 lists the TRGGRA, TRGGRB, TRGGRC, and TRGGRD Register Functions.

**Remark** i = A, B, C, D

**Table 9 - 4 TRGGRA, TRGGRB, TRGGRC, and TRGGRD Register Functions**

Mode, Function	Register	Setting	Function
Input capture	TRGGRA	TRGIOR (TRGIOA2 = 1) TRGMR (TRGPWM = 0)	Input capture register (stores value of TRG register)
	TRGGRB	TRGIOR (TRGIOB2 = 1) TRGMR (TRGPWM = 0)	Input capture register (stores value of TRG register)
Output compare	TRGGRA	TRGIOR (TRGIOA2 = 0) TRGMR (TRGPWM = 0)	Output compare register (stores compare value with TRG register and outputs set value to TRGIOA at compare match)
	TRGGRB	TRGIOR (TRGIOB2 = 0) TRGMR (TRGPWM = 0)	Output compare register (stores compare value with TRG register and outputs set value to TRGIOB at compare match)
PWM	TRGGRA	TRGMR (TRGPWM = 1)	Output compare register (outputs high level to TRGIOA at compare match)
	TRGGRB		Output compare register (outputs low level to TRGIOA at compare match)
Common	TRGGRC	TRGIOR (TRGBUFA = 0)	Not used
	TRGGRD	TRGIOR (TRGBUFB = 0)	Not used
	TRGGRC	TRGIOR (TRGBUFA = 1)	Buffer register for TRGGRA (transfers from/to TRGGRA) <ul style="list-style-type: none"> <li>• When TRGIOA2 = 1</li> <li>  Input capture signal: Receive previous input capture value from TRGGRA</li> <li>• When TRGIOA2 = 0</li> <li>  TRG and TRGGRA compare match: Send next expected compare value to TRGGRA</li> </ul>
	TRGGRD	TRGIOR (TRGBUFB = 1)	Buffer register for TRGGRB (transfers from/to TRGGRB) <ul style="list-style-type: none"> <li>• When TRGIOB2 = 1</li> <li>  Input capture signal: Receive previous input capture value from TRGGRB</li> <li>• When TRGIOB2 = 0</li> <li>  TRG and TRGGRB compare match: Send next expected compare value to TRGGRB</li> </ul>

**Caution** When the setting of bits TRGTCK2 to TRGTCK0 in the TRGCR register is 000B (fCLK) and the compare value is set to 0000H, a request signal to the DTC and the ELC is generated only once immediately after the count starts. When the compare value is 0001H or higher, a request signal is generated each time a compare match occurs.

### 9.3.10 Port mode registers 0, 5 (PM0, PM5)

These registers set input/output of ports 0, 5 in 1-bit units.

When using the ports (P50/TRGIOA, P51/TRGIOB) to be shared with the timer output pin for timer output, set the port mode register (PMxx) bit and port register (Pxx) bit corresponding to each port to 0.

Example When using P50/TRGIOA for timer output  
 Set the PM50 bit of port mode register 5 to 0.  
 Set the P50 bit of port register 5 to 0.

When using the ports (P50/TRGIOA, P51/TRGIOB) to be shared with the timer input pin for timer input, set the port mode register (PMxx) bit corresponding to each port to 1. At this time, the port register (Pxx) bit may be 0 or 1.

Example When using P50/TRGIOA for timer input  
 Set the PM50 bit of port mode register 5 to 1.  
 Set the P50 bit of port register 5 to 0 or 1.

The PM0 and PM5 registers can be set by a 1-bit or 8-bit memory manipulation instruction.  
 Reset signal generation sets these registers to FFH.

**Figure 9 - 11 Format of Port mode registers 0, 5 (PM0, PM5) (100-pin products)**

Address: FFF20H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM0	1	PM06	PM05	PM04	PM03	PM02	PM01	PM00

Address: FFF25H After reset: FFH R/W

Symbol	7	6	5	4	3	2	1	0
PM5	PM57	PM56	PM55	PM54	PM53	PM52	PM51	PM50
PMmn	Pmn pin I/O mode selection (m = 0, 5; n = 0 to 7)							
0	Output mode (output buffer on)							
1	Input mode (output buffer off)							

**Remark** The figure shown above presents the format of port mode registers 0 and 5 of the 100-pin products. The format of the port mode register of other products, see **Tables 4 - 7 to 4 - 13 PMxx, Pxx, PUxx, PIxx, POMxx, PMCxx registers and the bits mounted on each product.**

## 9.4 Timer RG Operation

### 9.4.1 Items Common to Multiple Modes and Functions

#### (1) Count Sources

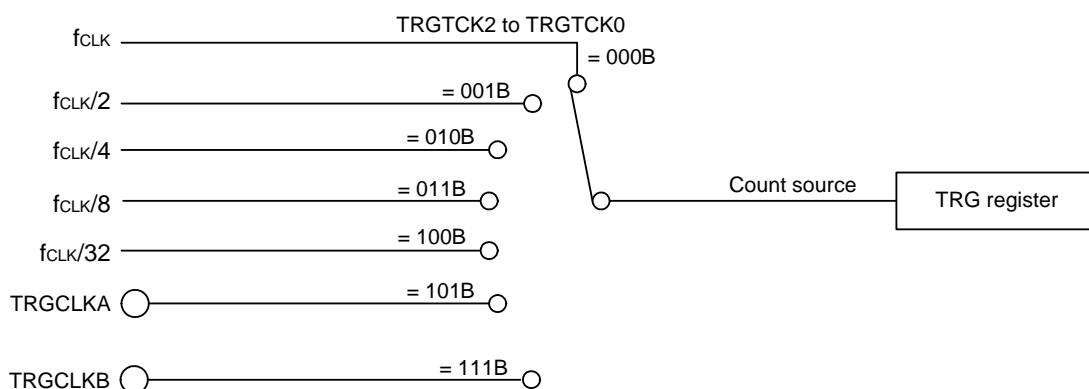
Table 9 - 5 lists the Count Source Selection and Figure 9 - 12 shows the Count Source Block Diagram.

When phase counting mode is selected, the settings of bits TRGTCK0 to TRGTCK2 and bits TRGCKEG0 and TRGCKEG1 in the TRGCR register are disabled.

**Table 9 - 5 Count Source Selection**

Count Source	Selection Method
fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32	The count source is selected by bits TRGTCK0 to TRGTCK2 in the TRGCR register.
External signal input to TRGCLKA or TRGCLKB pin	Bits TRGTCK2 to TRGTCK0 in the TRGCR register are set to 101B (TRGCLKA input) or 111B (TRGCLKB input). The active edge is selected by bits TRGCKEG0 and TRGCKEG1 in the TRGCR register. The corresponding bit of the port mode register is set to 1 (input mode).

**Figure 9 - 12 Count Source Block Diagram**



Remark TRGTCK0 to TRGTCK2: Bits in TRGCR register

The pulse width of an external clock input to the TRGCLKj pin ( $j = A$  or  $B$ ) should be set to three cycles or more of the timer RG operating clock (fCLK).

## (2) Buffer Operation

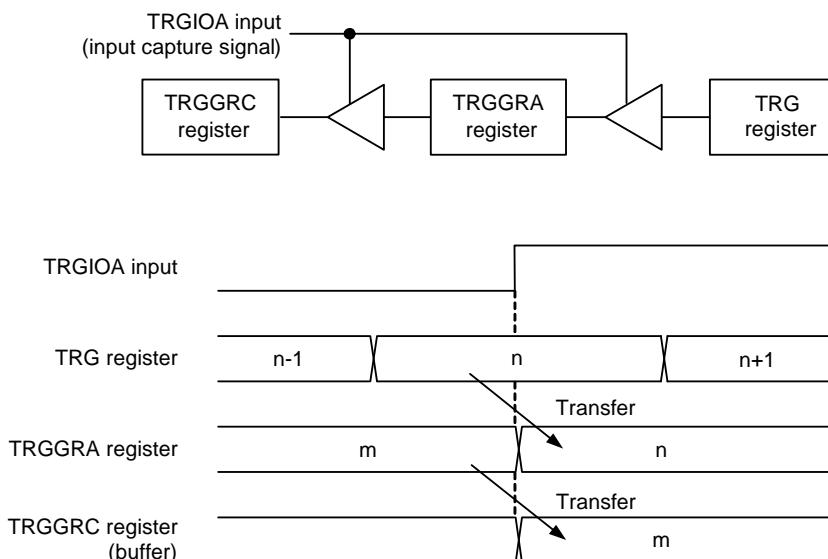
The TRGBUFA or TRGBUFB bit in the TRGIOR register can be used to select the TRGGRC or TRGGRD register as the buffer register for the TRGGRA or TRGGRB register.

- Buffer register for TRGGRA register: TRGGRC register
- Buffer register for TRGGRB register: TRGGRD register

Buffer operation differs depending on the mode. Table 9 - 6 lists the Buffer Operation in Each Mode, Figure 9 - 13 shows the Buffer Operation for Input Capture Function and Figure 9 - 14 shows the Buffer Operation for Output Compare Function.

**Table 9 - 6 Buffer Operation in Each Mode**

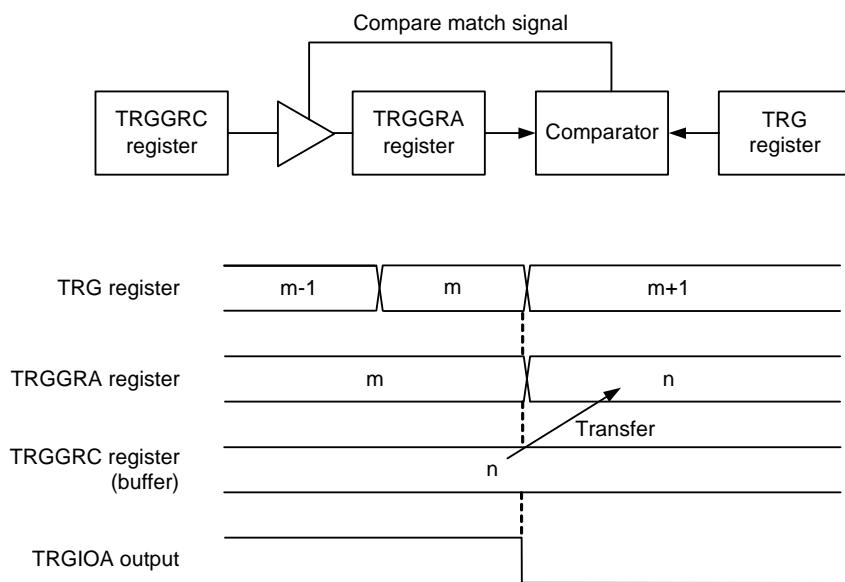
Function, Mode	Transfer Timing	Transfer Destination Register
Input capture function	Input capture signal input	The content of the TRGGRA (TRGGRB) register is transferred to the buffer register.
Output compare function	Compare match between the TRG register and the TRGGRA (TRGGRB) register	The content of the buffer register is transferred to the TRGGRA (TRGGRB) register.
PWM mode		

**Figure 9 - 13 Buffer Operation for Input Capture Function**

The above diagram applies under the following conditions

- The TRGBUFA bit in the TRGIOR register is set to 1 (TRGGRC register is used as buffer register for TRGGRA register).
- Bits TRGIOA2 to TRGIOA0 in the TRGIOR register are set to 100B (input capture at the rising edge).

Figure 9 - 14 Buffer Operation for Output Compare Function



The above diagram applies under the following conditions

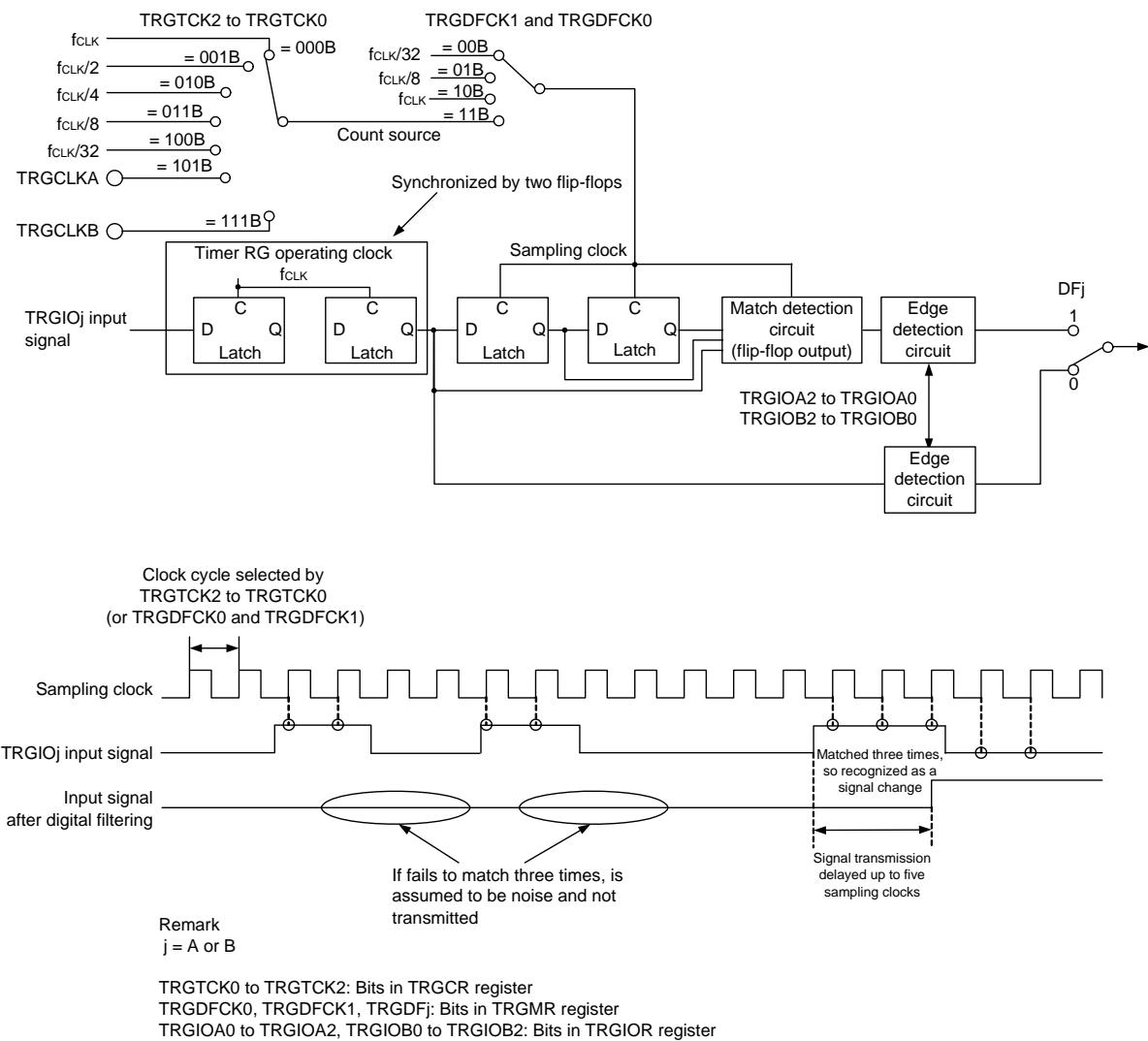
- The TRGBUFA bit in the TRGIOR register is set to 1 (TRGGRC register is used as buffer register for TRGGRA register).
- Bits TRGIOA2 to TRGIOA0 in the TRGIOR register are set to 001B (low output by compare match).

## (3) Digital Filter

The TRGIOj input ( $j = A$  or  $B$ ) is sampled, and when the sampled input level matches three times, its level is determined. Select the digital filter function and sampling clock using the TRGMR register.

Figure 9 - 15 shows a Block Diagram of Digital Filter.

**Figure 9 - 15 Block Diagram of Digital Filter**



## (4) Event Input from Event Link Controller (ELC)

Timer RG performs input capture operation B by event input from the ELC. The TRGIMFB bit in the TRGSR register is set to 1 at this time.

To use this function, select the input capture function of timer mode/phase counting mode, and set the TRGELCICE bit in the TRGMR register to 1. This function is disabled in other modes (the output compare function of timer mode/phase counting mode and PWM mode).

## Setting procedure

- <1> Set timer RG as the ELC event link destination.
- <2> Set the TRGELCICE bit in the TRGMR register to 1.

## (5) Event Output to Event Link Controller (ELC)

Table 9 - 7 lists the ELC Event Output according to TRGIMFA Bit. Table 9 - 8 lists the ELC Event Output according to TRGIMFB Bit.

**Table 9 - 7 ELC Event Output according to TRGIMFA Bit**

Mode, Function	ELC Source
Input capture function (TRGPWM = 0, TRGIOA2 = 1)	Detection of TRGIOA edge set by bits TRGIOA0 and TRGIOA1
Output compare function (TRGPWM = 0, TRGIOA2 = 0)	Compare match between registers TRG and TRGGRA
PWM mode (TRGPWM = 1)	Compare match between registers TRG and TRGGRA

**Remark** TRGPWM: Bit in TRGMR register

TRGIOA0, TRGIOA1, TRGIOA2: Bits in TRGIOR register

**Table 9 - 8 ELC Event Output according to TRGIMFB Bit**

Mode, Function	ELC Source
Input capture function (TRGPWM = 0, TRGIOB2 = 1)	Detection of TRGIOB edge set by bits TRGIOB0 and TRGIOB1
Output compare function (TRGPWM = 0, TRGIOB2 = 0)	Compare match between registers TRG and TRGGRB
PWM mode (TRGPWM = 1)	Compare match between registers TRG and TRGGRB

**Remark** TRGPWM: Bit in TRGMR register

TRGIOB0, TRGIOB1, TRGIOB2: Bits in TRGIOR register

### 9.4.2 Timer Mode (Input Capture Function)

The value of the TRG register can be transferred to registers TRGGRA and TRGGRB upon detecting the input edge of the input capture/output compare pins (TRGIOA and TRGIOB). The detection edge can be selected from the rising edge/falling edge/both edges.

The input capture function can be used for measuring pulse widths and periods.

Table 9 - 9 lists the Input Capture Function Specifications.

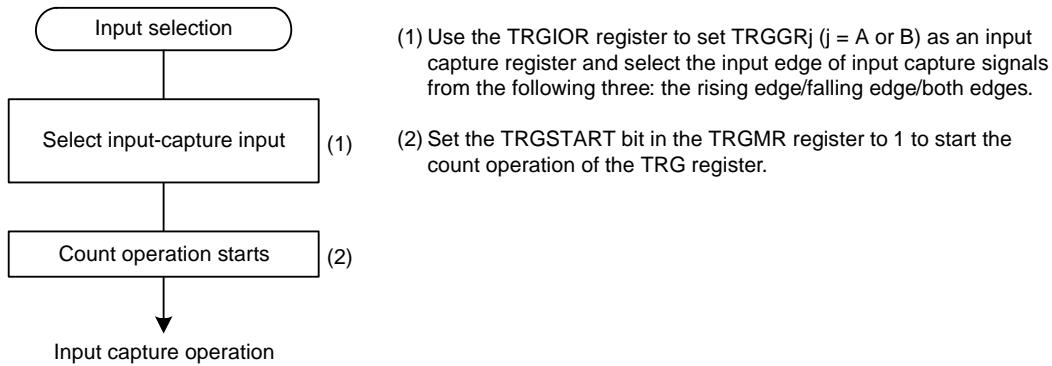
**Table 9 - 9 Input Capture Function Specifications**

Item	Specification
Count sources	fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRGCLKA or TRGCLKB pin (active edge selectable by a program)
Count operation	Increment
Count period	When bits TRGCCLR1 to TRGCCLR0 in the TRGCR register are set to 00B (free-running operation) $1/f_k \times 65,536 f_k$ : Frequency of count source
Count start condition	1 (count starts) is written to the TRGSTART bit in the TRGMR register.
Count stop condition	0 (count stops) is written to the TRGSTART bit in the TRGMR register.
Interrupt request generation timing	<ul style="list-style-type: none"> <li>• Input capture (active edge of TRGIOA and TRGIOB pin input)</li> <li>• TRG register overflow</li> </ul>
TRGIOA, TRGIOB pin function	I/O port or input-capture input (selectable for each pin)
TRGCLKA, TRGCLKB pin function	I/O port or external clock input
Read from timer	The count value can be read by reading the TRG register.
Write to timer	The TRG register can be written to.
Selectable functions	<ul style="list-style-type: none"> <li>• Input-capture input pin selection Either one or both of pins TRGIOA and TRGIOB</li> <li>• Active edge selection for input-capture input Rising edge, falling edge, or both rising and falling edges</li> <li>• Timing for setting the TRG register to 0000H At overflow or input capture</li> <li>• Buffer operation (see <b>9.4.1 (2) Buffer Operation</b>)</li> <li>• Digital filter (see <b>9.4.1 (3) Digital Filter</b>)</li> <li>• Input capture operation by event input signal (input capture) from ELC</li> </ul>

## (1) Procedure Example for Setting Input Capture Operation

Figure 9 - 16 shows a Procedure Example for Setting Input Capture Operation.

**Figure 9 - 16 Procedure Example for Setting Input Capture Operation**



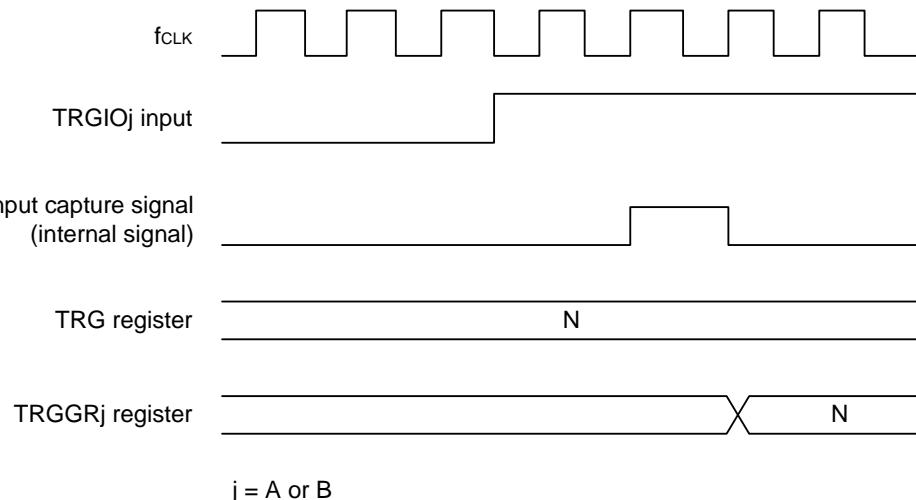
## (2) Input Capture Signal Timing

For input-capture input, the rising edge/falling edge/both edges can be selected by setting the TRGIOR register.

Figure 9 - 17 shows the Input-Capture Input Signal Timing.

The pulse width of input-capture input signals should be 1.5 f<sub>CLK</sub> or more for a single edge and 2.5 f<sub>CLK</sub> or more for both edges.

**Figure 9 - 17 Input-Capture Input Signal Timing**



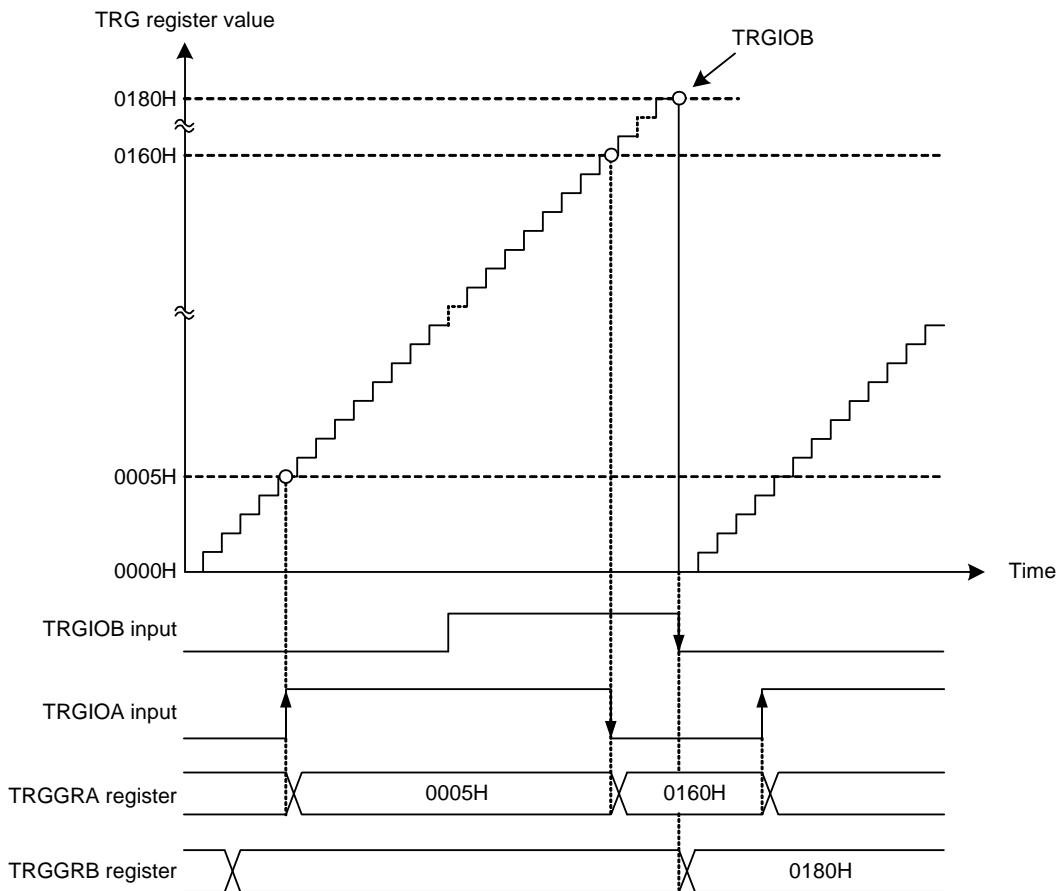
## (3) Operation Example

Figure 9 - 18 shows an Operation Example of Input Capture.

This example applies when both the rising/falling edges are selected as the input-capture input edge of the TRGIOA pin and the falling edge is selected as the input-capture input edge of the TRGIOB pin, and the TRG register is set to be cleared by the input capture to the TRGGRB register.

- (a) Use the TRGIOR register to set registers TRGGRA and TRGGRB as input capture registers and select the input edge of input capture signals from the following three: the rising edge/falling edge/both edges.
- (b) Set the TRGSTART bit in TRGMR to 1 and start the count operation of the TRG register.

**Figure 9 - 18 Operation Example of Input Capture**



By setting bits TRGCCLR0 and TRGCCLR1 in the TRGCR register, the count can be cleared by input capture A or B.

Figure 9 - 18 shows an operation example with bits TRGCCLR1 and TRGCCLR0 set to 10B. If the input capture operation has been set to clear the count during operation and is performed when the timer count value is FFFFH, depending on the timing between the count source and input capture operation interrupt flags bits TRGIMFA, TRGIMFB, and TRGOVF may be set to 1 simultaneously.

### 9.4.3 Timer Mode (Output Compare Function)

This mode (output compare function) detects when the contents of the TRG register and the TRGGRA or TRGGRB register match (compare match). When a match occurs, a signal is output from the TRGIOA or TRGIOB pin at a given level.

Table 9 - 10 lists the Output Compare Function Specifications.

**Table 9 - 10 Output Compare Function Specifications**

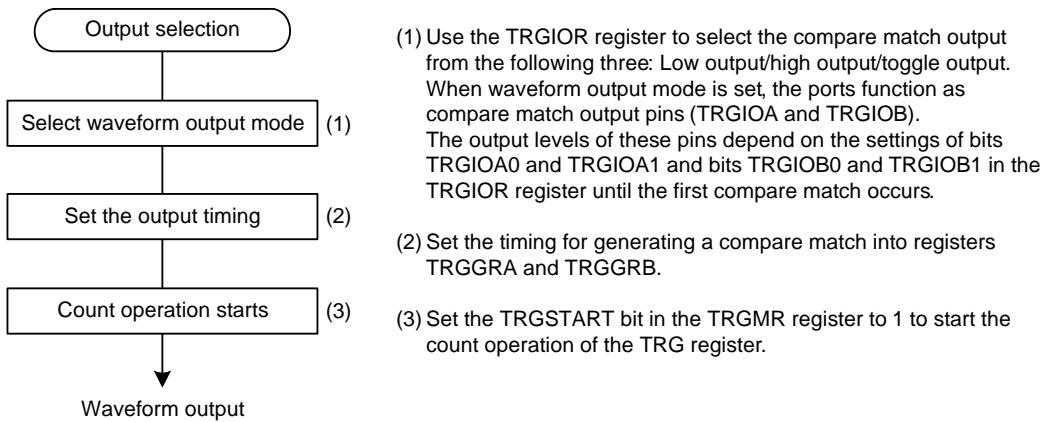
Item	Specification
Count sources	fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRGCLKj pin (active edge selectable by a program)
Count operation	Increment
Count periods	<ul style="list-style-type: none"> <li>When bits TRGCCLR1 and TRGCCLR0 in the TRGCR register are set to 00B (free-running operation)  <math>1/f_k \times 65,536 f_k</math>: Frequency of count source</li> <li>When bits TRGCCLR1 and TRGCCLR0 in the TRGCR register are set to 01B or 10B (TRG is set to 0000H by compare match with TRGGRj)  <math>1/f_k \times (n + 1)</math>  n: Value set in the TRGGRj register</li> </ul>
Waveform output timing	Compare match (contents of registers TRG and TRGGRj match)
Count start condition	1 (count starts) is written to the TRGSTART bit in the TRGMR register.
Count stop condition	0 (count stops) is written to the TRGSTART bit in the TRGMR register.
Interrupt request generation timing	<ul style="list-style-type: none"> <li>Compare match (contents of registers TRG and TRGGRj match)</li> <li>TRG register overflow</li> </ul>
TRGIOA, TRGIOB pin function	I/O port or output-compare output (selectable for each pin)
TRGCLKA, TRGCLKB pin function	I/O port or external clock input
Read from timer	The count value can be read by reading the TRG register.
Write to timer	The TRG register can be written to.
Selectable functions	<ul style="list-style-type: none"> <li>Output-compare output pin selection Either one or both of pins TRGIOA and TRGIOB</li> <li>Output level selection at compare match Low output, high output, or inverted output level</li> <li>Timing for setting the TRG register to 0000H Overflow or compare match with the TRGGRj register</li> <li>Buffer operation (see <b>9.4.1 (2) Buffer Operation</b>)</li> </ul>

**Remark** j = A or B

## (1) Procedure Example for Setting Waveform Output by Compare Match

Figure 9 - 19 shows a Procedure Example for Setting Waveform Output by Compare Match.

**Figure 9 - 19 Procedure Example for Setting Waveform Output by Compare Match**

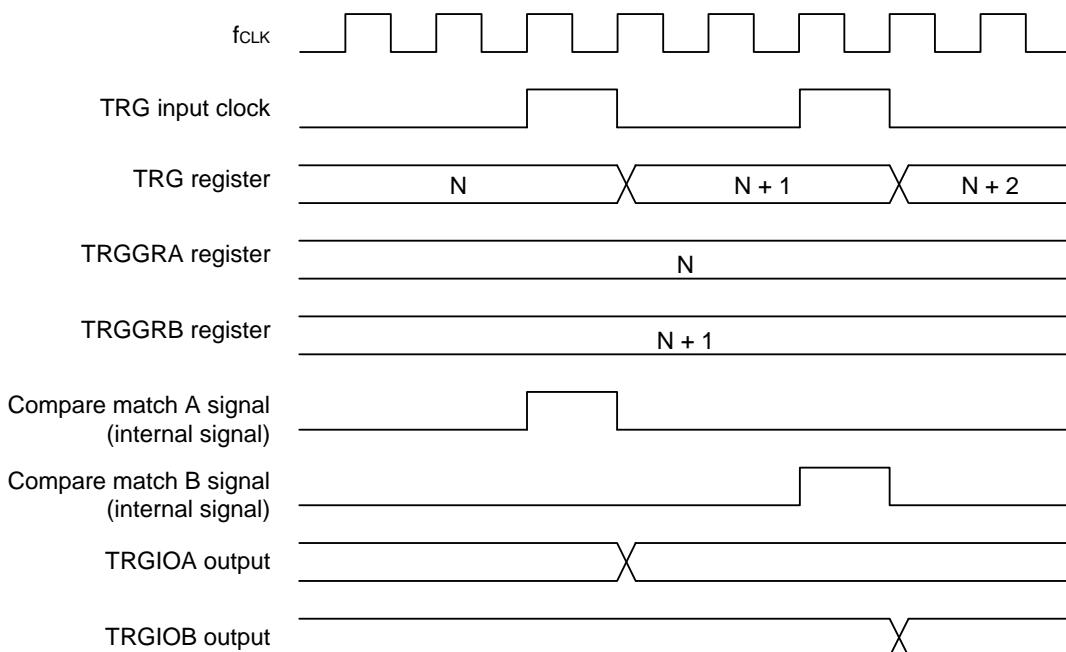


## (2) Output-Compare Output Timing

A compare match signal is generated at the last state when the TRG register and the TRGGRA or TRGGRB register match (at the timing for updating the count value that the TRG register matches). When the compare match signal is generated, the output value set by the TRGIOR register is output to the output-compare output pin (TRGIOA or TRGIOB). After the TRG register and the TRGGRA or TRGGRB register match, no compare match signal is generated until the TRG input clock is generated.

Figure 9 - 20 shows the Output-Compare Output Timing.

**Figure 9 - 20 Output-Compare Output Timing**



### (3) Operation Example

Figure 9 - 21 shows an Operation Example of Low Output and High Output.

This example applies when the TRG register is set for free-running operation, and low output is set at compare match A, and high output is set at compare match B. When the set level and the pin level match, the pin level does not change.

**Figure 9 - 21 Operation Example of Low Output and High Output**

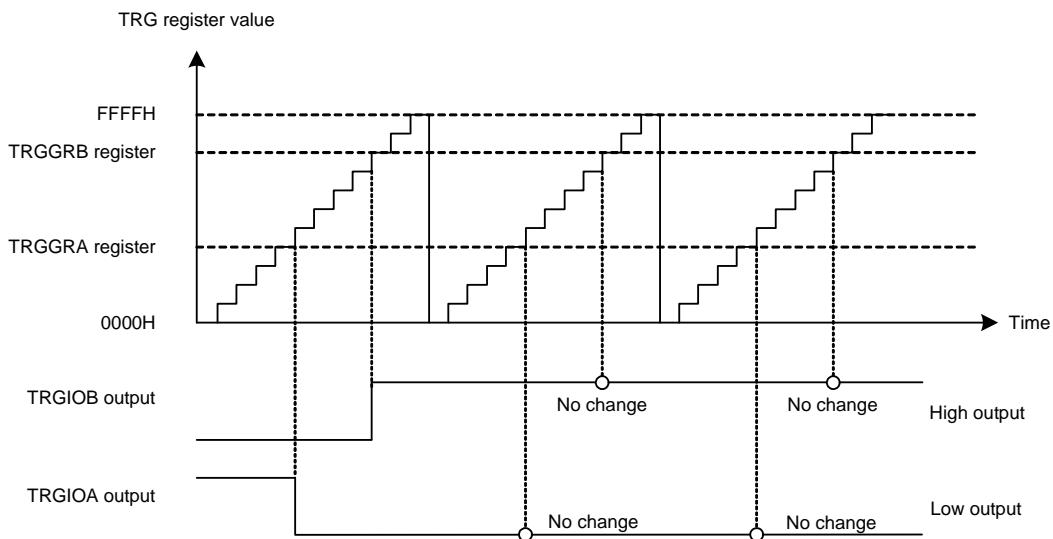


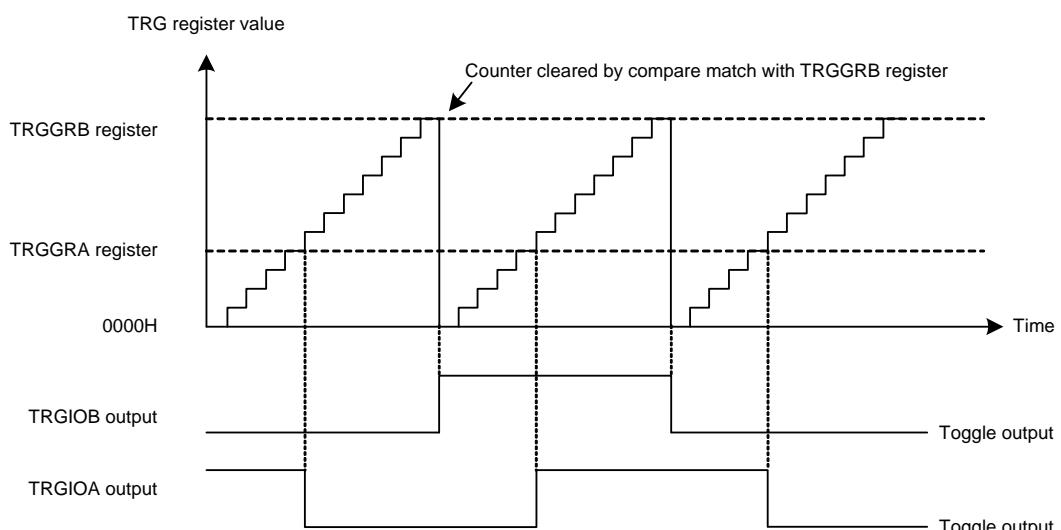
Figure 9 - 22 shows the Operation Example of Toggle Output. This example applies when the TRG register is set for period counting operation (counter clear at compare match B), and toggle output is set at both compare match A and B.

- (a) Use the TRGIOR register to select the compare match output from the following three: Low output/high output/toggle output. When waveform output mode is set, the ports function as compare match output pins (TRGIOA and TRGIOB).
- (b) Set the timing for generating a compare match into registers TRGGRA and TRGGRB.
- (c) Set the TRGSTART bit in the TRGMR register to 1 to start the count operation of the TRG register.

The compare match output pins (TRGIOA and TRGIOB) are not initialized by setting the TRGSTART bit to 0 during operation. To return to initial values, write to the TRGIOR register to initialize the output. (The output is only initialized when bits TRGIOA0, TRGIOA1, TRGIOB0, and TRGIOB1 in the TRGIOR register are set to low output or high output.) By setting bits TRGCCLR0 and TRGCCLR1 in the TRGCR register, the timer RG counter value is reset by an input capture/compare match (match with the TRGGRA or TRGGRB register). If the expected compare value is FFFFH at this time, FFFFH changes to 0000H, same as the overflow operation, and the TRGOVF bit is set to 1.

This operation is the same for modes where the output compare function is used on the timer RG counter value and expected compare value.

**Figure 9 - 22 Operation Example of Toggle Output**



#### 9.4.4 PWM Mode

In PWM mode, registers TRGGRA and TRGGRB are used as a pair and a PWM waveform is output from the TRGIOA output pin. The output setting by the TRGIOR register is invalid for the pins set to PWM mode. Set the high output timing for a PWM waveform into the TRGGRA register and the low output timing for a PWM waveform into the TRGGRB register.

By setting the compare match with either the TRGGRA or TRGGRB register as the counter clear source for the TRG register, a PWM waveform with duty cycle 0% to 100% can be output from the TRGIOA pin.

Table 9 - 11 lists the PWM Mode Specifications and Table 9 - 12 lists the Combination of PWM Output Pins and Registers. When the setting values in registers TRGGRA and TRGGRB are the same, the output value does not change even if a compare match occurs.

**Table 9 - 11 PWM Mode Specifications**

Item	Specification
Count sources	fCLK, fCLK/2, fCLK/4, fCLK/8, fCLK/32 External signal input to the TRGCLKj pin (active edge selectable by a program)
Count operation	Increment
PWM waveform	<ul style="list-style-type: none"> <li>• The high output timing of a PWM waveform is set into the TRGGRA register.</li> <li>• The low output timing of a PWM waveform is set into the TRGGRB register.</li> </ul>
Count start condition	1 (count starts) is written to the TRGSTART bit in the TRGMR register.
Count stop condition	0 (count stops) is written to the TRGSTART bit in the TRGMR register.
Interrupt request generation timing	<ul style="list-style-type: none"> <li>• Compare match (contents of registers TRG and TRGGRj match)</li> <li>• TRG register overflow</li> </ul>
TRGIOA pin function	PWM output
TRGIOB pin function	I/O port
TRGCLKA, TRGCLKB pin function	I/O port or external clock input
Read from timer	The count value can be read by reading the TRG register.
Write to timer	The TRG register can be written to.
Selectable functions	<ul style="list-style-type: none"> <li>• Timing for setting the TRG register to 0000H</li> <li>• Overflow or compare match with the TRGGRj register</li> <li>• Buffer operation (see <b>9.4.1 (2) Buffer Operation</b>)</li> </ul>

**Remark** j = A or B

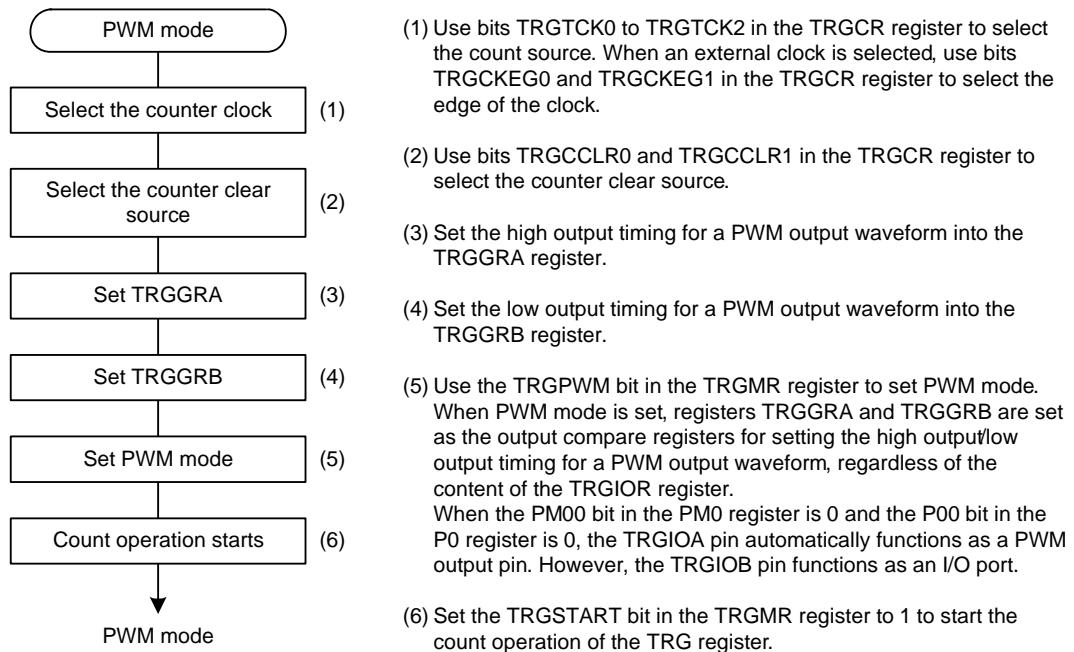
**Table 9 - 12 Combination of PWM Output Pins and Registers**

Output Pin	High Output	Low Output
TRGIOA	TRGGRA	TRGGRB
TRGIOB		I/O port function

## (1) Procedure Example for Setting PWM Mode

Figure 9 - 23 shows a Procedure Example for Setting PWM Mode.

**Figure 9 - 23 Procedure Example for Setting PWM Mode**



(2) Operation Example

Figure 9 - 24 shows an Operation Example (1) in PWM Mode.

When the PM00 bit in the PM0 register is 0 and the P00 bit in the P0 register is 0, the TRGIOA pin automatically functions as an output pin, and high output is set at the compare match with the TRGGRA register and low output is set at the compare match with the TRGGRB register. However, regardless of the setting of the TRGIOR register, the TRGIOB pin functions as an I/O port.

This example applies when the compare match with the TRGGRA or TRGGRB register is set as the counter clear source for the TRG register. The initial state of the TRGIOA pin depends only on the counter clear sources. This correspondence is shown in Table 9 - 13.

This initialization is performed when the TRGSTART bit in the TRGMR register is 0 (count stops).

**Table 9 - 13 Correspondence between Initial State of TRGIOA Pin and Counter Clear Sources**

Counter Clear Source	Initial State of TRGIOA Pin
Compare match with TRGGRA register	High
Compare match with TRGGRB register	Low

When bits TRGCCLR1 and TRGCCLR0 in the TRGCR register are set to 00B (clear disabled), the initial state of the TRGIOA pin becomes high.

**Figure 9 - 24 Operation Example (1) in PWM Mode**

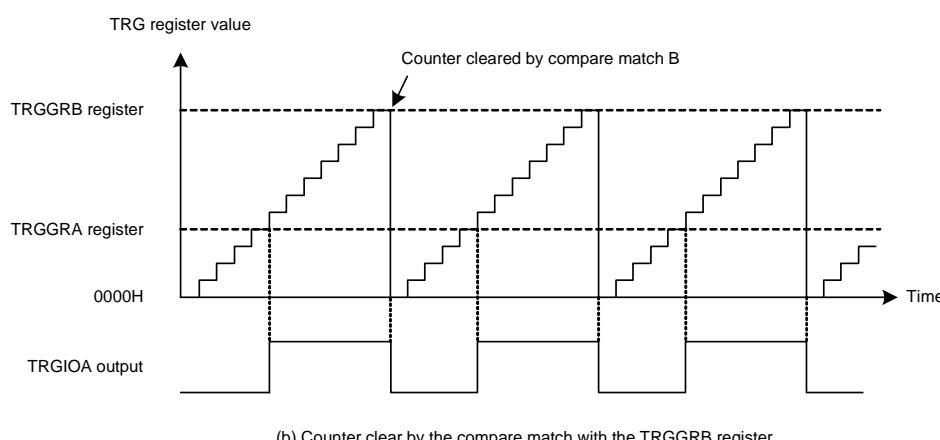
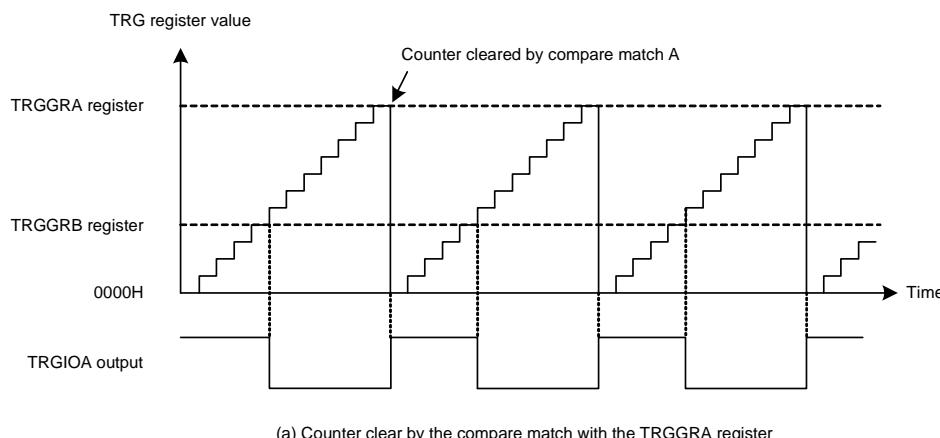


Figure 9 - 25 shows an example for outputting a PWM waveform with duty cycle 0% and duty cycle 100%. A PWM waveform is set to duty cycle 0% when the compare match with the TRGGRB register is set as the counter clear source with the following:

- Value set in TRGGRA register > Value set in TRGGRB register

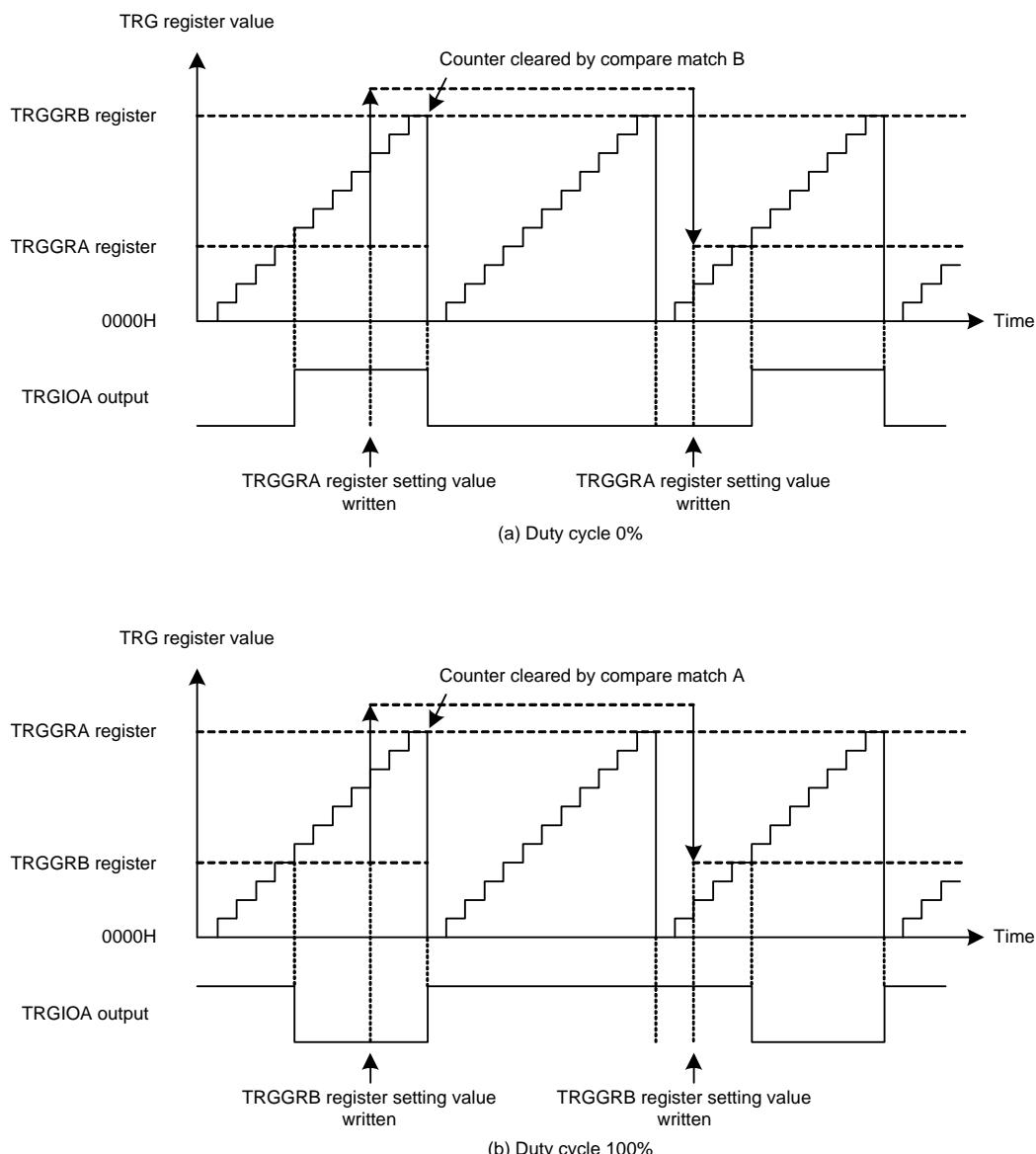
A PWM waveform is set to duty cycle 100% when the compare match with TRGGRA register is set as the counter clear source with the following:

- Value set in TRGGRB register > Value set in TRGGRA register

Output value is unchanged even if a compare match is generated with the following:

- Value set in TRGGRA register = Value set in TRGGRB register

**Figure 9 - 25 Operation Example (2) in PWM Mode**



### 9.4.5 Phase Counting Mode

In phase counting mode, a phase difference between external input signals from two pins TRGCLKA and TRGCLKB is detected and the TRG register is incremented/decremented.

When phase counting mode is set when bits PM00 and PM01 in the PM0 register are 1, regardless of the settings of bits TRGTCK0 to TRGTCK2 and bits TRGCKEG0 and TRGCKEG1 in the TRGCR register, pins TRGCLKA and TRGCLKB automatically function as external clock input pins and the TRG register is incremented/decremented by bits CNTEN0 to CNTEN7 in the TRGCNTC register. However, bits TRGCCLR0 and TRGCCLR1 in the TRGCR register and registers TRGIOR, TRGIER, TRGSR, TRGGRA, and TRGGRB are enabled. This allows the input capture/output compare functions, PWM output function, and interrupt sources to be used.

The TRG register operates counting at both the rising/falling edges of pins TRGCLKA and TRGCLKB by bits CNTEN0 to CNTEN7.

Table 9 - 14 lists the Phase Counting Mode Specifications and Table 9 - 15 lists the Increment/Decrement Conditions for TRG Register.

**Table 9 - 14 Phase Counting Mode Specifications**

Item	Specification
Count source	External signal input to the TRGCLKj pin
Count operations	Increment/decrement
Count start condition	1 (count starts) is written to the TRGSTART bit in the TRGMR register.
Count stop condition	0 (count stops) is written to the TRGSTART bit in the TRGMR register.
Interrupt request generation timing	<ul style="list-style-type: none"> <li>• Input capture (active edge of TRGIOj input)</li> <li>• Compare match (contents of registers TRG and TRGGRj match)</li> <li>• TRG register overflow</li> <li>• TRG register underflow</li> </ul>
TRGIOA pin function	I/O port, input-capture input, output-compare output, or PWM output
TRGIOB pin function	I/O port, input-capture input, or output-compare output
TRGCLKA, TRGCLKB pin function	External clock input
Read from timer	The count value can be read by reading the TRG register.
Write to timer	The TRG register can be written to.
Selectable functions	<ul style="list-style-type: none"> <li>• Selection of counter increment/decrement conditions Selectable by bits CNTEN0 to CNTEN7 in the TRGCNTC register.</li> <li>• Input capture/output compare functions and PWM function can be used.</li> </ul>

**Remark** j = A or B

**Table 9 - 15 Increment/Decrement Conditions for TRG Register**

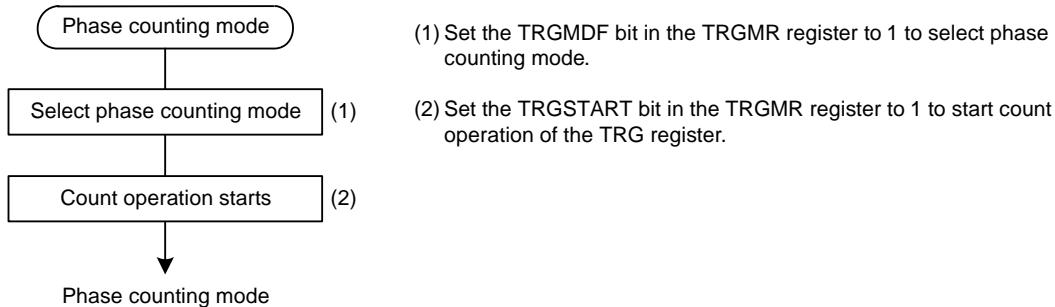
TRGCLKB pin	↑	High	↓	Low	High	↓	Low	↑
TRGCLKA pin	Low	↑	High	↓	↓	Low	↑	High
Bits CNTEN7 to CNTEN0 in TRGCNTC register	CNTEN7	CNTEN6	CNTEN5	CNTEN4	CNTEN3	CNTEN2	CNTEN1	CNTEN0
Count direction Note	+1	+1	+1	+1	-1	-1	-1	-1

**Note** The count direction when each bit in the TRGCNTC register is 1 (decrement or increment) is shown. When a bit is 0 (disabled), count is not performed.

## (1) Procedure Example for Setting Phase Counting Mode

Figure 9 - 26 shows a Procedure Example for Setting Phase Counting Mode.

**Figure 9 - 26 Procedure Example for Setting Phase Counting Mode**



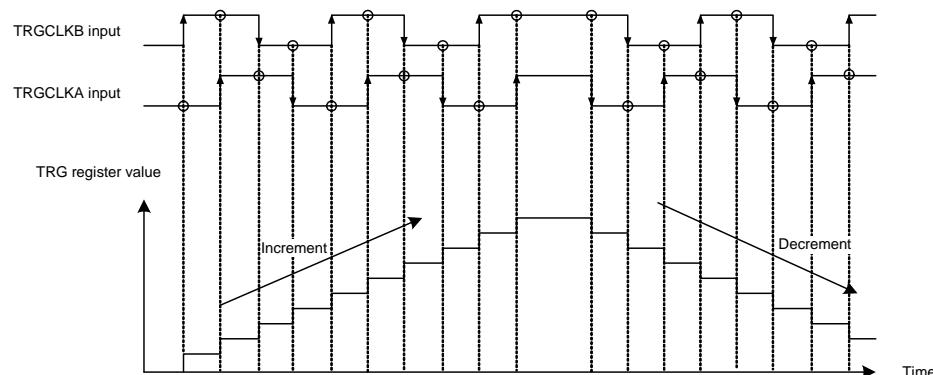
## (2) Operation Example

Figures 9 - 27 to 9 - 30 show Operation Examples in Phase Counting Mode.

In phase counting mode, the TRG register is incremented/decremented at both the rising(↑)/falling(↓) edges of pins TRGCLKA and TRGCLKB by bits CNTEN0 to CNTEN7 in the TRGCNTC register.

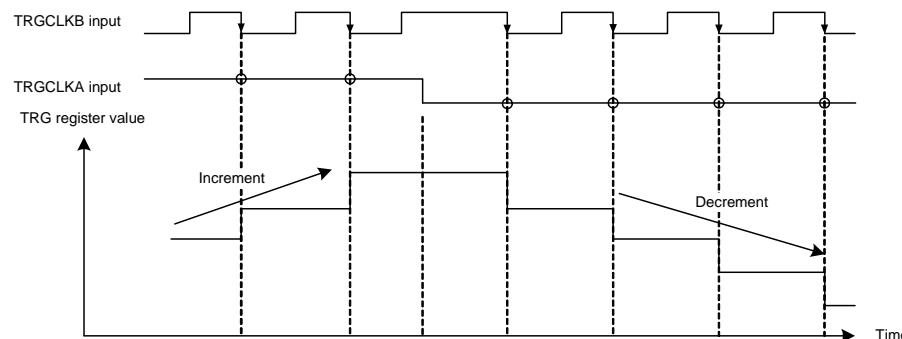
**Figure 9 - 27 Operation Example 1 in Phase Counting Mode**

- When the TRGCNTC register value is FFH



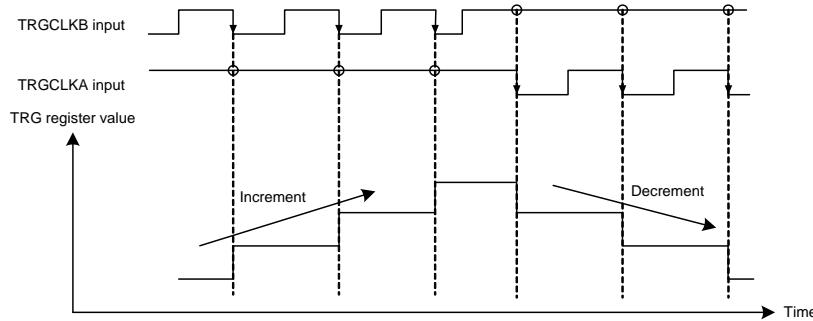
**Figure 9 - 28 Operation Example 2 in Phase Counting Mode**

- When the TRGCNTC register value is 24H

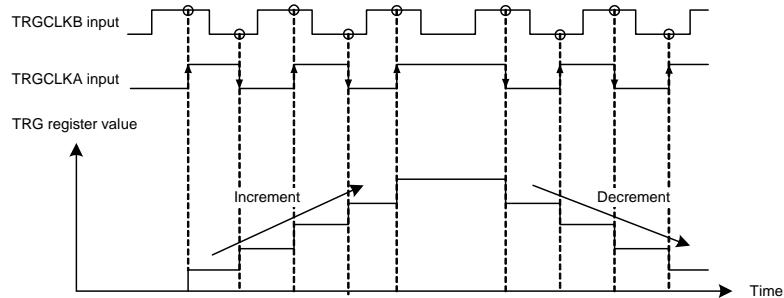


**Figure 9 - 29 Operation Example 3 in Phase Counting Mode**

- When the TRGCNTC register value is 28H

**Figure 9 - 30 Operation Example 4 in Phase Counting Mode**

- When the TRGCNTC register value is 5AH

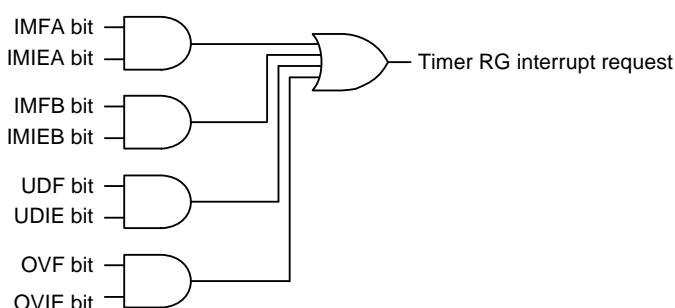


## 9.5 Timer RG Interrupt

Timer RG generates the timer RG interrupt request from four sources. Table 9 - 16 lists the Registers Associated with Timer RG Interrupt and Figure 9 - 31 shows the Timer RG Interrupt Block Diagram.

**Table 9 - 16 Registers Associated with Timer RG Interrupt**

	Timer RG Status Register	Timer RG Interrupt Enable Register	Interrupt Request Flag (Register)	Interrupt Mask Flag (Register)	Priority Specification Flag (Register)
Timer RG	TRGSR	TRGIER	TRGIF (IF2H)	TRGMK (MK2H)	TRGPR0 (PR02H) TRGPR1 (PR12H)

**Figure 9 - 31 Timer RG Interrupt Block Diagram**

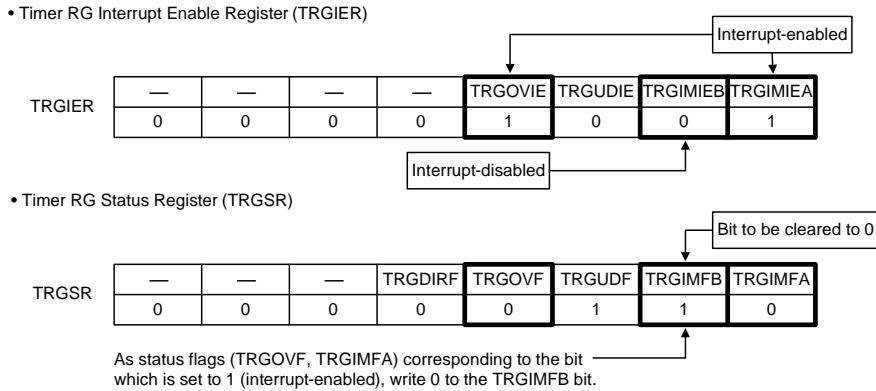
IMFA, IMFB, UDF, OVF: Bits in TRGSR register

IMIEA, IMIEB, UDIE, OVIE: Bits in TRGSR register

Since the interrupt source (timer RG interrupt) is generated by a combination of multiple interrupt request sources for timer RG, the following differences from other maskable interrupts except timer RD interrupt apply:

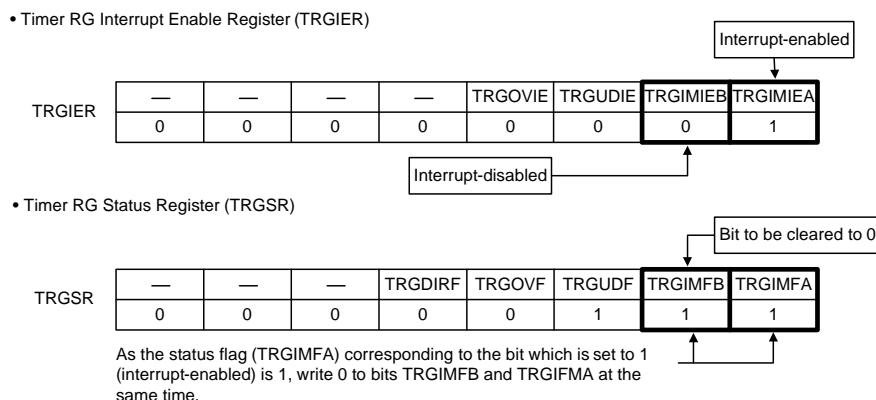
- When a bit in the TRGSR register is 1 and the corresponding bit in the TRGIER register is 1 (interrupt enabled), the TRGIF bit in the IF2H register is set to 1 (interrupt requested).
- If multiple bits in the TRGIER register are set to 1, use the TRGSR register to determine the source of the interrupt request.
- Since the bits in the TRGSR register are not automatically set to 0 even if the interrupt is acknowledged, set the corresponding bit to 0 in the interrupt routine.
- When status flags of interrupt sources (applicable status flags) of timer RG are set to 0 and their interrupts are disabled in the timer RG interrupt enable register (TRGIER), use either one of the following methods (a) to (c).
  - (a) Set 00H (all interrupts disabled) to the TRGIER register and write 0 to applicable status flags.
  - (b) When there are bits set to 1 (interrupt-enabled) in timer RG interrupt enable register (TRGIER) and status flags of interrupt sources related to their bits are 0, write 0 to applicable status flags.

Example: To clear the TRGIMFB bit to 0 when bits TRGIMIEA and TRGOVIE are set to 1 (interrupt-enabled) and the TRGIMIEB bit is set to 0 (interrupt-disabled).



- (c) When there are bits set to 1 (interrupt-enabled) in the timer RG interrupt enable register (TRGIER) and status flags of interrupt sources related to their bits are 1, write 0 to these status flags and applicable status flags at the same time.

Example: To clear the TRGIMFB bit to 0 when the TRGIMIEA bit is set to 1 (interrupt-enabled) and the TRGIMIEB bit is set to 0 (interrupt-disabled).



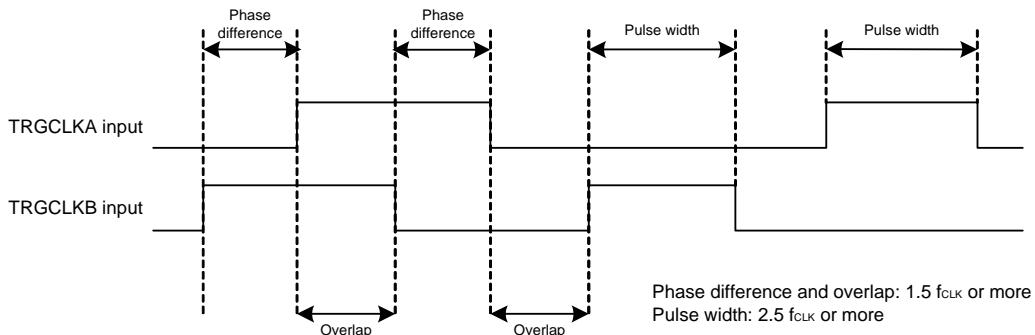
## 9.6 Cautions for Timer RG

### 9.6.1 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

The phase difference and overlap between external input signals from pins TRGCLKA and TRGCLKB should be 1.5 f<sub>CLK</sub> or more, respectively. The pulse width should be 2.5 f<sub>CLK</sub> or more.

Figure 9 - 32 shows the Phase Difference, Overlap, and Pulse Width in Phase Counting Mode.

**Figure 9 - 32 Phase Difference, Overlap, and Pulse Width in Phase Counting Mode**



### 9.6.2 Mode Switching

- When switching modes during operation, set the TRGSTART bit in the TRGMR register to 0 (count stops) before switching.
- After switching modes, set the TRGIF bit to 0 before starting operation.

Refer to **CHAPTER 21 INTERRUPT FUNCTIONS** for details.

### 9.6.3 Count Source Switching

- Stop the count before switching the count source Note.

Changing procedure

- Set the TRGSTART bit in the TRGMR register to 0 (count stops).
- Change bits TRGTCK0 to TRGTCK2 in the TRGCR register.

**Note** The registers and bits that cannot be rewritten during count operation are as follows:

- All bits except TRGSTART in the TRGMR register
- The TRGCNTC register
- The TRGCR register
- The TRGIOR register

### 9.6.4 Procedure for Setting Pins TRGIOA and TRGIOB

To output from pins TRGIOA and TRGIOB, use the following setting procedure:

Changing procedure

- (1) Set the mode and the initial value/output enabled (in order to make the initial value and enable settings using the same SFRs).
- (2) Set the port register bits corresponding to pins TRGIOA and TRGIOB to 0.
- (3) Set the port mode register bits corresponding to pins TRGIOA and TRGIOB to output mode (output is started from pins TRGIOA and TRGIOB).
- (4) Start the count (TRGSTART in TRGMR register = 1).

To change the port mode register bits corresponding to pins TRGIOA and TRGIOB from output mode to input mode, use the following setting procedure:

- (1) Set the port mode register bits corresponding to pins TRGIOA and TRGIOB to input mode (input is started from pins TRGIOA and TRGIOB).
- (2) Set to the input capture function.
- (3) Start the count (TRGSTART in TRGMR register = 1).

When switching pins TRGIOA and TRGIOB from output mode to input mode, input capture operation may be performed depending on the states of these pins. When the digital filter is not used, edge detection is performed after two or more cycles of the CPU clock have elapsed. When the digital filter is enabled, edge detection is performed after up to five cycles of the digital filter sampling clock.

### 9.6.5 External Clock TRGCLKA, TRGCLKB

The pulse width of an external clock input to the TRGCLK<sub>j</sub> pin (<sub>j</sub> = A or B) should be set to three cycles or more of the timer RG operating clock (fCLK).

### 9.6.6 SFR Read/Write Access

When setting timer RG, set the TRGEN bit in the PER1 register to 1 first. If the TRGEN bit is 0, writes to the timer RG control registers are ignored and all the read values are the initial values (except for the port registers and the port mode registers).

#### (1) TRGMR Register

Use the following setting procedure when switching the digital filter clock.

- (a) With the TRGSTART bit set to 0 (count stops), set bits TRGDFA and TRGDFB (digital filter function select bits of pins TRGIOA and TRGIOB) in the TRGMR register, and bits TRGDFCK0 and TRGDFCK1 (clock select bits used by digital filter function) in the TRGMR register.
- (b) Set the TRGSTART bit to 1.

However, when the digital filter is not set and TRGDFCK1 and TRGDFCK0 = 00B remain unchanged after a reset, the setting can be performed in a single step.

Besides external input pins (TRGIOA and TRGIOB), event input from the ELC can also be selected as an operating source for input capture. To use this function, set the TRGELCICE bit in the TRGMR register to 1, and set the input capture function (the rising edge as the active edge for input capture (TRGIOB2 to TRGIOB0 = 100B)).

This function is disabled in PWM mode and the timer mode output compare function (TRGPWM = 1 and TRGIOB2 = 0).

#### (2) TRG Register

- Writing to the TRGMR register has priority over count reset operations generated by timer RG operating conditions.

### 9.6.7 Input Capture Operation when Count is Stopped

In input capture mode, an input capture interrupt request for the active edge of the TRGIOj input is also generated when the TRGSTART bit in the TRGMR register is 0 (count stops) if the edge selected by bits TRGIOj0 and TRGIOj1 in the TRGIOR register is input to the TRGIOj pin (j = A or B).

## CHAPTER 10 REAL-TIME CLOCK

### 10.1 Functions of Real-time Clock

The real-time clock has the following features.

- Having counters of year, month, week, day, hour, minute, and second, and can count up to 99 years.
- Constant-period interrupt function (period: 0.5 seconds, 1 second, 1 minute, 1 hour, 1 day, 1 month)
- Alarm interrupt function (alarm: week, hour, minute)
- Pin output function of 1 Hz (40, 44, 48, 52, 64, 80, and 100-pin products only)

The real-time clock interrupt signal (INTRTC) can be utilized for wakeup from STOP mode and triggering an A/D converter's SNOOZE mode.

**Caution** The count of year, month, week, day, hours, minutes and second can only be performed when a subsystem clock ( $f_{SUB} = 32.768\text{ kHz}$ ) is selected as the operation clock of the real-time clock. When the low-speed oscillation clock ( $f_{IL} = 15\text{ kHz}$ ) is selected, only the constant-period interrupt function is available. The 30-, 32-, and 36-pin products have the constant-period interrupt function only, because these products have no subsystem clock. However, the constant-period interrupt interval when  $f_{IL}$  is selected will be calculated with the constant-period (the value selected with RTCC0 register)  $\times f_{SUB}/f_{IL}$ .

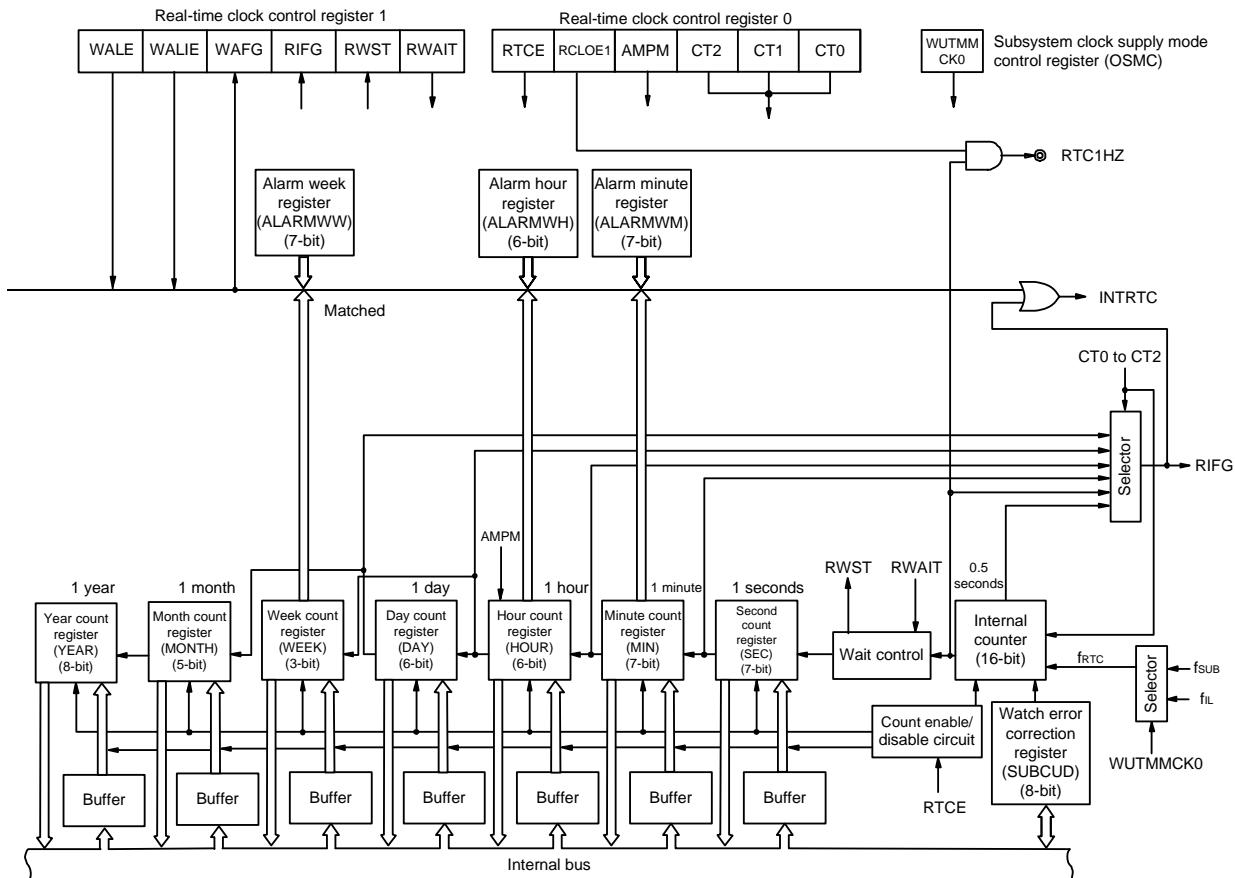
### 10.2 Configuration of Real-time Clock

The real-time clock includes the following hardware.

Table 10 - 1 Configuration of Real-time Clock

Item	Configuration
Counter	Internal counter (16-bit)
Control registers	Peripheral enable register 0 (PER0) Subsystem clock supply mode control register (OSMC) Real-time clock control register 0 (RTCC0) Real-time clock control register 1 (RTCC1) Second count register (SEC) Minute count register (MIN) Hour count register (HOUR) Day count register (DAY) Week count register (WEEK) Month count register (MONTH) Year count register (YEAR) Watch error correction register (SUBCUD) Alarm minute register (ALARMWM) Alarm hour register (ALARMWH) Alarm week register (ALARMWW)

Figure 10 - 1 Block Diagram of Real-time Clock



**Caution** The count of year, month, week, day, hour, minutes and second can only be performed when a subsystem clock ( $f_{SUB} = 32.768\text{ kHz}$ ) is selected as the operation clock of the real-time clock.

When the low-speed oscillation clock ( $f_{IL} = 15\text{ kHz}$ ) is selected, only the constant-period interrupt function is available. The 30-, 32-, and 36-pin products have the constant-period interrupt function only, because these products have no subsystem clock.

However, the constant-period interrupt interval when  $f_{IL}$  is selected will be calculated with the constant-period (the value selected with RTCC0 register)  $\times f_{SUB}/f_{IL}$ .

### 10.3 Registers Controlling Real-time Clock

The real-time clock is controlled by the following registers.

- Peripheral enable register 0 (PER0)
- Subsystem clock supply mode control register (OSMC)
- Real-time clock control register 0 (RTCC0)
- Real-time clock control register 1 (RTCC1)
- Second count register (SEC)
- Minute count register (MIN)
- Hour count register (HOUR)
- Day count register (DAY)
- Week count register (WEEK)
- Month count register (MONTH)
- Year count register (YEAR)
- Watch error correction register (SUBCUD)
- Alarm minute register (ALARMWM)
- Alarm hour register (ALARMWH)
- Alarm week register (ALARMWW)
- Port mode register 3 (PM3)
- Port register 3 (P3)

### 10.3.1 Peripheral enable register 0 (PER0)

This register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the real-time clock is used, be sure to set bit 7 (RTCEN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 2 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

PER0	RTCEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
RTCEN	Control of real-time clock (RTC) and 12-bit interval timer input clock supply							
0	Stops input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer cannot be written. • The real-time clock (RTC) and 12-bit interval timer are in the reset status.							
1	Input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer can be read/written.							

**Note**      80 and 100-pin products only.

**Caution 1.** When using the real-time clock, first set the RTCEN bit to 1 and then set the following registers, while oscillation of the count clock (fRTC) is stable. If RTCEN = 0, writing to the control registers of the real-time clock is ignored, and, even if the registers are read, only the default values are read (except for the subsystem clock supply mode control register (OSMC), port mode register 3 (PM3), port register 3 (P3)).

- Real-time clock control register 0 (RTCC0)
- Real-time clock control register 1 (RTCC1)
- Second count register (SEC)
- Minute count register (MIN)
- Hour count register (HOUR)
- Day count register (DAY)
- Week count register (WEEK)
- Month count register (MONTH)
- Year count register (YEAR)
- Watch error correction register (SUBCUD)
- Alarm minute register (ALARMWM)
- Alarm hour register (ALARMWH)
- Alarm week register (ALARMWW)

**Caution 2.** Subsystem clock supply to peripheral functions other than the real-time clock and 12-bit interval timer can be stopped in STOP mode or HALT mode when the subsystem clock is used, by setting the RTCLPC bit of the subsystem clock supply mode control register (OSMC) to 1.

**Caution 3.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

### 10.3.2 Subsystem clock supply mode control register (OSMC)

The WUTMMCK0 bit can be used to select the real-time clock count clock ( $f_{RTC}$ ).

In addition, by stopping clock functions that are unnecessary, the RTCLPC bit can be used to reduce power consumption. For details about setting the RTCLPC bit, see **CHAPTER 5 CLOCK GENERATOR**.

The OSMC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 3 Format of Subsystem clock supply mode control register (OSMC)**

Address: F00F3H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

WUTMMCK0	Selection of count clock ( $f_{RTC}$ ) for real-time clock, 12-bit interval timer, and timer RJ operation clock
0	Subsystem clock ( $f_{SUB}$ ) <ul style="list-style-type: none"> <li>The subsystem clock is selected as the count clock for the real-time clock and the 12-bit interval timer.</li> <li>The low-speed on-chip oscillator cannot be selected as the count source for timer RJ.</li> </ul>
1	Low-speed on-chip oscillator clock ( $f_{IL}$ ) <ul style="list-style-type: none"> <li>The low-speed on-chip oscillator clock is selected as the count clock for the real-time clock and the 12-bit interval timer.</li> <li>Either the low-speed on-chip oscillator or the subsystem clock can be selected as the count source for timer RJ.</li> </ul>

**Caution** The count of year, month, week, day, hour, minutes and second can only be performed when a subsystem clock ( $f_{SUB} = 32.768\text{ kHz}$ ) is selected as the operation clock of the real-time clock. When the low-speed oscillation clock ( $f_{IL} = 15\text{ kHz}$ ) is selected, only the constant-period interrupt function is available. The 30-, 32-, and 36-pin products have the constant-period interrupt function only, because these products have no subsystem clock.  
 However, the constant-period interrupt interval when  $f_{IL}$  is selected will be calculated with the constant-period (the value selected with RTCC0 register)  $\times f_{SUB}/f_{IL}$ .

### 10.3.3 Real-time clock control register 0 (RTCC0)

The RTCC0 register is an 8-bit register that is used to start or stop the real-time clock operation, control the RTC1HZ pin, and set a 12- or 24-hour system and the constant-period interrupt function.

The RTCC0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 4 Format of Real-time clock control register 0 (RTCC0)**

Address: FFF9DH      After reset: 00H      R/W

Symbol	<7>	6	<5>	4	3	2	1	0
--------	-----	---	-----	---	---	---	---	---

RTCC0	RTCE	0	RCLOE1 Note	0	AMPM	CT2	CT1	CT0
-------	------	---	-------------	---	------	-----	-----	-----

RTCE	Real-time clock operation control
0	Stops counter operation.
1	Starts counter operation.

RCLOE1 Note	RTC1HZ pin output control
0	Disables output of the RTC1HZ pin (1 Hz).
1	Enables output of the RTC1HZ pin (1 Hz).

AMPM	Selection of 12-/24-hour system
0	12-hour system (a.m. and p.m. are displayed.)
1	24-hour system

- Rewrite the AMPM bit value after setting the RWAIT bit (bit 0 of real-time clock control register 1 (RTCC1)) to 1. If the AMPM bit value is changed, the values of the hour count register (HOUR) change according to the specified time system.
- Table 10 - 2 shows the Displayed Time Digits.

CT2	CT1	CT0	Constant-period interrupt (INTRTC) selection
0	0	0	Does not use fixed-cycle interrupt function.
0	0	1	Once per 0.5 s (synchronized with second count up)
0	1	0	Once per 1 s (same time as second count up)
0	1	1	Once per 1 m (second 00 of every minute)
1	0	0	Once per 1 hour (minute 00 and second 00 of every hour)
1	0	1	Once per 1 day (hour 00, minute 00, and second 00 of every day)
1	1	x	Once per 1 month (Day 1, hour 00 a.m., minute 00, and second 00 of every month)

When changing the values of the CT2 to CT0 bits while the counter operates (RTCE = 1), rewrite the values of the CT2 to CT0 bits after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, after rewriting the values of the CT2 to CT0 bits, enable interrupt servicing after clearing the RIFG and RTCIF flags.

**Note** Set the RCLOE1 bit to 0 in 30-, 32-, and 36-pin products.

**Caution 1. Do not change the value of the RTCLOE1 bit when RTCE = 1.**

**Caution 2. 1 Hz is not output even if RCCOE1 is set to 1 when RTCE = 0.**

**Remark** x: Don't care

### 10.3.4 Real-time clock control register 1 (RTCC1)

The RTCC1 register is an 8-bit register that is used to control the alarm interrupt function and the wait time of the counter.

The RTCC1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 5 Format of Real-time clock control register 1 (RTCC1) (1/2)**

Address: FFF9EH      After reset: 00H      R/W

Symbol	<7>	<6>	5	<4>	<3>	2	<1>	<0>
--------	-----	-----	---	-----	-----	---	-----	-----

RTCC1	WALE	WALIE	0	WAFC	RIFG	0	RWST	RWAIT
-------	------	-------	---	------	------	---	------	-------

WALE	Alarm operation control
0	Match operation is invalid.
1	Match operation is valid.

When setting a value to the WALE bit while the counter operates (RTCE = 1) and WALIE = 1, rewrite the WALE bit after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, clear the WAFC and RTCIF flags after rewriting the WALE bit. When setting each alarm register (WALIE flag of real-time clock control register 1 (RTCC1), the alarm minute register (ALARMWM), the alarm hour register (ALARMWH), and the alarm week register (ALARMWW)), set match operation to be invalid ("0") for the WALE bit.

WALIE	Control of alarm interrupt (INTRTC) function operation
0	Does not generate interrupt on matching of alarm.
1	Generates interrupt on matching of alarm.

WAFC	Alarm detection status flag
0	Alarm mismatch
1	Detection of matching of alarm

This is a status flag that indicates detection of matching with the alarm. It is valid only when WALE = 1 and is set to "1" one cycle of fRTC after matching of the alarm is detected.

This flag is cleared when "0" is written to it. Writing "1" to it is invalid.

**Figure 10 - 6 Format of Real-time clock control register 1 (RTCC1) (2/2)**

RIFG	Constant-period interrupt status flag
0	Fixed-cycle interrupt is not generated.
1	Fixed-cycle interrupt is generated.
This flag indicates the status of generation of the fixed-cycle interrupt. When the fixed-cycle interrupt is generated, it is set to "1".	
This flag is cleared when "0" is written to it. Writing "1" to it is invalid.	
RWST	Wait status flag of real-time clock
0	Counter is operating.
1	Mode to read or write counter value
This status flag indicates whether the setting of the RWAIT bit is valid.	
Before reading or writing the counter value, confirm that the value of this flag is 1.	
RWAIT	Wait control of real-time clock
0	Sets counter operation.
1	Stops SEC to YEAR counters. Mode to read or write counter value
This bit controls the operation of the counter.	
Be sure to write "1" to it to read or write the counter value.	
As the internal counter (16-bit) is continuing to run, complete reading or writing within one second and turn back to 0.	
When RWAIT = 1, it takes up to one cycle of fRTC until the counter value can be read or written (RWST = 1).	
When the internal counter (16-bit) overflowed while RWAIT = 1, it keeps the event of overflow until RWAIT = 0, then counts up.	
However, when it wrote a value to second count register, it will not keep the overflow event.	

**Caution** If writing is performed to the RTCC1 register with a 1-bit manipulation instruction, the RIFG flag and WAFG flag may be cleared. Therefore, to perform writing to the RTCC1 register, be sure to use an 8-bit manipulation instruction. To prevent the RIFG flag and WAFG flag from being cleared during writing, disable writing by setting 1 to the corresponding bit. If the RIFG flag and WAFG flag are not used and the value may be changed, the RTCC1 register may be written by using a 1-bit manipulation instruction.

**Remark 1.** Fixed-cycle interrupts and alarm match interrupts use the same interrupt source (INTRTC). When using these two types of interrupts at the same time, which interrupt occurred can be judged by checking the fixed-cycle interrupt status flag (RIFG) and the alarm detection status flag (WAFG) upon INTRTC occurrence.

**Remark 2.** The internal counter (16 bits) is cleared when the second count register (SEC) is written.

### 10.3.5 Second count register (SEC)

The SEC register is an 8-bit register that takes a value of 0 to 59 (decimal) and indicates the count value of seconds.

It counts up when the internal counter (16-bit) overflows.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Set a decimal value of 00 to 59 to this register in BCD code.

The SEC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 7 Format of Second count register (SEC)**

Address: FFF92H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
SEC	0	SEC40	SEC20	SEC10	SEC8	SEC4	SEC2	SEC1

**Caution** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

**Remark** The internal counter (16 bits) is cleared when the second count register (SEC) is written.

### 10.3.6 Minute count register (MIN)

The MIN register is an 8-bit register that takes a value of 0 to 59 (decimal) and indicates the count value of minutes.

It counts up when the second counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Even if the second count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 00 to 59 to this register in BCD code.

The MIN register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 8 Format of Minute count register (MIN)**

Address: FFF93H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
MIN	0	MIN40	MIN20	MIN10	MIN8	MIN4	MIN2	MIN1

**Caution** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

### 10.3.7 Hour count register (HOUR)

The HOUR register is an 8-bit register that takes a value of 00 to 23 or 01 to 12 and 21 to 32 (decimal) and indicates the count value of hours.

It counts up when the minute counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Even if the minute count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Specify a decimal value of 00 to 23, 01 to 12, or 21 to 32 by using BCD code according to the time system specified using bit 3 (AMPM) of real-time clock control register 0 (RTCC0).

If the AMPM bit value is changed, the values of the HOUR register change according to the specified time system.

The HOUR register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 12H.

However, the value of this register is 00H if the AMPM bit (bit 3 of the RTCC0 register) is set to 1 after reset.

**Figure 10 - 9 Format of Hour count register (HOUR)**

Address: FFF94H      After reset: 12H      R/W

Symbol	7	6	5	4	3	2	1	0
HOUR	0	0	HOUR20	HOUR10	HOUR8	HOUR4	HOUR2	HOUR1

**Caution 1.** Bit 5 (HOUR20) of the HOUR register indicates AM(0)/PM(1) if AMPM = 0 (if the 12-hour system is selected).

**Caution 2.** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

Table 10 - 2 shows the relationship between the setting value of the AMPM bit, the hour count register (HOUR) value, and time.

**Table 10 - 2 Displayed Time Digits**

24-Hour Display (AMPM = 1)		12-Hour Display (AMPM = 1)	
Time	HOUR Register	Time	HOUR Register
0	00 H	12 a.m.	12 H
1	01 H	1 a.m.	01 H
2	02 H	2 a.m.	02 H
3	03 H	3 a.m.	03 H
4	04 H	4 a.m.	04 H
5	05 H	5 a.m.	05 H
6	06 H	6 a.m.	06 H
7	07 H	7 a.m.	07 H
8	08 H	8 a.m.	08 H
9	09 H	9 a.m.	09 H
10	10 H	10 a.m.	10 H
11	11 H	11 a.m.	11 H
12	12 H	12 p.m.	32 H
13	13 H	1 p.m.	21 H
14	14 H	2 p.m.	22 H
15	15 H	3 p.m.	23 H
16	16 H	4 p.m.	24 H
17	17 H	5 p.m.	25 H
18	18 H	6 p.m.	26 H
19	19 H	7 p.m.	27 H
20	20 H	8 p.m.	28 H
21	21 H	9 p.m.	29 H
22	22 H	10 p.m.	30 H
23	23 H	11 p.m.	31 H

The HOUR register value is set to 12-hour display when the AMPM bit is “0” and to 24-hour display when the AMPM bit is “1”.

In 12-hour display, the fifth bit of the HOUR register displays 0 for AM and 1 for PM.

### 10.3.8 Day count register (DAY)

The DAY register is an 8-bit register that takes a value of 1 to 31 (decimal) and indicates the count value of days.

It counts up when the hour counter overflows.

This counter counts as follows.

- 01 to 31 (January, March, May, July, August, October, December)
- 01 to 30 (April, June, September, November)
- 01 to 29 (February, leap year)
- 01 to 28 (February, normal year)

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Even if the hour count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 01 to 31 to this register in BCD code.

The DAY register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 01H.

**Figure 10 - 10 Format of Day count register (DAY)**

Address: FFF96H      After reset: 01H      R/W

Symbol	7	6	5	4	3	2	1	0
DAY	0	0	DAY20	DAY10	DAY8	DAY4	DAY2	DAY1

**Caution** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

### 10.3.9 Week count register (WEEK)

The WEEK register is an 8-bit register that takes a value of 0 to 6 (decimal) and indicates the count value of weekdays.

It counts up in synchronization with the day counter.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Set a decimal value of 00 to 06 to this register in BCD code.

The WEEK register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 11 Format of Week count register (WEEK)**

Address: FFF95H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
WEEK	0	0	0	0	0	WEEK4	WEEK2	WEEK1

**Caution 1.** The value corresponding to the month count register (MONTH) or the day count register (DAY) is not stored in the week count register (WEEK) automatically. After reset release, set the week count register as follow.

Day	WEEK
Sunday	00 H
Monday	01 H
Tuesday	02 H
Wednesday	03 H
Thursday	04 H
Friday	05 H
Saturday	06 H

**Caution 2.** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

### 10.3.10 Month count register (MONTH)

The MONTH register is an 8-bit register that takes a value of 1 to 12 (decimal) and indicates the count value of months.

It counts up when the day counter overflows.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Even if the day count register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 01 to 12 to this register in BCD code.

The MONTH register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 01H.

**Figure 10 - 12 Format of Month count register (MONTH)**

Address: FFF97H      After reset: 01H      R/W

Symbol	7	6	5	4	3	2	1	0
MONTH	0	0	0	MONTH10	MONTH8	MONTH4	MONTH2	MONTH1

**Caution** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

### 10.3.11 Year count register (YEAR)

The YEAR register is an 8-bit register that takes a value of 0 to 99 (decimal) and indicates the count value of years.

It counts up when the month count register (MONTH) overflows.

Values 00, 04, 08, ..., 92, and 96 indicate a leap year.

When data is written to this register, it is written to a buffer and then to the counter up to two cycles of fRTC later.

Even if the MONTH register overflows while this register is being written, this register ignores the overflow and is set to the value written. Set a decimal value of 00 to 99 to this register in BCD code.

The YEAR register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 13 Format of Year count register (YEAR)**

Address: FFF98H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
YEAR	YEAR80	YEAR40	YEAR20	YEAR10	YEAR8	YEAR4	YEAR2	YEAR1

**Caution** When it reads or writes from/to the register while the counter is in operation (RTCE = 1), follow the procedures described in 10.4.3 Reading/writing real-time clock.

### 10.3.12 Watch error correction register (SUBCUD)

This register is used to correct the watch with high accuracy when it is slow or fast by changing the value that overflows from the internal counter (16-bit) to the second count register (SEC) (reference value: 7FFFH).

The SUBCUD register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 10 - 14 Format of Watch error correction register (SUBCUD)**

Address: FFF99H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

SUBCUD	DEV	F6	F5	F4	F3	F2	F1	F0
--------	-----	----	----	----	----	----	----	----

DEV	Setting of watch error correction timing
0	Corrects watch error when the second digits are at 00, 20, or 40 (every 20 seconds).
1	Corrects watch error only when the second digits are at 00 (every 60 seconds).
Writing to the SUBCUD register at the following timing is prohibited.	
<ul style="list-style-type: none"> <li>When DEV = 0 is set: For a period of SEC = 00H, 20H, 40H</li> <li>When DEV = 1 is set: For a period of SEC = 00H</li> </ul>	

F6	Setting of watch error correction value
0	Increases by $\{(F5, F4, F3, F2, F1, F0) - 1\} \times 2$ .
1	Decreases by $\{(/F5, /F4, /F3, /F2, /F1, /F0) + 1\} \times 2$ .
When $(F6, F5, F4, F3, F2, F1, F0) = (*, 0, 0, 0, 0, 0, *)$ , the watch error is not corrected. * is 0 or 1.	
$/F5$ to $/F0$ are the inverted values of the corresponding bits (000011 when 111100).	
Range of correction value: (when F6 = 0) 2, 4, 6, 8, ..., 120, 122, 124 (when F6 = 1) -2, -4, -6, -8, ..., -120, -122, -124	

The range of value that can be corrected by using the watch error correction register (SUBCUD) is shown below.

	DEV = 0 (correction every 20 seconds)	DEV = 1 (correction every 60 seconds)
Correctable range	-189.2 ppm to 189.2 ppm	-63.1 ppm to 63.1 ppm
Maximum excludes quantization error	$\pm 1.53$ ppm	$\pm 0.51$ ppm
Minimum resolution	$\pm 3.05$ ppm	$\pm 1.02$ ppm

**Remark** If a correctable range is -63.1 ppm or lower and 63.1 ppm or higher, set 0 to DEV.

### 10.3.13 Alarm minute register (ALARMWM)

This register is used to set minutes of alarm.

The ALARMWM register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Caution** Set a decimal value of 00 to 59 to this register in BCD code. If a value outside the range is set, the alarm is not detected.

Figure 10 - 15 Format of Alarm minute register (ALARMWM)

Address: FFF9AH After reset: 00H R/W								
Symbol	7	6	5	4	3	2	1	0
ALARMWM	0	WM40	WM20	WM10	WM8	WM4	WM2	WM1

### 10.3.14 Alarm hour register (ALARMWH)

This register is used to set hours of alarm.

The ALARMWH register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 12H.

However, the value of this register is 00H if the AMPM bit is set to 1 after reset.

**Caution** Set a decimal value of 00 to 23, 01 to 12, or 21 to 32 to this register in BCD code. If a value outside the range is set, the alarm is not detected.

Figure 10 - 16 Format of Alarm hour register (ALARMWH)

Address: FFF9BH After reset: 12H R/W								
Symbol	7	6	5	4	3	2	1	0
ALARMWH	0	0	WH20	WH10	WH8	WH4	WH2	WH1

**Caution** Bit 5 (WH20) of the ALARMWH register indicates AM(0)/PM(1) if AMPM = 0 (if the 12-hour system is selected).

### 10.3.15 Alarm week register (ALARMWW)

This register is used to set date of alarm.

The ALARMWW register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

Figure 10 - 17 Format of Alarm week register (ALARMWW)

Address: FFF9CH After reset: 00H R/W								
Symbol	7	6	5	4	3	2	1	0
ALARMWW	0	WW6	WW5	WW4	WW3	WW2	WW1	WW0

Here is an example of setting the alarm.

Time of Alarm	Day							12-Hour Display				24-Hour Display			
	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Hour	Hour	Minute	Minute	Hour	Hour	Minute	Minute
	W	W	W	W	W	W	W	10	1	10	1	10	1	10	1
0	1	2	3	4	5	6									
Every day, 0:00 a.m.	1	1	1	1	1	1	1	1	2	0	0	0	0	0	0
Every day, 1:30 a.m.	1	1	1	1	1	1	1	0	1	3	0	0	1	3	0
Every day, 11:59 a.m.	1	1	1	1	1	1	1	1	1	5	9	1	1	5	9
Monday through Friday, 0:00 p.m.	0	1	1	1	1	1	0	3	2	0	0	1	2	0	0
Sunday, 1:30 p.m.	1	0	0	0	0	0	0	2	1	3	0	1	3	3	0
Monday, Wednesday, Friday, 11:59 p.m.	0	1	0	1	0	1	0	3	1	5	9	2	3	5	9

### 10.3.16 Port mode register 3 (PM3)

The PM3 register can be set by a 1-bit or 8-bit manipulation instruction.

Reset signal generation sets this register to FFH.

When using the port 3 as the RTC1HZ pin for output of 1 Hz, set the PM30 bit to 0.

Figure 10 - 18 Format of Port mode register 3 (PM3)

Address: FFF23H      After reset: FFH      R/W

Symbol	7	6	5	4	3	2	1	0
PM3	1	1	1	1	1	1	PM31	PM30

### 10.3.17 Port register 3 (P3)

The P3 register can be set by a 1-bit or 8-bit manipulation instruction.

Reset signal generation sets this register to 00H.

When using the port 3 as 1 Hz output to the RTC1Hz pin, set the P30 bit to 0.

Figure 10 - 19 Format of Port register 3 (P3)

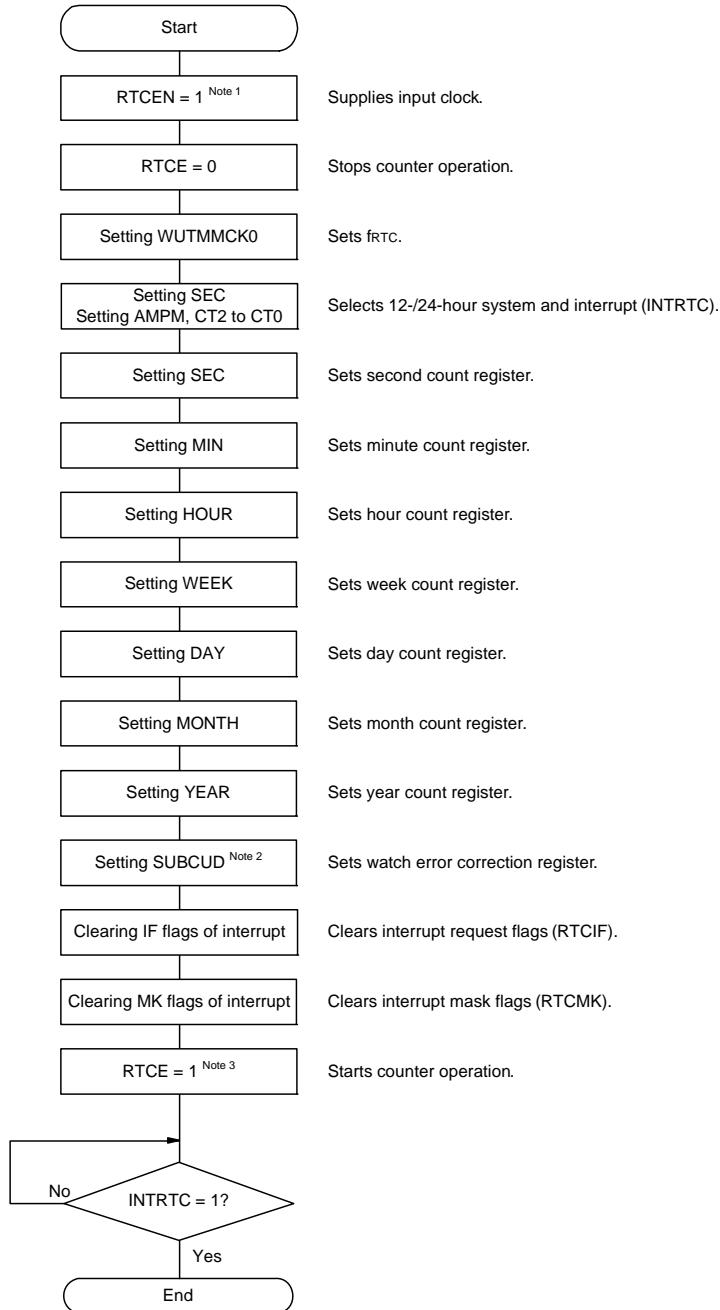
Address: FFF03H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
P3	0	0	0	0	0	0	P31	P30

## 10.4 Real-time Clock Operation

### 10.4.1 Starting operation of real-time clock

Figure 10 - 20 Procedure for Starting Operation of Real-time Clock



**Note 1.** First set the RTCEN bit to 1, while oscillation of the count clock (fRTC) is stable.

**Note 2.** Set up the SUBCUD register only if the watch error must be corrected. For details about how to calculate the correction value, see **10.4.6 Example of watch error correction of real-time clock**.

**Note 3.** Confirm the procedure described in **10.4.2 Shifting to HALT/STOP mode after starting operation** when shifting to HALT/STOP mode without waiting for INTRTC = 1 after RTCE = 1.

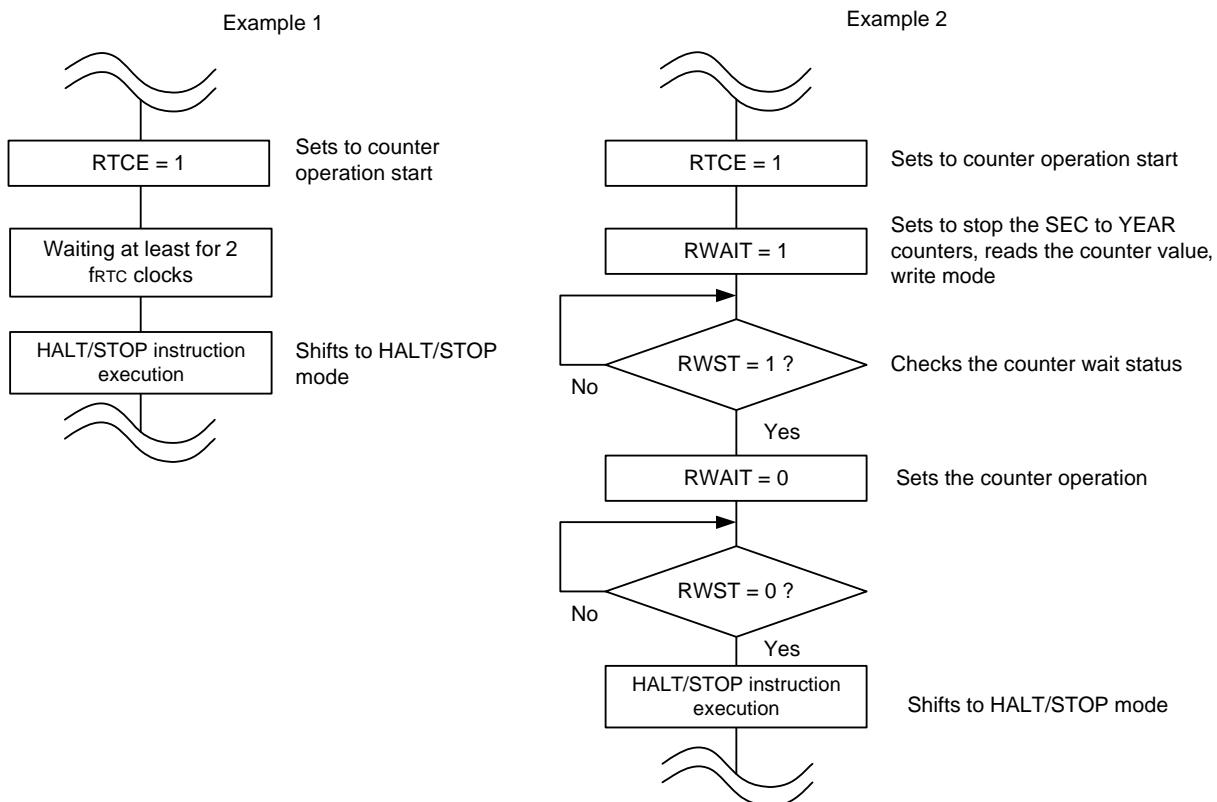
### 10.4.2 Shifting to HALT/STOP mode after starting operation

Perform one of the following processing when shifting to HALT/STOP mode immediately after setting the RTCE bit to 1.

However, after setting the RTCE bit to 1, this processing is not required when shifting to HALT/STOP mode after INTRTC interrupt has occurred.

- Shifting to HALT/STOP mode when at least two count clock (fRTC) have elapsed after setting the RTCE bit to 1 (see **Figure 10 - 21, Example 1**).
- Checking by polling the RWST bit to become 1, after setting the RTCE bit to 1 and then setting the RWAIT bit to 1. Afterward, setting the RWAIT bit to 0 and shifting to HALT/STOP mode after checking again by polling that the RWST bit has become 0 (see **Figure 10 - 21, Example 2**).

**Figure 10 - 21 Procedure for Shifting to HALT/STOP Mode After Setting RTCE bit to 1**

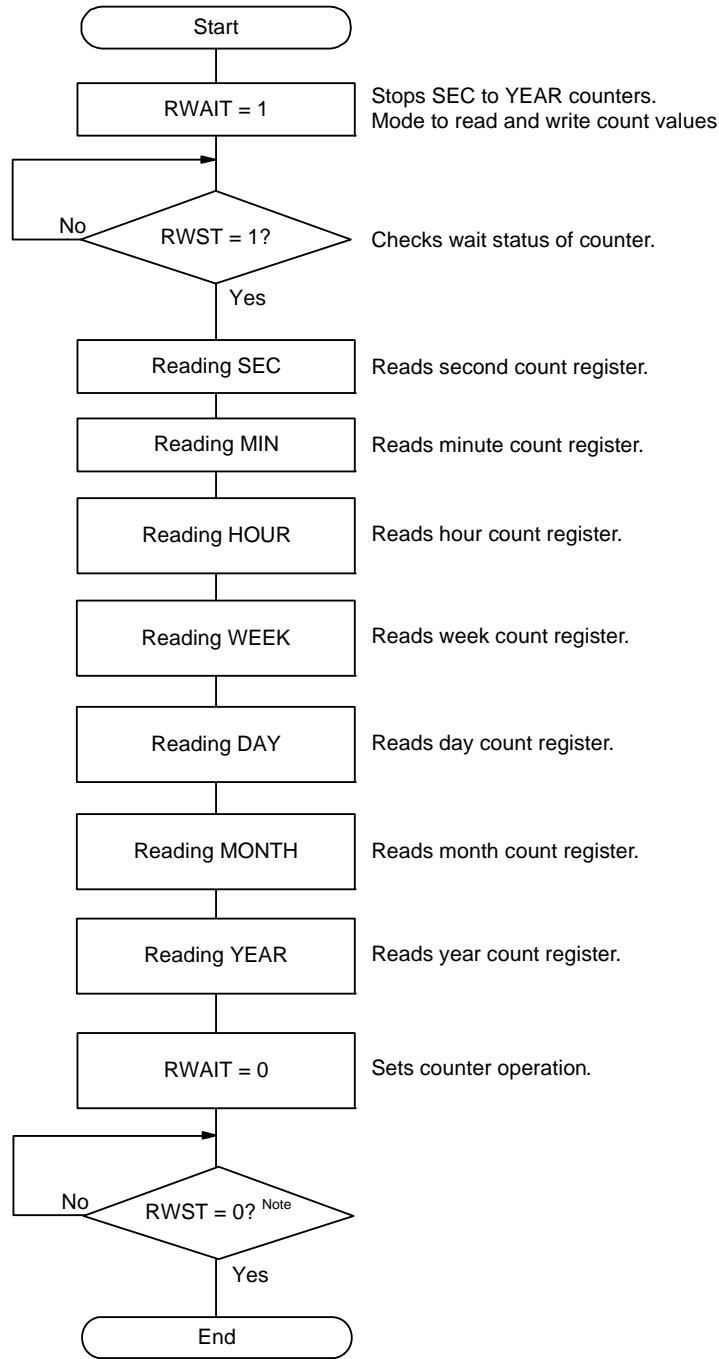


### 10.4.3 Reading/writing real-time clock

Read or write the counter after setting 1 to RWAIT first.

Set RWAIT to 0 after completion of reading or writing the counter.

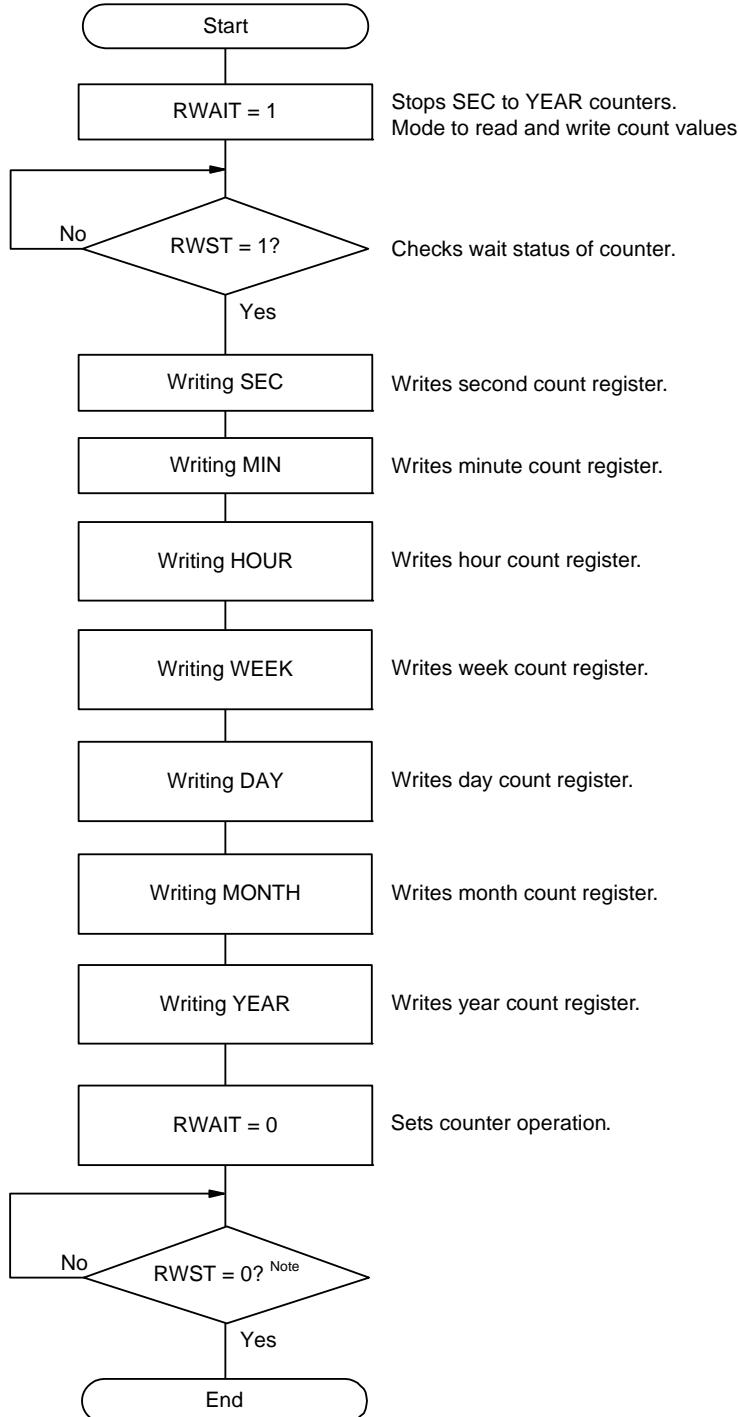
**Figure 10 - 22 Procedure for Reading Real-time Clock**



**Note** Be sure to confirm that RWST = 0 before setting HALT/STOP mode.

**Caution** Complete the series of process of setting the RWAIT bit to 1 to clearing the RWAIT bit to 0 within 1 second.

**Remark** The second count register (SEC), minute count register (MIN), hour count register (HOUR), week count register (WEEK), day count register (DAY), month count register (MONTH), and year count register (YEAR) may be read in any sequence. All the registers do not have to read and only some registers may be read.

**Figure 10 - 23 Procedure for Writing Real-time Clock**

**Note** Be sure to confirm that RWST = 0 before setting HALT/STOP mode.

**Caution 1.** Complete the series of operations of setting the RWAIT bit to 1 to clearing the RWAIT bit to 0 within 1 second.

**Caution 2.** When changing the values of the SEC, MIN, HOUR, WEEK, DAY, MONTH, and YEAR register while the counter operates (RTCE = 1), rewrite the values of the MIN register after disabling interrupt servicing INTRTC by using the interrupt mask flag register. Furthermore, clear the WAFG, RIFG and RTCIF flags after rewriting the MIN register.

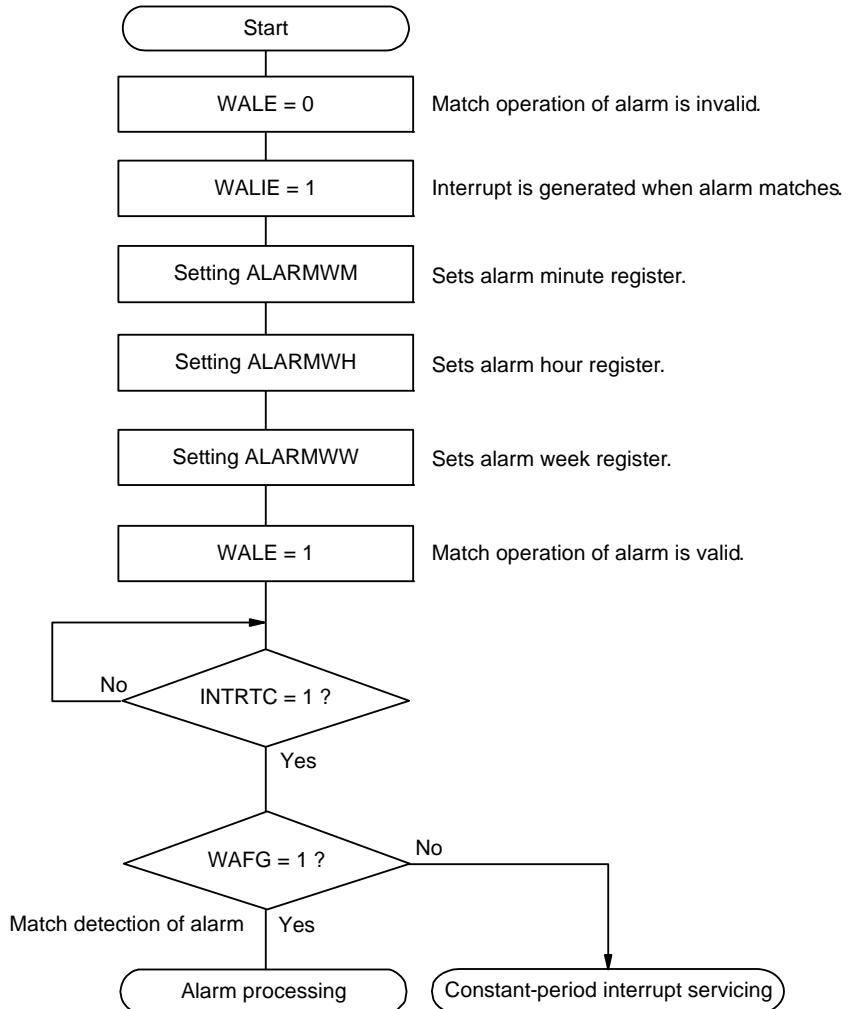
**Remark** The second count register (SEC), minute count register (MIN), hour count register (HOUR), week count register (WEEK), day count register (DAY), month count register (MONTH), and year count register (YEAR) may be written in any sequence.

All the registers do not have to be set and only some registers may be written.

#### 10.4.4 Setting alarm of real-time clock

Set time of alarm after setting 0 to WALE (alarm operation invalid) first.

Figure 10 - 24 Alarm Setting Procedure

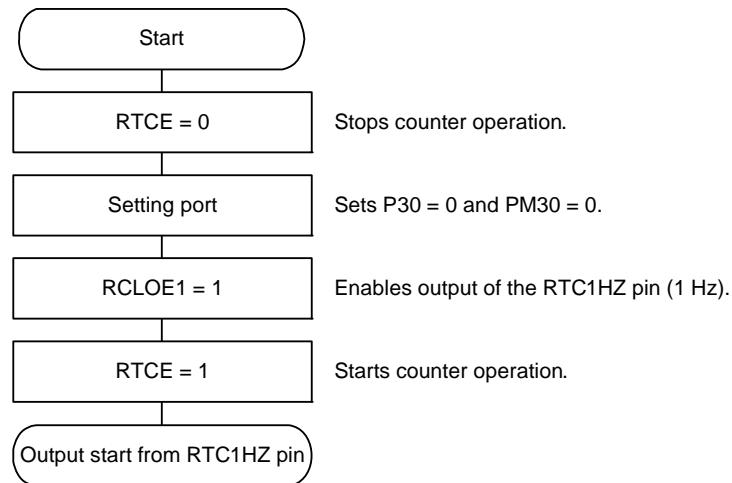


**Remark 1.** The alarm week register (ALARMWW), alarm hour register (ALARMWH), and alarm week register (ALARMWW) may be written in any sequence.

**Remark 2.** Fixed-cycle interrupts and alarm match interrupts use the same interrupt source (INTRTC). When using these two types of interrupts at the same time, which interrupt occurred can be judged by checking the fixed-cycle interrupt status flag (RIFG) and the alarm detection status flag (WAFG) upon INTRTC occurrence.

### 10.4.5 1 Hz output of real-time clock

Figure 10 - 25 1 Hz Output Setting Procedure



**Caution 1.** First set the RTCEN bit to 1, while oscillation of the count clock (fSUB) is stable.

**Caution 2.** The 1 Hz output function of the real-time clock is not provided in 30-, 32-, and 36-pin products.

### 10.4.6 Example of watch error correction of real-time clock

The watch can be corrected with high accuracy when it is slow or fast, by setting a value to the watch error correction register.

**Example of calculating the correction value**

The correction value used when correcting the count value of the internal counter (16-bit) is calculated by using the following expression.

Set the DEV bit to 0 when the correction range is -63.1 ppm or less, or 63.1 ppm or more.

(When DEV = 0)

$$\text{Correction value}^{\text{Note}} = \frac{\text{Number of correction counts in 1 minute}}{3} = (\text{Oscillation frequency} \div \text{Target frequency} - 1) \times 32768 \times 60 \div 3$$

(When DEV = 1)

$$\text{Correction value}^{\text{Note}} = \text{Number of correction counts in 1 minute} = (\text{Oscillation frequency} \div \text{Target frequency} - 1) \times 32768 \times 60$$

**Note** The correction value is the watch error correction value calculated by using bits 6 to 0 of the watch error correction register (SUBCUD).

(When F6 = 0) Correction value =  $\{(F_5, F_4, F_3, F_2, F_1, F_0) - 1\} \times 2$

(When F6 = 1) Correction value =  $-(\{/F_5, /F_4, /F_3, /F_2, /F_1, /F_0\} + 1) \times 2$

When  $(F_6, F_5, F_4, F_3, F_2, F_1, F_0)$  is (\*, 0, 0, 0, 0, 0, \*), watch error correction is not performed. “\*” is 0 or 1.

$/F_5$  to  $/F_0$  are bit-inverted values (000011 when 111100).

**Remark 1.** The correction value is 2, 4, 6, 8, ... 120, 122, 124 or -2, -4, -6, -8, ... -120, -122, -124.

**Remark 2.** The oscillation frequency is the count clock ( $f_{RTC}$ ).

It can be calculated from the output frequency of the RTC1HZ pin  $\times 32768$  when the watch error correction register is set to its initial value (00H).

**Remark 3.** The target frequency is the frequency resulting after correction performed by using the watch error correction register.

**Correction example 1**

Example of correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz - 131.2 ppm)

[Measuring the oscillation frequency]

The oscillation frequency Note of each product is measured by outputting about 32.768 kHz from the PCLBUZ0 pin, or by outputting about 1 Hz from the RTC1HZ pin when the watch error correction register (SUBCUD) is set to its initial value (00H).

**Note** See **10.4.5 1 Hz output of real-time clock** for the setting procedure of the RTC1Hz output, and see **12.4 Operations of Clock Output/Buzzer Output Controller** for the setting procedure of outputting about 32 kHz from the PCLBUZ0 pin.

[Calculating the correction value]

(When the output frequency from the PCLBUZ0 pin is 32772.3 Hz)

Assume the target frequency to be 32768 Hz (32772.3 Hz - 131.2 ppm) and DEV to be 0, because the correctable range of -131.2 ppm is -63.1 ppm or lower.

The expression for calculating the correction value when DEV is 0 is applied.

$$\begin{aligned}\text{Correction value} &= \text{Number of correction counts in 1 minute} \div 3 \\ &= (\text{Oscillation frequency} \div \text{target frequency} - 1) \times 32768 \times 60 \div 3 \\ &= (32772.3 \div 32768 - 1) \times 32768 \times 60 \div 3 \\ &= 86\end{aligned}$$

[Calculating the values to be set to (F6 to F0)]

(When the correction value is 86)

If the correction value is 0 or larger (when slowing), assume F6 to be 0.

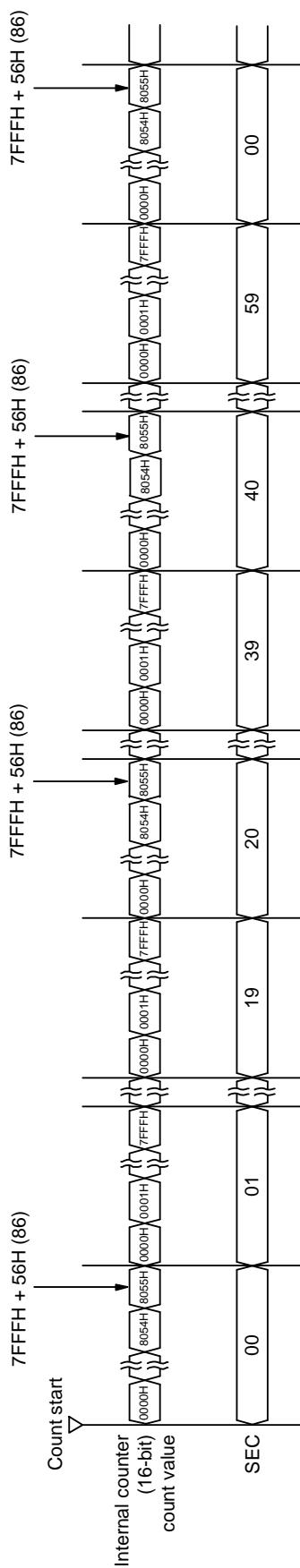
Calculate (F5, F4, F3, F2, F1, F0) from the correction value.

$$\begin{array}{ll}\{(F5, F4, F3, F2, F1, F0) - 1\} \times 2 & = 86 \\ (F5, F4, F3, F2, F1, F0) & = 44 \\ (F5, F4, F3, F2, F1, F0) & = (1, 0, 1, 1, 0, 0)\end{array}$$

Consequently, when correcting from 32772.3 Hz to 32768 Hz (32772.3 Hz - 131.2 ppm), setting the correction register such that DEV is 0 and the correction value is 86 (bits 6 to 0 of the SUBCUD register: 0101100) results in 32768 Hz (0 ppm).

Figure 10 - 26 shows the operation when (DEV, F6, F5, F4, F3, F2, F1, F0) is (0, 0, 1, 0, 1, 1, 0, 0).

**Figure 10 - 26 Correction Operation when  $(DEV, F6, F5, F4, F3, F2, F1, F0) = (0, 0, 1, 0, 1, 1, 0, 0)$**



**Correction example 2**

Example of correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm)

[Measuring the oscillation frequency]

The oscillation frequency Note of each product is measured by outputting about 1 Hz from the RTC1HZ pin when the watch error correction register (SUBCUD) is set to its initial value (00H).

**Note** See **10.4.5 1 Hz output of real-time clock** for the setting procedure of outputting about 1 Hz from the RTC1HZ pin.

[Calculating the correction value]

(When the output frequency from the RTCCL pin is 0.9999817 Hz)

$$\text{Oscillation frequency} = 32768 \times 0.9999817 \approx 32767.4 \text{ Hz}$$

Assume the target frequency to be 32768 Hz (32767.4 Hz + 18.3 ppm) and DEV to be 1.

The expression for calculating the correction value when DEV is 1 is applied.

$$\begin{aligned}\text{Correction value} &= \text{Number of correction counts in 1 minute} \\ &= (\text{Oscillation frequency} \div \text{Target frequency} - 1) \times 32768 \times 60 \\ &= (32767.4 \div 32768 - 1) \times 32768 \times 60 \\ &= -36\end{aligned}$$

[Calculating the values to be set to (F6 to F0)]

(When the correction value is -36)

If the correction value is 0 or less (when quickening), assume F6 to be 1.

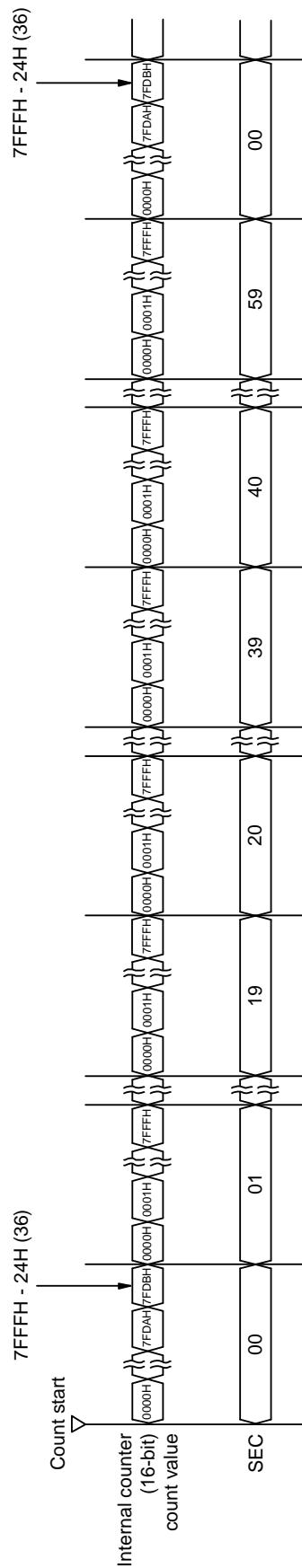
Calculate (F5, F4, F3, F2, F1, F0) from the correction value.

$$\begin{aligned}- \{(/F5, /F4, /F3, /F2, /F1, /F0) + 1\} \times 2 &= -36 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= 17 \\ (/F5, /F4, /F3, /F2, /F1, /F0) &= (0, 1, 0, 0, 0, 1) \\ (F5, F4, F3, F2, F1, F0) &= (1, 0, 1, 1, 1, 0)\end{aligned}$$

Consequently, when correcting from 32767.4 Hz to 32768 Hz (32767.4 Hz + 18.3 ppm), setting the correction register such that DEV is 1 and the correction value is -36 (bits 6 to 0 of the SUBCUD register: 1101110) results in 32768 Hz (0 ppm).

Figure 10 - 27 shows the Operation when (DEV, F6, F5, F4, F3, F2, F1, F0) = (1, 1, 1, 0, 1, 1, 1, 0).

Figure 10 - 27 Operation when  $(DEV, F6, F5, F4, F3, F2, F1, F0) = (1, 1, 1, 0, 1, 1, 1, 0)$



## CHAPTER 11 12-BIT INTERVAL TIMER

### 11.1 Functions of 12-bit Interval Timer

An interrupt (INTIT) is generated at any previously specified time interval. It can be utilized for wakeup from STOP mode and triggering an A/D converter's SNOOZE mode.

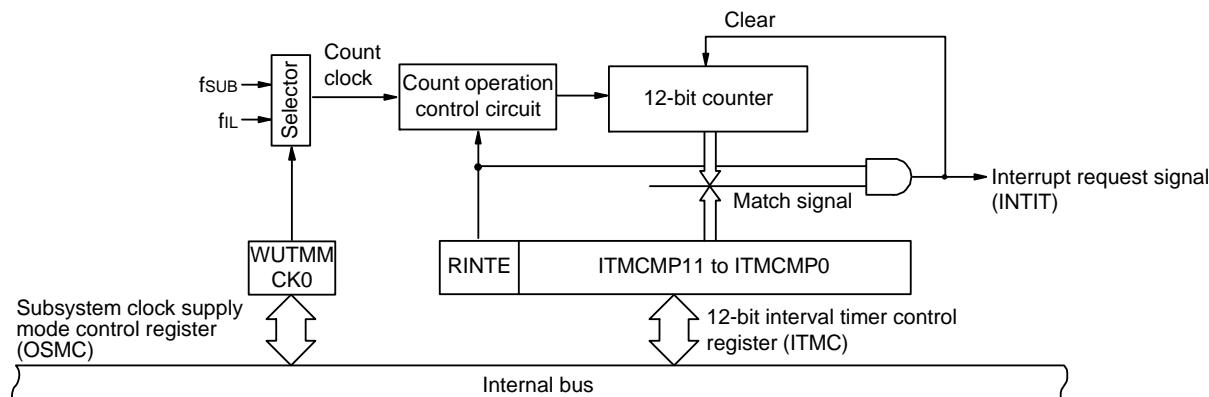
### 11.2 Configuration of 12-bit Interval Timer

The 12-bit interval timer includes the following hardware.

**Table 11 - 1 Configuration of 12-bit Interval Timer**

Item	Configuration
Counter	12-bit counter
Control registers	Peripheral enable register 0 (PER0)
	Subsystem clock supply mode control register (OSMC)
	12-bit interval timer control register (ITMC)

**Figure 11 - 1 Block Diagram of 12-bit Interval Timer**



**Note** The subsystem clock (fSUB) can be selected as the count clock in 40-, 44-, 48-, 52-, 64-, 80-, and 100-pin products.

## 11.3 Registers Controlling 12-bit Interval Timer

The 12-bit interval timer is controlled by the following registers.

- Peripheral enable register 0 (PER0)
- Subsystem clock supply mode control register (OSMC)
- 12-bit interval timer control register (ITMC)

### 11.3.1 Peripheral enable register 0 (PER0)

This register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the 12-bit interval timer is used, be sure to set bit 7 (RTCEN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 11 - 2 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PER0	RTCEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
RTCEN		Control of real-time clock (RTC) and 12-bit interval timer input clock supply						
0		Stops input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer cannot be written. • The real-time clock (RTC) and 12-bit interval timer are in the reset status.						
1		Enables input clock supply. • SFR used by the real-time clock (RTC) and 12-bit interval timer can be read/written.						

**Note**      80 and 100-pin products only.

**Caution 1.** When using the 12-bit interval timer, be sure to first set the RTCEN bit to 1 and then set the following register, while oscillation of the count clock is stable. If RTCEN = 0, writing to the control register controlling the 12-bit interval timer is ignored, and, even if the register is read, only the default value is read (except the subsystem clock supply mode control register (OSMC)).

- 12-bit interval timer control register (ITMC)

**Caution 2.** Clock supply to peripheral functions other than the real-time clock and 12-bit interval timer can be stopped in STOP mode or HALT mode when the subsystem clock is used, by setting the RTCLPC bit of the subsystem clock supply mode control register (OSMC) to 1.

**Caution 3.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

### 11.3.2 Subsystem clock supply mode control register (OSMC)

The WUTMMCK0 bit can be used to select the 12-bit interval timer, real-time clock, and timer RJ operation clock.

In addition, by stopping clock functions that are unnecessary, the RTCLPC bit can be used to reduce power consumption. For details about setting the RTCLPC bit, see **CHAPTER 5 CLOCK GENERATOR**.

The OSMC register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 11 - 3 Format of Subsystem clock supply mode control register (OSMC)**

Address: F00F3H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
OSMC	RTCLPC	0	0	WUTMMCK0	0	0	0	0

WUTMMCK0	Selection of operation clock for real-time clock, 12-bit interval timer, and timer RJ.
0	<p>Subsystem clock (<math>f_{SUB}</math>)</p> <ul style="list-style-type: none"> <li>The subsystem clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>The low-speed on-chip oscillator cannot be selected as the count source for timer RJ.</li> </ul>
1	<p>Low-speed on-chip oscillator clock (<math>f_{IL}</math>)</p> <ul style="list-style-type: none"> <li>The low-speed on-chip oscillator clock is selected as the operation clock for the real-time clock and the 12-bit interval timer.</li> <li>Either the low-speed on-chip oscillator or the subsystem clock can be selected as the count source for timer RJ.</li> </ul>

### 11.3.3 12-bit interval timer control register (ITMC)

This register is used to set up the starting and stopping of the 12-bit interval timer operation and to specify the timer compare value.

The ITMC register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0FFFH.

**Figure 11 - 4 Format of 12-bit interval timer control register (ITMC)**

Address: FFF90H      After reset: 0FFFH      R/W

Symbol      15      14      13      12      11 to 0

ITMC	RINTE	0	0	0	ITCMP11 to IT CMP0					
RINTE	12-bit interval timer operation control									
0	Count operation stopped (count clear)									
1	Count operation started									
ITCMP11 to IT CMP0	Specification of the 12-bit interval timer compare value									
001H	These bits generate a fixed-cycle interrupt (count clock cycles $\times$ (ITCMP setting + 1)).									
•										
•										
FFFH										
000H	Setting prohibit									
Example interrupt cycles when 001H or FFFFH is specified for IT CMP11 to IT CMP0										
<ul style="list-style-type: none"> <li>• IT CMP11 to IT CMP0 = 001H, count clock: when fsub = 32.768 kHz  <math>1/32.768 \text{ [kHz]} \times (1 + 1) = 0.06103515625 \text{ [ms]} \approx 61.03 \text{ [\mu s]}</math></li> <li>• IT CMP11 to IT CMP0 = FFFFH, count clock: when fsub = 32.768 kHz  <math>1/32.768 \text{ [kHz]} \times (4095 + 1) = 125 \text{ [ms]}</math></li> </ul>										

**Caution 1.** Before changing the RINTE bit from 1 to 0, use the interrupt mask flag register to disable the INTIT interrupt servicing. When the operation starts (from 0 to 1) again, clear the TMKAIF flag, and then enable the interrupt servicing.

**Caution 2.** The value read from the RINTE bit is applied one count clock cycle after setting the RINTE bit.

**Caution 3.** When setting the ITMC register after a transition to normal operation or SNOOZE mode from HALT mode or STOP mode, confirm that the value is reflected in the ITMC register or allow at least one cycle of the count clock to elapse before returning the processor to HALT mode or STOP mode.

**Caution 4.** Only change the setting of the IT CMP11 to IT CMP0 bits when RINTE = 0.

However, it is possible to change the settings of the IT CMP11 to IT CMP0 bits at the same time as when changing RINTE from 0 to 1 or 1 to 0.

## 11.4 12-bit Interval Timer Operation

### 11.4.1 12-bit interval timer operation timing

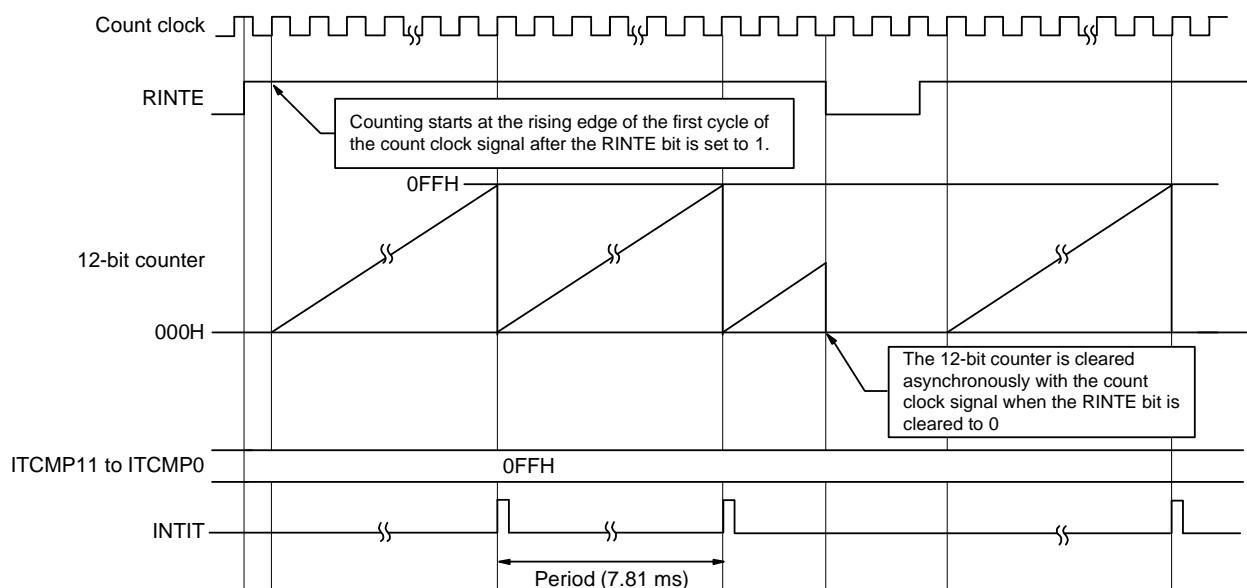
The count value specified for the ITCMP11 to ITCMP0 bits is used as an interval to operate an 12-bit interval timer that repeatedly generates interrupt requests (INTIT).

When the RINTE bit is set to 1, the 12-bit counter starts counting.

When the 12-bit counter value matches the value specified for the ITCMP11 to ITCMP0 bits, the 12-bit counter value is cleared to 0, counting continues, and an interrupt request signal (INTIT) is generated at the same time.

The basic operation of the 12-bit interval timer is as follows.

**Figure 11 - 5 12-bit Interval Timer Operation Timing (ITCMP11 to ITCMP0 = OFFH, count clock: f<sub>SUB</sub> = 32.768 kHz)**



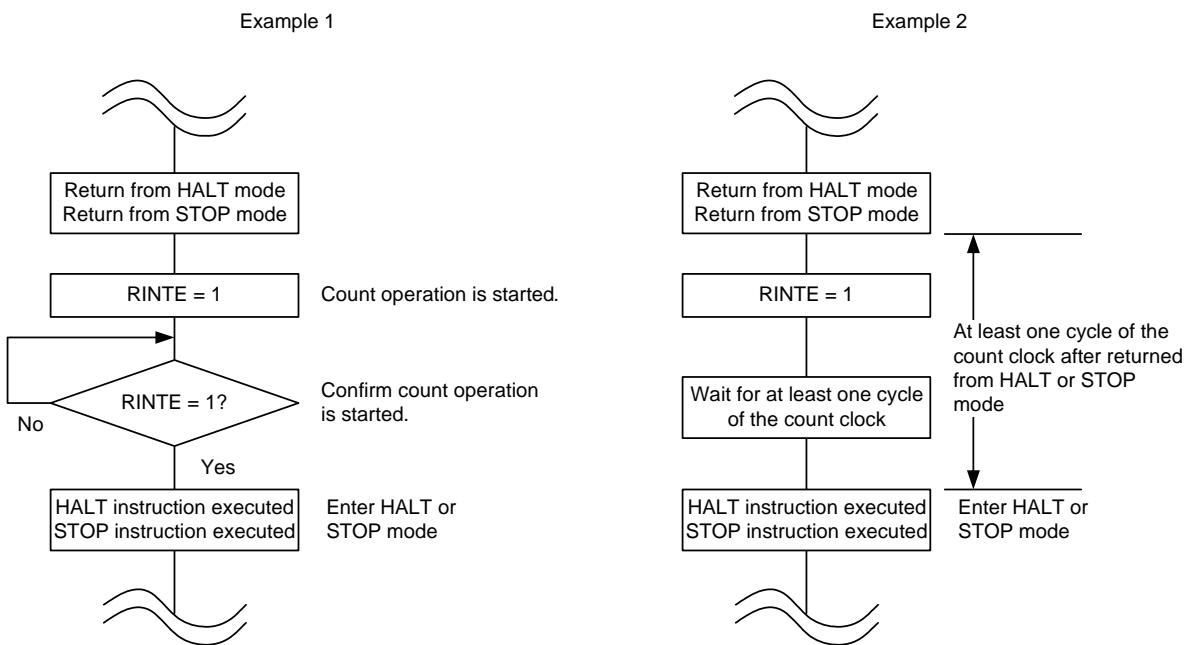
### 11.4.2 Start of count operation and re-enter to HALT/STOP mode after returned from HALT/STOP mode

When setting the RINTE bit after returned from HALT or STOP mode and entering HALT or STOP mode again, write 1 to the RINTE bit, and confirm the written value of the RINTE bit is reflected or wait for at least one cycle of the count clock.

Then, enter HALT or STOP mode.

- After setting RINTE to 1, confirm by polling that the RINTE bit has become 1, and then enter HALT or STOP mode (see **Example 1** in **Figure 11 - 6**).
- After setting RINTE to 1, wait for at least one cycle of the count clock and then enter HALT or STOP mode (see **Example 2** in **Figure 11 - 6**).

**Figure 11 - 6 Procedure of entering to HALT or STOP mode after setting RINTE to 1**



## CHAPTER 12 CLOCK OUTPUT/BUZZER OUTPUT CONTROLLER

**Caution** Most of the following descriptions in this chapter use the 64-pin as an example.

### 12.1 Functions of Clock Output/Buzzer Output Controller

The clock output controller is intended for clock output for supply to peripheral ICs. Buzzer output is a function to output a square wave of buzzer frequency.

One pin can be used to output a clock or buzzer sound.

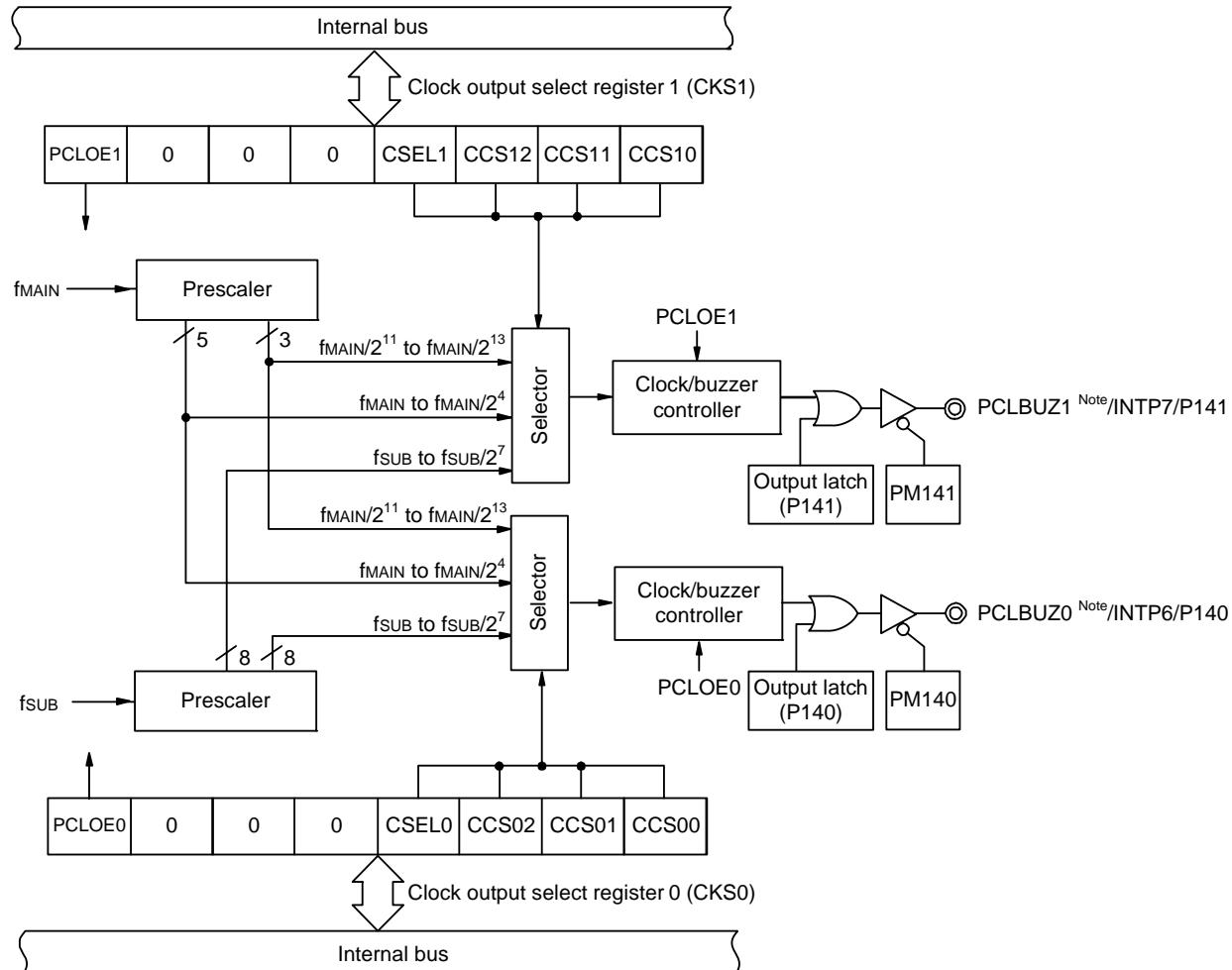
Two output pins, PCLBUZ0 and PCLBUZ1, are available.

The PCLBUZn pin outputs a clock selected by clock output select register n (CKSn).

Figure 12 - 1 shows the Block Diagram of Clock Output/Buzzer Output Controller.

**Caution** In HALT mode when RTCLPC in the subsystem clock supply mode control register (OSMC) = 1 and while the subsystem clock (fsub) is used for CPU operation, it is not possible to output the subsystem clock (fsub) from the PCLBUZn pin.

**Remark** n = 0, 1

**Figure 12 - 1 Block Diagram of Clock Output/Buzzer Output Controller**

**Note** For output frequencies available from PCLBUZ0 and PCLBUZ1, refer to **34.4 or 35.4 AC Characteristics**.

**Remark** The clock output/buzzer output pins in above diagram shows the information of 64- to 100-pins products with PIOR3 = 0 and PIOR4 = 0.

## 12.2 Configuration of Clock Output/Buzzer Output Controller

The clock output/buzzer output controller includes the following hardware.

**Table 12 - 1 Configuration of Clock Output/Buzzer Output Controller**

Item	Configuration
Control registers	Clock output select registers n (CKSn) Port mode registers 1, 3, 5, 14 (PM1, PM3, PM5, PM14) Port registers 1, 3, 5, 14 (P1, P3, P5, P14)

## 12.3 Registers Controlling Clock Output/Buzzer Output Controller

### 12.3.1 Clock output select registers n (CKSn)

These registers set output enable/disable for clock output or for the buzzer frequency output pin (PCLBUZn), and set the output clock.

Select the clock to be output from the PCLBUZn pin by using the CKSn register.

The CKSn register are set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 12 - 2 Format of Clock output select registers n (CKSn)**

Address: FFFA5H (CKS0), FFFA6H (CKS1) After reset: 00H R/W

Symbol	<7>	6	5	4	3	2	1	0
CKSn	PCLOEn	0	0	0	CSELn	CCSn2	CCSn1	CCSn0
PCLOEn	PCLBUZn pin output enable/disable specification							
0	Output disable (default)							
1	Output enable							
PCLBUZn pin output clock selection								
CSELn	CCSn2	CCSn1	CCSn0	f <sub>MAIN</sub>	f <sub>MAIN</sub> = 5 MHz	f <sub>MAIN</sub> = 10 MHz	f <sub>MAIN</sub> = 20 MHz	f <sub>MAIN</sub> = 32 MHz
0	0	0	0	f <sub>MAIN</sub>	5 MHz	10 MHz Note	Setting prohibited Note	Setting prohibited Note
0	0	0	1	f <sub>MAIN</sub> /2	2.5 MHz	5 MHz	10 MHz Note	16 MHz Note
0	0	1	0	f <sub>MAIN</sub> /2 <sup>2</sup>	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	f <sub>MAIN</sub> /2 <sup>3</sup>	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0	1	0	0	f <sub>MAIN</sub> /2 <sup>4</sup>	312.5 kHz	625 kHz	1.25 MHz	2 MHz
0	1	0	1	f <sub>MAIN</sub> /2 <sup>11</sup>	2.44 kHz	4.88 kHz	9.77 kHz	15.63 kHz
0	1	1	0	f <sub>MAIN</sub> /2 <sup>12</sup>	1.22 kHz	2.44 kHz	4.88 kHz	7.81 kHz
0	1	1	1	f <sub>MAIN</sub> /2 <sup>13</sup>	610 Hz	1.22 kHz	2.44 kHz	3.91 kHz
1	0	0	0	f <sub>SUB</sub>	32.768 kHz			
1	0	0	1	f <sub>SUB</sub> /2	16.384 kHz			
1	0	1	0	f <sub>SUB</sub> /2 <sup>2</sup>	8.192 kHz			
1	0	1	1	f <sub>SUB</sub> /2 <sup>3</sup>	4.096 kHz			
1	1	0	0	f <sub>SUB</sub> /2 <sup>4</sup>	2.048 kHz			
1	1	0	1	f <sub>SUB</sub> /2 <sup>5</sup>	1.024 kHz			
1	1	1	0	f <sub>SUB</sub> /2 <sup>6</sup>	512 Hz			
1	1	1	1	f <sub>SUB</sub> /2 <sup>7</sup>	256 Hz			

**Note** Use the output clock within a range of 16 MHz. See **34.4 or 35.4 AC Characteristics** for details.

**Caution 1.** Change the output clock after disabling clock output (PCLOEn = 0).

**Caution 2.** To shift to STOP mode when the main system clock is selected (CSELn = 0), set PCLOEn = 0 before executing the STOP instruction. When the subsystem clock is selected (CSELn = 1), PCLOEn = 1 can be set because the clock can be output while RTCLPC in the subsystem clock supply mode control register (OSMC) = 0 in STOP mode.

**Caution 3.** In HALT mode when RTCLPC in the subsystem clock supply mode control register (OSMC) = 1 and while the subsystem clock (f<sub>SUB</sub>) is used for CPU operation, it is not possible to output the subsystem clock (f<sub>SUB</sub>) from the PCLBUZn pin.

**Remark 1.** n = 0, 1

**Remark 2.** f<sub>MAIN</sub>: Main system clock frequency

f<sub>SUB</sub>: Subsystem clock frequency

### 12.3.2 Registers controlling port functions of pins to be used for clock or buzzer output

Using a port pin for clock or buzzer output requires setting of the registers that control the port functions multiplexed on the target pin (port mode register (PMxx), port register (Pxx)). For details, see **4.3.1 Port mode registers (PMxx)** and **4.3.2 Port registers (Pxx)**.

Specifically, using a port pin with a multiplexed clock or buzzer output function (e.g. P140/INTP6/PCLBUZ0, P141/INTP7/PCLBUZ1) for clock or buzzer output, requires setting the corresponding bits in the port mode register (PMxx) and port register (Pxx) to 0.

Example: When P140/INTP6/PCLBUZ0 is to be used for clock or buzzer output

Set the PM140 bit of port mode register 14 to 0.

Set the P140 bit of port register 14 to 0.

## 12.4 Operations of Clock Output/Buzzer Output Controller

One pin can be used to output a clock or buzzer sound.

The PCLBUZ0 pin outputs a clock/buzzer selected by the clock output select register 0 (CKS0).

The PCLBUZ1 pin outputs a clock/buzzer selected by the clock output select register 1 (CKS1).

### 12.4.1 Operation as output pin

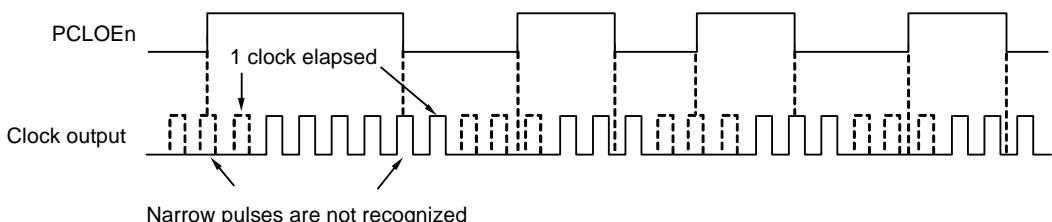
The PCLBUZn pin is output as the following procedures.

- <1> Set 0 in the bit of the port mode register (PMxx) and port register (Pxx) which correspond to the port which has a pin used as the PCLBUZ0 pin.
- <2> Select the output frequency with bits 0 to 3 (CCSn0 to CCSn2, CSELn) of the clock output select register (CKSn) of the PCLBUZn pin (output in disabled status).
- <3> Set bit 7 (PCLOEn) of the CKSn register to 1 to enable clock/buzzer output.

**Remark 1.** The controller used for outputting the clock starts or stops outputting the clock one clock after enabling or disabling clock output (PCLOEn bit) is switched. At this time, pulses with a narrow width are not output. Figure 12 - 3 shows enabling or stopping output using the PCLOEn bit and the timing of outputting the clock.

**Remark 2.** n = 0, 1

Figure 12 - 3 Timing of Outputting Clock from PCLBUZn Pin



## 12.5 Cautions of clock output/buzzer output controller

When the main system clock is selected for the PCLBUZn output (CSEL = 0), if STOP mode is entered within 1.5 clock cycles output from the PCLBUZn pin after the output is disabled (PCLOEn = 0), the PCLBUZn output width becomes shorter.

## CHAPTER 13 WATCHDOG TIMER

### 13.1 Functions of Watchdog Timer

The counting operation of the watchdog timer is set by the option byte (000C0H).

The watchdog timer operates on the low-speed on-chip oscillator clock ( $f_{IL}$ ).

The watchdog timer is used to detect an inadvertent program loop. If a program loop is detected, an internal reset signal is generated.

Program loop is detected in the following cases.

- If the watchdog timer counter overflows
- If a 1-bit manipulation instruction is executed on the watchdog timer enable register (WDTE)
- If data other than “ACH” is written to the WDTE register
- If data is written to the WDTE register during a window close period

When a reset occurs due to the watchdog timer, bit 4 (WDTRF) of the reset control flag register (RESF) is set to 1.

For details of the RESF register, see **CHAPTER 24 RESET FUNCTION**.

When  $75\% + 1/2 f_{IL}$  of the overflow time is reached, an interval interrupt can be generated.

## 13.2 Configuration of Watchdog Timer

The watchdog timer includes the following hardware.

**Table 13 - 1 Configuration of Watchdog Timer**

Item	Configuration
Counter	Internal counter (17 bits)
Control register	Watchdog timer enable register (WDTE)

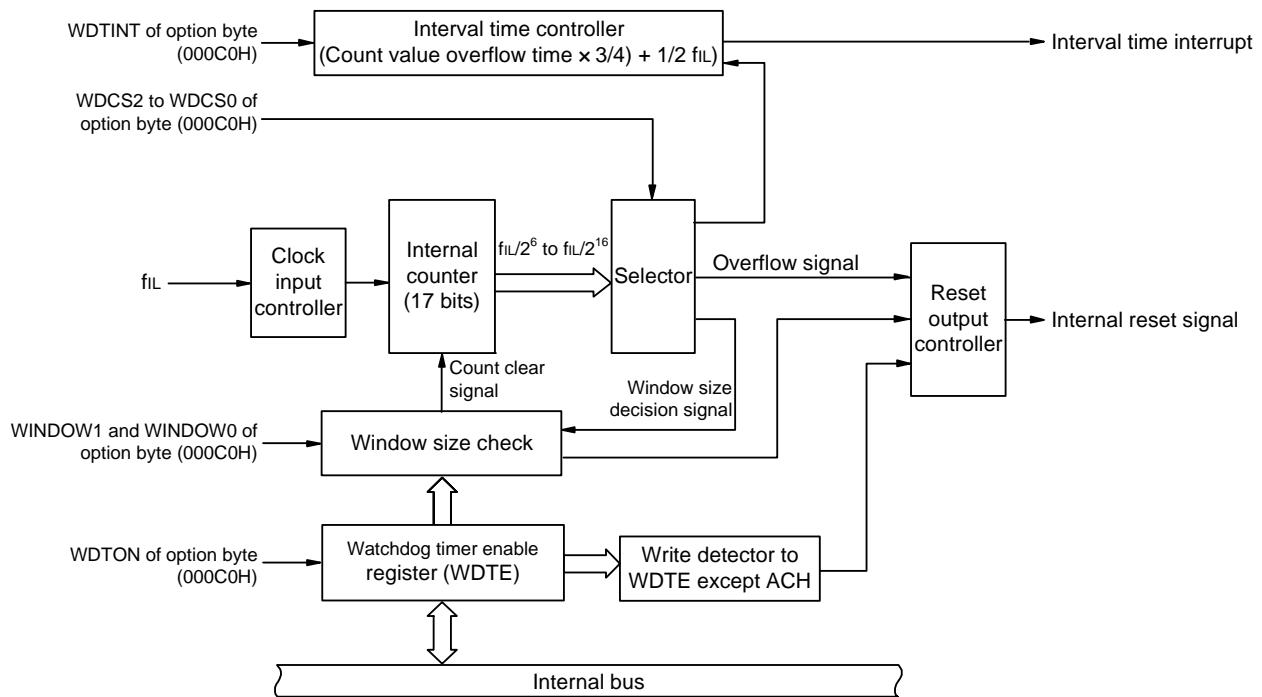
How the counter operation is controlled, overflow time, window open period, and interval interrupt are set by the option byte.

**Table 13 - 2 Setting of Option Bytes and Watchdog Timer**

Setting of Watchdog Timer	Option Byte (000C0H)
Watchdog timer interval interrupt	Bit 7 (WDTINT)
Window open period	Bits 6 and 5 (WINDOW1, WINDOW0)
Controlling counter operation of watchdog timer	Bit 4 (WDTON)
Overflow time of watchdog timer	Bits 3 to 1 (WDCS2 to WDCS0)
Controlling counter operation of watchdog timer (in HALT/STOP mode)	Bit 0 (WDSTBYON)

**Remark** For the option byte, see **CHAPTER 29 OPTION BYTE**.

**Figure 13 - 1 Block Diagram of Watchdog Timer**



**Remark** fIL: Low-speed on-chip oscillator clock

### 13.3 Register Controlling Watchdog Timer

The watchdog timer is controlled by the watchdog timer enable register (WDTE).

#### 13.3.1 Watchdog timer enable register (WDTE)

Writing “ACH” to the WDTE register clears the watchdog timer counter and starts counting again.

This register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to 9AH or 1AH Note.

**Figure 13 - 2 Format of Watchdog timer enable register (WDTE)**

Address: FFFABH	After reset: 9AH/1AH Note	R/W						
Symbol	7	6	5	4	3	2	1	0
WDTE								

**Note** The WDTE register reset value differs depending on the WDTON bit setting value of the option byte (000C0H). To operate watchdog timer, set the WDTON bit to 1.

WDTON Bit Setting Value	WDTE Register Reset Value
0 (watchdog timer count operation disabled)	1AH
1 (watchdog timer count operation enabled)	9AH

**Caution 1.** If a value other than “ACH” is written to the WDTE register, an internal reset signal is generated.

**Caution 2.** If a 1-bit memory manipulation instruction is executed for the WDTE register, an internal reset signal is generated.

**Caution 3.** The value read from the WDTE register is 9AH/1AH (this differs from the written value (ACH)).

## 13.4 Operation of Watchdog Timer

### 13.4.1 Controlling operation of watchdog timer

1. When the watchdog timer is used, its operation is specified by the option byte (000C0H).
  - Enable counting operation of the watchdog timer by setting bit 4 (WDTON) of the option byte (000C0H) to 1 (the counter starts operating after a reset release) (for details, see **CHAPTER 29**).

WDTON	Watchdog Timer Counter
0	Counter operation disabled (counting stopped after reset)
1	Counter operation enabled (counting started after reset)

- Set an overflow time by using bits 3 to 1 (WDCS2 to WDCS0) of the option byte (000C0H) (for details, see **13.4.2** and **CHAPTER 29**).
- Set a window open period by using bits 6 and 5 (WINDOW1 and WINDOW0) of the option byte (000C0H) (for details, see **13.4.3** and **CHAPTER 29**).

2. After a reset release, the watchdog timer starts counting.
3. By writing “ACH” to the watchdog timer enable register (WDTE) after the watchdog timer starts counting and before the overflow time set by the option byte, the watchdog timer is cleared and starts counting again.
4. After that, write the WDTE register the second time or later after a reset release during the window open period. If the WDTE register is written during a window close period, an internal reset signal is generated.
5. If the overflow time expires without “ACH” written to the WDTE register, an internal reset signal is generated.

An internal reset signal is generated in the following cases.

- If a 1-bit manipulation instruction is executed on the WDTE register
- If data other than “ACH” is written to the WDTE register

**Caution 1.**When data is written to the watchdog timer enable register (WDTE) for the first time after reset release, the watchdog timer is cleared in any timing regardless of the window open time, as long as the register is written before the overflow time, and the watchdog timer starts counting again.

**Caution 2.**After “ACH” is written to the WDTE register, an error of up to 2 clocks (fIL) may occur before the watchdog timer is cleared.

**Caution 3.**The watchdog timer can be cleared immediately before the count value overflows.

**Caution 4. The operation of the watchdog timer in the HALT and STOP modes differs as follows depending on the set value of bit 0 (WDSTBYON) of the option byte (000C0H).**

	WDSTBYON = 0	WDSTBYON = 1
In HALT mode	Watchdog timer operation stops.	Watchdog timer operation continues.
In STOP mode		
In SNOOZE mode		

If WDSTBYON = 0, the watchdog timer resumes counting after the HALT or STOP mode is released. At this time, the counter is cleared to 0 and counting starts.

When operating with the X1 oscillation clock after releasing the STOP mode, the CPU starts operating after the oscillation stabilization time has elapsed.

Therefore, if the period between the STOP mode release and the watchdog timer overflow is short, an overflow occurs during the oscillation stabilization time, causing a reset.

Consequently, set the overflow time in consideration of the oscillation stabilization time when operating with the X1 oscillation clock and when the watchdog timer is to be cleared after the STOP mode release by an interval interrupt.

### 13.4.2 Setting overflow time of watchdog timer

Set the overflow time of the watchdog timer by using bits 3 to 1 (WDCS2 to WDCS0) of the option byte (000C0H).

If an overflow occurs, an internal reset signal is generated. The present count is cleared and the watchdog timer starts counting again by writing "ACH" to the watchdog timer enable register (WDTE) during the window open period before the overflow time.

The following overflow times can be set.

**Table 13 - 3 Setting of Overflow Time of Watchdog Timer**

WDCS2	WDCS1	WDCS0	Overflow Time of Watchdog Timer (f <sub>L</sub> = 17.25 kHz (MAX.))
0	0	0	2 <sup>6</sup> /f <sub>L</sub> (3.71 ms)
0	0	1	2 <sup>7</sup> /f <sub>L</sub> (7.42 ms)
0	1	0	2 <sup>8</sup> /f <sub>L</sub> (14.84 ms)
0	1	1	2 <sup>9</sup> /f <sub>L</sub> (29.68 ms)
1	0	0	2 <sup>11</sup> /f <sub>L</sub> (118.72 ms)
1	0	1	2 <sup>13</sup> /f <sub>L</sub> (474.89 ms) Note
1	1	0	2 <sup>14</sup> /f <sub>L</sub> (949.79 ms) Note
1	1	1	2 <sup>16</sup> /f <sub>L</sub> (3799.18 ms) Note

**Note** When clearing the counter of the watchdog timer in any R5F104xF, R5F104xG, R5F104xH, or R5F104xJ product while the watchdog timer interval is set to 2<sup>13</sup>/f<sub>L</sub>, 2<sup>14</sup>/f<sub>L</sub>, or 2<sup>16</sup>/f<sub>L</sub> and watchdog timer interval interrupts are in use, follow the procedure described in **13.5 Restrictions on the Watchdog Timer**.

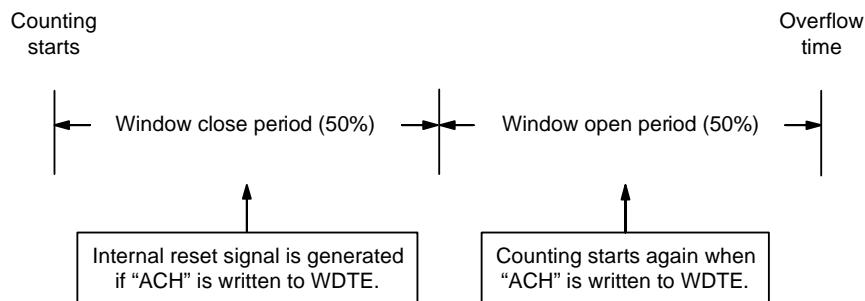
**Remark** f<sub>L</sub>: Low-speed on-chip oscillator clock frequency

### 13.4.3 Setting window open period of watchdog timer

Set the window open period of the watchdog timer by using bits 6 and 5 (WINDOW1, WINDOW0) of the option byte (000C0H). The outline of the window is as follows.

- If “ACH” is written to the watchdog timer enable register (WDTE) during the window open period, the watchdog timer is cleared and starts counting again.
- Even if “ACH” is written to the WDTE register during the window close period, an abnormality is detected and an internal reset signal is generated.

**Example:** If the window open period is 50%



**Caution** When data is written to the WDTE register for the first time after reset release, the watchdog timer is cleared in any timing regardless of the window open time, as long as the register is written before the overflow time, and the watchdog timer starts counting again.

The window open period can be set as follows.

**Table 13 - 4 Setting Window Open Period of Watchdog Timer**

WINDOW1	WINDOW0	Window Open Period of Watchdog Timer
0	0	Setting prohibited
0	1	50%
1	0	75%
1	1	100%

**Caution** When bit 0 (WDSTBYON) of the option byte (000C0H) = 0, the window open period is 100% regardless of the values of the WINDOW1 and WINDOW0 bits.

**Remark** If the overflow time is set to  $2^9/f_{IL}$ , the window close time and open time are as follows.

	Setting of Window Open Period		
	50%	75%	100%
Window close time	0 to 20.08 ms	0 to 10.04 ms	None
Window open time	20.08 to 29.68 ms	10.04 to 29.68 ms	0 to 29.68 ms

<When window open period is 50%>

- Overflow time:  
 $2^9/f_{IL} (\text{MAX.}) = 2^9/17.25 \text{ kHz} (\text{MAX.}) = 29.68 \text{ ms}$
- Window close time:  
 $0 \text{ to } 2^9/f_{IL} (\text{MIN.}) \times (1 - 0.5) = 0 \text{ to } 2^9/12.75 \text{ kHz} \times 0.5 = 0 \text{ to } 20.08 \text{ ms}$
- Window open time:  
 $2^9/f_{IL} (\text{MIN.}) \times (1 - 0.5) \text{ to } 2^9/f_{IL} (\text{MAX.}) = 2^9/12.75 \text{ kHz} \times 0.5 \text{ to } 2^9/17.25 \text{ kHz} = 20.08 \text{ to } 29.68 \text{ ms}$

#### 13.4.4 Setting watchdog timer interval interrupt

Setting bit 7 (WDTINT) of an option byte (000C0H) can generate an interval interrupt (INTWDTI) when  $75\% + 1/2 f_{IL}$  of the overflow time is reached.

**Table 13 - 5 Setting of Watchdog Timer Interval Interrupt**

WDTINT	Use of Watchdog Timer Interval Interrupt
0	Interval interrupt is not used.
1	Interval interrupt is generated when $75\% + 1/2 f_{IL}$ of overflow time is reached.

**Caution** When operating with the X1 oscillation clock after releasing the STOP mode, the CPU starts operating after the oscillation stabilization time has elapsed. Therefore, if the period between the STOP mode release and the watchdog timer overflow is short, an overflow occurs during the oscillation stabilization time, causing a reset. Consequently, set the overflow time in consideration of the oscillation stabilization time when operating with the X1 oscillation clock and when the watchdog timer is to be cleared after the STOP mode release by an interval interrupt.

**Remark** The watchdog timer continues counting even after INTWDTI is generated (until ACH is written to the watchdog timer enable register (WDTE)). If ACH is not written to the WDTE register before the overflow time, an internal reset signal is generated.

#### 13.5 Restrictions on the Watchdog Timer

When clearing the counter of the watchdog timer in applicable products, do so in accord with the procedure given as steps 1 to 5 below.

1. Set the WDTIMK bit in interrupt mask flag register 0 (MK0L) to 1 before clearing the counter for the watchdog timer.
2. Clear the counter of the watchdog timer.
3. Wait for at least 80  $\mu$ s.
4. Clear the WDTIIF bit in interrupt request flag register 0 (IF0L) to 0.
5. Clear the WDTIMK bit in interrupt mask flag register 0 (MK0L) to 0.

## CHAPTER 14 A/D CONVERTER

The number of analog input channels of the A/D converter differs, depending on the product.

	30, 32-pin	36-pin	40-pin	44, 48-pin	52, 64-pin	80-pin	100-pin
Analog input channels	8 ch (ANI0 to ANI3, ANI16 to ANI19)	8 ch (ANI0 to ANI5, ANI18, ANI19)	9 ch (ANI0 to ANI6, ANI18, ANI19)	10 ch (ANI0 to ANI7, ANI18, ANI19)	12 ch (ANI0 to ANI7, ANI16 to ANI19)	17 ch (ANI0 to ANI11, ANI16 to ANI20)	20 ch (ANI0 to ANI14, ANI16 to ANI20)

### 14.1 Function of A/D Converter

The A/D converter is a converter that converts analog input signals into digital values, and is configured to control analog inputs, including up to twenty channels of A/D converter analog inputs (ANI0 to ANI14 and ANI16 to ANI20). 10-bit or 8-bit resolution can be selected by the ADTYP bit of the A/D converter mode register 2 (ADM2).

The A/D converter has the following function.

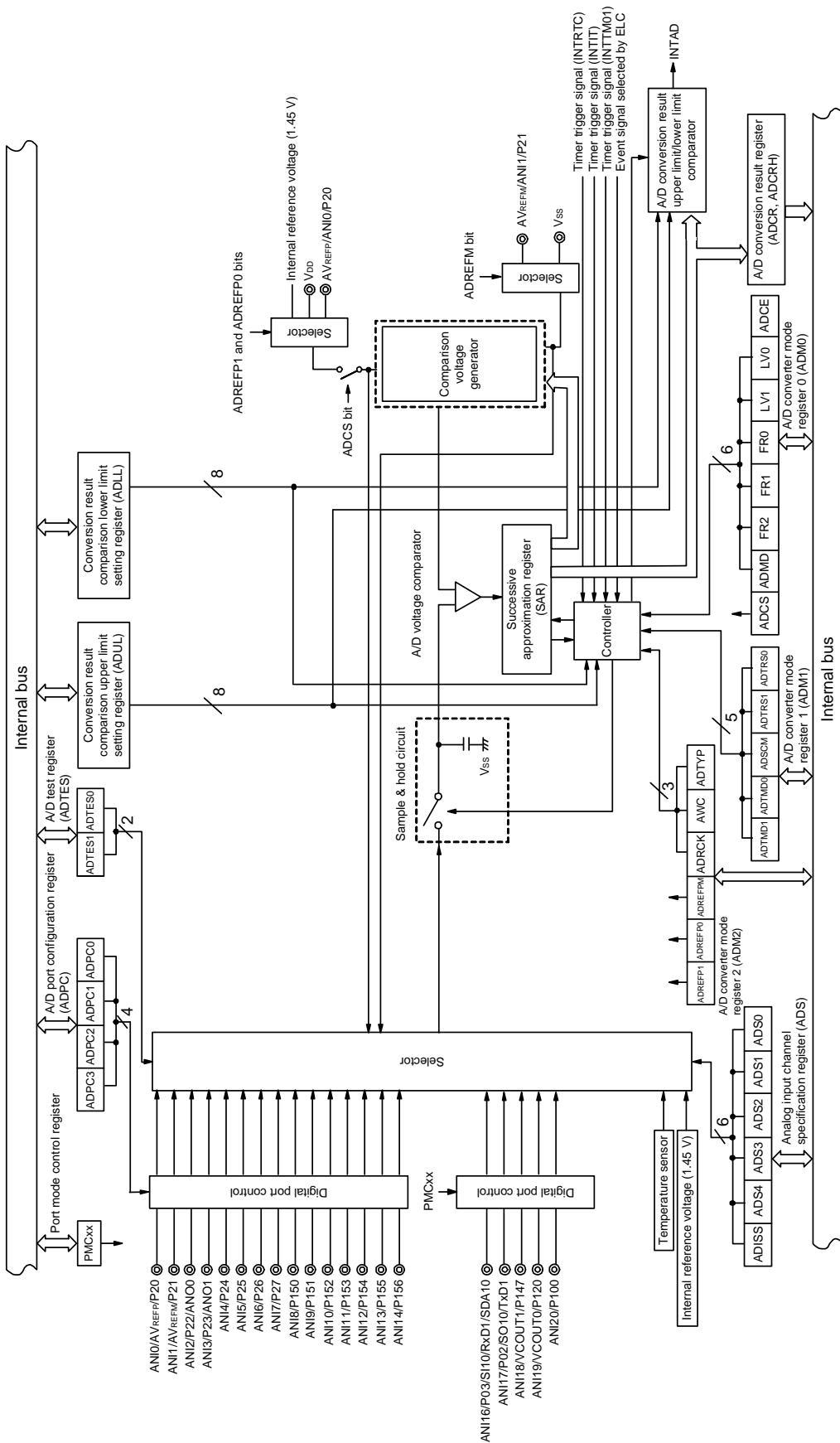
- 10-bit or 8-bit resolution A/D conversion

10-bit or 8-bit resolution A/D conversion is carried out repeatedly for one analog input channel selected from ANI0 to ANI14 and ANI16 to ANI20. Each time an A/D conversion operation ends, an interrupt request (INTAD) is generated (when in the select mode).

Various A/D conversion modes can be specified by using the mode combinations below.

Trigger mode	Software trigger	Conversion is started by software.
	Hardware trigger no-wait mode	Conversion is started by detecting a hardware trigger.
	Hardware trigger wait mode	The power is turned on by detecting a hardware trigger while the system is off and in the conversion standby state, and conversion is then started automatically after the stabilization wait time passes. When using the SNOOZE mode function, specify the hardware trigger wait mode.
Channel selection mode	Select mode	A/D conversion is performed on the analog input of one selected channel.
	Scan mode	A/D conversion is performed on the analog input of four channels in order. Four consecutive channels can be selected from ANI0 to ANI14 as analog input channels.
Conversion operation mode	One-shot conversion mode	A/D conversion is performed on the selected channel once.
	Sequential conversion mode	A/D conversion is sequentially performed on the selected channels until it is stopped by software.
Operation voltage mode	Standard 1 or standard 2 mode	Conversion is done in the operation voltage range of $2.7 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$ .
	Low voltage 1 or low voltage 2 mode	Conversion is done in the operation voltage range of $1.6 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$ . Select this mode for conversion at a low voltage. Because the operation voltage is low, it is internally boosted during conversion.
Sampling time selection	Sampling clock cycles: 7 fAD	The sampling time in standard 1 or low voltage 1 mode is seven cycles of the conversion clock (fAD). Select this mode when the output impedance of the analog input source is high and the sampling time should be long.
	Sampling clock cycles: 5 fAD	The sampling time in standard 2 or low voltage 2 mode is five cycles of the conversion clock (fAD). Select this mode when enough sampling time is ensured (for example, when the output impedance of the analog input source is low).

Figure 14 - 1 Block Diagram of A/D Converter



**Remark** Analog input pin for Figure 14 - 1 when a 100-pin product is used.

## 14.2 Configuration of A/D Converter

The A/D converter includes the following hardware.

(1) ANI0 to ANI14 and ANI16 to ANI20 pins

These are the analog input pins of the twenty channels of the A/D converter. They input analog signals to be converted into digital signals. Pins other than the one selected as the analog input pin can be used as I/O port pins.

(2) Sample & hold circuit

The sample & hold circuit samples each of the analog input voltages sequentially sent from the input circuit, and sends them to the A/D voltage comparator. This circuit also holds the sampled analog input voltage during A/D conversion.

(3) A/D voltage comparator

This A/D voltage comparator compares the voltage generated from the voltage tap of the comparison voltage generator with the analog input voltage. If the analog input voltage is found to be greater than the reference voltage ( $1/2 \text{ AVREF}$ ) as a result of the comparison, the most significant bit (MSB) of the successive approximation register (SAR) is set. If the analog input voltage is less than the reference voltage ( $1/2 \text{ AVREF}$ ), the MSB bit of the SAR is reset.

After that, bit 8 of the SAR register is automatically set, and the next comparison is made. The voltage tap of the comparison voltage generator is selected by the value of bit 9, to which the result has been already set.

Bit 9 = 0: ( $1/4 \text{ AVREF}$ )

Bit 9 = 1: ( $3/4 \text{ AVREF}$ )

The voltage tap of the comparison voltage generator and the analog input voltage are compared and bit 8 of the SAR register is manipulated according to the result of the comparison.

Analog input voltage  $\geq$  Voltage tap of comparison voltage generator: Bit 8 = 1

Analog input voltage  $\leq$  Voltage tap of comparison voltage generator: Bit 8 = 0

Comparison is continued like this to bit 0 of the SAR register.

When performing A/D conversion at a resolution of 8 bits, the comparison continues until bit 2 of the SAR register.

**Remark** AVREF: The + side reference voltage of the A/D converter. This can be selected from AVREFP, the internal reference voltage (1.45 V), and VDD.

(4) Comparison voltage generator

The comparison voltage generator generates the comparison voltage input from an analog input pin.

(5) Successive approximation register (SAR)

The SAR register is a register that sets voltage tap data whose values from the comparison voltage generator match the voltage values of the analog input pins, 1 bit at a time starting from the most significant bit (MSB).

If data is set in the SAR register all the way to the least significant bit (LSB) (end of A/D conversion), the contents of the SAR register (conversion results) are held in the A/D conversion result register (ADCR). When all the specified A/D conversion operations have ended, an A/D conversion end interrupt request signal (INTAD) is generated.

(6) 10-bit A/D conversion result register (ADCR)

The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCR register holds the A/D conversion result in its higher 10 bits (the lower 6 bits are fixed to 0).

(7) 8-bit A/D conversion result register (ADCRH)

The A/D conversion result is loaded from the successive approximation register to this register each time A/D conversion is completed, and the ADCRH register stores the higher 8 bits of the A/D conversion result.

(8) Controller

This circuit controls the conversion time of an input analog signal that is to be converted into a digital signal, as well as starting and stopping of the conversion operation. When A/D conversion has been completed, this controller generates INTAD through the A/D conversion result upper limit/lower limit comparator.

(9) AVREFP pin

This pin inputs an external reference voltage (AVREFP).

If using AVREFP as the + side reference voltage of the A/D converter, set the ADREFP1 and ADREFP0 bits of A/D converter mode register 2 (ADM2) to 0 and 1, respectively.

The analog signals input to ANI2 to ANI14 and ANI16 to ANI20 are converted to digital signals based on the voltage applied between AVREFP and the – side reference voltage (AVREFM/Vss).

In addition to AVREFP, it is possible to select VDD or the internal reference voltage (1.45 V) as the + side reference voltage of the A/D converter.

(10) AVREFM pin

This pin inputs an external reference voltage (AVREFM). If using AVREFM as the - side reference voltage of the A/D converter, set the ADREFM bit of the ADM2 register to 1.

In addition to AVREFM, it is possible to select Vss as the – side reference voltage of the A/D converter.

### 14.3 Registers Controlling A/D Converter

The A/D converter is controlled by the following registers.

- Peripheral enable register 0 (PER0)
- A/D converter mode register 0 (ADM0)
- A/D converter mode register 1 (ADM1)
- A/D converter mode register 2 (ADM2)
- 10-bit A/D conversion result register (ADCR)
- 8-bit A/D conversion result register (ADCRH)
- Analog input channel specification register (ADS)
- Conversion result comparison upper limit setting register (ADUL)
- Conversion result comparison lower limit setting register (ADLL)
- A/D test register (ADTES)
- A/D port configuration register (ADPC)
- Port mode control registers 0, 10, 12, and 14 (PMC0, PMC10, PMC12, PMC14)
- Port mode registers 0, 2, 10, 12, 14, and 15 (PM0, PM2, PM10, PM12, PM14, PM15)

### 14.3.1 Peripheral enable register 0 (PER0)

This register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the A/D converter is used, be sure to set bit 5 (ADCEN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 2 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

PER0	RTCEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
ADCEN	Control of A/D converter input clock supply							
0	Stops input clock supply. • SFR used by the A/D converter cannot be written. • The A/D converter is in the reset status.							
1	Enables input clock supply. • SFR used by the A/D converter can be read/written.							

**Note**      80 and 100-pin products only.

**Caution 1.** When setting the A/D converter, be sure to set the following registers first while the ADCEN bit is set to 1.

If ADCEN = 0, the values of the A/D converter control registers are cleared to their initial values and writing to them is ignored (except for port mode registers 0, 2, 10, 12, 14, and 15 (PM0, PM2, PM10, PM12, PM14, PM15), port mode control registers 0, 10, 12, and 14 (PMC0, PMC10, PMC12, PMC14), and A/D port configuration register (ADPC)).

- A/D converter mode register 0 (ADM0)
- A/D converter mode register 1 (ADM1)
- A/D converter mode register 2 (ADM2)
- 10-bit A/D conversion result register (ADCR)
- 8-bit A/D conversion result register (ADCRH)
- Analog input channel specification register (ADS)
- Conversion result comparison upper limit setting register (ADUL)
- Conversion result comparison lower limit setting register (ADLL)
- A/D test register (ADTES)

**Caution 2.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

### 14.3.2 A/D converter mode register 0 (ADM0)

This register sets the conversion time for analog input to be A/D converted, and starts/stops conversion.

The ADM0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 3 Format of A/D converter mode register 0 (ADM0)**

Address: FFF30H      After reset: 00H      R/W

Symbol	<7>	6	5	4	3	2	1	<0>
--------	-----	---	---	---	---	---	---	-----

ADM0	ADCS	ADMD	FR2 Note 1	FR1 Note 1	FR0 Note 1	LV1 Note 1	LV0 Note 1	ADCE
------	------	------	------------	------------	------------	------------	------------	------

ADCS	A/D conversion operation control						
0	Stops conversion operation [When read] Conversion stopped/standby status						
1	Enables conversion operation [When read] While in the software trigger mode:      Conversion operation status While in the hardware trigger wait mode:      A/D power supply stabilization wait status + conversion operation status						

ADMD	Specification of the A/D conversion channel selection mode						
0	Select mode						
1	Scan mode						

ADCE	A/D voltage comparator operation control Note 2						
0	Stops A/D voltage comparator operation						
1	Enables A/D voltage comparator operation						

**Note 1.** For details of the FR2 to FR0, LV1, LV0 bits, and A/D conversion, see **Tables 14 - 3 to 14 - 6 A/D Conversion Time Selection**.

**Note 2.** While in the software trigger mode or hardware trigger no-wait mode, the operation of the A/D voltage comparator is controlled by the ADCS and ADCE bits, and it takes 1  $\mu$ s from the start of operation for the operation to stabilize. Therefore, when the ADCS bit is set to 1 after 1  $\mu$ s or more has elapsed from the time ADCE bit is set to 1, the conversion result at that time has priority over the first conversion result. Otherwise, ignore data of the first conversion.

**Caution 1. Change the ADMD, FR2 to FR0, LV1, and LV0 bits while conversion is stopped (ADCS = 0, ADCE = 0).**

**Caution 2. Do not set the ADCS bit to 1 and the ADCE bit to 0 at the same time.**

**Caution 3. Do not change the ADCS and ADCE bits from 0 to 1 at the same time by using an 8-bit manipulation instruction. Be sure to set these bits in the order described in 14.7 A/D Converter Setup Flowchart.**

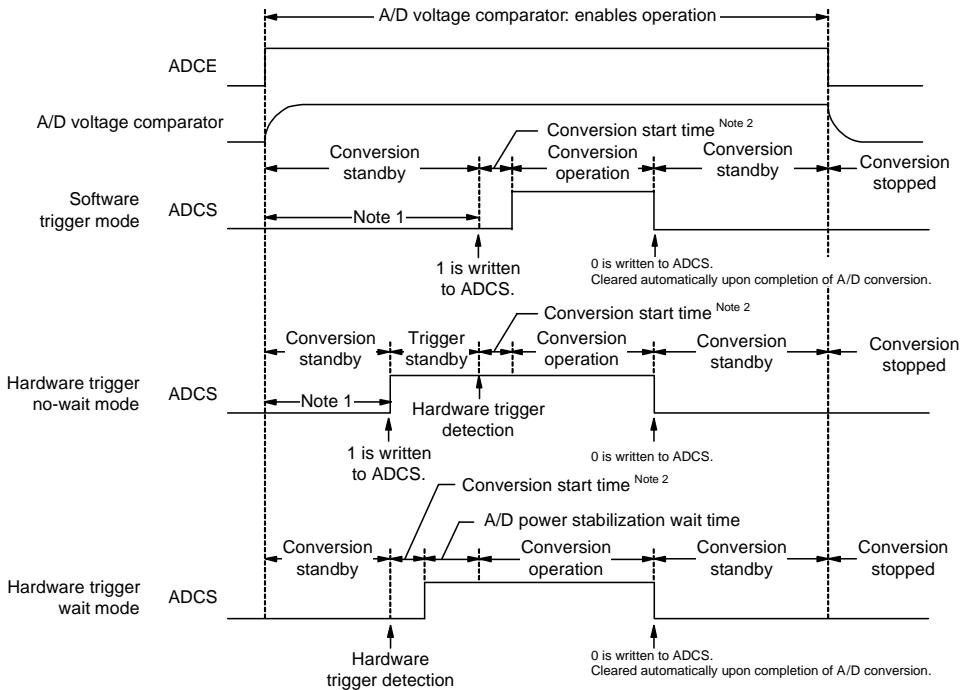
**Table 14 - 1 Settings of ADCS and ADCE Bits**

ADCS	ADCE	A/D Conversion Operation
0	0	Conversion stopped state
0	1	Conversion standby state
1	0	Setting prohibited
1	1	Conversion-in-progress state

**Table 14 - 2 Setting and Clearing Conditions for ADCS Bit**

A/D Conversion Mode			Set Conditions	Clear Conditions
Software trigger	Select mode	Sequential conversion mode	When 1 is written to ADCS	When 0 is written to ADCS
		One-shot conversion mode		<ul style="list-style-type: none"> <li>• When 0 is written to ADCS</li> <li>• The bit is automatically cleared to 0 when A/D conversion ends.</li> </ul>
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		<ul style="list-style-type: none"> <li>• When 0 is written to ADCS</li> <li>• The bit is automatically cleared to 0 when conversion ends on the specified four channels.</li> </ul>
Hardware trigger no-wait mode	Select mode	Sequential conversion mode	When 0 is written to ADCS	When 0 is written to ADCS
		One-shot conversion mode		When 0 is written to ADCS
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		When 0 is written to ADCS
Hardware trigger wait mode	Select mode	Sequential conversion mode	When a hardware trigger is input	When 0 is written to ADCS
		One-shot conversion mode		<ul style="list-style-type: none"> <li>• When 0 is written to ADCS</li> <li>• The bit is automatically cleared to 0 when A/D conversion ends.</li> </ul>
	Scan mode	Sequential conversion mode		When 0 is written to ADCS
		One-shot conversion mode		<ul style="list-style-type: none"> <li>• When 0 is written to ADCS</li> <li>• The bit is automatically cleared to 0 when conversion ends on the specified four channels.</li> </ul>

Figure 14 - 4 Timing Chart When A/D Voltage Comparator Is Used



**Note 1.** While in the software trigger mode or hardware trigger no-wait mode, the time from the rising of the ADCE bit to the falling of the ADCS bit must be  $1 \mu\text{s}$  or longer to stabilize the internal circuit.

**Note 2.** The following time is the maximum amount of time necessary to start conversion.

ADM0			Conversion Clock (fAD)	Conversion Start Time (Number of fCLK Clocks)	
FR2	FR1	FR0		Software trigger mode/ Hardware trigger no wait mode	Hardware trigger wait mode
0	0	0	fCLK/64	63	1
0	0	1	fCLK/32	31	
0	1	0	fCLK/16	15	
0	1	1	fCLK/8	7	
1	0	0	fCLK/6	5	
1	0	1	fCLK/5	4	
1	1	0	fCLK/4	3	
1	1	1	fCLK/2	1	

However, for the second and subsequent conversion in sequential conversion mode and for conversion of the channel specified by scan 1, 2, and 3 in scan mode, the conversion start time and stabilization wait time for A/D power supply do not occur after a hardware trigger is detected.

**Caution 1.** If using the hardware trigger wait mode, setting the ADCS bit to 1 is prohibited (but the bit is automatically switched to 1 when the hardware trigger signal is detected). However, it is possible to clear the ADCS bit to 0 to specify the A/D conversion standby status.

**Caution 2.** While in the one-shot conversion mode of the hardware trigger no-wait mode, the ADCS flag is not automatically cleared to 0 when A/D conversion ends. Instead, 1 is retained.

**Caution 3.** Only rewrite the value of the ADCE bit when ADCS = 0 (while in the conversion stopped/conversion standby status).

**Caution 4.** To complete A/D conversion, specify at least the following time as the hardware trigger interval:

Hardware trigger no wait mode: 2 fCLK clock + Conversion start time + A/D conversion time

Hardware trigger wait mode: 2 fCLK clock + Conversion start time + A/D power supply stabilization wait time + A/D conversion time

**Remark** fCLK: CPU/peripheral hardware clock frequency

**Table 14 - 3 A/D Conversion Time Selection (1/4)**

**(1) When there is no A/D power supply stabilization wait time Normal mode 1, 2  
(software trigger mode/hardware trigger no-wait mode)**

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion Clock (fAD)	Number of Conversion Clock Note	Conversion Time	Conversion Time at 10-Bit Resolution					
									2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V					
FR2	FR1	FR0	LV1	LV0		fCLK = 1 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 16 MHz	fCLK = 32 MHz				
0	0	0	0	0	Normal 1	fCLK/64	19 fAD (number of sampling clock: 7 fAD)	1216/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	76 μs	38 μs	
0	0	1				fCLK/32		608/fCLK		76 μs	38 μs	19 μs		
0	1	0				fCLK/16		304/fCLK		76 μs	38 μs	19 μs	9.5 μs	
0	1	1				fCLK/8		152/fCLK		38 μs	19 μs	9.5 μs	4.75 μs	
1	0	0				fCLK/6		114/fCLK		28.5 μs	14.25 μs	7.125 μs	3.5625 μs	
1	0	1				fCLK/5		95/fCLK		95 μs	23.75 μs	11.875 μs	5.938 μs	
1	1	0				fCLK/4		76/fCLK		76 μs	19 μs	9.5 μs	4.75 μs	
1	1	1				fCLK/2		38/fCLK		38 μs	9.5 μs	4.75 μs	2.375 μs	
0	0	0				0		1088/fCLK		Setting prohibited	Setting prohibited	68 μs	34 μs	
0	0	1	1	1	Normal 2	fCLK/64	17 fAD (number of sampling clock: 5 fAD)	544/fCLK			68 μs	34 μs	17 μs	
0	1	0				fCLK/32		272/fCLK			68 μs	34 μs	17 μs	8.5 μs
0	1	1				fCLK/16		136/fCLK			34 μs	17 μs	8.5 μs	4.25 μs
1	0	0				fCLK/8		102/fCLK			25.5 μs	12.75 μs	6.375 μs	3.1875 μs
1	0	1				fCLK/6		85/fCLK			85 μs	21.25 μs	10.625 μs	5.3125 μs
1	1	0				fCLK/5		68/fCLK			68 μs	17 μs	8.5 μs	4.25 μs
1	1	1				fCLK/4		34/fCLK			34 μs	8.5 μs	4.25 μs	2.125 μs
						fCLK/2								Setting prohibited

**Note** These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (fAD).

**Caution 1.** The A/D conversion time must also be within the relevant range of conversion times (t<sub>CONV</sub>) described in 34.6.1 or 35.6.1 A/D converter characteristics.

**Caution 2.** Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 3.** The above conversion time does not include conversion state time. Conversion state time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.

**Remark** fCLK: CPU/peripheral hardware clock frequency

**Table 14 - 4 A/D Conversion Time Selection (2/4)**

**(2) When there is no A/D power supply stabilization wait time Low-voltage mode 1, 2  
(software trigger mode/hardware trigger no-wait mode)**

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion Clock (fAD)	Number of Conversion Clock Note 4	Conversion Time	Conversion Time at 10-Bit Resolution							
FR2	FR1	FR0	LV1	LV0					1.6 V ≤ VDD ≤ 5.5 V	Note 1	Note 2	Note 3				
									fCLK = 1 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 16 MHz	fCLK = 32 MHz			
0	0	0	1	0	Low-voltage 1	fCLK/64	19 fAD (number of sampling clock: 7 fAD)	1216/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	76 μs	38 μs			
0	0	1				fCLK/32		608/fCLK		76 μs	38 μs	19 μs				
0	1	0				fCLK/16		304/fCLK		76 μs	38 μs	9.5 μs				
0	1	1				fCLK/8		152/fCLK		38 μs	19 μs	9.5 μs	4.75 μs			
1	0	0				fCLK/6		114/fCLK		28.5 μs	14.25 μs	7.125 μs	3.5625 μs			
1	0	1				fCLK/5		95/fCLK		95 μs	23.75 μs	11.875 μs	5.938 μs			
1	1	0				fCLK/4		76/fCLK		76 μs	19 μs	9.5 μs	4.75 μs			
1	1	1				fCLK/2		38/fCLK		38 μs	9.5 μs	4.75 μs	Setting prohibited			
0	0	0	1	1	Low-voltage 2	fCLK/64	17 fAD (number of sampling clock: 5 fAD)	1088/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	68 μs	34 μs			
0	0	1				fCLK/32		544/fCLK		68 μs	34 μs	17 μs				
0	1	0				fCLK/16		272/fCLK		68 μs	34 μs	17 μs	8.5 μs			
0	1	1				fCLK/8		136/fCLK		34 μs	17 μs	8.5 μs	4.25 μs			
1	0	0				fCLK/6		102/fCLK		25.5 μs	12.75 μs	6.375 μs	3.1875 μs			
1	0	1				fCLK/5		85/fCLK		85 μs	21.25 μs	10.625 μs	5.3125 μs			
1	1	0				fCLK/4		68/fCLK		68 μs	17 μs	8.5 μs	4.25 μs			
1	1	1				fCLK/2		34/fCLK		34 μs	8.5 μs	4.25 μs	Setting prohibited			

**Note 1.** 1.8 V ≤ VDD ≤ 5.5 V

**Note 2.** 2.4 V ≤ VDD ≤ 5.5 V

**Note 3.** 2.7 V ≤ VDD ≤ 5.5 V

**Note 4.** These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (fAD).

**Caution 1.** The A/D conversion time must also be within the relevant range of conversion times (tCONV) described in 34.6.1 or 35.6.1 A/D converter characteristics.

**Caution 2.** Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 3.** The above conversion time does not include conversion state time. Conversion state time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.

**Remark** fCLK: CPU/peripheral hardware clock frequency

**Table 14 - 5 A/D Conversion Time Selection (3/4)**

**(3) When there is A/D power supply stabilization wait time Normal mode 1, 2  
(hardware trigger wait mode Note 1)**

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion Clock (fAD)	Number of A/D Power Supply Stabilization Wait Clock	Number of Conversion Clock Note 2	A/D Power Supply Stabilization Wait Time + Conversion Time	A/D Power Supply Stabilization Wait Time + Conversion Time at 10-Bit Resolution				
FR2	FR1	FR0	LV1	LV0						2.7 V ≤ VDD ≤ 5.5 V				
fCLK = 1 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 16 MHz	fCLK = 32 MHz										
0	0	0	0	0	Normal 1	fCLK/64	8 fAD	19 fAD (number of sampling clock: 7 fAD)	1728/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	108 μs	54 μs
0	0	1												
0	1	0												
0	1	1												
1	0	0												
1	0	1												
1	1	0												
1	1	1												
0	0	0	0	0										
0	0	1			Normal 2	fCLK/64	8 fAD	17 fAD (number of sampling clock: 5 fAD)	1600/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	100 μs	50 μs
0	1	0												
0	1	1												
1	0	0												
1	0	1												
1	1	0												
1	1	1												

**Note 1.** For the second and subsequent conversion in sequential conversion mode and for conversion of the channel specified by scan 1, 2, and 3 in scan mode, the conversion start time and stabilization wait time for A/D power supply do not occur after a hardware trigger is detected (see **Table 14 - 3**).

**Note 2.** These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (fAD).

**Caution 1. The A/D conversion time must also be within the relevant range of conversion times (tconv) described in 34.6.1 or 35.6.1 A/D converter characteristics.**

Note that the conversion time (tconv) does not include the A/D power supply stabilization wait time.

**Caution 2. Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0, ADCE = 0).**

**Caution 3. The above conversion time does not include conversion state time. Conversion state time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.**

**Caution 4. When hardware trigger wait mode, specify the conversion time, including the A/D power supply stabilization wait time from the hardware trigger detection.**

**Remark** fCLK: CPU/peripheral hardware clock frequency

**Table 14 - 6 A/D Conversion Time Selection (4/4)**

**(4) When there is A/D power supply stabilization wait time Low-voltage mode 1, 2  
(hardware trigger wait mode Note 1)**

A/D Converter Mode Register 0 (ADM0)					Mode	Conversion Clock (fAD)	Number of A/D Power Supply Stabilization Wait Clock	Number of Conversion Clock Note 5	A/D Power Supply Stabilization Wait Time + Conversion Time	A/D Power Supply Stabilization Wait Time + Conversion Time at 10-Bit Resolution				
FR2	FR1	FR0	LV1	LV0						1.6 V ≤ VDD ≤ 5.5 V	Note 2	Note 3	Note 4	
fCLK = 1 MHz	fCLK = 4 MHz	fCLK = 8 MHz	fCLK = 16 MHz	fCLK = 32 MHz										
0	0	0	1	0	Low-voltage1	fCLK/64	2 fAD	19 fAD (number of sampling clock: 7 fAD)	1344/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	84 µs	42 µs
0	0	1				fCLK/32			672/fCLK		84 µs	42 µs	21 µs	
0	1	0				fCLK/16			336/fCLK		84 µs	42 µs	10.5 µs	
0	1	1				fCLK/8			168/fCLK		42 µs	21 µs	10.5 µs	5.25 µs
1	0	0				fCLK/6			126/fCLK		31.25 µs	15.75 µs	7.875 µs	3.9375 µs
1	0	1				fCLK/5			105/fCLK		105 µs	26.25 µs	13.125 µs	6.5625 µs
1	1	0				fCLK/4			84/fCLK		84 µs	21 µs	10.5 µs	5.25 µs
1	1	1				fCLK/2			42/fCLK		42 µs	10.5 µs	5.25 µs	2.625 µs
0	0	0	1	1	Low-voltage2	fCLK/64	2 fAD	17 fAD (number of sampling clock: 5 fAD)	1216/fCLK	Setting prohibited	Setting prohibited	Setting prohibited	76 µs	38 µs
0	0	1				fCLK/32			608/fCLK		76 µs	38 µs	19 µs	
0	1	0				fCLK/16			304/fCLK		76 µs	38 µs	19 µs	9.5 µs
0	1	1				fCLK/8			152/fCLK		38 µs	19 µs	9.5 µs	4.75 µs
1	0	0				fCLK/6			114/fCLK		28.5 µs	14.25 µs	7.125 µs	3.5625 µs
1	0	1				fCLK/5			95/fCLK		96 µs	23.75 µs	11.875 µs	5.938 µs
1	1	0				fCLK/4			76/fCLK		76 µs	19 µs	9.5 µs	4.75 µs
1	1	1				fCLK/2			38/fCLK		38 µs	9.5 µs	4.75 µs	2.375 µs

**Note 1.** For the second and subsequent conversion in sequential conversion mode and for conversion of the channel specified by scan 1, 2, and 3 in scan mode, the conversion start time and stabilization wait time for A/D power supply do not occur after a hardware trigger is detected (see **Table 14 - 4**).

**Note 2.**  $1.8 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$

**Note 3.**  $2.4 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$

**Note 4.**  $2.7 \text{ V} \leq \text{VDD} \leq 5.5 \text{ V}$

**Note 5.** These are the numbers of clock cycles when conversion is with 10-bit resolution. When eight-bit resolution is selected, the values are shorter by two cycles of the conversion clock (fAD).

**Caution 1. The A/D conversion time must also be within the relevant range of conversion times (tCONV) described in 34.6.1 or 35.6.1 A/D converter characteristics.**

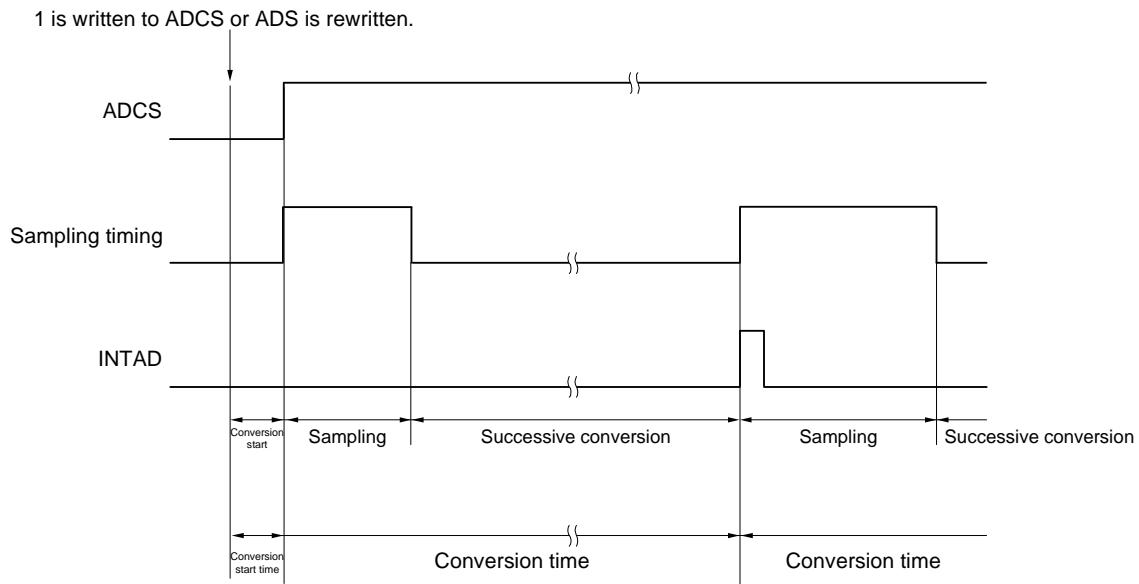
Note that the conversion time (tCONV) does not include the A/D power supply stabilization wait time.

**Caution 2. Rewrite the FR2 to FR0, LV1 and LV0 bits to other than the same data while conversion is stopped (ADCS = 0, ADCE = 0).**

**Caution 3. The above conversion time does not include conversion state time. Conversion state time add in the first conversion. Select conversion time, taking clock frequency errors into consideration.**

**Caution 4. When hardware trigger wait mode, specify the conversion time, including the A/D power supply stabilization wait time from the hardware trigger detection.**

**Remark** fCLK: CPU/peripheral hardware clock frequency

**Figure 14 - 5 A/D Converter Sampling and A/D Conversion Timing (Example for Software Trigger Mode)**

### 14.3.3 A/D converter mode register 1 (ADM1)

This register is used to specify the A/D conversion trigger, conversion mode, and hardware trigger signal.

The ADM1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 6 Format of A/D converter mode register 1 (ADM1)**

Address: FFF32H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

ADM1	ADTMD1	ADTMD0	ADSCM	0	0	0	ADTRS1	ADTRS0
------	--------	--------	-------	---	---	---	--------	--------

ADTMD1	ADTMD0	Selection of the A/D conversion trigger mode
0	0	Software trigger mode
0	1	
1	0	Hardware trigger no-wait mode
1	1	Hardware trigger wait mode

ADSCM	Specification of the A/D conversion mode
0	Sequential conversion mode
1	One-shot conversion mode

ADTRS1	ADTRS0	Selection of the hardware trigger signal
0	0	End of timer channel 1 count or capture interrupt signal (INTTM01)
0	1	Event signal selected by ELC
1	0	Real-time clock interrupt signal (INTRTC)
1	1	12-bit interval timer interrupt signal (INTIT)

**Caution 1.** Rewrite the value of the ADM1 register while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 2.** To complete A/D conversion, specify at least the following time as the hardware trigger interval:

Hardware trigger no wait mode: 2 fCLK clock + conversion start time + A/D conversion time

Hardware trigger wait mode: 2 fCLK clock + conversion start time + A/D power supply stabilization wait time + A/D conversion time

**Caution 3.** In modes other than SNOOZE mode, input of the next INTRTC or INTIT will not be recognized as a valid hardware trigger for up to four fCLK cycles after the first INTRTC or INTIT is input.

**Remark**    fCLK: CPU/peripheral hardware clock frequency

### 14.3.4 A/D converter mode register 2 (ADM2)

This register is used to select the + side or - side reference voltage of the A/D converter, check the upper limit and lower limit A/D conversion result values, select the resolution, and specify whether to use the SNOOZE mode.

The ADM2 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 7 Format of A/D converter mode register 2 (ADM2) (1/2)**

Address: F0010H      After reset: 00H      R/W

Symbol	7	6	5	4	<3>	<2>	1	<0>
--------	---	---	---	---	-----	-----	---	-----

ADM2	ADREFP1	ADREFP0	ADREFM	0	ADRCK	AWC	0	ADTYP
------	---------	---------	--------	---	-------	-----	---	-------

ADREFP1	ADREFP0	Selection of the + side reference voltage source of the A/D converter
0	0	Supplied from Vdd
0	1	Supplied from P20/AVREFP/ANIO
1	0	Supplied from the internal reference voltage (1.45 V) Note
1	1	Setting prohibited

- When ADREFP1 or ADREFP0 bit is rewritten, this must be configured in accordance with the following procedures.

- (1) Set ADCE = 0
- (2) Change the values of ADREFP1 and ADREFP0
- (3) Reference voltage stabilization wait time (A)
- (4) Set ADCE = 1
- (5) Reference voltage stabilization wait time (B)

When ADREFP1 and ADREFP0 are set to 1 and 0, the setting is changed to A = 5 µs, B = 1 µs.

When ADREFP1 and ADREFP0 are set to 0 and 0 or 0 and 1, A needs no wait and B = 1 µs.

- When ADREFP1 and ADREFP0 are set to 1 and 0, respectively, A/D conversion cannot be performed on the temperature sensor output voltage and internal reference voltage.

Be sure to perform A/D conversion while ADISS = 0.

ADREFM	Selection of the - side reference voltage of the A/D converter
0	Supplied from Vss
1	Supplied from P21/AVREFM/ANI1

**Note** Operation is possible only in HS (high-speed main) mode.

**Caution 1. Rewrite the value of the ADM2 register while conversion is stopped (ADCS = 0, ADCE = 0).**

**Caution 2. When entering STOP mode or HALT mode while the CPU is operating on the subsystem clock, do not set ADREFP1 to 1. When selecting internal reference voltage (ADREFP1, ADREFP0 = 1, 0), the current value of A/D converter reference voltage current (IADREF) shown in 34.3.2 or 35.3.2 Supply current characteristics is added.**

**Caution 3. When using AVREFP and AVREFM, specify ANI0 and ANI1 as the analog input channels and specify input mode by using the port mode register.**

**Figure 14 - 8 Format of A/D converter mode register 2 (ADM2) (2/2)**

Address: F0010H After reset: 00H R/W

Symbol 7 6 5 4 &lt;3&gt; &lt;2&gt; 1 &lt;0&gt;

ADM2	ADREFP1	ADREFP0	ADREFM	0	ADRCK	AWC	0	ADTYP
------	---------	---------	--------	---	-------	-----	---	-------

ADRCK	Checking the upper limit and lower limit conversion result values
0	The interrupt signal (INTAD) is output when the ADLL register $\leq$ the ADCR register $\leq$ the ADUL register (AREA1).
1	The interrupt signal (INTAD) is output when the ADCR register $<$ the ADLL register (AREA2) or the ADUL register $<$ the ADCR register (AREA3).

Figure 14 - 9 shows the generation range of the interrupt signal (INTAD) for AREA1 to AREA3.

AWC	Specification of the SNOOZE mode
0	Do not use the SNOOZE mode function.
1	Use the SNOOZE mode function.

When there is a hardware trigger signal in the STOP mode, the STOP mode is exited, and A/D conversion is performed without operating the CPU (the SNOOZE mode).

- The SNOOZE mode function can only be specified when the high-speed on-chip oscillator clock is selected for the CPU/peripheral hardware clock (fCLK). If any other clock is selected, specifying this mode is prohibited.
- Using the SNOOZE mode function in the software trigger mode or hardware trigger no-wait mode is prohibited.
- Using the SNOOZE mode function in the sequential conversion mode is prohibited.
- When using the SNOOZE mode function, specify a hardware trigger interval of at least “shift time to SNOOZE mode Note + conversion start time + A/D power supply stabilization wait time + A/D conversion time + 2 fCLK clock”
- Even when using SNOOZE mode, be sure to set the AWC bit to 0 in normal operation and change it to 1 just before shifting to STOP mode.

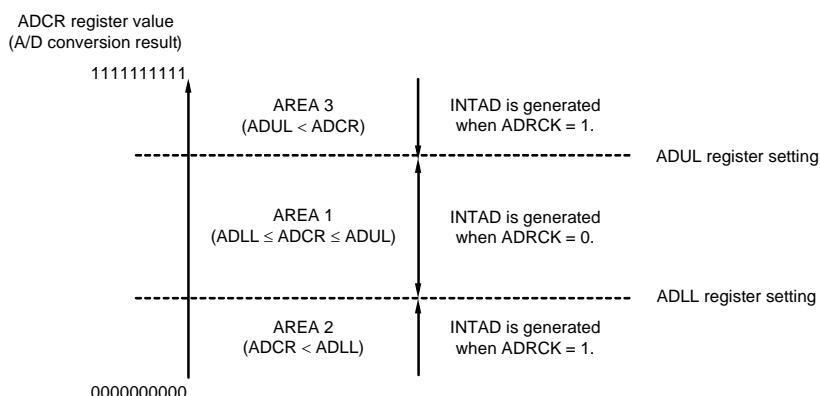
Also, be sure to change the AWC bit to 0 after returning from STOP mode to normal operation.

If the AWC bit is left set to 1, A/D conversion will not start normally in spite of the subsequent SNOOZE or normal operation mode.

ADTYP	Selection of the A/D conversion resolution
0	10-bit resolution
1	8-bit resolution

**Note** Refer to “Transition time from STOP mode to SNOOZE mode” in **23.3.3 SNOOZE mode**.

**Caution** Only rewrite the value of the ADM2 register while conversion operation is stopped (which is indicated by the ADCS and ADCE bits of A/D converter mode register 0 (ADM0) being 0).

**Figure 14 - 9 ADRCK Bit Interrupt Signal Generation Range**

**Remark** If INTAD does not occur, the A/D conversion result is not stored in the ADCR or ADCRH register.

### 14.3.5 10-bit A/D conversion result register (ADCR)

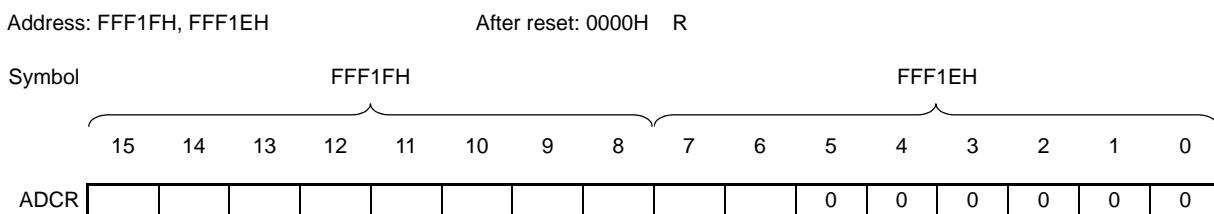
This register is a 16-bit register that stores the A/D conversion result in the select mode. The lower 6 bits are fixed to 0. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR). The higher 8 bits of the conversion result are stored in FFF1FH and the lower 2 bits are stored in the higher 2 bits of FFF1EH Note.

The ADCR register can be read by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Note** If the A/D conversion result is outside the range specified by using the A/D conversion comparison function (the value specified by the ADRCK bit of the ADM2 register and ADUL/ADLL registers; see **Figure 14 - 9**), the result is not stored.

**Figure 14 - 10 Format of 10-bit A/D conversion result register (ADCR)**



**Caution 1.** When 8-bit resolution A/D conversion is selected (when the ADTYP bit of A/D converter mode register 2 (ADM2) is 1) and the ADCR register is read, 0 is read from the lower two bits (bits 7 and 6 of the ADCR register).

**Caution 2.** When the ADCR register is accessed in 16-bit units, the higher 10 bits of the conversion result are read in order starting at bit 15.

#### 14.3.6 8-bit A/D conversion result register (ADCRH)

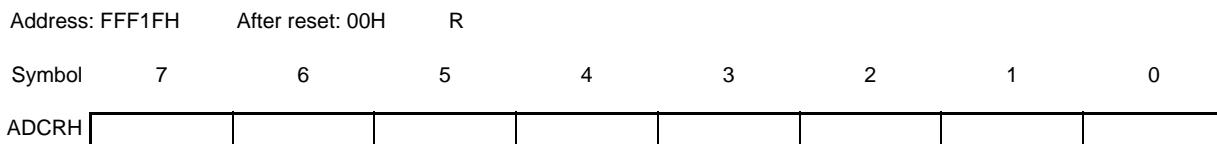
This register is an 8-bit register that stores the A/D conversion result. The higher 8 bits of 10-bit resolution are stored. Note

The ADCRH register can be read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Note** If the A/D conversion result is outside the range specified by using the A/D conversion comparison function (the value specified by the ADRCK bit of the ADM2 register and ADUL/ADLL registers; see **Figure 14 - 9**), the result is not stored.

**Figure 14 - 11 Format of 8-bit A/D conversion result register (ADCRH)**



**Caution** When writing to the A/D converter mode register 0 (ADM0), Analog input channel specification register (ADS), and A/D port configuration register (ADPC), the contents of the ADCRH register may become undefined. Read the conversion result following conversion completion before writing to the ADM0, ADS, and ADPC registers. Using timing other than the above may cause an incorrect conversion result to be read.

### 14.3.7 Analog input channel specification register (ADS)

This register specifies the input channel of the analog voltage to be A/D converted.

The ADS register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 12 Format of Analog input channel specification register (ADS) (1/2)**

Address: FFF31H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADS	ADISS	0	0	ADS4	ADS3	ADS2	ADS1	ADS0

- Select mode (ADMD = 0)

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	Analog input channel	Input source
0	0	0	0	0	0	ANIO	P20/ANI0/AVREFP pin
0	0	0	0	0	1	ANI1	P21/ANI1/AVREFM pin
0	0	0	0	1	0	ANI2	P22/ANI2 pin
0	0	0	0	1	1	ANI3	P23/ANI3 pin
0	0	0	1	0	0	ANI4	P24/ANI4 pin
0	0	0	1	0	1	ANI5	P25/ANI5 pin
0	0	0	1	1	0	ANI6	P26/ANI6 pin
0	0	0	1	1	1	ANI7	P27/ANI7 pin
0	0	1	0	0	0	ANI8	P150/ANI8 pin
0	0	1	0	0	1	ANI9	P151/ANI9 pin
0	0	1	0	1	0	ANI10	P152/ANI10 pin
0	0	1	0	1	1	ANI11	P153/ANI11 pin
0	0	1	1	0	0	ANI12	P154/ANI12 pin
0	0	1	1	0	1	ANI13	P155/ANI13 pin
0	0	1	1	1	0	ANI14	P156/ANI14 pin
0	1	0	0	0	0	ANI16	P03/ANI16 pin Note 1
0	1	0	0	0	1	ANI17	P02/ANI17 pin Note 2
0	1	0	0	1	0	ANI18	P147/ANI18 pin
0	1	0	0	1	1	ANI19	P120/ANI19 pin
0	1	0	1	0	0	ANI20	P100/ANI20 pin
1	0	0	0	0	0	—	Temperature sensor output voltage Notes 3, 4
1	0	0	0	0	1	—	Internal reference voltage (1.45 V) Note 4
Other than the above						Setting prohibited	

**Note 1.** 30-, 32-pin products: P01/ANI16 pin

**Note 2.** 30-, 32-pin products: P00/ANI17 pin

**Note 3.** If the internal reference voltage (1.45 V) is selected as the reference voltage of comparator 0 or comparator 1 in products with 96 KB or more code flash memory, the temperature sensor output cannot be selected.

**Note 4.** Operation is possible only in HS (high-speed main) mode.

(Cautions are listed on the next page.)

**Figure 14 - 13 Format of Analog input channel specification register (ADS) (2/2)**

Address: FFF31H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
ADS	ADISS	0	0	ADS4	ADS3	ADS2	ADS1	ADS0

- Scan mode (ADMD = 1)

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	Analog input channel			
						Scan 0	Scan 1	Scan 2	Scan 3
0	0	0	0	0	0	ANI0	ANI1	ANI2	ANI3
0	0	0	0	0	1	ANI1	ANI2	ANI3	ANI4
0	0	0	0	1	0	ANI2	ANI3	ANI4	ANI5
0	0	0	0	1	1	ANI3	ANI4	ANI5	ANI6
0	0	0	1	0	0	ANI4	ANI5	ANI6	ANI7
0	0	0	1	0	1	ANI5	ANI6	ANI7	ANI8
0	0	0	1	1	0	ANI6	ANI7	ANI8	ANI9
0	0	0	1	1	1	ANI7	ANI8	ANI9	ANI10
0	0	1	0	0	0	ANI8	ANI9	ANI10	ANI11
0	0	1	0	0	1	ANI9	ANI10	ANI11	ANI12
0	0	1	0	1	0	ANI10	ANI11	ANI12	ANI13
0	0	1	0	1	1	ANI11	ANI12	ANI13	ANI14
Other than the above						Setting prohibited			

**Caution 1.** Be sure to clear bits 5 and 6 to 0.

**Caution 2.** Set a channel to be set the analog input by ADPC and PMCx registers in the input mode by using port mode registers 0, 2, 10, 12, 14, and 15 (PM0, PM2, PM10, PM12, PM14, PM15).

**Caution 3.** Do not set the pin that is set by the A/D port configuration register (ADPC) as digital I/O by the ADS register.

**Caution 4.** Do not set the pin that is set by Port mode control registers 0, 10, 12, and 14 (PMC0, PMC10, PMC12, PMC14) as digital I/O by the ADS register.

**Caution 5.** Rewrite the value of the ADISS bit while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 6.** If using AVREFP as the + side reference voltage of the A/D converter, do not select ANI0 as an A/D conversion channel.

**Caution 7.** If using AVREFM as the – side reference voltage of the A/D converter, do not select ANI1 as an A/D conversion channel.

**Caution 8.** If the ADISS bit is set to 1, the internal reference voltage (1.45 V) cannot be used for the + side reference voltage. After the ADISS bit is set to 1, the initial conversion result cannot be used.

For the setting flow, see 14.7.4 Setup when temperature sensor output voltage/internal reference voltage is selected (example for software trigger mode and one-shot conversion mode).

**Caution 9.** Do not set the ADISS bit to 1 when shifting to STOP mode, or to HALT mode while the CPU is operating on the subsystem clock. When the ADISS bit is set to 1, the A/D converter reference voltage current (IADREF) indicated in 34.3.2 or 35.3.2 Supply current characteristics will be added.

### 14.3.8 Conversion result comparison upper limit setting register (ADUL)

This register is used to specify the setting for checking the upper limit of the A/D conversion results.

The A/D conversion results and ADUL register value are compared, and interrupt signal (INTAD) generation is controlled in the range specified for the ADRCK bit of A/D converter mode register 2 (ADM2) (shown in **Figure 14 - 9**).

The ADUL register can be set by an 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Caution 1.** When 10-bit resolution A/D conversion is selected, the higher eight bits of the 10-bit A/D conversion result register (ADCR) are compared with the ADUL and ADLL registers.

**Caution 2.** Only write new values to the ADUL and ADLL registers while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 3.** The setting of the ADUL and ADLL registers must be greater than that of the ADLL register.

**Figure 14 - 14 Format of Conversion result comparison upper limit setting register (ADUL)**

Address: F0011H      After reset: FFH      R/W

Symbol	7	6	5	4	3	2	1	0
ADUL	ADUL7	ADUL6	ADUL5	ADUL4	ADUL3	ADUL2	ADUL1	ADUL0

### 14.3.9 Conversion result comparison lower limit setting register (ADLL)

This register is used to specify the setting for checking the lower limit of the A/D conversion results.

The A/D conversion results and ADLL register value are compared, and interrupt signal (INTAD) generation is controlled in the range specified for the ADRCK bit of A/D converter mode register 2 (ADM2) (shown in **Figure 14 - 9**).

The ADLL register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 15 Format of Conversion result comparison lower limit setting register (ADLL)**

Address: F0012H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADLL	ADLL7	ADLL6	ADLL5	ADLL4	ADLL3	ADLL2	ADLL1	ADLL0

**Caution 1.** When 10-bit resolution A/D conversion is selected, the higher eight bits of the 10-bit A/D conversion result register (ADCR) are compared with the ADUL and ADLL registers.

**Caution 2.** Only write new values to the ADUL and ADLL registers while conversion is stopped (ADCS = 0, ADCE = 0).

**Caution 3.** The setting of the ADUL and ADLL registers must be greater than that of the ADLL register.

### 14.3.10 A/D test register (ADTES)

This register is used to select the + side reference voltage or - side reference voltage for the converter, an analog input channel (ANIx<sub>x</sub>), the temperature sensor output voltage, or the internal reference voltage (1.45 V) as the target for A/D conversion.

When using this register to test the converter, set as follows.

- For zero-scale measurement, select the - side reference voltage as the target for conversion.
- For full-scale measurement, select the + side reference voltage as the target for conversion.

The ADTES register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 14 - 16 Format of A/D test register (ADTES)**

Address: F0013H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADTES	0	0	0	0	0	0	ADTES1	ADTES0

		A/D conversion target		
ADTES1	ADTES0			
0	0	ANIx <sub>x</sub> /temperature sensor output voltage Note/internal reference voltage (1.45 V) Note (This is specified using the analog input channel specification register (ADS).)		
1	0	The - side reference voltage (selected by the ADREFM bit of the ADM2 register)		
1	1	The + side reference voltage (selected by the ADREFP1 or ADREFP0 bit of the ADM2 register)		
Other than the above		Setting prohibited		

**Note**      The temperature sensor output voltage and internal reference voltage (1.45 V) can be selected only in the HS (high-speed main) mode.

### 14.3.11 Registers controlling port function of analog input pins

Set up the registers for controlling the functions of the ports shared with the analog input pins of the A/D converter (port mode registers (PMxx), port mode control registers (PMCxx), and A/D port configuration register (ADPC)). For details, see **4.3.1 Port mode registers (PMxx)**, **4.3.6 Port mode control registers (PMCxx)**, and **4.3.7 A/D port configuration register (ADPC)**.

When using the ANI0 to ANI14 pins for analog input of the A/D converter, set the port mode register (PMxx) bit corresponding to each port to 1 and select analog input through the A/D port configuration register (ADPC).

When using the ANI16 to ANI20 pins for analog input of the A/D converter, set the port mode register (PMxx) bit and port mode control register (PMCxx) bit corresponding to each port to 1.

## 14.4 A/D Converter Conversion Operations

The A/D converter conversion operations are described below.

<1> The voltage input to the selected analog input channel is sampled by the sample & hold circuit.

<2> When sampling has been done for a certain time, the sample & hold circuit is placed in the hold state and the sampled voltage is held until the A/D conversion operation has ended.

<3> Bit 9 of the successive approximation register (SAR) is set. The series resistor string voltage tap is set to (1/2) AVREF by the tap selector.

<4> The voltage difference between the series resistor string voltage tap and sampled voltage is compared by the voltage comparator. If the analog input is greater than (1/2) AVREF, the MSB bit of the SAR register remains set to 1. If the analog input is smaller than (1/2) AVREF, the MSB bit is reset to 0.

<5> Next, bit 8 of the SAR register is automatically set to 1, and the operation proceeds to the next comparison.

The series resistor string voltage tap is selected according to the preset value of bit 9, as described below.

- Bit 9 = 1: (3/4) AVREF
- Bit 9 = 0: (1/4) AVREF

The voltage tap and sampled voltage are compared and bit 8 of the SAR register is manipulated as follows.

- Sampled voltage  $\geq$  Voltage tap: Bit 8 = 1
- Sampled voltage < Voltage tap: Bit 8 = 0

<6> Comparison is continued in this way up to bit 0 of the SAR register.

<7> Upon completion of the comparison of 10 bits, an effective digital result value remains in the SAR register, and the result value is transferred to the A/D conversion result register (ADCR, ADCRH) and then latched Note 1.

At the same time, the A/D conversion end interrupt request (INTAD) can also be generated.

<8> Repeat steps <1> to <7>, until the ADCS bit is cleared to 0 Note 2.

To stop the A/D converter, clear the ADCS bit to 0.

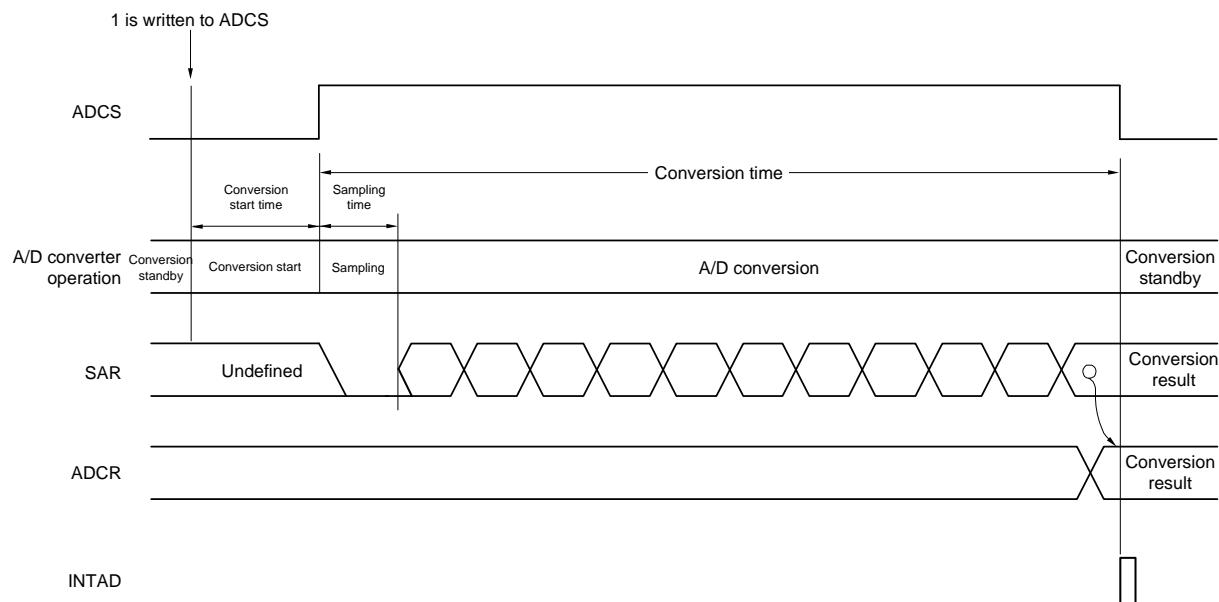
**Note 1.** If the A/D conversion result is outside the A/D conversion result range specified by the ADRCK bit and the ADUL and ADLL registers (see **Figure 14 - 9**), the A/D conversion result interrupt request signal is not generated and no A/D conversion results are stored in the ADCR and ADCRH registers.

**Note 2.** While in the sequential conversion mode, the ADCS flag is not automatically cleared to 0. This flag is not automatically cleared to 0 while in the one-shot conversion mode of the hardware trigger no-wait mode, either. Instead, 1 is retained.

**Remark 1.** Two types of the A/D conversion result registers are available.

- ADCR register (16 bits): Store 10-bit A/D conversion value
- ADCRH register (8 bits): Store 8-bit A/D conversion value

**Remark 2.** AVREF: The + side reference voltage of the A/D converter. This can be selected from AVREFP, the internal reference voltage (1.45 V), and VDD.

**Figure 14 - 17 Conversion Operation of A/D Converter (Software Trigger Mode)**

In one-shot conversion mode, the ADCS bit is automatically cleared to 0 after completion of A/D conversion.

In sequential conversion mode, A/D conversion operations proceed continuously until the software clears bit 7 (ADCS) of the A/D converter mode register 0 (ADM0) to 0.

Writing to the analog input channel specification register (ADS) during A/D conversion interrupts the current conversion after which A/D conversion of the analog input specified by the ADS register proceeds. Data from the A/D conversion that was in progress are discarded.

Reset signal generation clears the A/D conversion result register (ADCR, ADCRH) to 0000H or 00H.

## 14.5 Input Voltage and Conversion Results

The relationship between the analog input voltage input to the analog input pins (ANI0 to ANI14, ANI16 to ANI20) and the theoretical A/D conversion result (stored in the 10-bit A/D conversion result register (ADCR)) is shown by the following expression.

$$\text{SAR} = \text{INT} \left( \frac{V_{\text{AIN}}}{AV_{\text{REF}}} \times 1024 + 0.5 \right)$$

$$\text{ADCR} = \text{SAR} \times 64$$

or

$$\left( \frac{\text{ADCR}}{64} - 0.5 \right) \times \frac{AV_{\text{REF}}}{1024} \leq V_{\text{AIN}} < \left( \frac{\text{ADCR}}{64} + 0.5 \right) \times \frac{AV_{\text{REF}}}{1024}$$

where, INT(): Function which returns integer part of value in parentheses

V<sub>AIN</sub>: Analog input voltage

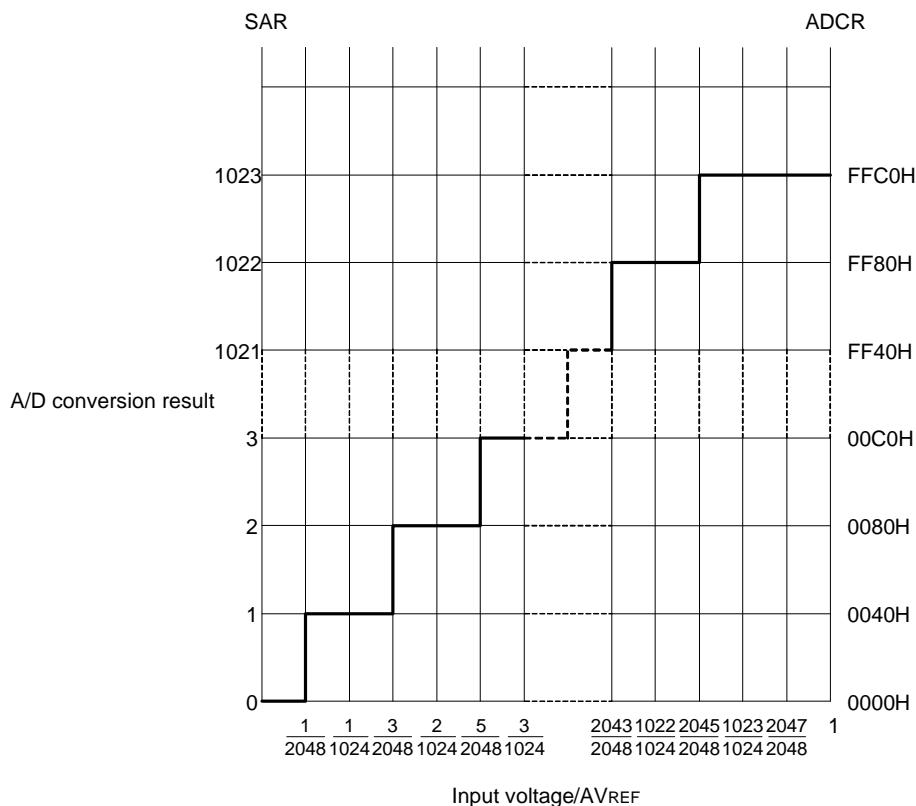
AV<sub>REF</sub>: AV<sub>REF</sub> pin voltage

ADCR: A/D conversion result register (ADCR) value

SAR: Successive approximation register

Figure 14 - 18 shows the Relationship Between Analog Input Voltage and A/D Conversion Result.

**Figure 14 - 18 Relationship Between Analog Input Voltage and A/D Conversion Result**



**Remark** AV<sub>REF</sub>: The + side reference voltage of the A/D converter. This can be selected from AV<sub>REFP</sub>, the internal reference voltage (1.45 V), and V<sub>DD</sub>.

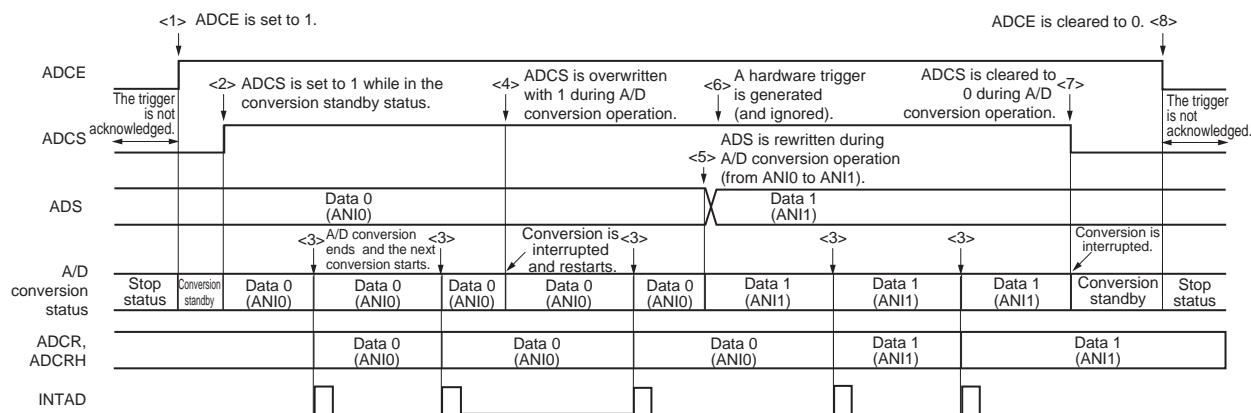
## 14.6 A/D Converter Operation Modes

The operation of each A/D converter mode is described below. In addition, the procedure for specifying each mode is described in **14.7 A/D Converter Setup Flowchart**.

### 14.6.1 Software trigger mode (select mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to perform the A/D conversion of the analog input specified by the analog input channel specification register (ADS).
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion ends, the next A/D conversion immediately starts.
- <4> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <6> Even if a hardware trigger is input during conversion operation, A/D conversion does not start.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status.
- <8> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start.

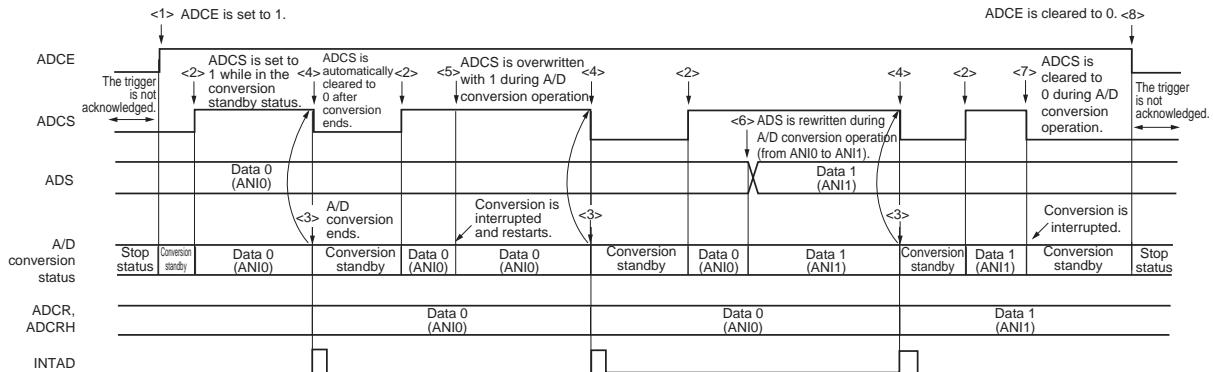
**Figure 14 - 19 Example of Software Trigger Mode (Select Mode, Sequential Conversion Mode) Operation Timing**



### 14.6.2 Software trigger mode (select mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to perform the A/D conversion of the analog input specified by the analog input channel specification register (ADS).
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated.
- <4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the system enters the A/D conversion standby status.
- <5> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status.
- <8> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start. In addition, A/D conversion does not start even if a hardware trigger is input while in the A/D conversion standby status.

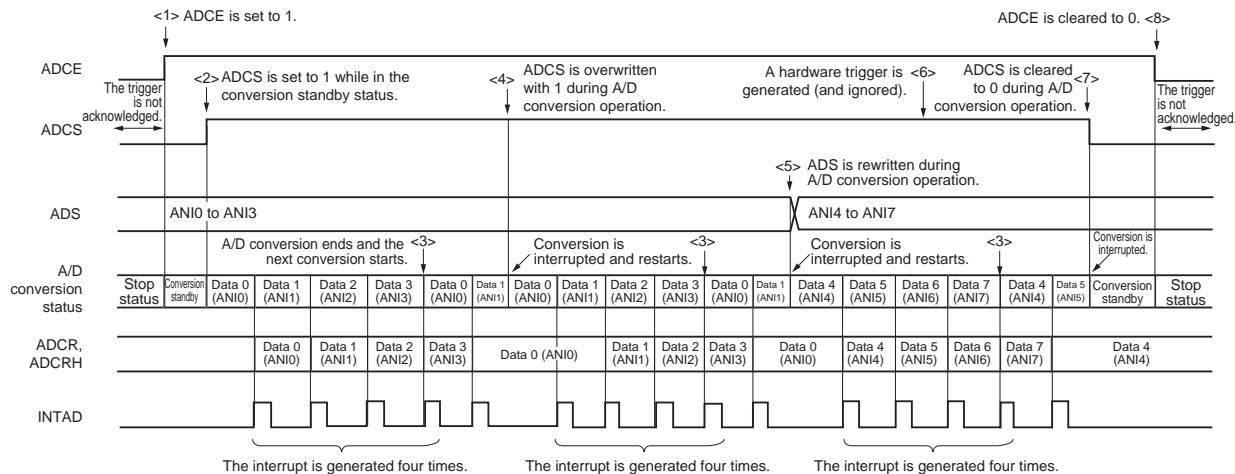
**Figure 14 - 20 Example of Software Trigger Mode (Select Mode, One-Shot Conversion Mode) Operation Timing**



### 14.6.3 Software trigger mode (scan mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to perform A/D conversion on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <3> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion of the four channels ends, the A/D conversion of the channel following the specified channel automatically starts (until all four channels are finished).
- <4> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <6> Even if a hardware trigger is input during conversion operation, A/D conversion does not start.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status.
- <8> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start.

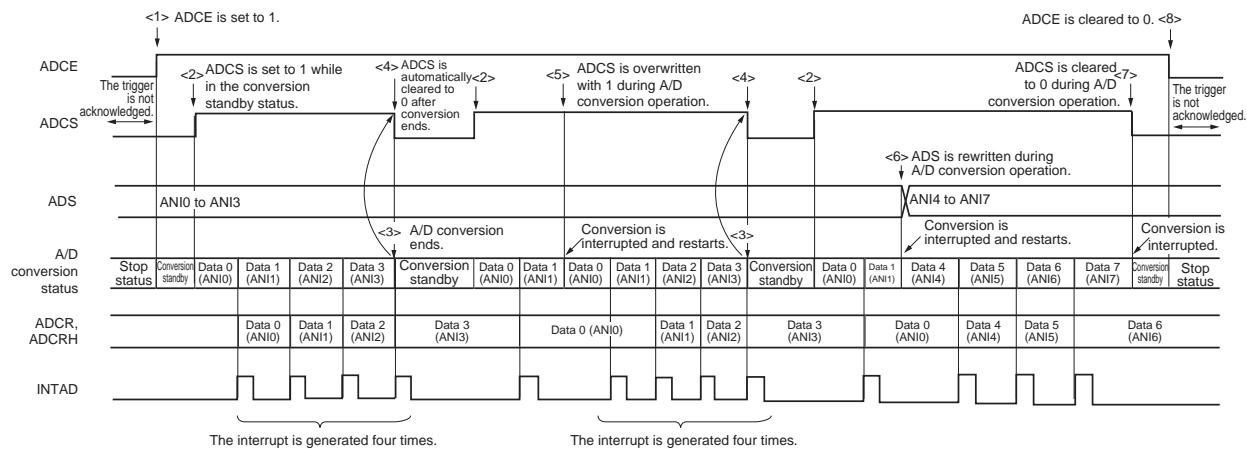
**Figure 14 - 21 Example of Software Trigger Mode (Scan Mode, Sequential Conversion Mode) Operation Timing**



#### 14.6.4 Software trigger mode (scan mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to perform A/D conversion on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <3> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated.
- <4> After A/D conversion of the four channels ends, the ADCS bit is automatically cleared to 0, and the system enters the A/D conversion standby status.
- <5> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status.
- <8> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start. In addition, A/D conversion does not start even if a hardware trigger is input while in the A/D conversion standby status.

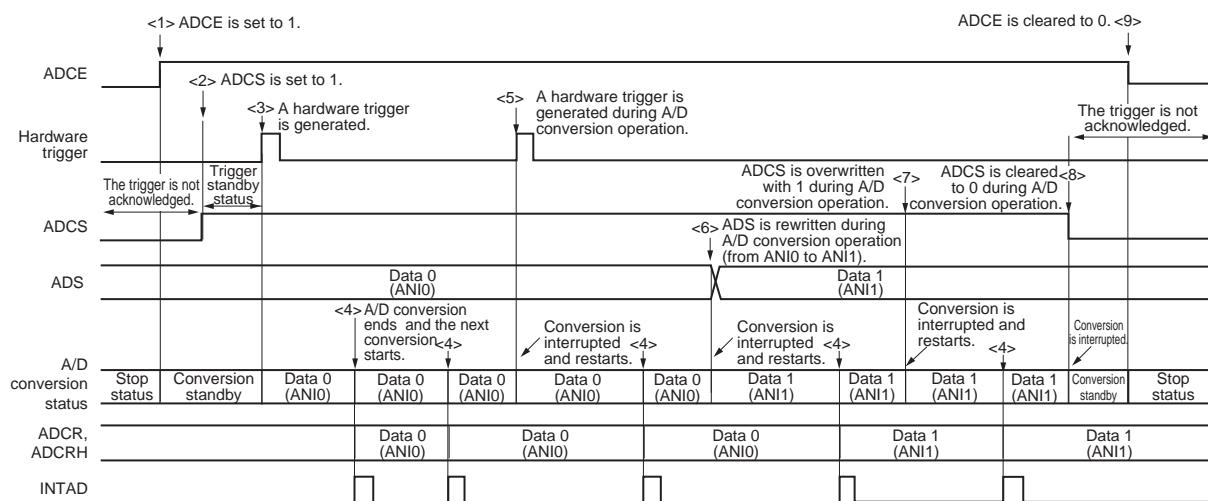
**Figure 14 - 22 Example of Software Trigger Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing**



### 14.6.5 Hardware trigger no-wait mode (select mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time ( $1 \mu\text{s}$ ), the ADCS bit of the ADM0 register is set to 1 to place the system in the hardware trigger standby status (and conversion does not start at this stage). Note that, while in this status, A/D conversion does not start even if ADCS is set to 1.
- <3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS).
- <4> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion ends, the next A/D conversion immediately starts.
- <5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status. However, the A/D converter does not stop in this status.
- <9> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCS = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

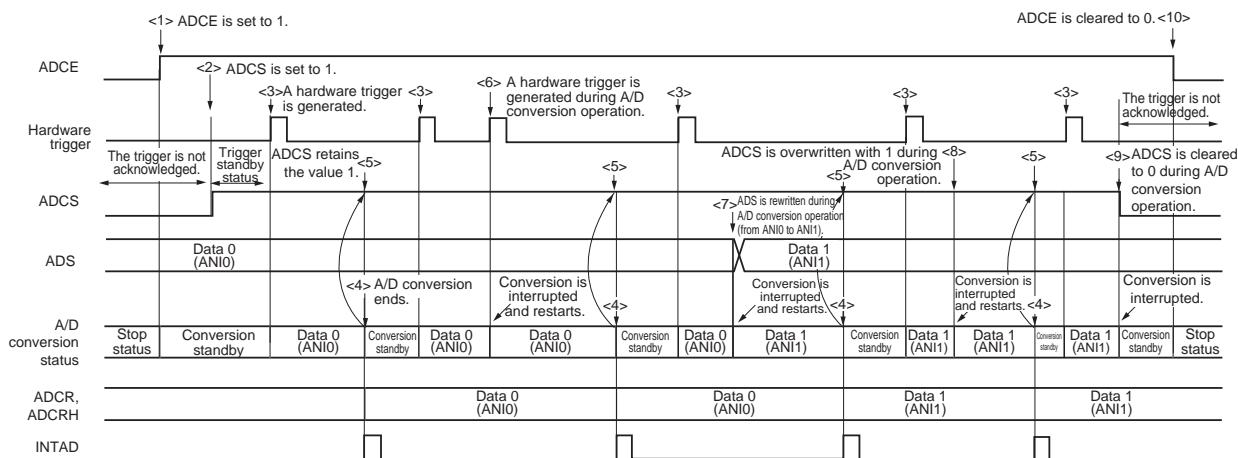
**Figure 14 - 23 Example of Hardware Trigger No-Wait Mode (Select Mode, Sequential Conversion Mode) Operation Timing**



### 14.6.6 Hardware trigger no-wait mode (select mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to place the system in the hardware trigger standby status (and conversion does not start at this stage). Note that, while in this status, A/D conversion does not start even if ADCS is set to 1.
- <3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS).
- <4> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated.
- <5> After A/D conversion ends, the ADCS bit remains set to 1, and the system enters the A/D conversion standby status.
- <6> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <7> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <8> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <9> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status. However, the A/D converter does not stop in this status.
- <10> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCS = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

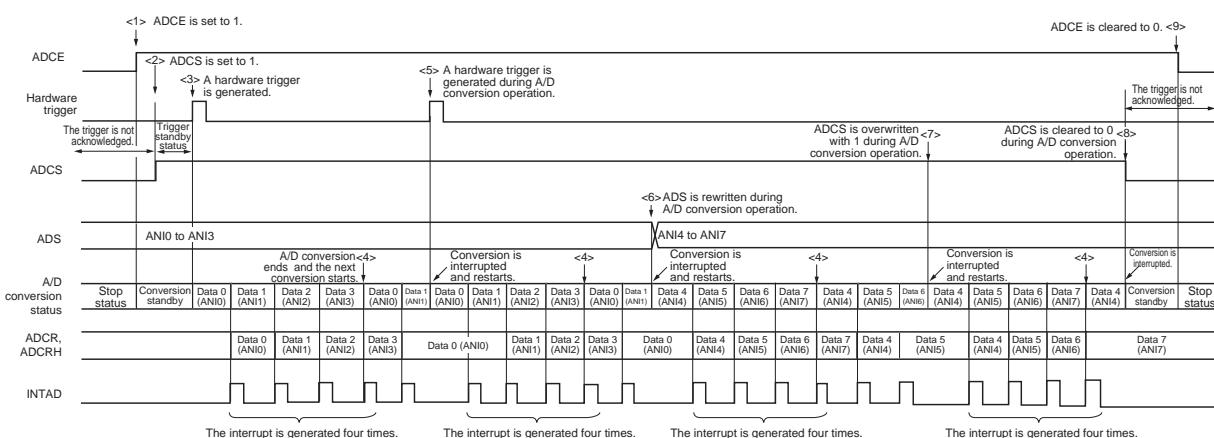
**Figure 14 - 24 Example of Hardware Trigger No-Wait Mode (Select Mode, One-Shot Conversion Mode) Operation Timing**



### 14.6.7 Hardware trigger no-wait mode (scan mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time (1  $\mu$ s), the ADCS bit of the ADM0 register is set to 1 to place the system in the hardware trigger standby status (and conversion does not start at this stage). Note that, while in this status, A/D conversion does not start even if ADCS is set to 1.
- <3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <4> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion of the four channels ends, the A/D conversion of the channel following the specified channel automatically starts.
- <5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status. However, the A/D converter does not stop in this status.
- <9> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCE = 0, specifying 1 for ADCS is ignored and A/D conversion does not start.

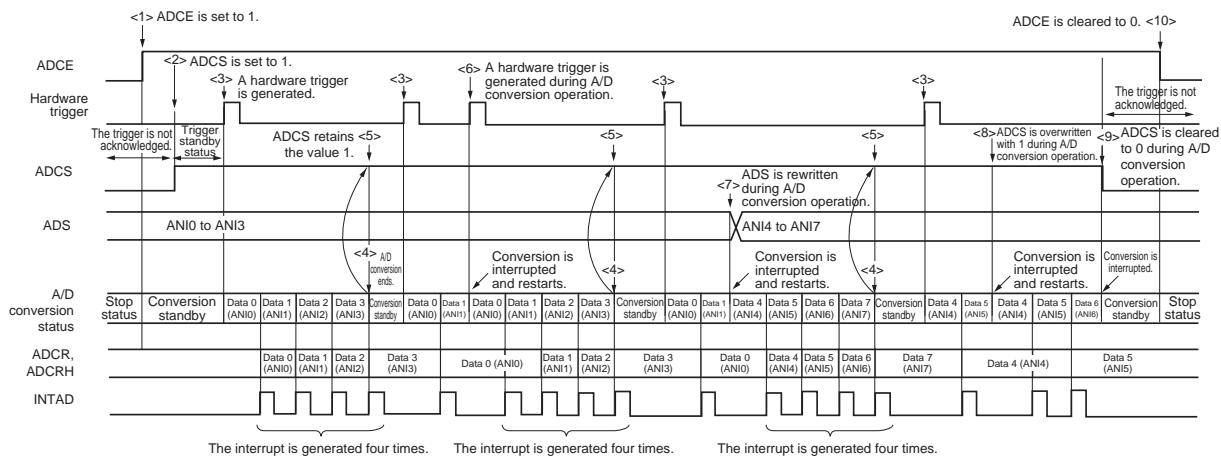
**Figure 14 - 25 Example of Hardware Trigger No-Wait Mode (Scan Mode, Sequential Conversion Mode) Operation Timing**



### 14.6.8 Hardware trigger no-wait mode (scan mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> After the software counts up to the stabilization wait time ( $1 \mu\text{s}$ ), the ADCS bit of the ADM0 register is set to 1 to place the system in the hardware trigger standby status (and conversion does not start at this stage). Note that, while in this status, A/D conversion does not start even if ADCS is set to 1.
- <3> If a hardware trigger is input while ADCS = 1, A/D conversion is performed on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <4> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated.
- <5> After A/D conversion of the four channels ends, the ADCS bit remains set to 1, and the system enters the A/D conversion standby status.
- <6> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <7> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <8> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <9> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, and the system enters the A/D conversion standby status. However, the A/D converter does not stop in this status.
- <10> When ADCE is cleared to 0 while in the A/D conversion standby status, the A/D converter enters the stop status. When ADCS = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

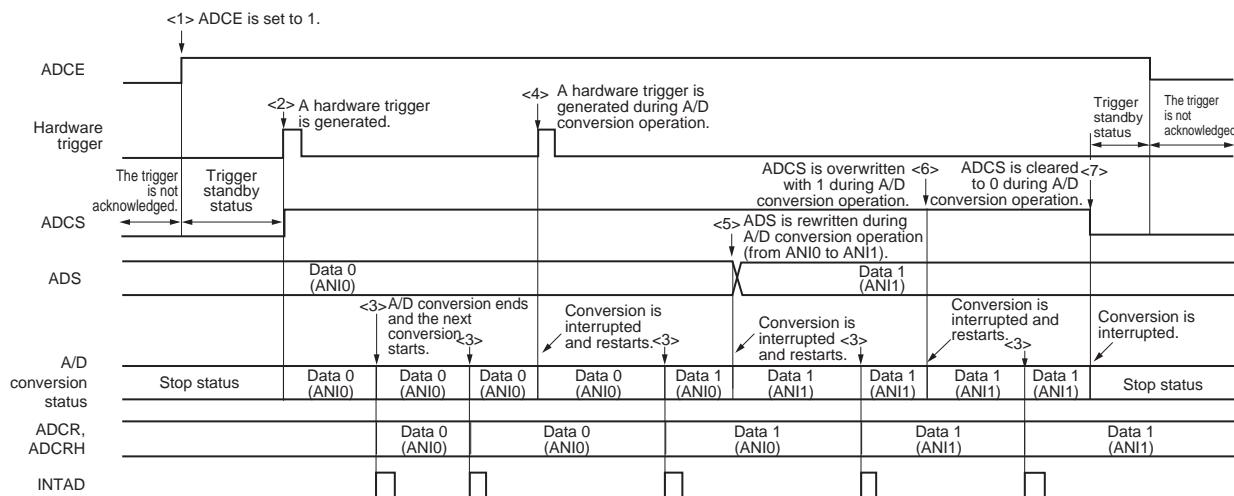
**Figure 14 - 26 Example of Hardware Trigger No-Wait Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing**



### 14.6.9 Hardware trigger wait mode (select mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the hardware trigger standby status.
- <2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input.
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion ends, the next A/D conversion immediately starts. (At this time, no hardware trigger is necessary.)
- <4> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <6> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

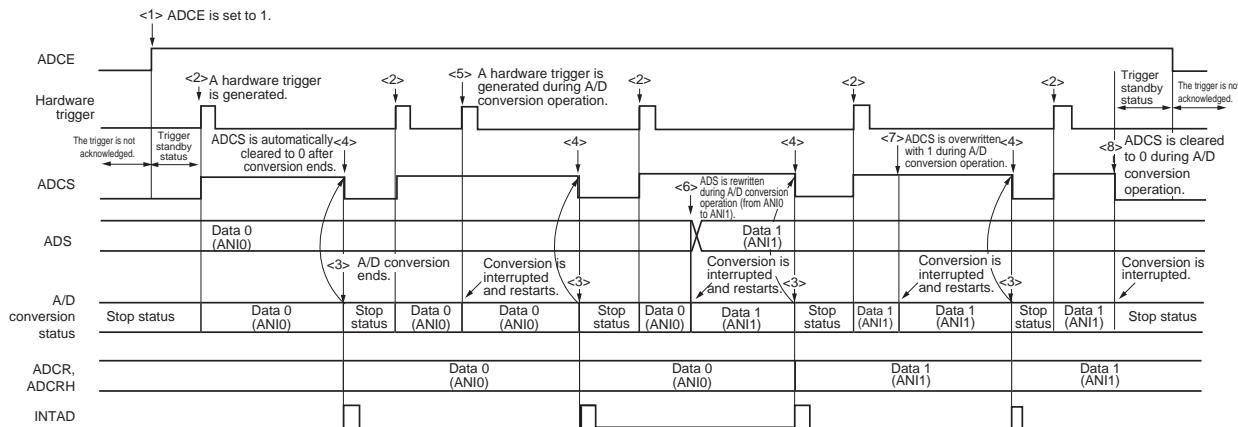
**Figure 14 - 27 Example of Hardware Trigger Wait Mode (Select Mode, Sequential Conversion Mode) Operation Timing**



### 14.6.10 Hardware trigger wait mode (select mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the hardware trigger standby status.
- <2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the analog input specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input.
- <3> When A/D conversion ends, the conversion result is stored in the A/D conversion result register (ADCR, ADCRH), and the A/D conversion end interrupt request signal (INTAD) is generated.
- <4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the A/D converter enters the stop status.
- <5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the analog input respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is initialized.
- <8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

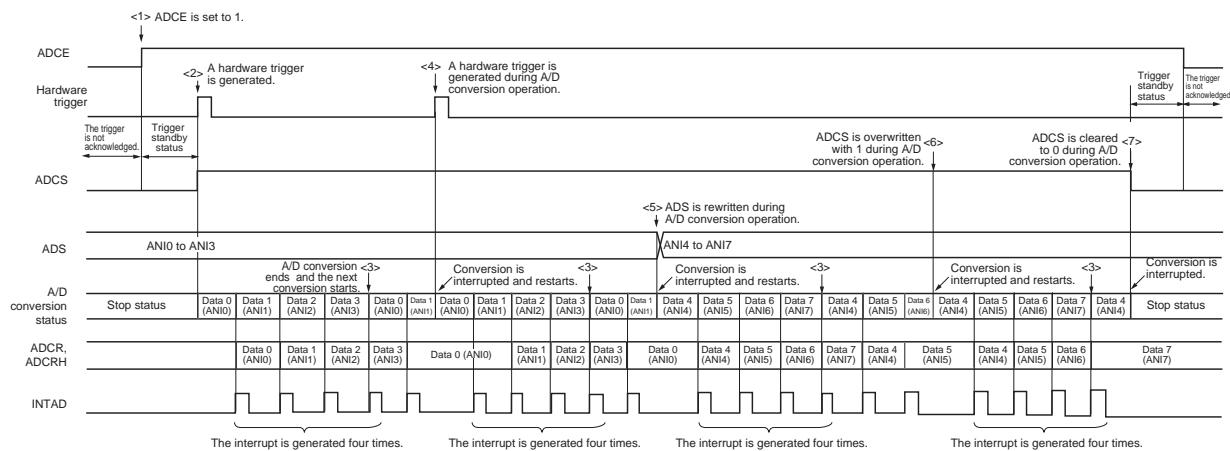
**Figure 14 - 28 Example of Hardware Trigger Wait Mode (Select Mode, One-Shot Conversion Mode) Operation Timing**



### 14.6.11 Hardware trigger wait mode (scan mode, sequential conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input. A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <3> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated. After A/D conversion of the four channels ends, the A/D conversion of the channel following the specified channel automatically starts.
- <4> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <5> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <6> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <7> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

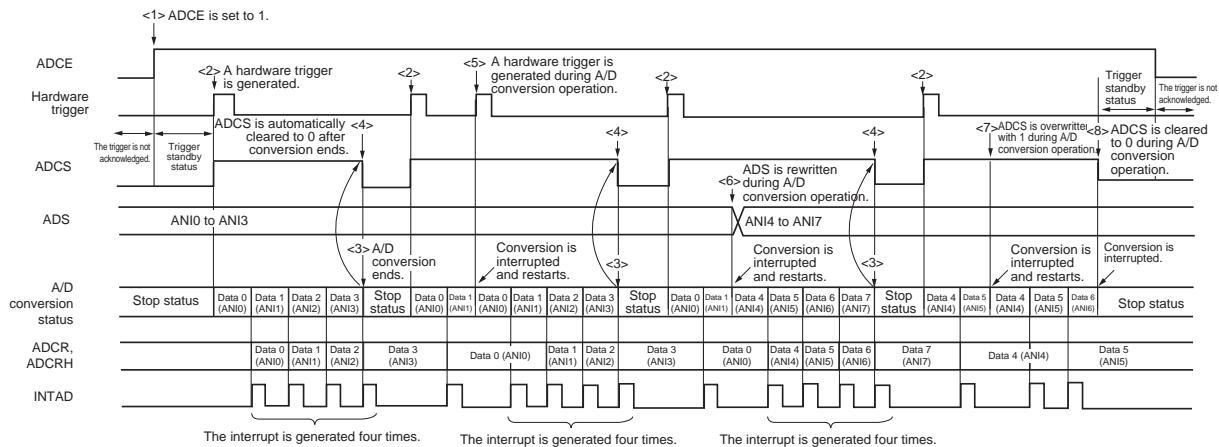
**Figure 14 - 29 Example of Hardware Trigger Wait Mode (Scan Mode, Sequential Conversion Mode) Operation Timing**



### 14.6.12 Hardware trigger wait mode (scan mode, one-shot conversion mode)

- <1> In the stop status, the ADCE bit of A/D converter mode register 0 (ADM0) is set to 1, and the system enters the A/D conversion standby status.
- <2> If a hardware trigger is input while in the hardware trigger standby status, A/D conversion is performed on the four analog input channels specified by scan 0 to scan 3, which are specified by the analog input channel specification register (ADS). The ADCS bit of the ADM0 register is automatically set to 1 according to the hardware trigger input. A/D conversion is performed on the analog input channels in order, starting with that specified by scan 0.
- <3> A/D conversion is sequentially performed on the four analog input channels, the conversion results are stored in the A/D conversion result register (ADCR, ADCRH) each time conversion ends, and the A/D conversion end interrupt request signal (INTAD) is generated.
- <4> After A/D conversion ends, the ADCS bit is automatically cleared to 0, and the A/D converter enters the stop status.
- <5> If a hardware trigger is input during conversion operation, the current A/D conversion is interrupted, and conversion restarts at the first channel. The partially converted data is discarded.
- <6> When the value of the ADS register is rewritten or overwritten during conversion operation, the current A/D conversion is interrupted, and A/D conversion is performed on the first channel respecified by the ADS register. The partially converted data is discarded.
- <7> When ADCS is overwritten with 1 during conversion operation, the current A/D conversion is interrupted, and conversion restarts. The partially converted data is discarded.
- <8> When ADCS is cleared to 0 during conversion operation, the current A/D conversion is interrupted, the system enters the hardware trigger standby status, and the A/D converter enters the stop status. When ADCE = 0, inputting a hardware trigger is ignored and A/D conversion does not start.

**Figure 14 - 30 Example of Hardware Trigger Wait Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing**

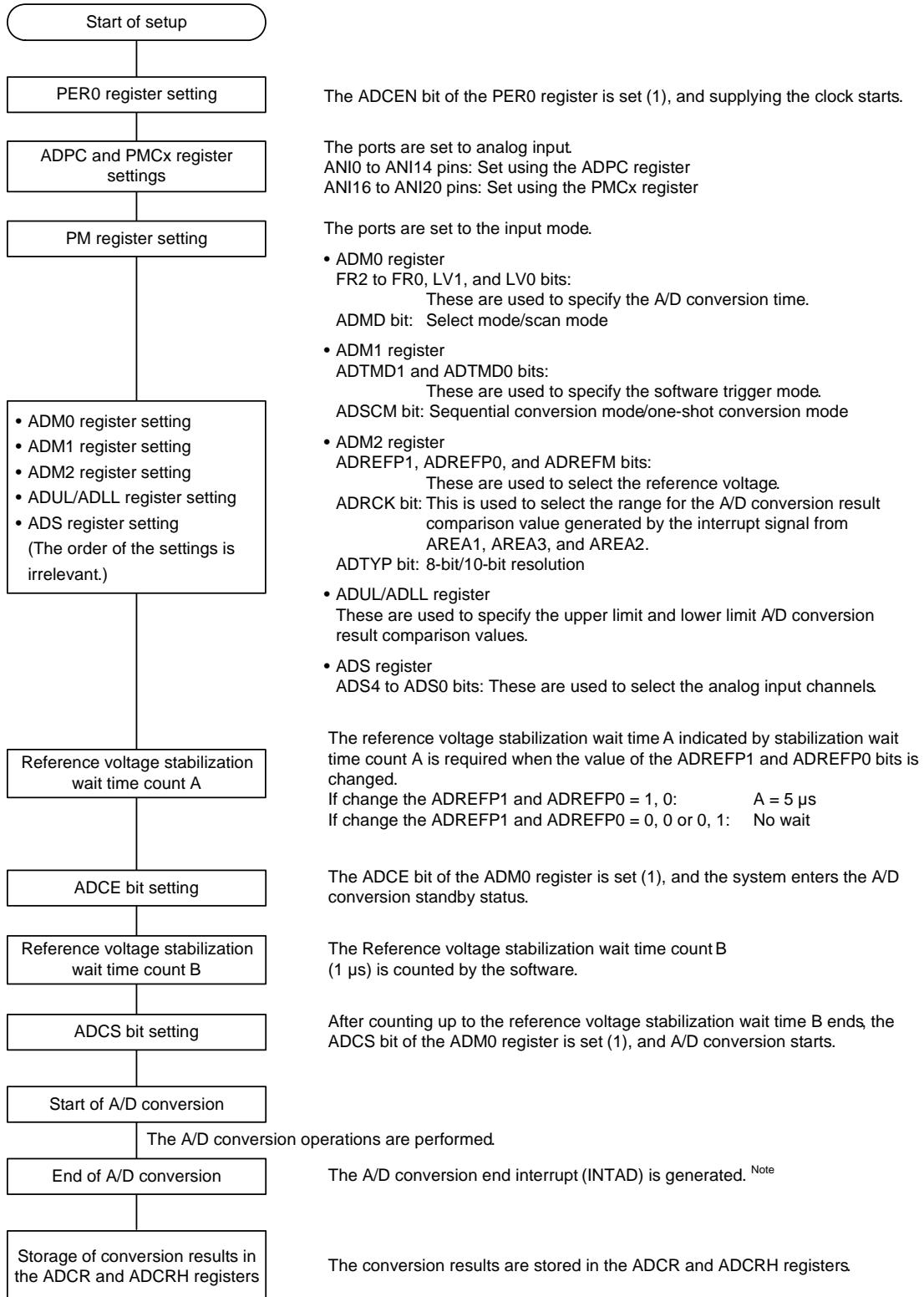


## 14.7 A/D Converter Setup Flowchart

The A/D converter setup flowchart in each operation mode is described below.

### 14.7.1 Setting up software trigger mode

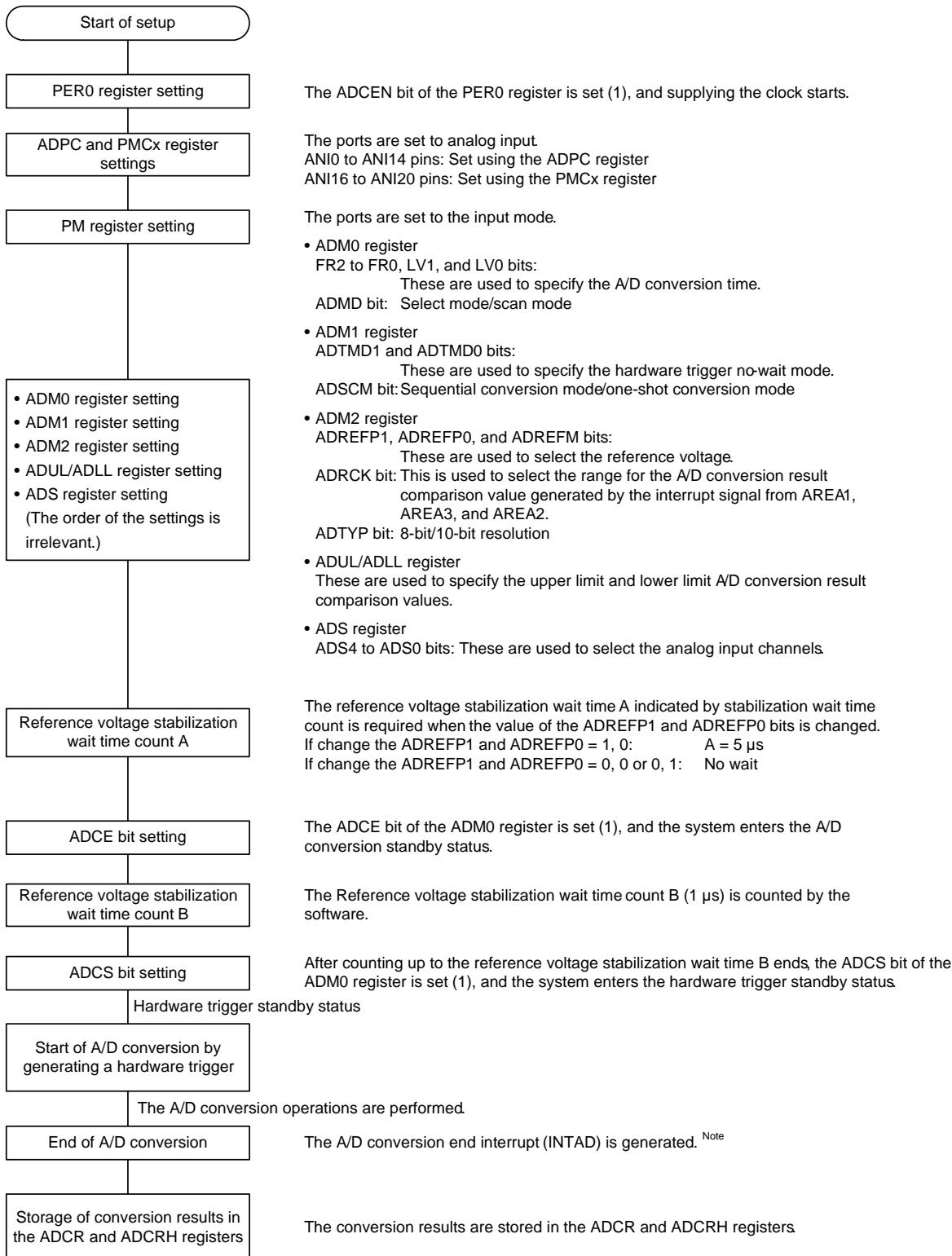
Figure 14 - 31 Setting up Software Trigger Mode



**Note** Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR, ADCRH register.

### 14.7.2 Setting up hardware trigger no-wait mode

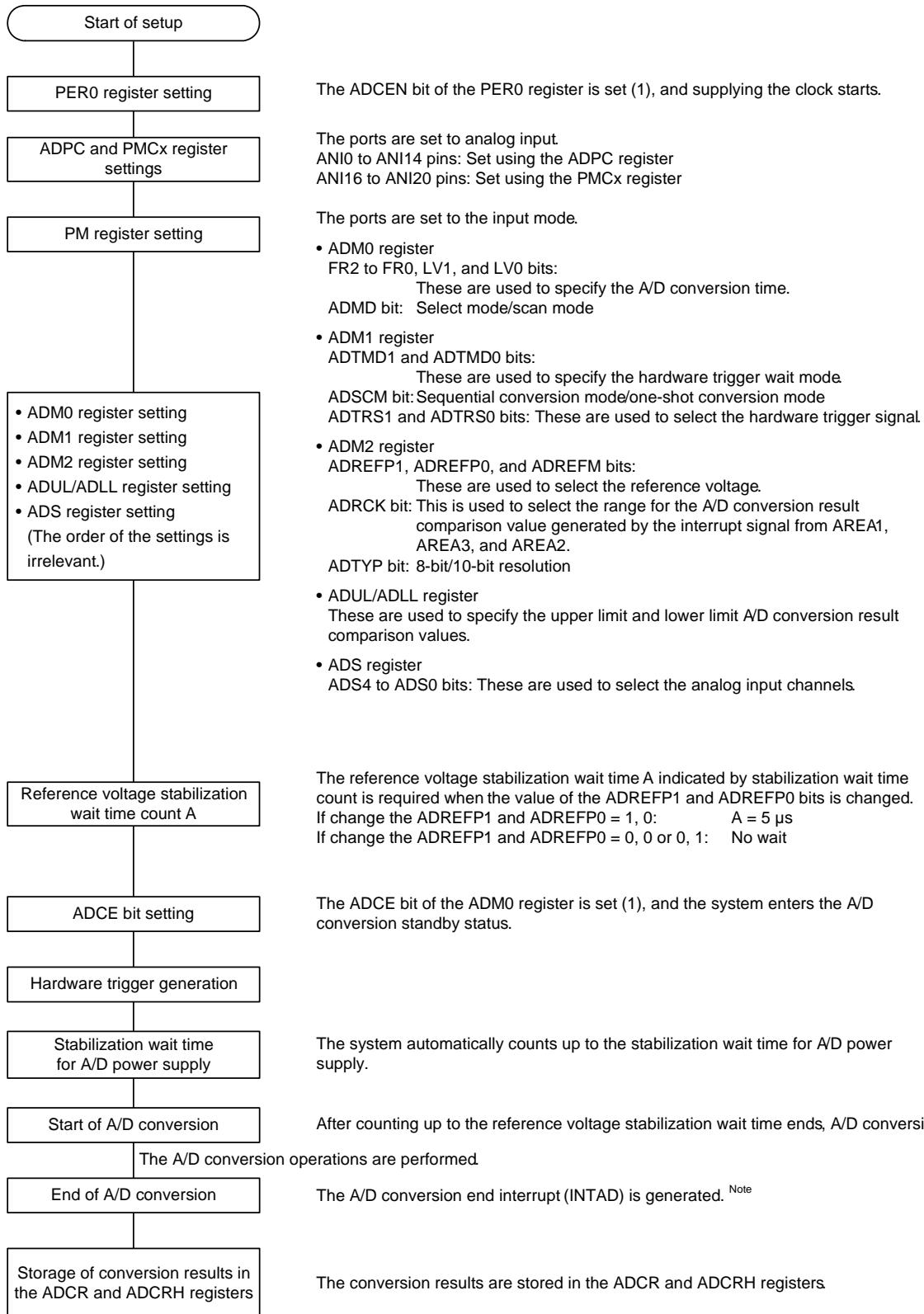
Figure 14 - 32 Setting up Hardware Trigger No-Wait Mode



**Note** Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR, ADCRH register.

### 14.7.3 Setting up hardware trigger wait mode

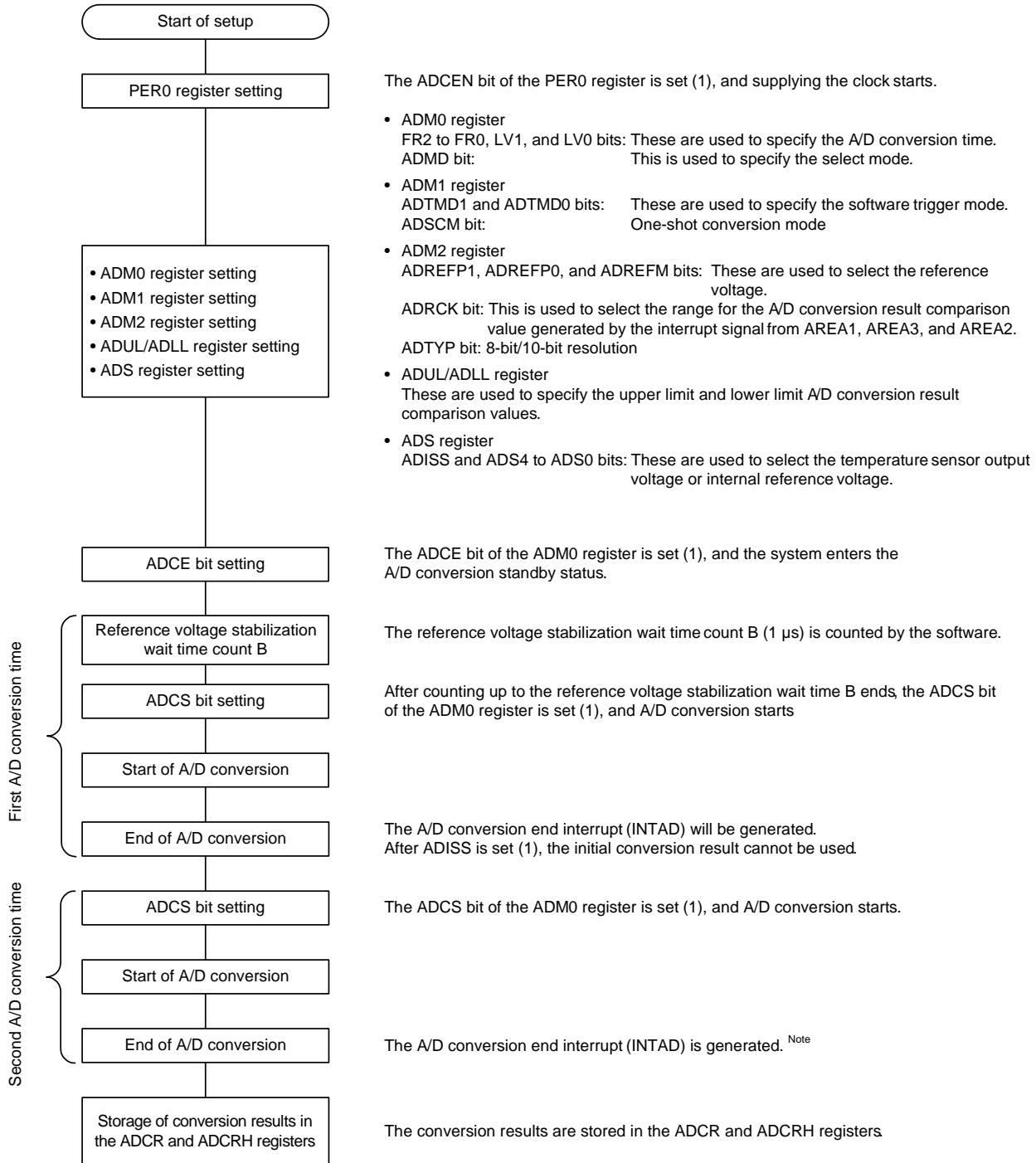
Figure 14 - 33 Setting up Hardware Trigger Wait Mode



**Note** Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR, ADCRH register.

#### 14.7.4 Setup when temperature sensor output voltage/internal reference voltage is selected (example for software trigger mode and one-shot conversion mode)

Figure 14 - 34 Setup when temperature sensor output voltage/internal reference voltage is selected

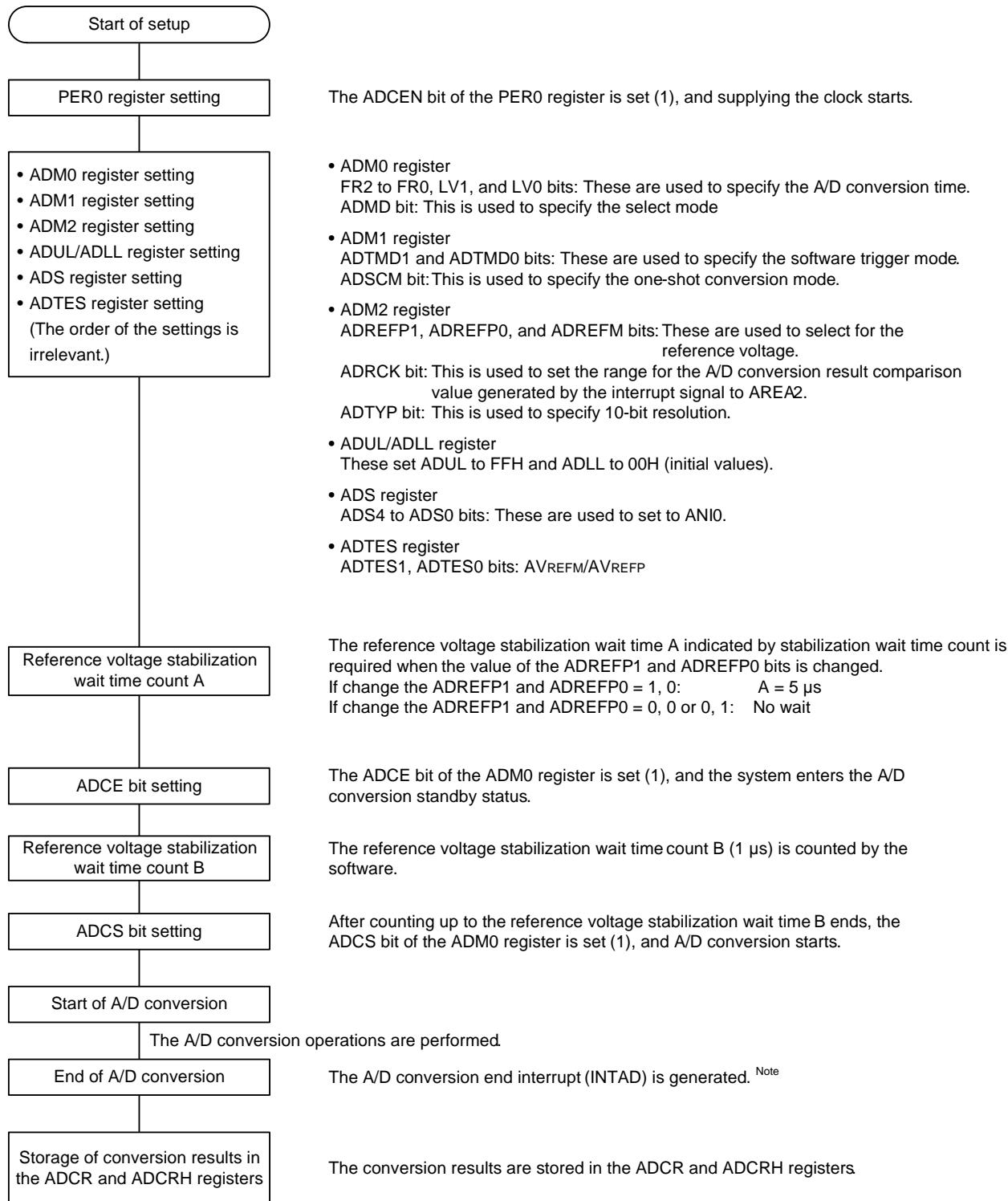


**Note** Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR, ADCRH register.

**Caution** This setting can be used only in HS (high-speed main) mode.

### 14.7.5 Setting up test mode

Figure 14 - 35 Setting up Test Mode



**Note** Depending on the settings of the ADRCK bit and ADUL/ADLL register, there is a possibility of no interrupt signal being generated. In this case, the results are not stored in the ADCR, ADCRH register.

**Caution** For the procedure for testing the A/D converter, see 27.3.8 A/D test function.

## 14.8 SNOOZE Mode Function

In the SNOOZE mode, A/D conversion is triggered by inputting a hardware trigger in the STOP mode. Normally, A/D conversion is stopped while in the STOP mode, but, by using the SNOOZE mode, A/D conversion can be performed without operating the CPU by inputting a hardware trigger. This is effective for reducing the operation current.

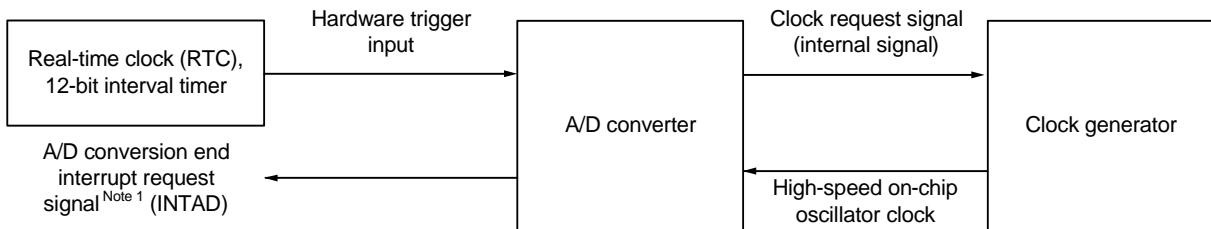
If the A/D conversion result range is specified using the ADUL and ADLL registers, A/D conversion results can be judged at a certain interval of time in SNOOZE mode. Using this function enables power supply voltage monitoring and input key judgment based on A/D inputs.

In the SNOOZE mode, only the following two conversion modes can be used:

- Hardware trigger wait mode (select mode, one-shot conversion mode)
- Hardware trigger wait mode (scan mode, one-shot conversion mode)

**Caution** That the SNOOZE mode can only be specified when the high-speed on-chip oscillator clock is selected for  $f_{CLK}$ .

Figure 14 - 36 Block Diagram When Using SNOOZE Mode Function



When using the SNOOZE mode function, the initial setting of each register is specified before switching to the STOP mode. (For details about these settings, see 14.7.3 Setting up hardware trigger wait mode Note 2.) At this time, bit 2 (AWC) of A/D converter mode register 2 (ADM2) is set to 1. After the initial settings are specified, bit 0 (ADCE) of A/D converter mode register 0 (ADM0) is set to 1.

If a hardware trigger is input after switching to the STOP mode, the high-speed on-chip oscillator clock is supplied to the A/D converter. After supplying this clock, the system automatically counts up to the A/D power supply stabilization wait time, and then A/D conversion starts.

The SNOOZE mode operation after A/D conversion ends differs depending on whether an interrupt signal is generated Note 1.

**Note 1.** Depending on the setting of the A/D conversion result comparison function (ADRCK bit, ADUL/ADLL register), there is a possibility of no interrupt signal being generated.

**Note 2.** Be sure to set the ADM1 register to E1H, E2H or E3H.

**Remark** The hardware trigger is event selected by ELC, INTRTC or INTIT.

Specify the hardware trigger by using the A/D Converter Mode Register 1 (ADM1).

- (1) If an interrupt is generated after A/D conversion ends

If the A/D conversion result value is inside the range of values specified by the A/D conversion result comparison function (which is set up by using the ADRCK bit and ADUL/ADLL register), the A/D conversion end interrupt request signal (INTAD) is generated.

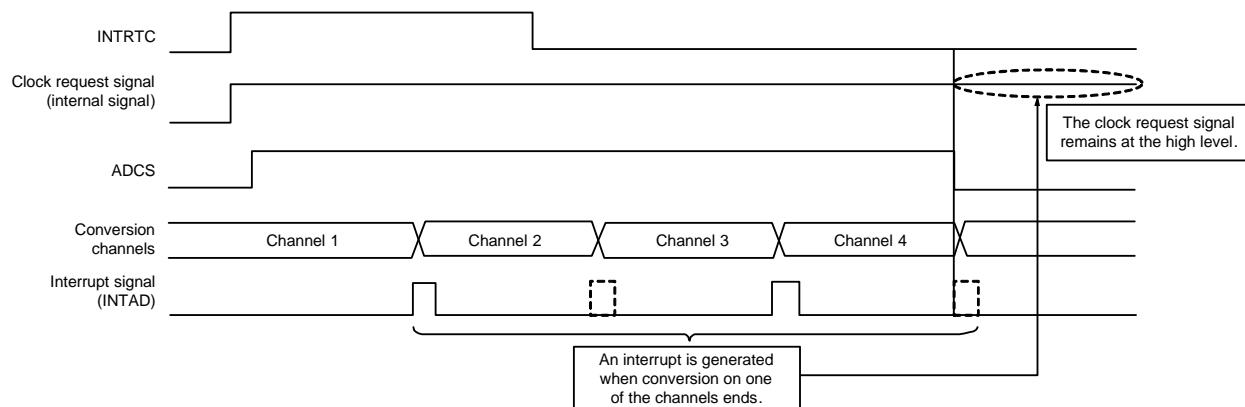
- While in the select mode

When A/D conversion ends and an A/D conversion end interrupt request signal (INTAD) is generated, the A/D converter returns to normal operation mode from SNOOZE mode. At this time, be sure to clear bit 2 (AWC = 0: SNOOZE mode release) of the A/D converter mode register 2 (ADM2). If the AWC bit is left set to 1, A/D conversion will not start normally in the subsequent SNOOZE or normal operation mode.

- While in the scan mode

If even one A/D conversion end interrupt request signal (INTAD) is generated during A/D conversion of the four channels, the clock request signal remains at the high level, and the A/D converter switches from the SNOOZE mode to the normal operation mode. At this time, be sure to clear bit 2 (AWC = 0: SNOOZE mode release) of A/D converter mode register 2 (ADM2) to 0. If the AWC bit is left set to 1, A/D conversion will not start normally in the subsequent SNOOZE or normal operation mode.

**Figure 14 - 37 Operation Example When Interrupt Is Generated After A/D Conversion Ends (While in Scan Mode)**



## (2) If no interrupt is generated after A/D conversion ends

If the A/D conversion result value is outside the range of values specified by the A/D conversion result comparison function (which is set up by using the ADRCK bit and ADUL/ADLL register), the A/D conversion end interrupt request signal (INTAD) is not generated.

## • While in the select mode

If the A/D conversion end interrupt request signal (INTAD) is not generated after A/D conversion ends, the clock request signal (an internal signal) is automatically set to the low level, and supplying the high-speed on-chip oscillator clock stops. If a hardware trigger is input later, A/D conversion work is again performed in the SNOOZE mode.

## • While in the scan mode

If the A/D conversion end interrupt request signal (INTAD) is not generated even once during A/D conversion of the four channels, the clock request signal (an internal signal) is automatically set to the low level after A/D conversion of the four channels ends, and supplying the high-speed on-chip oscillator clock stops. If a hardware trigger is input later, A/D conversion work is again performed in the SNOOZE mode.

**Figure 14 - 38 Operation Example When No Interrupt Is Generated After A/D Conversion Ends (While in Scan Mode)**

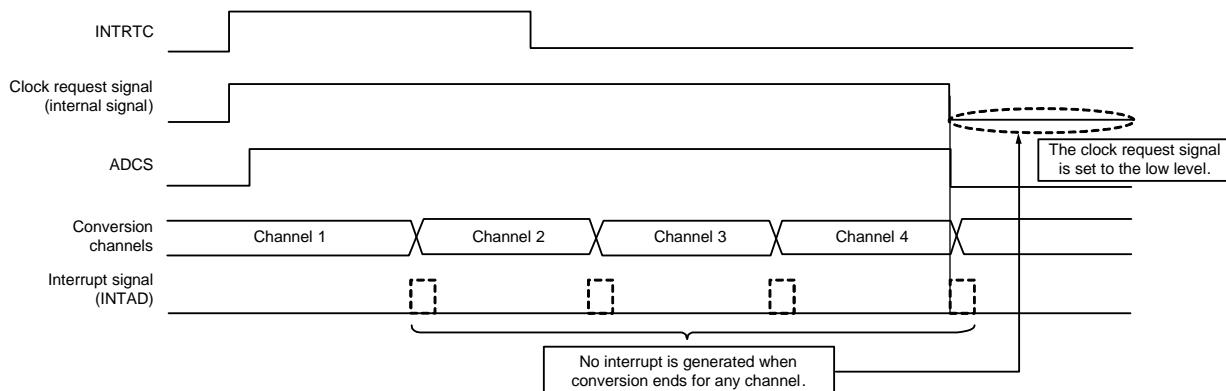
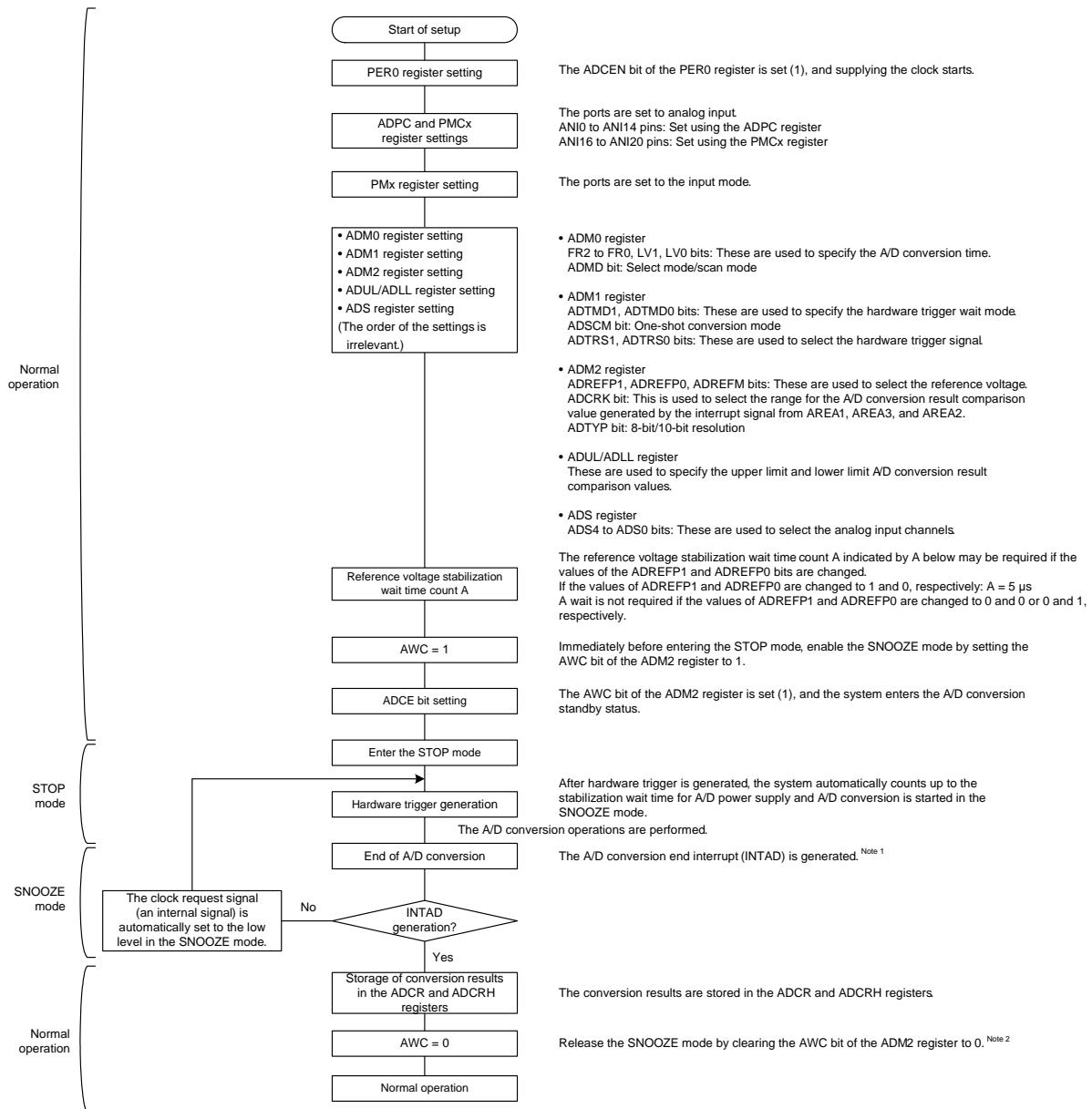


Figure 14 - 39 Flowchart for Setting up SNOOZE Mode



**Note 1.** If the A/D conversion end interrupt request signal (INTAD) is not generated by setting ADRCK bit and ADUL/ADLL register, the result is not stored in the ADCR and ADCRH registers.

The system enters the STOP mode again. If a hardware trigger is input later, A/D conversion operation is again performed in the SNOOZE mode.

**Note 2.** If the AWC bit is left set to 1, A/D conversion will not start normally in spite of the subsequent SNOOZE or normal operation mode. Be sure to clear the AWC bit to 0.

## 14.9 How to Read A/D Converter Characteristics Table

Here, special terms unique to the A/D converter are explained.

### (1) Resolution

This is the minimum analog input voltage that can be identified. That is, the percentage of the analog input voltage per bit of digital output is called 1LSB (Least Significant Bit). The percentage of 1LSB with respect to the full scale is expressed by %FSR (Full Scale Range).

1LSB is as follows when the resolution is 10 bits.

$$\begin{aligned} 1 \text{ LSB} &= 1/2^{10} = 1/1024 \\ &= 0.098\% \text{FSR} \end{aligned}$$

Accuracy has no relation to resolution, but is determined by overall error.

### (2) Overall error

This shows the maximum error value between the actual measured value and the theoretical value.

Zero-scale error, full-scale error, integral linearity error, and differential linearity errors that are combinations of these express the overall error.

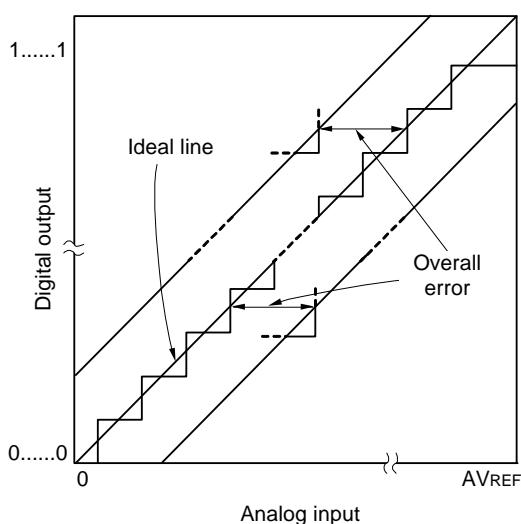
Note that the quantization error is not included in the overall error in the characteristics table.

### (3) Quantization error

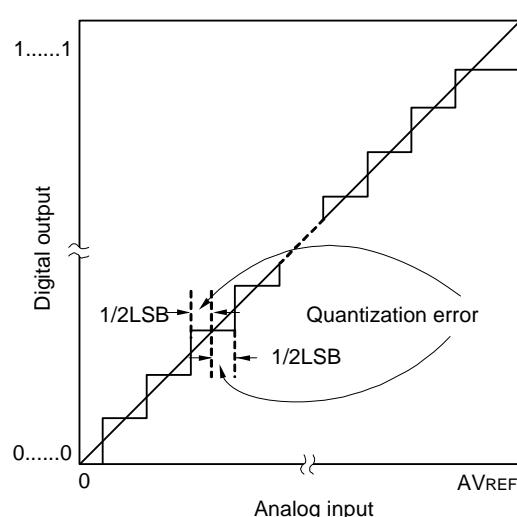
When analog values are converted to digital values, a  $\pm 1/2$ LSB error naturally occurs. In an A/D converter, an analog input voltage in a range of  $\pm 1/2$ LSB is converted to the same digital code, so a quantization error cannot be avoided.

Note that the quantization error is not included in the overall error, zero-scale error, full-scale error, integral linearity error, and differential linearity error in the characteristics table.

**Figure 14 - 40 Overall Error**



**Figure 14 - 41 Quantization Error**



## (4) Zero-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (1/2LSB) when the digital output changes from 0.....000 to 0.....001.

If the actual measurement value is greater than the theoretical value, it shows the difference between the actual measurement value of the analog input voltage and the theoretical value (3/2LSB) when the digital output changes from 0.....001 to 0.....010.

## (5) Full-scale error

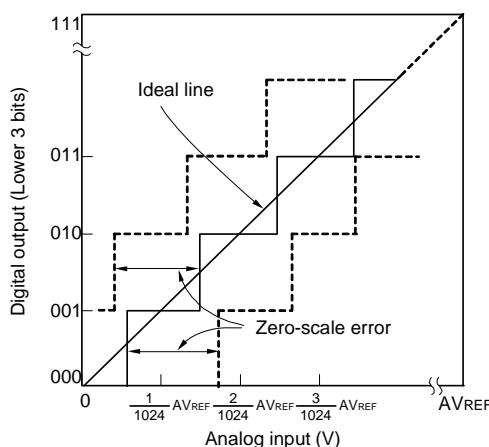
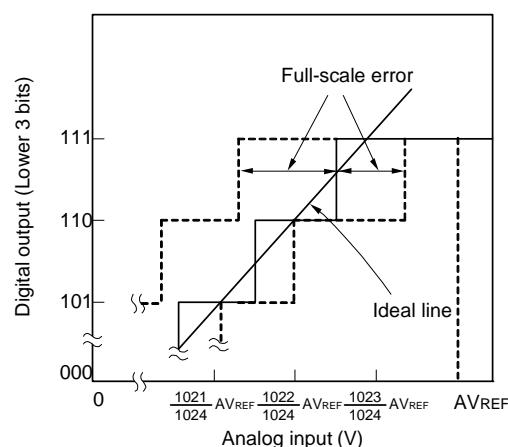
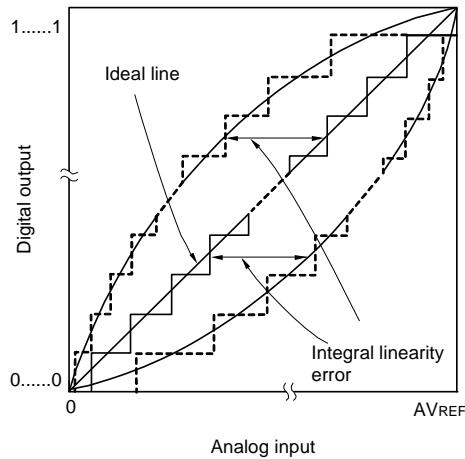
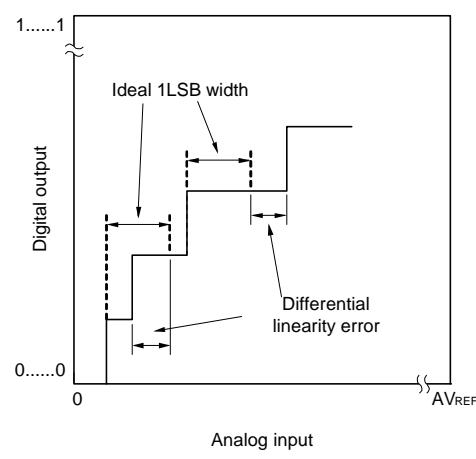
This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (Full-scale – 3/2LSB) when the digital output changes from 1.....110 to 1.....111.

## (6) Integral linearity error

This shows the degree to which the conversion characteristics deviate from the ideal linear relationship. It expresses the maximum value of the difference between the actual measurement value and the ideal straight line when the zero-scale error and full-scale error are 0.

## (7) Differential linearity error

While the ideal width of code output is 1LSB, this indicates the difference between the actual measurement value and the ideal value.

**Figure 14 - 42 Zero-Scale Error****Figure 14 - 43 Full-Scale Error****Figure 14 - 44 Integral Linearity Error****Figure 14 - 45 Differential Linearity Error**

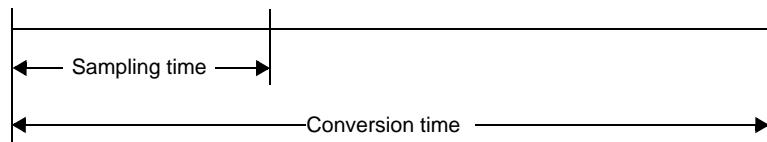
(8) Conversion time

This expresses the time from the start of sampling to when the digital output is obtained.

The sampling time is included in the conversion time in the characteristics table.

(9) Sampling time

This is the time the analog switch is turned on for the analog voltage to be sampled by the sample & hold circuit.



## 14.10 Cautions for A/D Converter

### (1) Operating current in STOP mode

Shift to STOP mode after stopping the A/D converter (by setting bit 7 (ADCS) of A/D converter mode register 0 (ADM0) to 0). The operating current can be reduced by setting bit 0 (ADCE) of the ADM0 register to 0 at the same time.

To restart from the standby status, clear bit 0 (ADIF) of interrupt request flag register 1H (IF1H) to 0 and start operation.

### (2) Input range of ANI0 to ANI14 and ANI16 to ANI20 pins

Observe the rated range of the ANI0 to ANI14 and ANI16 to ANI20 pins input voltage. If a voltage exceeding VDD and AVREFP or below Vss and AVREFM (even in the range of absolute maximum ratings) is input to an analog input channel, the converted value of that channel becomes undefined. In addition, the converted values of the other channels may also be affected.

When internal reference voltage (1.45 V) is selected reference voltage for the + side of the A/D converter, do not input voltage exceeding internal reference voltage (1.45 V) to a pin selected by the ADS register. However, it is no problem that a pin not selected by the ADS register is input voltage exceeding the internal reference voltage (1.45 V).

**Caution Internal reference voltage (1.45 V) can be used only in HS (high-speed main) mode.**

### (3) Conflicting operations

<1> Conflict between the A/D conversion result register (ADCR, ADCRH) write and the ADCR or ADCRH register read by instruction upon the end of conversion

The ADCR or ADCRH register read has priority. After the read operation, the new conversion result is written to the ADCR or ADCRH registers.

<2> Conflict between the ADCR or ADCRH register write and the A/D converter mode register 0 (ADM0) write, the analog input channel specification register (ADS), or A/D port configuration register (ADPC) write upon the end of conversion

The ADM0, ADS, or ADPC registers write has priority. The ADCR or ADCRH register write is not performed, nor is the conversion end interrupt signal (INTAD) generated.

### (4) Noise countermeasures

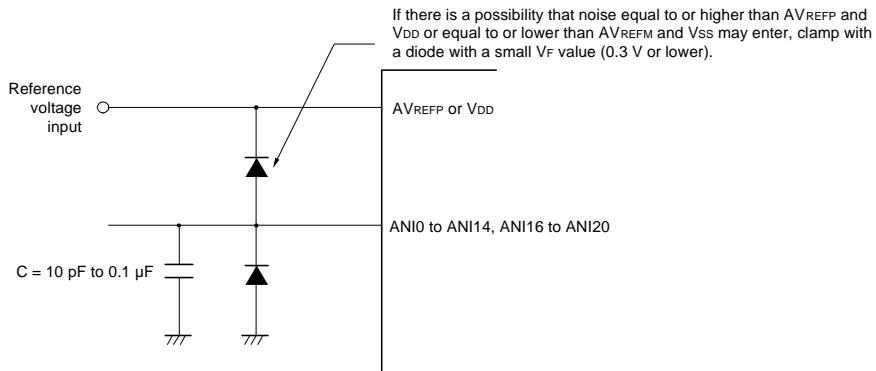
To maintain the 10-bit resolution, attention must be paid to noise input to the AVREFP, VDD, ANI0 to ANI14, and ANI16 to ANI20 pins.

<1> Connect a capacitor with a low equivalent resistance and a good frequency response to the power supply.

<2> The higher the output impedance of the analog input source, the greater the influence. To reduce the noise, connecting external capacitor as shown in Figure 14 - 46 is recommended.

<3> Do not switch these pins with other pins during conversion.

<4> The accuracy is improved if the HALT mode is set immediately after the start of conversion.

**Figure 14 - 46 Analog Input Pin Connection**

#### (5) Analog input (ANIn) pins

- <1> The analog input pins (ANI0 to ANI14, ANI16 to ANI20) are also used as input port pins (P20 to P27, P03, P02, P147, P120, P100, P150 to P156).

When A/D conversion is performed with any of the ANI0 to ANI14 and ANI16 to ANI20 pins selected, do not change to output value P20 to P27, P03, P02, P147, P120, P100, and P150 to P156 while conversion is in progress; otherwise the conversion resolution may be degraded.

- <2> If a pin adjacent to a pin that is being A/D converted is used as a digital I/O port pin, the A/D conversion result might differ from the expected value due to a coupling noise. Be sure to prevent such a pulse from being input or output.

#### (6) Input impedance of analog input (ANIn) pins

This A/D converter charges a sampling capacitor for sampling during sampling time.

Therefore, only a leakage current flows when sampling is not in progress, and a current that charges the capacitor flows during sampling. Consequently, the input impedance fluctuates depending on whether sampling is in progress, and on the other states.

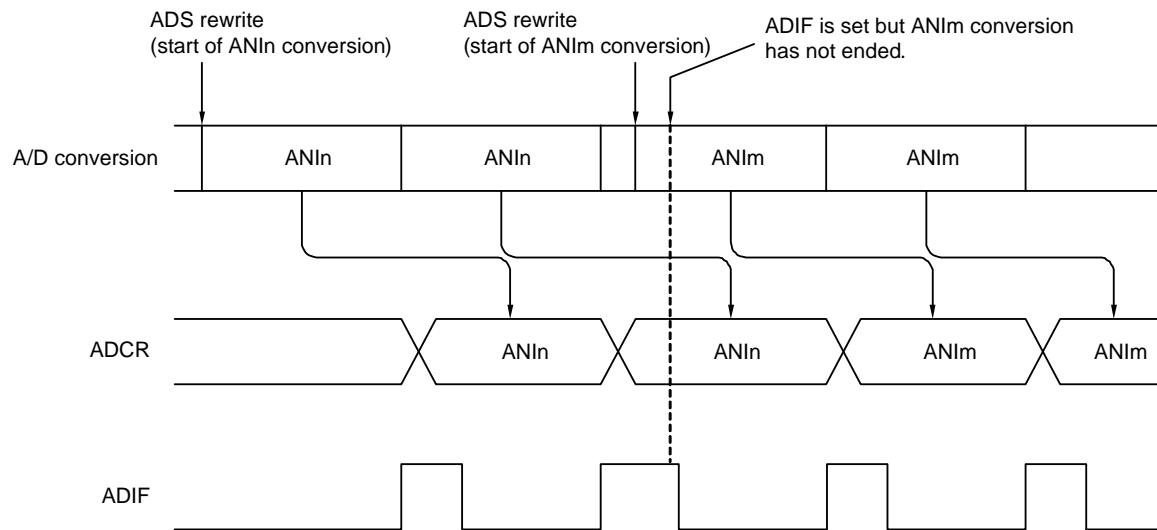
To make sure that sampling is effective, however, we recommend using the converter with analog input sources that have output impedances no greater than 1 kΩ. If a source has a higher output impedance, lengthen the sampling time or connect a larger capacitor (with a value of about 0.1 μF) to the pin from among ANI0 to ANI14 and ANI16 to ANI20 to which the source is connected (see **Figure 14 - 46**). The sampling capacitor may be being charged while the setting of the ADCS bit is 0 and immediately after sampling is restarted and so is not defined at these times. Accordingly, the state of conversion is undefined after charging starts in the next round of conversion after the value of the ADCS bit has been 1 or when conversion is repeated. Thus, to secure full charging regardless of the size of fluctuations in the analog signal, ensure that the output impedances of the sources of analog inputs are low or secure sufficient time for the completion of conversion.

#### (7) Interrupt request flag (ADIF)

The interrupt request flag (ADIF) is not cleared even if the analog input channel specification register (ADS) is changed.

Therefore, if an analog input pin is changed during A/D conversion, the A/D conversion result and ADIF flag for the pre-change analog input may be set just before the ADS register rewrite. Caution is therefore required since, at this time, when ADIF flag is read immediately after the ADS register rewrite, ADIF flag is set despite the fact A/D conversion for the post-change analog input has not ended.

When A/D conversion is stopped and then resumed, clear ADIF flag before the A/D conversion operation is resumed.

**Figure 14 - 47 Timing of A/D Conversion End Interrupt Request Generation**

## (8) Conversion results just after A/D conversion start

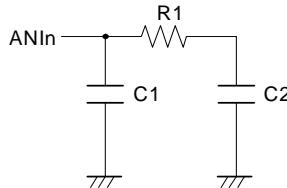
While in the software trigger mode or hardware trigger no-wait mode, the first A/D conversion value immediately after A/D conversion starts may not fall within the rating range if the ADCS bit is set to 1 within 1  $\mu$ s after the ADCE bit was set to 1. Take measures such as polling the A/D conversion end interrupt request (INTAD) and removing the first conversion result.

## (9) A/D conversion result register (ADCR, ADCRH) read operation

When a write operation is performed to A/D converter mode register 0 (ADM0), analog input channel specification register (ADS), A/D port configuration register (ADPC), and port mode control register (PMCxx), the contents of the ADCR and ADCRH registers may become undefined. Read the conversion result following conversion completion before writing to the ADM0, ADS, ADPC, or PMC register. Using a timing other than the above may cause an incorrect conversion result to be read.

## (10) Internal equivalent circuit

The equivalent circuit of the analog input block is shown below.

**Figure 14 - 48 Internal Equivalent Circuit of ANIn Pin****Table 14 - 7 Resistance and Capacitance Values of Equivalent Circuit (Reference Values)**

AVREFP, VDD	ANIn Pins	R1 [kΩ]	C1 [pF]	C2 [pF]
3.6 V ≤ VDD ≤ 5.5 V	ANI0 to ANI14	14	8	2.5
	ANI16 to ANI20	18	8	7.0
2.7 V ≤ VDD < 3.6 V	ANI0 to ANI14	39	8	2.5
	ANI16 to ANI20	53	8	7.0
1.8 V ≤ VDD < 2.7 V	ANI0 to ANI14	231	8	2.5
	ANI16 to ANI20	321	8	7.0
1.6 V ≤ VDD < 2.7 V	ANI0 to ANI14	632	8	2.5
	ANI16 to ANI20	902	8	7.0

**Remark** The resistance and capacitance values shown in Table 14 - 7 are not guaranteed values.

## (11) Starting the A/D converter

Start the A/D converter after the AVREFP and VDD voltages stabilize.

## (12) Temperature sensor output

If the internal reference voltage (1.45 V) is selected as the reference voltage of comparator 0 or comparator 1 in products with 96 KB or more code flash memory, the temperature sensor output cannot be selected.

## CHAPTER 15 D/A CONVERTER

The number of D/A converter channels differs depending on the product.

**Table 15 - 1 Output Pin of D/A Converter**

D/A output pins	R5F104xA (x = A to C, E to G)	R5F104xC (x = A to C, E to G, J, L)	R5F104xD (x = A to C, E to G, J, L)	R5F104xE (x = A to C, E to G, J, L)	R5F104AF	R5F104xF (x = B, C, E to G, J, L, M, P)	R5F104AG	R5F104xG (x = B, C, E to G, J, L, M, P)	R5F104xH (x = E to G, J, L, M, P)	R5F104xJ (x = F, G, J, L, M, P)	R5F104xK (x = G, L, M, P)	R5F104xL (x = G, L, M, P)
ANO0	—	—	—	—	✓	✓	✓	✓	✓	✓	✓	✓
ANO1	—	—	—	—	—	✓	—	✓	✓	✓	✓	✓

**Caution** Most of the following descriptions in this chapter use the 100-pin as an example.

### 15.1 Functions of D/A Converter

The D/A converter is an 8-bit resolution converter that converts digital inputs into analog signals. It is used to control analog outputs for two independent channels (ANO0, ANO1).

The D/A converter has the following features.

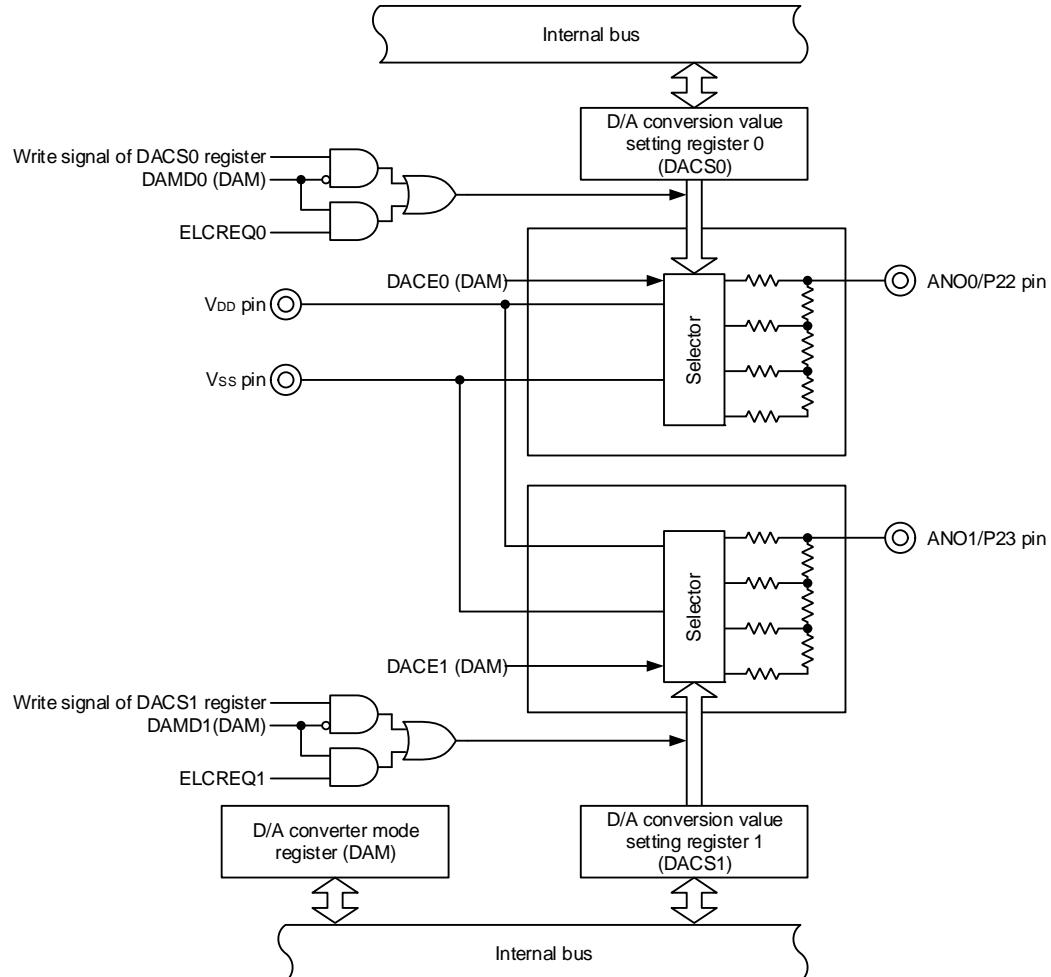
- 8-bit resolution × 2 channels
- R-2R ladder method
- Output analog voltage
  - 8-bit resolution:  $V_{DD} \times m8/256$  ( $m8$ : Value set to DACSi register)
- Operation mode
  - Normal mode
  - Real-time output mode

**Remark** i = 0, 1

## 15.2 Configuration of D/A Converter

Figure 15 - 1 shows the Block Diagram of D/A Converter.

**Figure 15 - 1 Block Diagram of D/A Converter**



**Remark** ELCREQ0 and ELCREQ1 are trigger signals (event signals from the ELC) that are used in the real-time output mode.

## 15.3 Registers Controlling D/A Converter

The D/A converter is controlled by the following registers.

- Peripheral enable register 1 (PER1)
- D/A converter mode register (DAM)
- D/A conversion value setting registers 0, 1 (DACS0, DACS1)
- Event output destination select register n (ELSELRn), n = 00 to 25

### 15.3.1 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable use of each peripheral hardware macro. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the D/A converter is used, be sure to set bit 7 (DACEN) of this register to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 15 - 2 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN

DACEN Note	Control of D/A converter input clock
0	Stops input clock supply. • SFR used by the D/A converter cannot be written. • The D/A converter is in the reset status.
1	Supplies input clock. • SFR used by the D/A converter can be read/written.

**Note** Only for products with 96 KB or more code flash memory.

**Caution 1. When setting the D/A converter, be sure to set DACEN to 1 first.**

If DACEN = 0, writing to a control register of the D/A converter is ignored, and all read values are default values (except for port mode register 2 (PM2), and port register 2 (P2)).

**Caution 2. Be sure to clear the following bits to 0.**

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 15.3.2 D/A converter mode register (DAM)

This register controls the operation of the D/A converter.

The DAM register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 15 - 3 Format of D/A converter mode register (DAM)**

Address: FFF36H After reset: 00H R/W

Symbol	7	6	<5>	<4>	3	2	1	0
DAM	0	0	DACE1	DACE0	0	0	DAMD1	DAMD0

DACEi	D/A conversion operation control
0	Stops D/A conversion operation
1	Enables D/A conversion operation

DAMD <sub>i</sub>	D/A converter operation mode selection
0	Normal mode
1	Real-time output mode

**Remark** i = 0, 1

### 15.3.3 D/A conversion value setting register i (DACS<sub>i</sub>) (i = 0, 1)

This register is used to set the analog voltage value to be output to the ANO0 and ANO1 pins when the D/A converter is used.

The DACS<sub>i</sub> register can be read by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 15 - 4 Format of D/A conversion value setting register i (DACS<sub>i</sub>) (i = 0, 1)**

Address: FFF34H (DACS0), FFF35H (DACS1) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
DACS <sub>i</sub>	DACS <sub>i</sub> 7	DACS <sub>i</sub> 6	DACS <sub>i</sub> 5	DACS <sub>i</sub> 4	DACS <sub>i</sub> 3	DACS <sub>i</sub> 2	DACS <sub>i</sub> 1	DACS <sub>i</sub> 0

**Remark** The relation between the resolution and analog output voltage (VANO<sub>i</sub>) of the D/A converter are as follows.  
 $VANO_i = VDD \times (DACS_i)/256$

When the D/A converter is not used, set the DACE<sub>i</sub> bit to 0 (output disable) and set the DACS<sub>i</sub> register to 00H to prevent current from flowing into the R-2R resistor ladder to reduce unnecessary current consumption.

### 15.3.4 Event output destination select register n (ELSELRn), n = 00 to 25

When the real-time output mode of the D/A converter is used, D/A conversion is performed using an event signal from the event link controller as an activation trigger.

For details, see **20.3.1 Event output destination select register n (ELSELRn) (n = 00 to 25)**.

### 15.3.5 Registers controlling port functions of analog input pins

Set the registers (port mode register (PMxx) and A/D port configuration register (ADPC)) that control the port functions shared with the analog output of the D/A converter.

For details, see **4.3.1 Port mode registers (PMxx)** and **4.3.7 A/D port configuration register (ADPC)**.

When using the ANO0 and ANO1 pins for analog output of the D/A converter, set the port mode register (PMxx) bit corresponding to each port to 1, and specify analog output using the A/D port configuration register (ADPC).

## 15.4 Operations of D/A Converter

### 15.4.1 Operation in Normal Mode

D/A conversion is performed using write operation to the DACSi register as the trigger. The setting method is described below.

- <1> Set the DACEN bit of the PER1 register (peripheral enable register 1) to 1 to start the supply of the input clock to the D/A converter.
- <2> Use the ADPC register (port configuration register) to set the ports to analog pins.
- <3> Set the DAMDi bit of the DAM register (D/A converter mode register) to 0 (normal mode).
- <4> Set the analog voltage value to be output to the ANO<sub>i</sub> pin to the DACSi register (D/A conversion value setting register i).

Steps <1> and <4> above constitute the initial settings.

- <5> Set the DACEi bit of the DAM register to 1 (D/A conversion enable).  
D/A conversion starts, and then, after the settling time elapses, the analog voltage set in step <4> is output to the ANO<sub>i</sub> pin.
- <6> To perform subsequent D/A conversions, write to the DACSi register.

The previous D/A conversion result is held until the next D/A conversion is performed.

When the DACEi bit of the DAM register is set to 0 (D/A conversion operation stop), D/A conversion stops.

**Caution 1. Even if 1, 0, and then 1 is set to the DACEi bit, the analog voltage set by the DACSi register is output to the ANO<sub>i</sub> pin when a settling time has elapsed after 1 is set for the last time.**

**Caution 2. If the DACSi register is rewritten during the settling time, D/A conversion is aborted and reconversion by using the rewritten values starts.**

**Remark**    i = 0, 1

### 15.4.2 Operation in Real-Time Output Mode

D/A conversion is performed on each channel using the event signals from the ELC as triggers. The setting method is described below.

- <1> Set the DACEN bit of the PER1 register (peripheral enable register 1) to 1 to start the supply of the input clock to the D/A converter.
- <2> Use the ADPC register (port configuration register) to set the ports to analog pins.
- <3> Set the DAMDi bit of the DAM register (D/A converter mode register) to 0 (normal mode).
- <4> Set the analog voltage value to be output to the ANO<sub>i</sub> pin to the DACSi register (D/A conversion value setting register i).
- <5> Set the DACEi bit of the DAM register to 1 (D/A conversion enable).  
D/A conversion starts, and then, after the settling time elapses, the analog voltage set in step <3> is output to the ANO<sub>i</sub> pin.
- <6> Use the event output destination select register (ELSEL<sub>Rn</sub>; n = 00 to 25) to set the trigger signal used for real-time output mode.
- <7> Set the DAMDi bit of the DAM register to 1 (real-time output mode).
- <8> Start the operation of the event source.

Steps <1> to <8> above constitute the initial settings.

- <9> Upon generation of the trigger signals used for real-time output mode, D/A conversion starts and the analog voltage set in step <4> will be output to the ANO<sub>i</sub> pin after a settling time has elapsed.  
Set the analog voltage value to be output to the ANO<sub>i</sub> pin, to the DACSi register before performing the next D/A conversion (trigger signal used for real-time output mode is generated).

Set the analog voltage value to be output to the ANO<sub>i</sub> pin, to the DACSi register before performing the next D/A conversion (trigger signal used for real-time output mode is generated).

When the DACEi bit of the DAM register is set to 0 (D/A conversion operation stop), D/A conversion stops.

**Caution 1. Even if 1, 0, and then 1 is set to the DACEi bit, the analog voltage set by the DACSi register is output to the ANO<sub>i</sub> pin when a settling time has elapsed after 1 is set for the last time.**

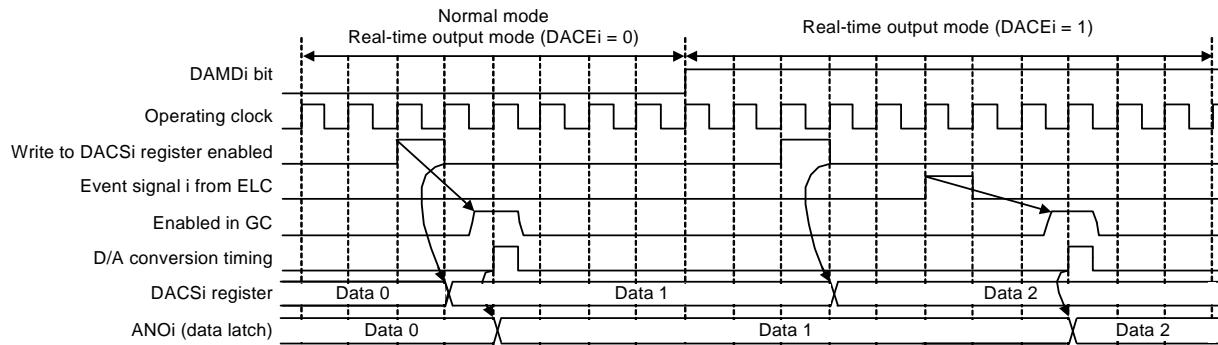
**Caution 2. Set the interval between each generation of the trigger signal used for real-time output mode of the same channel to longer than the settling time. If a trigger signal used for real-time output mode is generated during the settling time, D/A conversion is aborted and reconversion starts.**

**Caution 3. Set the interval between each generation of the trigger signal used for real-time output mode of the same channel to longer than the three clocks of fCLK. When a trigger is generated consecutively at intervals of three or fewer fCLK clock cycles, D/A conversion is performed using only the first trigger.**

### 15.4.3 Timing for Outputting D/A Conversion Value

Figure 15 - 5 shows the Timing for Outputting D/A Conversion Value.

**Figure 15 - 5 Timing for Outputting D/A Conversion Value**



**Remark**     $i = 0, 1$

- Normal mode and real-time output mode (when conversion operation is disabled)  
The value is written to the data latch after one cycle of the operating clock when the DACSi register is written.
- Real-time output mode (when conversion operation is enabled)  
The value is written to the data latch (output from the ANO $i$  pin) after three cycles of the operating clock when the event signal from the ELC is accepted.

## 15.5 Cautions for D/A Converter

Observe the following cautions when using the D/A converter.

- (1) The digital port I/O function, which is the alternate function of the ANO0 and ANO1 pins, does not operate if the ports are set to analog pins by using the ADPC register (port configuration register).  
When the P2 register is read while the ports are set to analog pins by using the ADPC register, 0 is read in the input mode and the set value of the P2 register is read in the output mode. If the digital output mode is set, no output data is output to pins.
- (2) The operation of the D/A converter continues in the HALT and STOP modes. To lower the power consumption, therefore, clear the DACEi bit to 0, and execute the HALT or STOP instruction after stopping the operation of the D/A converter.

**Remark** i = 0, 1

- (3) To stop the real-time output mode (including when changing to normal mode), one of the following procedures must be used:
  - Wait for at least three clocks after stopping the trigger output source and then set bits DACEi and DAMDi to 0.
  - After setting bits DACEi and DAMDi, set the DACEN bit of the PER1 register to 0 (DAC stop).  
When the DACEN bit is set to 0, all the registers in the DAC are cleared, so the settings of the SFRs are required to start the operation again.
- (4) When D/A conversion operation is enabled, do not perform A/D conversions from the analog input pins multiplexed with the ANO0 and ANO1 pins.
- (5) In real-time output mode, set the value of the DACSi register before a trigger signal used for real-time output mode is generated. Do not change the set value of the DACSi register while the trigger signal is output.
- (6) Since the output impedance of the D/A converter is high, no current can be taken out from the ANO0 or ANO1 pin. If the input impedance of the load is low, insert a follower amplifier between the load and the ANO0 and ANO1 pins before use. In addition, the wiring length between the follower amplifier and the load must be as short as possible due to the high output impedance. If the wiring length is long, take measures such as placing a ground pattern around the wiring area.
- (7) When entering STOP mode while real-time output mode is enabled, disable linking of ELC events before entering STOP mode.

## CHAPTER 16 COMPARATOR

Products with 96 KB or more code flash memory have two comparator channels.

### 16.1 Functions of Comparator

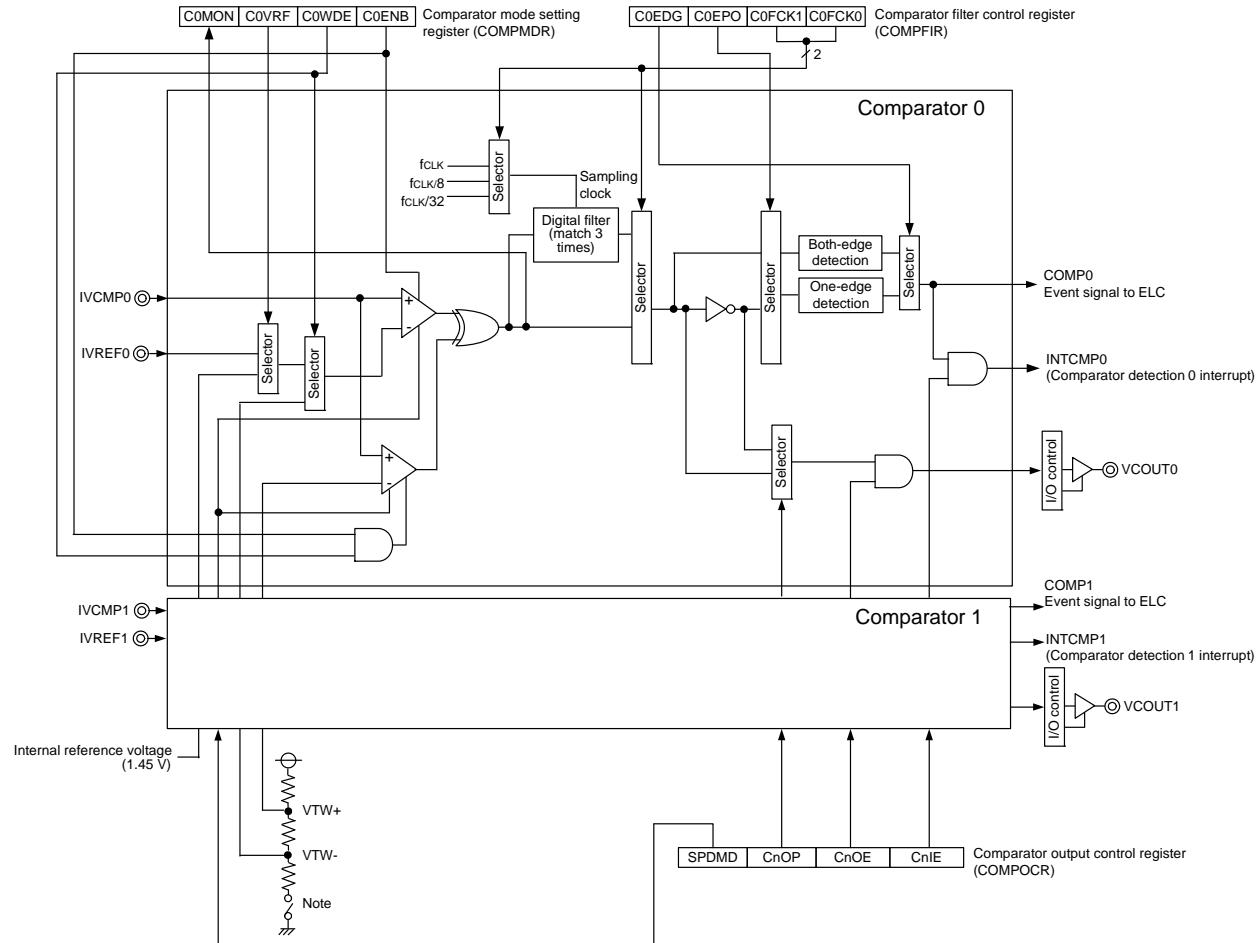
The comparator has the following functions.

- Comparator high-speed mode, comparator low-speed mode, or comparator window mode can be selected.
- The external reference voltage input or internal reference voltage can be selected as the reference voltage.
- The canceling width of the noise canceling digital filter can be selected.
- An interrupt signal can be generated by detecting an active edge of the comparator output.
- An event link controller (ELC) event signal can be output by detecting an active edge of the comparator output.

## 16.2 Configuration of Comparator

Figure 16 - 1 shows the Comparator Block Diagram.

**Figure 16 - 1 Comparator Block Diagram**



**Note** When setting either the C0WDE bit or C1WDE bit, or both bits to 1, this switch is turned ON, and the division resistor to generate the comparison voltage becomes enabled.

**Remark** n = 0, 1

## 16.3 Registers Controlling Comparator

Table 16 - 1 lists the Registers Controlling Comparator.

**Table 16 - 1 Registers Controlling Comparator**

Register Name	Symbol
Peripheral enable register 1	PER1
Comparator mode setting register	COMPMDR
Comparator filter control register	COMPFIR
Comparator output control register	COMPOCR
Port mode control register 1	PMC1
Port mode registers 1, 12, 14	PM1, PM12, PM14
Port registers 1, 12, 14	P1, P12, P14

### 16.3.1 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable use of each peripheral hardware macro. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When the Comparator is used, be sure to set bit 5 (CMPEN) of this register to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 16 - 2 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
--------	-----	-----	-----	-----	-----	---	---	-----

PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN
CMPEN Note	Control of Comparator input clock							
0	Stops input clock supply. <ul style="list-style-type: none"> <li>SFR used by the Comparator cannot be written.</li> <li>The Comparator is in the reset status.</li> </ul>							
1	Supplies input clock. <ul style="list-style-type: none"> <li>SFR used by the Comparator can be read/written.</li> </ul>							

**Note** Only for products with 96 KB or more code flash memory.

**Caution 1.** When setting the comparator, be sure to set the CMPEN bit to 1 first. If CMPEN = 0, writing to a control register of the comparator is ignored, and all read values are default values (except for port mode registers 1, 12, 14 (PM1, PM12, PM14), and port registers 1, 12, 14 (P1, P12, P14)).

**Caution 2.** Be sure to clear the following bits to 0.

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 16.3.2 Comparator mode setting register (COMPMDR)

Figure 16 - 3 Format of Comparator mode setting register (COMPMDR)

Address: F0340H		After reset: 00H		R/W					
Symbol	<7>	6	5	<4>	<3>	2	1	<0>	
COMPMDR	C1MON	C1VRF	C1WDE	C1ENB	C0MON	C0VRF	C0WDE	C0ENB	
C1MON	Comparator 1 monitor flag Notes 3, 7								
0	In standard mode: VCMP1 < comparator 1 reference voltage or comparator 1 stopped In window mode: IVCMP1 < low-voltage reference or IVCMP1 > high-voltage reference								
1	In standard mode: IVCMP1 > comparator 1 reference voltage In window mode: Low-voltage reference < IVCMP1 < high-voltage reference								
C1VRF	Comparator 1 reference voltage selection Notes 1, 4, 5, 6								
0	Comparator 1 reference voltage is IVREF1 input								
1	Comparator 1 reference voltage is internal reference voltage (1.45 V)								
C1WDE	Comparator 1 window mode selection Note 2								
0	Comparator 1 standard mode								
1	Comparator 1 window mode								
C1ENB	Comparator 1 operation enable								
0	Comparator 1 operation disabled								
1	Comparator 1 operation enabled								
C0MON	Comparator 0 monitor flag Notes 3, 7								
0	In standard mode: VCMP0 < comparator 0 reference voltage or comparator 0 stopped In window mode: VCMP0 < low-voltage reference or IVCMP0 > high-voltage reference								
1	In standard mode: VCMP0 > comparator 0 reference voltage In window mode: Low-voltage reference < IVCMP0 < high-voltage reference								

(Notes are listed on the next page.)

C0VRF	Comparator 0 reference voltage selection Notes 1, 4, 5, 6
0	Comparator 0 reference voltage is IVREF0 input
1	Comparator 0 reference voltage is internal reference voltage (1.45 V)

C0WDE	Comparator 0 window mode selection Note 2
0	Comparator 0 standard mode
1	Comparator 0 window mode

C0ENB	Comparator 0 operation enable
0	Comparator 0 operation disabled
1	Comparator 0 operation enabled

**Note 1.** Valid only when standard mode is selected. In window mode, the reference voltage in the comparator is selected regardless of the setting of this bit.

**Note 2.** Window mode cannot be set when low-speed mode is selected (the SPDMD bit in the COMPOCR register is 0).

**Note 3.** The initial value is 0 immediately after a reset is released. However, the value is undefined when C0ENB is set to 0 and C1ENB is set to 0 after operation of the comparator is enabled once.

**Note 4.** The internal reference voltage (1.45 V) can be selected in HS (high-speed main) mode. When the internal reference voltage (1.45 V) is selected in HS (high-speed main) mode, the temperature sensor output cannot be A/D converted by the A/D converter.

**Note 5.** Do not select the internal reference voltage in STOP mode.

**Note 6.** Do not select the internal reference voltage when the subsystem clock (f<sub>Ex</sub>) is selected as the CPU clock and the high-speed system clock (f<sub>MX</sub>) and the high-speed on-chip oscillator clock (f<sub>iH</sub>) are both stopped.

**Note 7.** The value written to this bit is ignored.

### 16.3.3 Comparator filter control register (COMP FIR)

Figure 16 - 4 Format of Comparator filter control register (COMP FIR)

Address: F0341H After reset: 00H R/W														
Symbol	7	6	5	4	3	2	1	0						
COMP FIR	C1EDG	C1EPO	C1FCK1	C1FCK0	C0EDG	C0EPO	C0FCK1	C0FCK0						
C1EDG		Comparator 1 edge detection selection Note 1												
0		Interrupt request by comparator 1 one-edge detection												
1		Interrupt request by comparator 1 both-edge detection												
C1EPO		Comparator 1 edge polarity switching Note 1												
0		Interrupt request at comparator 1 rising edge												
1		Interrupt request at comparator 1 falling edge												
C1FCK1		C1FCK0	Comparator 1 filter selection Note 1											
0		0	No comparator 1 filter											
0		1	Comparator 1 filter enabled, sampling at fCLK											
1		0	Comparator 1 filter enabled, sampling at fCLK/8											
1		1	Comparator 1 filter enabled, sampling at fCLK/32											
C0EDG		Comparator 0 edge detection selection Note 2												
0		Interrupt request by comparator 0 one-edge detection												
1		Interrupt request by comparator 0 both-edge detection												
C0EPO		Comparator 0 edge polarity switching Note 2												
0		Interrupt request at comparator 0 rising edge												
1		Interrupt request at comparator 0 falling edge												
C0FCK1		C0FCK0	Comparator 0 filter selection Note 2											
0		0	No comparator 0 filter											
0		1	Comparator 0 filter enabled, sampling at fCLK											
1		0	Comparator 0 filter enabled, sampling at fCLK/8											
1		1	Comparator 0 filter enabled, sampling at fCLK/32											

- Note 1.** If bits C1FCK1 to C1FCK0, C1EPO, and C1EDG are changed, a comparator 1 interrupt and an event signal to the ELC may be generated. Change these bits only after setting the ESELRL25 register for the ELC to 0 (not linked to comparator 1 output). In addition, clear bit 0 (CMPIF1) in interrupt request flag register 2H (IF2H) to 0. If bits C1FCK1 to C1FCK0 are changed from 00B (no comparator 1 filter) to a value other than 00B (comparator 1 filter enabled), allow the time for sampling four times to elapse until the filter output is updated, and then use the comparator 1 interrupt request or the event signal to the ELC.

**Note 2.** If bits C0FCK1 to C0FCK0, C0EPO, and C0EDG are changed, a comparator 0 interrupt and an event signal to the ELC may be generated. Change these bits only after setting the ELSELR24 register for the ELC to 0 (not linked to comparator 0 output). In addition, clear bit 7 (CMPIF0) in interrupt request flag register 2L (IF2L) to 0.

If bits C0FCK1 to C0FCK0 are changed from 00B (no comparator 0 filter) to a value other than 00B (comparator 0 filter enabled), allow the time for sampling four times to elapse until the filter output is updated, and then use the comparator 0 interrupt request or the event signal to the ELC.

### 16.3.4 Comparator output control register (COMPOCR)

Figure 16 - 5 Format of Comparator output control register (COMPOCR)

Symbol	<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
COMPOCR	SPDMD	C1OP	C1OE	C1IE	0	C0OP	C0OE	C0IE
SPDMD	Comparator speed selection Note 1							
0	Comparator low-speed mode							
1	Comparator high-speed mode							
C1OP	VCOUT1 output polarity selection							
0	Comparator 1 output is output to VCOUT1							
1	Inverted comparator 1 output is output to VCOUT1							
C1OE	VCOUT1 pin output enable							
0	Comparator 1 VCOUT1 pin output disabled							
1	Comparator 1 VCOUT1 pin output enabled							
C1IE	Comparator 1 interrupt request enable Note 2							
0	Comparator 1 interrupt request disabled							
1	Comparator 1 interrupt request enabled							
C0OP	VCOUT0 output polarity selection							
0	Comparator 0 output is output to VCOUT0							
1	Inverted comparator 0 output is output to VCOUT0							
C0OE	VCOUT0 pin output enable							
0	Comparator 0 VCOUT0 pin output disabled							
1	Comparator 0 VCOUT0 pin output enabled							
C0IE	Comparator 0 interrupt request enable Note 3							
0	Comparator 0 interrupt request disabled							
1	Comparator 0 interrupt request enabled							

- Note 1.** When rewriting the SPDMD bit, be sure to set the CiENB bit (*i* = 0 or 1) in the COMPMDR register to 0 in advance.
- Note 2.** If C1IE is changed from 0 (interrupt request disabled) to 1 (interrupt request enabled), since bit 0 (CMPIF1) in interrupt request flag register 2H (IF2H) may set to 1 (interrupt requested), clear bit 0 (CMPIF1) in interrupt request flag register 2H (IF2H) to 0 before using an interrupt.
- Note 3.** If C0IE is changed from 0 (interrupt request disabled) to 1 (interrupt request enabled), since bit 7 (CMPIFO) in interrupt request flag register 2L (IF2L) may set to 1 (interrupt requested), clear bit 7 (CMPIFO) in interrupt request flag register 2L (IF2L) to 0 before using an interrupt.

### 16.3.5 Registers controlling port functions of analog input pins

When using the IVCMP0, IVCMP1, IVREF0, and IVREF1 pins for analog input of the comparator, set the port mode register (PMxx) bit and port mode control register (PMCxx) bit to 1.

When using the VCOUT0 and VCOUT1 functions, set the registers (port mode register (PMxx) and port register (Pxx) that control the port functions shared with the target channels. For details, see **4.3.1 Port mode registers (PMxx)** and **4.3.2 Port registers (Pxx)**.

## 16.4 Operation

Comparator 0 and comparator 1 operate independently. Their setting methods and operations are the same. Table 16 - 2 lists the Procedure for Setting Comparator Associated Registers.

**Table 16 - 2 Procedure for Setting Comparator Associated Registers**

Step	Register	Bit	Setting Value		
1	PER1	CMPEN	1 (input clock supply)		
2	PMC1	PMC1n	Select the function of pins IVCMPi and IVREFi. Set the PMC1n bit to 1 (analog input). Set the PM1n bit to 1 (input mode).		
	PM1	PM1n			
3	COMPOCR	SPDMD	Select the comparator response speed (0: Low-speed mode/1: High-speed mode). Note 1		
4	COMPMDR	CiWDE	0 (standard mode)		1 (window mode) Note 2
		CiVRF	0 (Reference = IVREFi input)	1 (Reference = internal reference voltage (1.45 V)) Note 5	Window comparator operation (reference = internal VREF)
		CiENB	1 (operation enabled)		
5	Wait for comparator stabilization time tCMP.				
6	COMPFIR	CiFCK1, CiFCK0	Select whether the digital filter is used or not and the sampling clock.		
		CiEOP, CIEDG	Select the edge detection condition for an interrupt request (rising edge/falling edge/both edges).		
7	COMPOCR	CiOP, CiOE	Set the VCOUTi output (select the polarity and set output enabled or disabled). Refer to <b>16.4.4 Comparator i Output (i = 0 or 1)</b> .		
		CiIE	Set the interrupt request output enabled or disabled. Refer to <b>16.4.4 Comparator i Output (i = 0 or 1)</b> .		
8	PR2L, PR2H Note 3	CMPPR0i, CMPPR1i	When using an interrupt: Select the interrupt priority level.		
9	MK2L, MK2H Note 3	CMPMKi	When using an interrupt: Select the interrupt masking.		
10	IF2L, IF2H Note 3	CMPIFi	When using an interrupt: 0 (no interrupt requested: initialization) Note 4		

**Note 1.** Comparator 0 and comparator 1 cannot be set independently.

**Note 2.** Can be set in high-speed mode (SPDMD = 1).

**Note 3.** PR2L, MK2L, and IF2L are the interrupt control registers for comparator 0. PR2H, MK2H, and IF2H are the interrupt control registers for comparator 1.

**Note 4.** After the setting of the comparator, an unnecessary interrupt may occur until operation becomes stable, so initialize the interrupt flag.

**Note 5.** Can be set in HS (high-speed main) mode.

**Remark** i = 0, 1, n = 2, 3, 6, 7

Figures 16 - 6 and 16 - 7 show comparator i ( $i = 0$  or  $1$ ) operation examples. In standard mode, the CiMON bit in the COMPMDR register is set to 1 when the analog input voltage is higher than the reference input voltage, and the CiMON bit is set to 0 when the analog input voltage is lower than the reference input voltage.

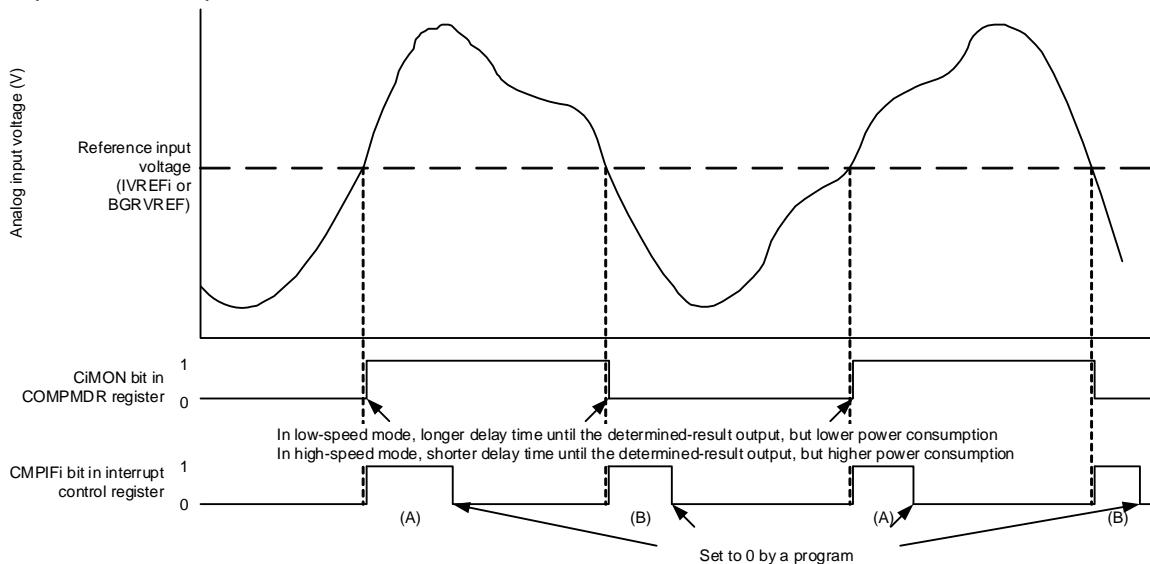
In window mode, the CiMON bit in the COMPMDR register is set to 1 when the analog input voltage meets the following condition, and the CiMON bit is set to 0 when the analog input voltage does not meet the following condition:

"Low-voltage reference voltage < analog input voltage < high-voltage reference voltage"

When using the comparator i interrupt, set CiE in the COMPOCR register to 1 (interrupt request enabled). If the comparison result changes at this time, a comparator i interrupt request is generated. For details on interrupt requests, refer to **16.4.2 Comparator i ( $i = 0$  or  $1$ ) Interrupts**.

**Figure 16 - 6 Comparator i ( $i = 0$  or  $1$ ) Operation Example in Standard Mode**

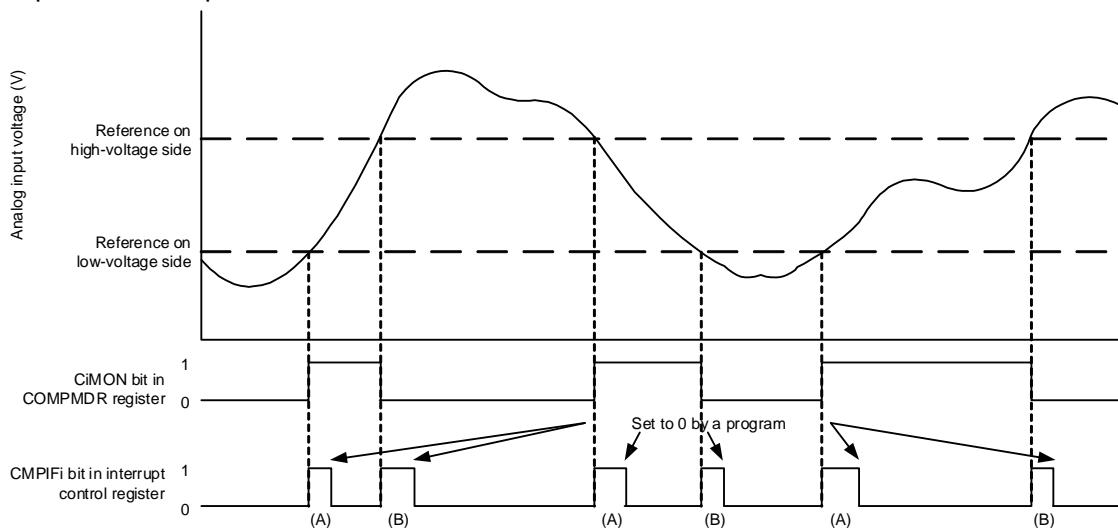
- Operation example in standard mode



**Caution** The above diagram applies when CiFCK1 to CiFCK0 in the COMPFIR register = 00B (no filter) and CiEDG = 1 (both edges). When CiEDG = 0 and CiEPO = 0 (rising edge), CMPIFI changes as shown by (A) only. When CiEDG = 0 and CiEPO = 1 (falling edge), CMPIFI changes as shown by (B) only.

**Figure 16 - 7 Comparator i (i = 0 or 1) Operation Example in Window Mode**

- Operation example in window mode



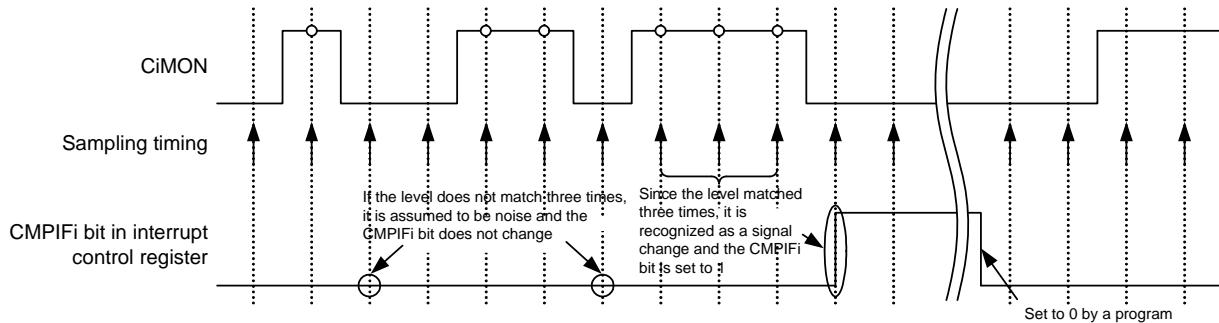
**Caution** The above diagram applies when CiFCK1 to CiFCK0 in the COMPFIR register = 00B (no filter) and CiEDG = 1 (both edges). When CiEDG = 0 and CiEPO = 0 (rising edge), CMPIFI changes as shown by (A) only. When CiEDG = 0 and CiEPO = 1 (falling edge), CMPIFI changes as shown by (B) only.

### 16.4.1 Comparator i Digital Filter ( $i = 0$ or $1$ )

Comparator  $i$  contains a digital filter. The sampling clock can be selected by bits CiFCK1 - CiFCK0 in the COMPFIR register. The comparator  $i$  output signal is sampled every sampling clock, and when the level matches three times, that value is determined as the digital filter output at the next sampling clock.

Figure 16 - 8 shows the Comparator  $i$  ( $i = 0$  or  $1$ ) Digital Filter and Interrupt Operation Example.

**Figure 16 - 8 Comparator  $i$  ( $i = 0$  or  $1$ ) Digital Filter and Interrupt Operation Example**



**Caution** The above operation example applies when bits CiFCK1 to CiFCK0 in the COMPFIR register is 01B, 10B, or 11B (digital filter enabled).

### 16.4.2 Comparator $i$ ( $i = 0$ or $1$ ) Interrupts

The comparator generates interrupt requests from two sources, comparator 0 and comparator 1. The comparator  $i$  interrupt each uses a priority level specification flag, an interrupt mask flag, an interrupt request flag, and a single vector.

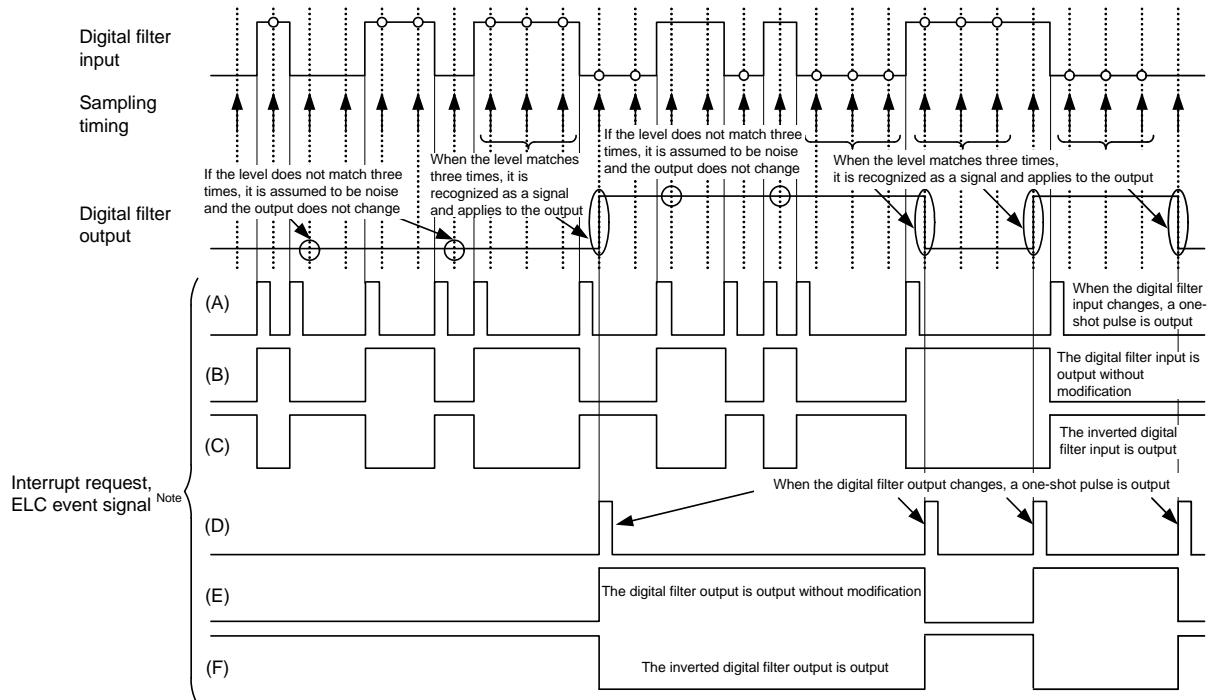
When using the comparator  $i$  interrupt, set the CiE bit in the COMPOCR register to 1 (interrupt request output enabled). The condition for interrupt request generation can be set by the COMPFIR register. The comparator outputs can also be passed through the digital filter. Three different sampling clocks can be selected for the digital filter.

For details on the register setting and interrupt request generation, refer to **16.3.3 Comparator filter control register (COMPFIR)** and **16.3.4 Comparator output control register (COMPOCR)**.

### 16.4.3 Event signal output to event link controller (ELC)

An event signal to the ELC is generated by detecting the edge for the digital filter output set by the COMPFIR register, which is the same as the condition for interrupt request generation. However, unlike interrupt requests, the event signal to the ELC are always output regardless of the CiE bit in the COMPOCR register. Set registers ESEL24 and ESEL25 for the ELC to select the event output destination and to stop linking events.

**Figure 16 - 9 Digital Filter and Interrupt Request/Event Signal Output to the ELC Operation**



**Note** When the CiE bit ( $i = 0, 1$ ) is 1, the same waveform is generated for an interrupt request and an ELC event signal.  
When the CiE bit ( $i = 0, 1$ ) is 0, the value is fixed at 0 for an interrupt request only.

The waveforms of (A), (B), and (C) are shown for an operation example when the CiFCK bits ( $i = 0, 1$ ) in the COMPFIR register are 00B (no digital filter). The waveforms (D), (E), and (F) are shown for an operation example when the CiFCK bits ( $i = 0, 1$ ) in the COMPFIR register are 01B, 10B, or 11B (digital filter enabled).

(A) and (D) apply when the CiEDG bit is set to 1 (both edges), (B) and (E) when the CiEDG bit is 0 and the CiEPO bit is 0 (rising edge), and (C) and (F) when the CiEDG bit is 0 and the CiEPO bit is 1 (falling edge).

#### 16.4.4 Comparator i Output (i = 0 or 1)

The comparison result from the comparator can be output to external pins. Bits CiOP and CiOE in the COMPOCR register can be used to set the output polarity (non-inverted output or inverted output) and output enabled or disabled. For the correspondence between the register setting and the comparator output, refer to [16.3.4 Comparator output control register \(COMPOCR\)](#).

To output the comparator comparison result to the VCOUT<sub>i</sub> output pin, use the following procedure to set the ports. Note that the ports are set to input after reset.

- <1> Set the mode for the comparator (Steps 2 to 5 as listed in Table 16 - 2 Procedure for Setting Comparator Associated Registers).
- <2> Set the VCOUT<sub>i</sub> output for the comparator (set the COMPOCR register to select the polarity and enable the output).
- <3> Set the corresponding port mode control register bit for the VCOUT<sub>i</sub> output pin to 0.
- <4> Set the corresponding port register bit for the VCOUT<sub>i</sub> output pin to 0.
- <5> Set the corresponding port mode register for the VCOUT<sub>i</sub> output pin to output (start outputting from the pin).

#### 16.4.5 Stopping or Supplying Comparator Clock

To stop the comparator clock by setting peripheral enable register 1 (PER1), use the following procedure:

- <1> Set the CiENB bit in the COMPMDR register to 0 (stop the comparator).
- <2> Set the CMPIFi bit in registers IF2L and IF2H to 0 (clear any unnecessary interrupt before stopping the comparator).
- <3> Set the CMPEN bit in the PER1 register to 0.

When the clock is stopped by setting PER1, all the internal registers in the comparator are initialized. To use the comparator again, follow the procedure in Table 16 - 2 to set the registers.

**Caution 1.** The temperature sensor output cannot be A/D converted by the A/D converter while the comparator n reference voltage select bit (CnVRF) in the comparator mode setting register (COMPMDR) is 1 (comparator n reference voltage is internal reference voltage (1.45 V)).

**Caution 2.** When DTC activation is enabled under either of the following conditions, a DTC transfer is started and an interrupt is generated after completion of the transfer.

- The comparator is set to an interrupt request on one-edge detection (CnEDG = 0), an interrupt request at the rising edge for the comparator, and IVCMP > IVREF (or internal reference voltage: 1.45 V)
- The comparator is set to an interrupt request on one-edge detection (CnEDG = 0), an interrupt request at the falling edge for the comparator, and IVCMP < IVREF (or internal reference voltage: 1.45 V)

## CHAPTER 17 SERIAL ARRAY UNIT

Serial array unit has four serial channels. All channels can achieve UART, and only channel 0 can achieve 3-wire serial (CSI) and simplified I<sup>2</sup>C.

Function assignment of each channel supported by the RL78/G14 is as shown below.

- 30, 32-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	—		—

- 36, 40, 44-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 48, 52-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 64-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	CSI10	UART1	IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 80, 100-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	CSI10	UART1	IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21
	2	CSI30	UART3	IIC30
	3	CSI31		IIC31

When "UART0" is used for channels 0 and 1 of the unit 0, CSI00 and CSI01 cannot be used, but CSI10, UART1, or IIC10 can be used for channels 2 and 3.

**Caution** Most of the following descriptions in this chapter use the units and channels of the 100-pin products as an example.

## 17.1 Functions of Serial Array Unit

Each serial interface supported by the RL78/G14 has the following features.

### 17.1.1 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)

Data is transmitted or received in synchronization with the serial clock (SCK) output from the master channel.

3-wire serial communication is clocked communication performed by using three communication lines: one for the serial clock (SCK), one for transmitting serial data (SO), one for receiving serial data (SI).

For details about the settings, see **17.5 Operation of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) Communication**.

[Data transmission/reception]

- Data length of 7 or 8 bits
- Phase control of transmit/receive data
- MSB/LSB first selectable

[Clock control]

- Master/slave selection
- Phase control of I/O clock
- Setting of transfer period by prescaler and internal counter of each channel
- Maximum transfer rate Note

During master communication: Max. fCLK/2 (CSI00 only)

Max. fCLK/4

During slave communication: Max. fmCK/6

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt

[Error detection flag]

- Overrun error

CSIs of following channels supports the SNOOZE mode. When SCK input is detected while in the STOP mode, the SNOOZE mode makes data reception that does not require the CPU possible. Only following CSIs can be specified.

- 30 to 64-pin products: CSI00
- 80, 100-pin products: CSI00 and CSI20

**Note** Use the clocks within a range satisfying the SCK cycle time ( $t_{KCY}$ ) characteristics. For details, see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**.

### 17.1.2 UART (UART0 to UART3)

This is a start-stop synchronization function using two lines: serial data transmission (Tx) and serial data reception (Rx) lines. By using these two communication lines, each data frame, which consist of a start bit, data, parity bit, and stop bit, is transferred asynchronously (using the internal baud rate) between the microcontroller and the other communication party. Full-duplex UART communication can be performed by using a channel dedicated to transmission (even-numbered channel) and a channel dedicated to reception (odd-numbered channel). The LIN-bus can be implemented by using timer array unit with an external interrupt (INTP0).

For details about the settings, see **17.7 Operation of UART (UART0 to UART3) Communication**.

[Data transmission/reception]

- Data length of 7, 8, or 9 bits Note
- Select the MSB/LSB first
- Level setting of transmit/receive data and select of reverse
- Parity bit appending and parity check functions
- Stop bit appending

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt
- Error interrupt in case of framing error, parity error, or overrun error

[Error detection flag]

- Framing error, parity error, or overrun error

In addition, UARTs of following channels supports the SNOOZE mode. When Rx input is detected while in the STOP mode, the SNOOZE mode makes data reception that does not require the CPU possible. Only the following UARTs can be specified when FRQSEL4 in the option byte (000C2H) = 0 in the SNOOZE mode.

- |                          |                 |
|--------------------------|-----------------|
| • 30 to 64-pin products: | UART0           |
| • 80, 100-pin products:  | UART0 and UART2 |

The LIN-bus is accepted in UART0 (0 and 1 channels of unit 0).

[LIN-bus functions]

- Wakeup signal detection
- Break field (BF) detection
- Sync field measurement, baud rate calculation

} Using the external interrupt (INTP0) and timer array unit

**Note** Only the following UARTs can be specified for the 9-bit data length.

- 30 to 64-pin products: UART0
- 80 to 100-pin products: UART0 and UART2

### 17.1.3 Simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31)

This is a clocked communication function to communicate with two or more devices by using two lines: serial clock (SCL) and serial data (SDA). This simplified I<sup>2</sup>C is designed for single communication with a device such as EEPROM, flash memory, or A/D converter, and therefore, it functions only as a master.

Make sure by using software, as well as operating the control registers, that the AC specifications of the start and stop conditions are observed.

For details about the settings, see **17.9 Operation of Simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) Communication**.

[Data transmission/reception]

- Master transmission, master reception (only master function with a single master)
- ACK output function Note and ACK detection function
- Data length of 8 bits (When an address is transmitted, the address is specified by the higher 7 bits, and the least significant bit is used for R/W control.)
- Manual generation of start condition and stop condition

[Interrupt function]

- Transfer end interrupt

[Error detection flag]

- ACK error or overrun error

\* [Functions not supported by simplified I<sup>2</sup>C]

- Slave transmission, slave reception
- Arbitration loss detection function
- Wait detection functions

**Note** When receiving the last data, ACK will not be output if 0 is written to the SOEmn bit (serial output enable register m (SOEm)) and serial communication data output is stopped. See the processing flow in **17.9.3 (2)** for details.

**Remark** To use an I<sup>2</sup>C bus of full function, see **CHAPTER 18 SERIAL INTERFACE IICA**.

## 17.2 Configuration of Serial Array Unit

The serial array unit includes the following hardware.

**Table 17 - 1 Configuration of Serial Array Unit**

Item	Configuration
Shift register	8 bits or 9 bits Note 1
Buffer register	Lower 8 bits or 9 bits of serial data register mn (SDRmn) Notes 1, 2
Serial clock I/O	SCK00, SCK01, SCK10, SCK11, SCK20, SCK21, SCK30, SCK31 pins (for 3-wire serial I/O), SCL00, SCL01, SCL10, SCL11, SCL20, SCL21, SCL30, SCL31 pins (for simplified I <sup>2</sup> C)
Serial data input	SI00, SI01, SI10, SI11, SI20, SI21, SI30, SI31 pins (for 3-wire serial I/O), RxD0 pin (for UART supporting LIN-bus), RxD1, RxD2, RxD3 pins (for UART)
Serial data output	SO00, SO01, SO10, SO11, SO20, SO21, SO30, SO31 pins (for 3-wire serial I/O), TxD0 pin (for UART supporting LIN-bus), TxD1, TxD2, TxD3 pins (for UART)
Serial data I/O	SDA00, SDA01, SDA10, SDA11, SDA20, SDA21, SDA30, SDA31 pins (for simplified I <sup>2</sup> C)
Slave select input	$\overline{SSI00}$ pin (for slave select input function)
Control registers	<ul style="list-style-type: none"> <li>&lt;Registers of unit setting block&gt;           <ul style="list-style-type: none"> <li>• Peripheral enable register 0 (PER0)</li> <li>• Serial clock select register m (SPSm)</li> <li>• Serial channel enable status register m (SEm)</li> <li>• Serial channel start register m (SSm)</li> <li>• Serial channel stop register m (STm)</li> <li>• Serial output enable register m (SOEm)</li> <li>• Serial output register m (SOm)</li> <li>• Serial output level register m (SOLm)</li> <li>• Serial standby control register m (SSCm)</li> <li>• Input switch control register (ISC)</li> <li>• Noise filter enable register 0 (NFEN0)</li> </ul> </li> <li>&lt;Registers of each channel&gt;           <ul style="list-style-type: none"> <li>• Serial data register mn (SDRmn)</li> <li>• Serial mode register mn (SMRmn)</li> <li>• Serial communication operation setting register mn (SCRmn)</li> <li>• Serial status register mn (SSRmn)</li> <li>• Serial flag clear trigger register mn (SIRmn)</li> </ul> </li> <li>&lt;Port registers&gt;           <ul style="list-style-type: none"> <li>• Port input mode registers 0, 1, 3, 5 (PIM0, PIM1, PIM3, PIM5)</li> <li>• Port output mode registers 0, 1, 3, 5, 7 (POM0, POM1, POM3, POM5, POM7)</li> <li>• Port mode registers 0, 1, 3, 5 to 7 (PM0, PM1, PM3, PM5 to PM7)</li> <li>• Port registers 0, 1, 3, 5 to 7 (P0, P1, P3, P5 to P7)</li> </ul> </li> </ul>

(Notes and Remark are listed on the next page.)

**Note 1.** The number of bits used as the shift register and buffer register differs depending on the unit and channel.

- 30 to 64 -pin products and mn = 00, 01: lower 9 bits
- 80, 100-pin products and mn = 00, 01, 10, 11: lower 9 bits
- Other than above: lower 8 bits

**Note 2.** The lower 8 bits of serial data register mn (SDRmn) can be read or written as the following SFR, depending on the communication mode.

- CSIp communication ..... SIOp (CSIp data register)
- UARTq reception ..... RXDq (UARTq receive data register)
- UARTq transmission ..... TXDq (UARTq transmit data register)
- IICr communication ..... SIOR (IICr data register)

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), q: UART number (q = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)

Figure 17 - 1 shows the Block Diagram of Serial Array Unit 0.

**Figure 17 - 1 Block Diagram of Serial Array Unit 0**

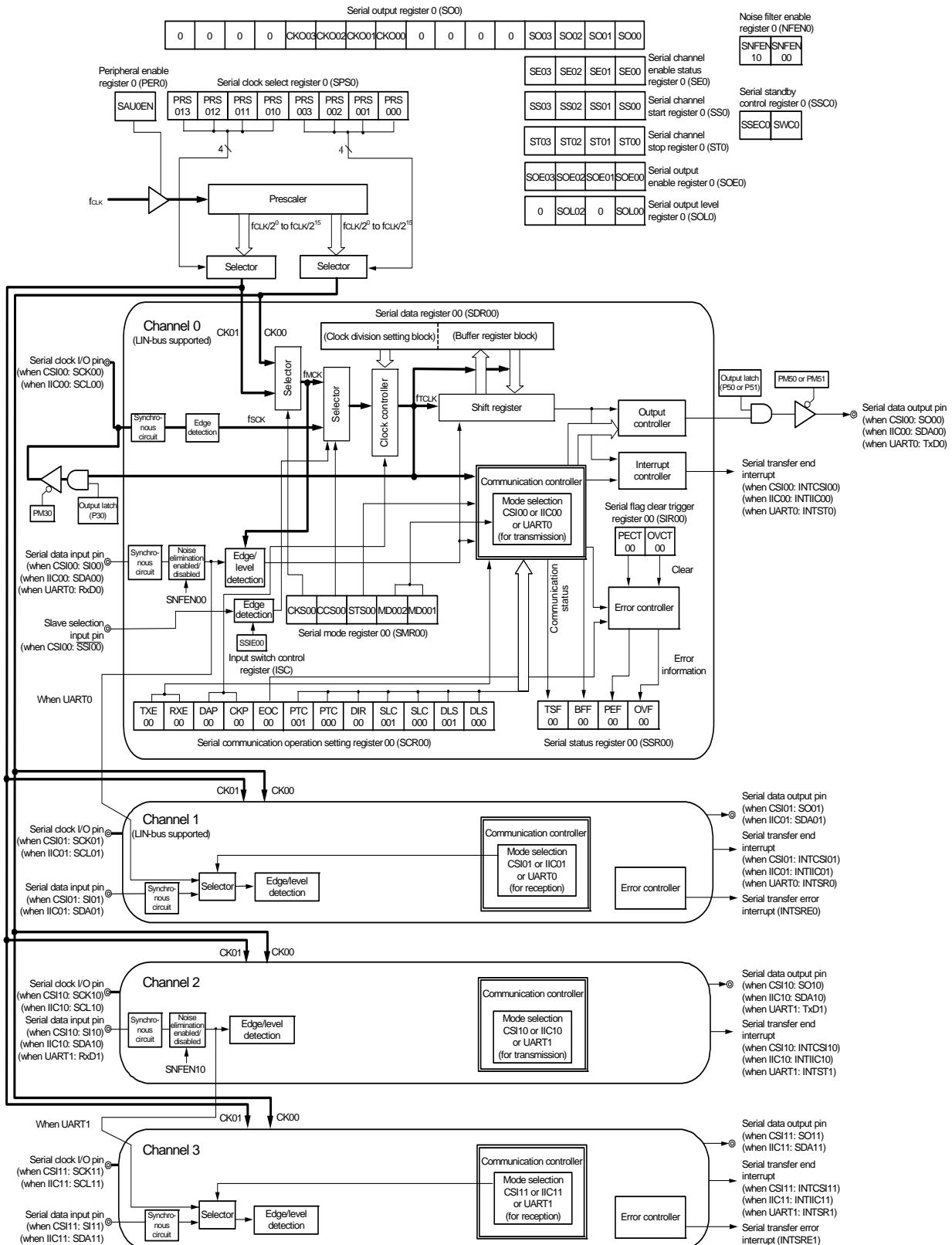
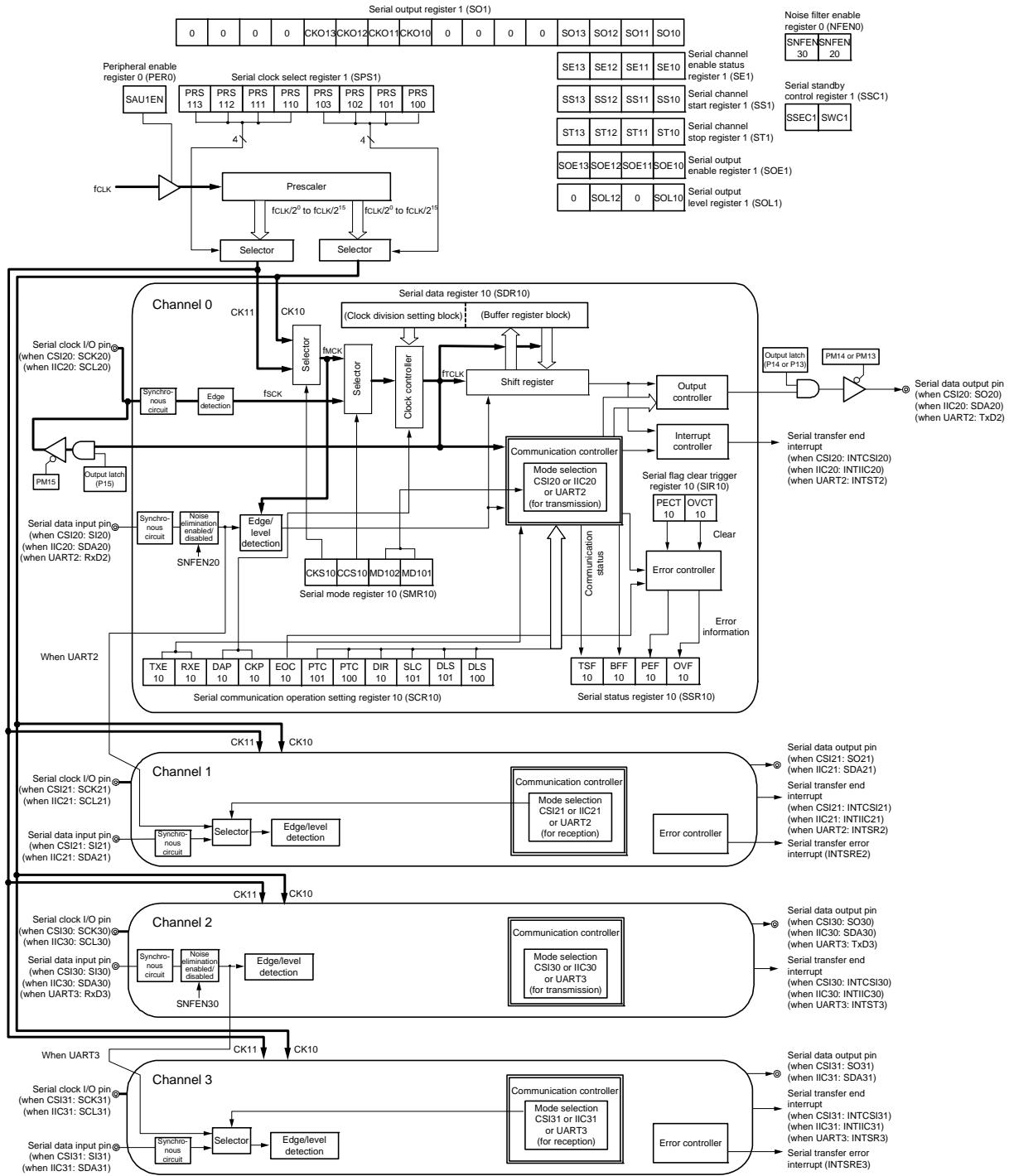


Figure 17 - 2 shows the Block Diagram of Serial Array Unit 1.

**Figure 17 - 2 Block Diagram of Serial Array Unit 1**



### 17.2.1 Shift register

This is a 9-bit register that converts parallel data into serial data or vice versa.

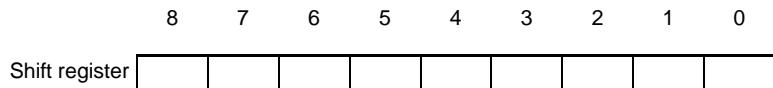
In case of the UART communication of nine bits of data, nine bits (bits 0 to 8) are used <sup>Note 1</sup>.

During reception, it converts data input to the serial pin into parallel data.

When data is transmitted, the value set to this register is output as serial data from the serial output pin.

The shift register cannot be directly manipulated by program.

To read or write the shift register, use the lower 8/9 bits of serial data register mn (SDRmn).



### 17.2.2 Lower 8/9 bits of the serial data register mn (SDRmn)

The SDRmn register is the transmit/receive data register (16 bits) of channel n. Bits 8 to 0 (lower 9 bits) <sup>Note 1</sup> or bits 7 to 0 (lower 8 bits) function as a transmit/receive buffer register, and bits 15 to 9 are used as a register that sets the division ratio of the operation clock (fMCK).

When data is received, parallel data converted by the shift register is stored in the lower 8/9 bits. When data is to be transmitted, set transmit data to be transferred to the shift register to the lower 8/9 bits.

The data stored in the lower 8/9 bits of this register is as follows, depending on the setting of bits 0 and 1 (DLSmn0, DLSmn1) of serial communication operation setting register mn (SCRmn), regardless of the output sequence of the data.

- 7-bit data length (stored in bits 0 to 6 of SDRmn register)
- 8-bit data length (stored in bits 0 to 7 of SDRmn register)
- 9-bit data length (stored in bits 0 to 8 of SDRmn register) <sup>Note 1</sup>

The SDRmn register can be read or written in 16-bit units.

The lower 8/9 bits of the SDRmn register can be read or written <sup>Note 2</sup> as the following SFR, depending on the communication mode.

- CSIp communication..... SIOp (CSIp data register)
- UARTq reception ..... RXDq (UARTq receive data register)
- UARTq transmission ..... TXDq (UARTq transmit data register)
- IICr communication..... SIOR (IICr data register)

Reset signal generation clears the SDRmn register to 0000H.

**Note 1.** Only following UARTs can be specified for the 9-bit data length.

- 30 to 64-pin products: UART0
- 80, 100-pin products: UART0, UART2

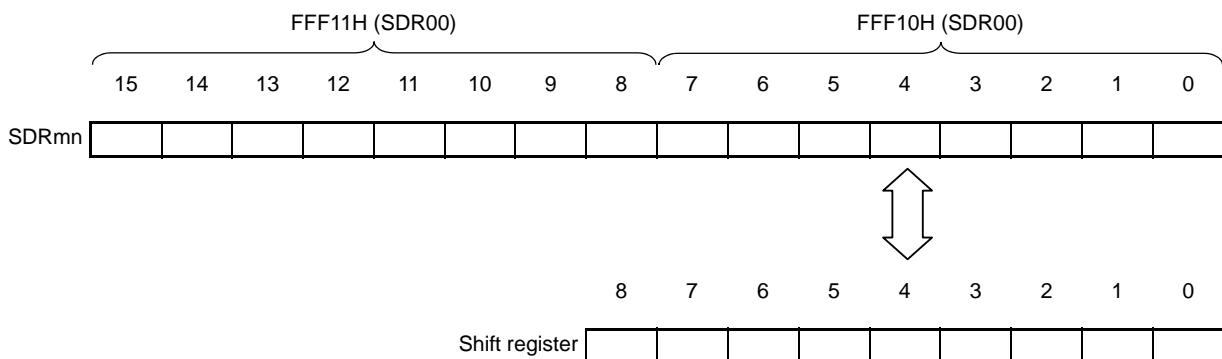
**Note 2.** When operation is stopped (SEmn = 0), do not rewrite SDRmn[7:0] by an 8-bit memory manipulation instruction (SDRmn[15:9] are all cleared to 0).

**Remark 1.** After data is received, "0" is stored in bits 0 to 8 in bit portions that exceed the data length.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), q: UART number (q = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)

**Figure 17 - 3 Format of Serial data register mn (SDRmn) (mn = 00, 01, 10, 11)**

Address: FFF10H, FFF11H (SDR00), FFF12H, FFF13H (SDR01) After reset: 0000H R/W  
 FFF48H, FFF49H (SDR10) Note, FFF4AH, FFF4BH (SDR11) Note

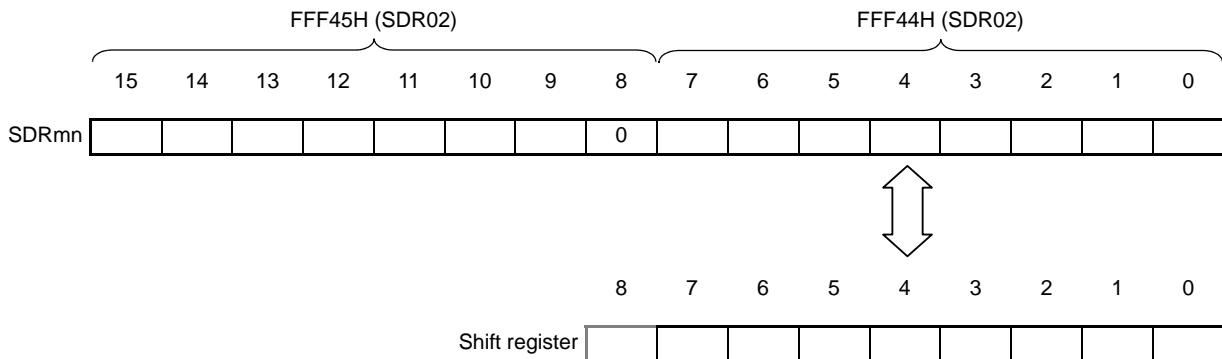


**Note** 80, 100-pin products

**Remark** For the function of the higher 7 bits of the SDRmn register, see 17.3 Registers Controlling Serial Array Unit.

**Figure 17 - 4 Format of Serial data register mn (SDRmn) (mn = 02, 03, 10, 11, 12, 13)**

Address: FFF44H, FFF45H (SDR02), FFF46H, FFF47H (SDR03), After reset: 0000H R/W  
 FFF48H, FFF49H (SDR10) Note 1, FFF4AH, FFF4BH (SDR11) Note 1  
 FFF14H, FFF15H (SDR12) Note 2, FFF16H, FFF17H (SDR13) Note 2



**Note 1.** 30 to 64-pin products

**Note 2.** 80, 100-pin products

**Caution** Be sure to clear bit 8 to “0”.

**Remark** For the function of the higher 7 bits of the SDRmn register, see 17.3 Registers Controlling Serial Array Unit.

### 17.3 Registers Controlling Serial Array Unit

Serial array unit is controlled by the following registers.

- Peripheral enable register 0 (PER0)
- Serial clock select register m (SPSm)
- Serial mode register mn (SMRmn)
- Serial communication operation setting register mn (SCRmn)
- Serial data register mn (SDRmn)
- Serial flag clear trigger register mn (SIRmn)
- Serial status register mn (SSRmn)
- Serial channel start register m (SSm)
- Serial channel stop register m (STm)
- Serial channel enable status register m (SEm)
- Serial output enable register m (SOEm)
- Serial output level register m (SOLm)
- Serial output register m (SOm)
- Serial standby control register m (SSCm)
- Input switch control register (ISC)
- Noise filter enable register 0 (NFEN0)
- Port input mode registers 0, 1, 3, 5 (PIM0, PIM1, PIM3, PIM5)
- Port output mode registers 0, 1, 3, 5, 7 (POM0, POM1, POM3, POM5, POM7)
- Port mode registers 0, 1, 3, 5 to 7 (PM0, PM1, PM3, PM5 to PM7)
- Port registers 0, 1, 3, 5 to 7 (P0, P1, P3, P5 to P7)

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.1 Peripheral enable register 0 (PER0)

PER0 is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When serial array unit 0 is used, be sure to set bit 2 (SAU0EN) of this register to 1.

When serial array unit 1 is used, be sure to set bit 3 (SAU1EN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears the PER0 register to 00H.

**Figure 17 - 5 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PER0	RTCEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
SAUmEN	Control of serial array unit m input clock supply							
0	Stops supply of input clock. • SFR used by serial array unit m cannot be written. • Serial array unit m is in the reset status.							
1	Enables input clock supply. • SFR used by serial array unit m can be read/written.							

**Note**      80 and 100-pin products only.

**Caution 1.** When setting serial array unit m, be sure to first set the following registers with the SAUmEN bit set to 1. If SAUmEN = 0, writing to a control register of serial array unit m is ignored, and, even if the register is read, only the default value is read (except for the input switch control register (ISC), noise filter enable register 0 (NFEN0), port input mode registers 0, 1, 3, 5 (PIM0, PIM1, PIM3, PIM5), port output mode registers 0, 1, 3, 5, 7 (POM0, POM1, POM3, POM5, POM7), port mode registers 0, 1, 3, 5 to 7 (PM0, PM1, PM3, PM5 to PM7), port mode control registers 0, 1, 10, 12, 14 (PMC0, PMC1, PMC10, PMC12, PMC14), and port registers 0, 1, 3, 5 to 7 (P0, P1, P3, P5 to P7).

- Serial clock select register m (SPSm)
- Serial mode register mn (SMRmn)
- Serial communication operation setting register mn (SCRmn)
- Serial data register mn (SDRmn)
- Serial flag clear trigger register mn (SIRmn)
- Serial status register mn (SSRmn)
- Serial channel start register m (SSm)
- Serial channel stop register m (STm)
- Serial channel enable status register m (SEm)
- Serial output enable register m (SOEm)
- Serial output level register m (SOLm)
- Serial output register m (SOm)
- Serial standby control register m (SSCm)

**Caution 2.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

### 17.3.2 Serial clock select register m (SPSm)

The SPSm register is a 16-bit register that is used to select two types of operation clocks (CKm0, CKm1) that are commonly supplied to each channel. CKm1 is selected by bits 7 to 4 of the SPSm register, and CKm0 is selected by bits 3 to 0.

Rewriting the SPSm register is prohibited when the register is in operation (when SEmn = 1).

The SPSm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SPSm register can be set with an 8-bit memory manipulation instruction with SPSmL.

Reset signal generation clears the SPSm register to 0000H.

**Figure 17 - 6 Format of Serial clock select register m (SPSm)**

Address: F0126H, F0127H (SPS0), F0166H, F0167H (SPS1) After reset: 0000H R/W

Symbol 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0

SPSm	0	0	0	0	0	0	0	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00
------	---	---	---	---	---	---	---	---------	---------	---------	---------	---------	---------	---------	---------

PRS mk3	PRS mk2	PRS mk1	PRS mk0	Section of operation clock (CKmk) Note					
				fCLK = 2 MHz	fCLK = 5 MHz	fCLK = 10 MHz	fCLK = 20 MHz	fCLK = 32 MHz	
0	0	0	0	fCLK	2 MHz	5 MHz	10 MHz	20 MHz	32 MHz
0	0	0	1	fCLK/2	1 MHz	2.5 MHz	5 MHz	10 MHz	16 MHz
0	0	1	0	fCLK/2 <sup>2</sup>	500 kHz	1.25 MHz	2.5 MHz	5 MHz	8 MHz
0	0	1	1	fCLK/2 <sup>3</sup>	250 kHz	625 kHz	1.25 MHz	2.5 MHz	4 MHz
0	1	0	0	fCLK/2 <sup>4</sup>	125 kHz	313 kHz	625 kHz	1.25 MHz	2 MHz
0	1	0	1	fCLK/2 <sup>5</sup>	62.5 kHz	156 kHz	313 kHz	625 kHz	1 MHz
0	1	1	0	fCLK/2 <sup>6</sup>	31.3 kHz	78.1 kHz	156 kHz	313 kHz	500 kHz
0	1	1	1	fCLK/2 <sup>7</sup>	15.6 kHz	39.1 kHz	78.1 kHz	156 kHz	250 kHz
1	0	0	0	fCLK/2 <sup>8</sup>	7.81 kHz	19.5 kHz	39.1 kHz	78.1 kHz	125 kHz
1	0	0	1	fCLK/2 <sup>9</sup>	3.91 kHz	9.77 kHz	19.5 kHz	39.1 kHz	62.5 kHz
1	0	1	0	fCLK/2 <sup>10</sup>	1.95 kHz	4.88 kHz	9.77 kHz	19.5 kHz	31.3 kHz
1	0	1	1	fCLK/2 <sup>11</sup>	977 Hz	2.44 kHz	4.88 kHz	9.77 kHz	15.6 kHz
1	1	0	0	fCLK/2 <sup>12</sup>	488 Hz	1.22 kHz	2.44 kHz	4.88 kHz	7.8 kHz
1	1	0	1	fCLK/2 <sup>13</sup>	244 Hz	610 Hz	1.22 kHz	2.44 kHz	3.9 kHz
1	1	1	0	fCLK/2 <sup>14</sup>	122 Hz	305 Hz	610 Hz	1.22 kHz	1.95 kHz
1	1	1	1	fCLK/2 <sup>15</sup>	61 Hz	153 Hz	305 Hz	610 Hz	977 Hz

**Note** When changing the clock selected for fCLK (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (STm) = 000FH) the operation of the serial array unit (SAU).

**Caution** Be sure to clear bits 15 to 8 to “0”.

**Remark 1.** fCLK: CPU/peripheral hardware clock frequency

**Remark 2.** m: Unit number (m = 0, 1)

**Remark 3.** k = 0, 1

### 17.3.3 Serial mode register mn (SMRmn)

The SMRmn register is a register that sets an operation mode of channel n. It is also used to select an operation clock (fmck), specify whether the serial clock (fsck) may be input or not, set a start trigger, an operation mode (CSI, UART, or simplified I<sup>2</sup>C), and an interrupt source. This register is also used to invert the level of the receive data only in the UART mode.

Rewriting the SMRmn register is prohibited when the register is in operation (when SEmn = 1). However, the MDmn0 bit can be rewritten during operation.

The SMRmn register can be set by a 16-bit memory manipulation instruction.

Reset signal generation sets the SMRmn register to 0020H.

**Figure 17 - 7 Format of Serial mode register mn (SMRmn) (1/2)**

Address: F0110H, F0111H (SMR00) to F0116H, F0117H (SMR03), After reset: 0020H R/W  
F0150H, F0151H (SMR10) to F0156H, F0157H (SMR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKS mn	CCS mn	0	0	0	0	0	STS mn Note 2	0	SIS mn0 Note 2	1	0	0	MD mn2	MD mn1	MD mn0

CKS mn	Selection of operation clock (fmck) of channel n
0	Operation clock CKm0 set by the SPSm register
1	Operation clock CKm1 set by the SPSm register
Operation clock (fmck) is used by the edge detector. In addition, depending on the setting of the CCSmn bit and the higher 7 bits of the SDRmn register, a transfer clock (ftclk) is generated.	

CCS mn	Selection of transfer clock (ftclk) of channel n
0	Divided operation clock fmck specified by the CKSmn bit
1	Clock input fsck from the SCKp pin (slave transfer in CSI mode)
Transfer clock ftclk is used for the shift register, communication controller, output controller, interrupt controller, and error controller. When CCSmn = 0, the division ratio of operation clock (fmck) is set by the higher 7 bits of the SDRmn register.	

STS mn Note 2	Selection of start trigger source
0	Only software trigger is valid (selected for CSI, UART transmission, and simplified I <sup>2</sup> C).
1	Valid edge of the RxDq pin (selected for UART reception)
Transfer is started when the above source is satisfied after 1 is set to the SSm register.	

**Note 1.** SMR00 to SMR03, SMR10, SMR11: All products  
SMR12, SMR13: 80-, 100-pin products

**Note 2.** The SMR01, SMR03, SMR11, and SMR13 registers only.

**Caution** Be sure to clear bits 13 to 9, 7, 4, and 3 (or bits 13 to 6, 4, and 3 for the SMR00, SMR02, SMR10, or SMR12 register) to “0”. Be sure to set bit 5 to “1”.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), q: UART number (q = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)

**Figure 17 - 8 Format of Serial mode register mn (SMRmn) (2/2)**

Address: F0110H, F0111H (SMR00) to F0116H, F0117H (SMR03), After reset: 0020H R/W  
 F0150H, F0151H (SMR10) to F0156H, F0157H (SMR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SMRmn	CKS mn	CCS mn	0	0	0	0	0	STS mn Note 2	0	SIS mn0 Note 2	1	0	0	MD mn2	MD mn1	MD mn0
SIS mn0 Note 2		Controls inversion of level of receive data of channel n in UART mode														
0		Falling edge is detected as the start bit. The input communication data is captured as is.														
1		Rising edge is detected as the start bit. The input communication data is inverted and captured.														
MD mn2		Setting of operation mode of channel n														
0		CSI mode														
0		UART mode														
1		Simplified I2C mode														
1		Setting prohibited														
MD mn0		Selection of interrupt source of channel n														
0		Transfer end interrupt														
1		Buffer empty interrupt (Occurs when data is transferred from the SDRmn register to the shift register.)														
For successive transmission, the next transmit data is written by setting the MDmn0 bit to 1 when SDRmn data has run out.																

**Note 1.** SMR00 to SMR03, SMR10, SMR11: All products  
 SMR12, SMR13: 80-, 100-pin products

**Note 2.** The SMR01, SMR03, SMR11, and SMR13 registers only.

**Caution** Be sure to clear bits 13 to 9, 7, 4, and 3 (or bits 13 to 6, 4, and 3 for the SMR00, SMR02, SMR10, or SMR12 register) to "0". Be sure to set bit 5 to "1".

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to 3), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ ), q: UART number ( $q = 0$  to 3), r: IIC number ( $r = 00, 01, 10, 11, 20, 21, 30, 31$ )

### 17.3.4 Serial communication operation setting register mn (SCRmn)

The SCRmn register is a communication operation setting register of channel n. It is used to set a data transmission/reception mode, phase of data and clock, whether an error signal is to be masked or not, parity bit, start bit, stop bit, and data length.

Rewriting the SCRmn register is prohibited when the register is in operation (when SEMn = 1).

The SCRmn register can be set by a 16-bit memory manipulation instruction.

Reset signal generation sets the SCRmn register to 0087H.

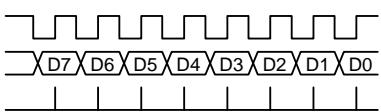
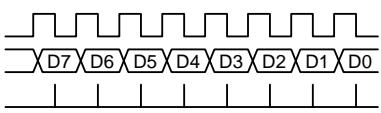
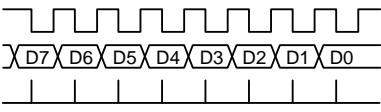
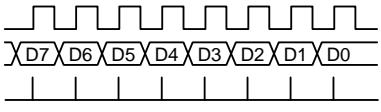
**Figure 17 - 9 Format of Serial communication operation setting register mn (SCRmn) (1/2)**

Address: F0118H, F0119H (SCR00) to F011EH, F011FH (SCR03), After reset: 0087H R/W  
 F0158H, F0159H (SCR10) to F015EH, F015FH (SCR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--------	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

SCRmn	TXE mn	RXE mn	DAP mn	CKP mn	0	EOC mn	PTC mn1	PTC mn0	DIR mn	0	SLCm n1 Note 2	SLC mn0	0	1	DLSm n1 Note 3	DLS mn0
	Setting of operation mode of channel n															

TXE mn	RXE mn	Setting of operation mode of channel n													
0	0	Disable communication.													
0	1	Reception only													
1	0	Transmission only													
1	1	Transmission/reception													

DAP mn	CKP mn	Selection of data and clock phase in CSI mode														Type
0	0			SCKp												1
0	1			SCKp												2
1	0			SCKp												3
1	1			SCKp												4
Be sure to set DAPmn, CKPmn = 0, 0 in the UART mode and simplified I <sup>2</sup> C mode.																

EOC mn	Mask control of error interrupt signal (INTSREx (x = 0 to 3))														
0	Disables generation of error interrupt INTSREx (INTSRx is generated).														
1	Enables generation of error interrupt INTSREx (INTSRx is not generated if an error occurs).														
Set EOCmn = 0 in the CSI mode, simplified I <sup>2</sup> C mode, and during UART transmission Note 4.															

**Note 1.** SCR00 to SCR03, SCR10, SCR11: All products  
 SCR12, SCR13: 80-, 100-pin products

**Note 2.** The SCR00, SCR02, SCR10, and SCR12 registers only.

**Note 3.** The SCR00 and SCR01 registers and SCR10 and SCR11 registers for 80, 100-pin products only.  
 Others are fixed to 1.

**Note 4.** When using CSImn not with EOCmn = 0, error interrupt INTSREn may be generated.

**Caution** Be sure to clear bits 3, 6, and 11 to "0" (Also clear bit 5 of the SCR01, SCR03, SCR11, or SCR13 register to 0). Be sure to set bit 2 to "1".

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)

**Figure 17 - 10 Format of Serial communication operation setting register mn (SCRmn) (2/2)**

Address: F0118H, F0119H (SCR00) to F011EH, F011FH (SCR03), After reset: 0087H R/W  
 F0158H, F0159H (SCR10) to F015EH, F015FH (SCR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCRmn	TXE mn	RXE mn	DAP mn	CKP mn	0	EOC mn	PTC mn1	PTC mn0	DIR mn	0	SLCm n1 Note 2	SLC mn0	0	1	DLSm n1 Note 3	DLS mn0

PTC mn1	PTC mn0	Setting of parity bit in UART mode					
		Transmission			Reception		
0	0	Does not output the parity bit.				Receives without parity	
0	1	Outputs 0 parity Note 4.				No parity judgment	
1	0	Outputs even parity.				Judged as even parity.	
1	1	Outputs odd parity.				Judges as odd parity.	

Be sure to set PTCmn1, PTCmn0 = 0, 0 in the CSI mode and simplified I<sup>2</sup>C mode.

DIR mn	Selection of data transfer sequence in CSI and UART modes					
0	Inputs/outputs data with MSB first.					
1	Inputs/outputs data with LSB first.					

Be sure to clear DIRmn = 0 in the simplified I<sup>2</sup>C mode.

SLCmn1 Note 2	SLC mn0	Setting of stop bit in UART mode					
0	0	No stop bit					
0	1	Stop bit length = 1 bit					
1	0	Stop bit length = 2 bits (mn = 00, 02, 10, 12 only)					
1	1	Setting prohibited					

When the transfer end interrupt is selected, the interrupt is generated when all stop bits have been completely transferred.

Set 1 bit (SLCmn1, SLCmn0 = 0, 1) during UART reception and in the simplified I<sup>2</sup>C mode.

Set no stop bit (SLCmn1, SLCmn0 = 0, 0) in the CSI mode.

Set 1 bit (SLCmn1, SLCmn0 = 0, 1) or 2 bits (SLCmn1, SLCmn0 = 1, 0) during UART transmission.

DLSmn1 Note 3	DLS mn0	Setting of data length in CSI and UART modes					
0	1	9-bit data length (stored in bits 0 to 8 of the SDRmn register) (settable in UART mode only)					
1	0	7-bit data length (stored in bits 0 to 6 of the SDRmn register)					
1	1	8-bit data length (stored in bits 0 to 7 of the SDRmn register)					
Other than above		Setting prohibited					

Be sure to set DLSmn1, DLSmn0 = 1, 1 in the simplified I<sup>2</sup>C mode.

**Note 1.** SCR00 to SCR03, SCR10, SCR11: All products

SCR12, SCR13: 80-, 100-pin products

**Note 2.** The SCR00, SCR02, SCR10, and SCR12 registers only.

**Note 3.** The SCR00 and SCR01 registers and SCR10 and SCR11 registers for 80, 100-pin products only.  
 Others are fixed to 1.

**Note 4.** 0 is always added regardless of the data contents.

**Caution** Be sure to clear bits 3, 6, and 11 to "0" (Also clear bit 5 of the SCR01, SCR03, SCR11, or SCR13 register to 0). Be sure to set bit 2 to "1".

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)

### 17.3.5 Serial data register mn (SDRmn)

The SDRmn register is the transmit/receive data register (16 bits) of channel n.

Bits 8 to 0 (lower 9 bits) of SDR00, SDR01, SDR10 Note 1, SDR11 Note 1 or bits 7 to 0 (lower 8 bits) of SDR02, SDR03, SDR10 Note 2, SDR11 Note 2, SDR12, and SDR13 function as a transmit/receive buffer register, and bits 15 to 9 (higher 7 bits) are used as a register that sets the division ratio of the operation clock (fmck).

If the CCSmn bit of serial mode register mn (SMRmn) is cleared to 0, the clock set by dividing the operating clock by the higher 7 bits of the SDRmn register is used as the transfer clock.

If the CCSmn bit of serial mode register mn (SMRmn) is set to 1, set bits 15 to 9 (upper 7 bits) of SDR00, SDR01, SDR10 Note 1, and SDR11 Note 1 to 0000000B. The input clock fsck (slave transfer in CSI mode) from the SCKp pin is used as the transfer clock.

The lower 8/9 bits of the SDRmn register function as a transmit/receive buffer register. During reception, the parallel data converted by the shift register is stored in the lower 8/9 bits, and during transmission, the data to be transmitted to the shift register is set to the lower 8/9 bits.

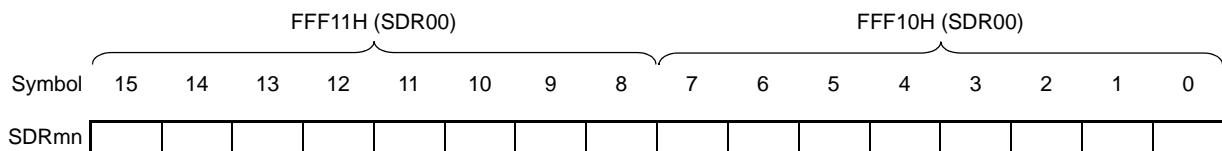
The SDRmn register can be read or written in 16-bit units.

However, the higher 7 bits can be written or read only when the operation is stopped ( $SEmn = 0$ ). During operation ( $SEmn = 1$ ), a value is written only to the lower 8/9 bits of the SDRmn register. When the SDRmn register is read during operation, the higher 7 bits are always read as 0.

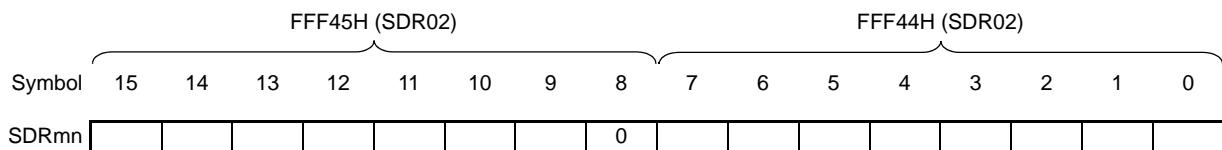
Reset signal generation clears the SDRmn register to 0000H.

**Figure 17 - 11 Format of Serial data register mn (SDRmn)**

Address: FFF10H, FFF11H (SDR00), FFF12H, FFF13H (SDR01) After reset: 0000H R/W  
FFF48H, FFF49H (SDR10) Note 1, FFF4AH, FFF4BH (SDR11) Note 1



Address: FFF44H, FFF45H (SDR02), FFF46H, FFF47H (SDR03) After reset: 0000H R/W  
FFF48H, FFF49H (SDR10) Note 2, FFF4AH, FFF4BH (SDR11) Note 2  
FFF14H, FFF15H (SDR12) Note 1, FFF16H, FFF17H (SDR13) Note 1



SDRmn[15:9]								Transfer clock set by dividing the operating clock							
0	0	0	0	0	0	0	0	fmck/2							
0	0	0	0	0	0	0	1	fmck/4							
0	0	0	0	0	0	1	0	fmck/6							
0	0	0	0	0	0	1	1	fmck/8							
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮							
1	1	1	1	1	1	1	0	fmck/254							
1	1	1	1	1	1	1	1	fmck/256							

**Note 1.** 80, 100-pin products

**Note 2.** 30 to 64-pin products

(Cautions and Remarks are listed on the next page.)

**Caution 1.** Be sure to clear bit 8 of the SDR02, SDR03, SDR12 or SDR13 register, and SDR10, or SDR11 register of 30 to 64-pin products to “0”.

**Caution 2.** Setting SDRmn[15:9] = (0000000B, 0000001B) is prohibited when UART is used.

**Caution 3.** Setting SDRmn[15:9] = 0000000B is prohibited when simplified I<sup>2</sup>C is used. Set SDRmn[15:9] to 0000001B or greater.

**Caution 4.** When operation is stopped (SEmn = 0), do not rewrite SDRmn [7:0] by an 8-bit memory manipulation instruction (SDRmn [15:9] are all cleared to 0).

**Remark 1.** For the function of the lower 8/9 bits of the SDRmn register, see **17.2 Configuration of Serial Array Unit**.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.6 Serial flag clear trigger register mn (SIRmn)

The SIRmn register is a trigger register that is used to clear each error flag of channel n.

When each bit (FECTmn, PECTmn, OVCTmn) of this register is set to 1, the corresponding bit (FEFmn, PEFmn, OVFmn) of serial status register mn is cleared to 0. Because the SIRmn register is a trigger register, it is cleared immediately when the corresponding bit of the SSRmn register is cleared.

The SIRmn register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SIRmn register can be set with an 8-bit memory manipulation instruction with SIRmnL.

Reset signal generation clears the SIRmn register to 0000H.

**Figure 17 - 12 Format of Serial flag clear trigger register mn (SIRmn)**

Address: F0108H, F0109H (SIR00) to F010EH, F010FH (SIR03), After reset: 0000H R/W  
F0148H, F0149H (SIR10) to F014EH, F014FH (SIR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SIRmn	0	0	0	0	0	0	0	0	0	0	0	0	0	FEC Tmn Note 2	PEC Tmn	OVC Tmn
FEC Tmn Note 2	Clear trigger of framing error of channel n															
0	Not cleared															
1	Clears the FEFmn bit of the SSRmn register to 0.															
PEC Tmn	Clear trigger of parity error flag of channel n															
0	Not cleared															
1	Clears the PEFmn bit of the SSRmn register to 0.															
OVC Tmn	Clear trigger of overrun error flag of channel n															
0	Not cleared															
1	Clears the OVFmn bit of the SSRmn register to 0.															

**Note 1.** SIR00 to SIR03, SIR10, SIR11: All products  
SIR12, SIR13: 80-, 100-pin products

**Note 2.** The SIR01, SIR03, SIR11, and SIR13 registers only.

**Caution** Be sure to clear bits 15 to 3 (or bits 15 to 2 for the SIR00, SIR02, SIR10, or SIR12 register) to "0".

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Remark 2.** When the SIRmn register is read, 0000H is always read.

### 17.3.7 Serial status register mn (SSRmn)

The SSRmn register is a register that indicates the communication status and error occurrence status of channel n. The errors indicated by this register are a framing error, parity error, and overrun error.

The SSRmn register can be read by a 16-bit memory manipulation instruction.

The lower 8 bits of the SSRmn register can be set with an 8-bit memory manipulation instruction with SSRmnL.

Reset signal generation clears the SSRmn register to 0000H.

**Figure 17 - 13 Format of Serial status register mn (SSRmn) (1/2)**

Address: F0100H, F0101H (SSR00) to F0106H, F0107H (SSR03), After reset: 0000H R

F0140H, F0141H (SSR10) to F0146H, F0147H (SSR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSRmn	0	0	0	0	0	0	0	0	0	TSF mn	BFF mn	0	0	FEF mn Note 2	PEF mn	OVF mn

TSF mn	Communication status indication flag of channel n
0	Communication is stopped or suspended.
1	Communication is in progress.
<Clear conditions>	
<ul style="list-style-type: none"> <li>The STmn bit of the STm register is set to 1 (communication is stopped) or the SSmn bit of the SSm register is set to 1 (communication is suspended).</li> <li>Communication ends.</li> </ul>	
<Set condition>	
<ul style="list-style-type: none"> <li>Communication starts.</li> </ul>	

BFF mn	Buffer register status indication flag of channel n
0	Valid data is not stored in the SDRmn register.
1	Valid data is stored in the SDRmn register.
<Clear conditions>	
<ul style="list-style-type: none"> <li>Transferring transmit data from the SDRmn register to the shift register ends during transmission.</li> <li>Reading receive data from the SDRmn register ends during reception.</li> <li>The STmn bit of the STm register is set to 1 (communication is stopped) or the SSmn bit of the SSm register is set to 1 (communication is enabled).</li> </ul>	
<Set conditions>	
<ul style="list-style-type: none"> <li>Transmit data is written to the SDRmn register while the TXEmn bit of the SCRmn register is set to 1 (transmission or transmission and reception mode in each communication mode).</li> <li>Receive data is stored in the SDRmn register while the RXEmn bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).</li> <li>A reception error occurs.</li> </ul>	

**Note 1.** SSR00 to SSR03, SSR10, SSR11: All products  
SSR12, SSR13: 80-, 100-pin products

**Note 2.** The SSR01, SSR03, SSR11, and SSR13 registers only.

**Caution** When the CSI is performing reception operations in the SNOOZE mode (SWCm = 1), the BFFmn flag will not change.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Figure 17 - 14 Format of Serial status register mn (SSRmn) (2/2)**

Address: F0100H, F0101H (SSR00) to F0106H, F0107H (SSR03), After reset: 0000H R  
 F0140H, F0141H (SSR10) to F0146H, F0147H (SSR13) Note 1

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
--------	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

SSRmn	0	0	0	0	0	0	0	0	0	TSF mn	BFF mn	0	0	FEF mn Note 2	PEF mn	OVF mn
-------	---	---	---	---	---	---	---	---	---	--------	--------	---	---	---------------	--------	--------

FEF mn Note 2	Framing error detection flag of channel n
0	No error occurs.
1	An error occurs (during UART reception).
<Clear condition>	
• 1 is written to the FECTmn bit of the SIRmn register.	
<Set condition>	
• A stop bit is not detected when UART reception ends.	

PEF mn	Parity error detection flag of channel n
0	No error occurs.
1	An error occurs (during UART reception) or ACK is not detected (during I <sup>2</sup> C transmission).
<Clear condition>	
• 1 is written to the PECTmn bit of the SIRmn register.	
<Set condition>	
• The parity of the transmit data and the parity bit do not match when UART reception ends (parity error).	
• No ACK signal is returned from the slave channel at the ACK reception timing during I <sup>2</sup> C transmission (ACK is not detected).	

OVF mn	Overrun error detection flag of channel n
0	No error occurs.
1	An error occurs
<Clear condition>	
• 1 is written to the OVCTmn bit of the SIRmn register.	
<Set condition>	
• Even though receive data is stored in the SDRmn register, that data is not read and transmit data or the next receive data is written while the RXEmn bit of the SCRmn register is set to 1 (reception or transmission and reception mode in each communication mode).	
• Transmit data is not ready for slave transmission or transmission and reception in CSI mode.	

**Note 1.** SSR00 to SSR03, SSR10, SSR11: All products  
 SSR12, SSR13: 80-, 100-pin products

**Note 2.** The SSR01, SSR03, SSR11, and SSR13 registers only.

**Caution 1.** If data is written to the SDRmn register when BFFmn = 1, the transmit/receive data stored in the register is discarded and an overrun error (OVEmn = 1) is detected.

**Caution 2.** When the CSI is performing reception operations in the SNOOZE mode (SWCm = 1), the OVFmn flag will not change.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.8 Serial channel start register m (SSm)

The SSm register is a trigger register that is used to enable starting communication/count by each channel.

When 1 is written a bit of this register (SSmn), the corresponding bit (SEmn) of serial channel enable status register m (SEm) is set to 1 (Operation is enabled). Because the SSmn bit is a trigger bit, it is cleared immediately when SEmn = 1.

The SSm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SSm register can be set with an 1-bit or 8-bit memory manipulation instruction with SSmL. Reset signal generation clears the SSm register to 0000H.

**Figure 17 - 15 Format of Serial channel start register m (SSm)**

Address: F0122H, F0123H (SS0)

After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS0	0	0	0	0	0	0	0	0	0	0	0	0	SS03	SS02	SS01	SS00

Address: F0162H, F0163H (SS1)

After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SS1	0	0	0	0	0	0	0	0	0	0	0	0	SS13	SS12	SS11	SS10

SSm n	Operation start trigger of channel n
0	No trigger operation
1	Sets the SEmn bit to 1 and enters the communication wait status Note.

**Note** If set the SSmn = 1 to during a communication operation, will wait status to stop the communication.

At this time, holding status value of control register and shift register, SCKmn and SOmn pins, and FEFmn, PEFmn, OVFmn flags.

**Caution 1.** Be sure to clear bits 15 to 4 of the SS0 register, bits 15 to 2 of the SS1 register for 30 to 64-pin products and bits 15 to 4 of the SS1 register for 80 to 100-pin products to “0”.

**Caution 2.** For the UART reception, set the RXEmn bit of SCRmn register to 1, and then be sure to set SSmn to 1 after 4 or more fmck clocks have elapsed.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Remark 2.** When the SSm register is read, 0000H is always read.

### 17.3.9 Serial channel stop register m (STm)

The STm register is a trigger register that is used to enable stopping communication/count by each channel.

When 1 is written a bit of this register (STmn), the corresponding bit (SEmn) of serial channel enable status register m (SEm) is cleared to 0 (operation is stopped). Because the STmn bit is a trigger bit, it is cleared immediately when SEMn = 0.

The STm register can set written by a 16-bit memory manipulation instruction.

The lower 8 bits of the STm register can be set with a 1-bit or 8-bit memory manipulation instruction with STmL.

Reset signal generation clears the STm register to 0000H.

**Figure 17 - 16 Format of Serial channel stop register m (STm)**

Address: F0124H, F0125H (ST0)

After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ST0	0	0	0	0	0	0	0	0	0	0	0	0	ST03	ST02	ST01	ST00

Address: F0164H, F0165H (ST1) After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ST1	0	0	0	0	0	0	0	0	0	0	0	0	ST13	ST12	ST11	ST10

STm n	Operation stop trigger of channel n
0	No trigger operation
1	Clears the SEMn bit to 0 and stops the communication operation Note.

**Note** Holding status value of the control register and shift register, the SCKmn and SOmn pins, and FEFmn, PEFmn, OVFmn flags.

**Caution** Be sure to clear bits 15 to 4 of the ST0 register, bits 15 to 2 of the ST1 register for 30 to 64-pin products and bits 15 to 4 of the ST1 register for 80 to 100-pin products to “0”.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Remark 2.** When the STm register is read, 0000H is always read.

### 17.3.10 Serial channel enable status register m (SEm)

The SEm register indicates whether data transmission/reception operation of each channel is enabled or stopped.

When 1 is written a bit of serial channel start register m (SSm), the corresponding bit of this register is set to 1.

When 1 is written a bit of serial channel stop register m (STm), the corresponding bit is cleared to 0.

Channel n that is enabled to operate cannot rewrite by software the value of the CKOmn bit (serial clock output of channel n) of serial output register m (SOm) to be described below, and a value reflected by a communication operation is output from the serial clock pin.

Channel n that stops operation can set the value of the CKOmn bit of the SOm register by software and output its value from the serial clock pin. In this way, any waveform, such as that of a start condition/stop condition, can be created by software.

The SEm register can be read by a 16-bit memory manipulation instruction.

The lower 8 bits of the SEm register can be set with a 1-bit or 8-bit memory manipulation instruction with SEmL.

Reset signal generation clears the SEm register to 0000H.

**Figure 17 - 17 Format of Serial channel enable status register m (SEm)**

Address: F0120H, F0121H (SE0)

After reset: 0000H R

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE0	0	0	0	0	0	0	0	0	0	0	0	0	SE03	SE02	SE01	SE00

Address: F0160H, F0161H (SE1)

After reset: 0000H R

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SE1	0	0	0	0	0	0	0	0	0	0	0	0	SE13	SE12	SE11	SE10

SEm n	Indication of operation enable/stop status of channel n
0	Operation stops
1	Operation is enabled.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.11 Serial output enable register m (SOEm)

The SOEm register is a register that is used to enable or stop output of the serial communication operation of each channel.

Channel n that enables serial output cannot rewrite by software the value of the SOmn bit of serial output register m (SOm) to be described below, and a value reflected by a communication operation is output from the serial data output pin.

For channel n, whose serial output is stopped, the SOmn bit value of the SOm register can be set by software, and that value can be output from the serial data output pin. In this way, any waveform of the start condition and stop condition can be created by software.

The SOEm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SOEm register can be set with a 1-bit or 8-bit memory manipulation instruction with SOEmL.

Reset signal generation clears the SOEm register to 0000H.

**Figure 17 - 18 Format of Serial output enable register m (SOEm)**

Address: F012AH, F012BH After reset: 0000H R/W																
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE0	0	0	0	0	0	0	0	0	0	0	0	0	SOE 03	SOE 02	SOE 01	SOE 00
Address: F016AH, F016BH After reset: 0000H R/W																
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOE1	0	0	0	0	0	0	0	0	0	0	0	0	SOE 13	SOE 12	SOE 11	SOE 10
SOE mn	Serial output enable/stop of channel n															
0	Stops output by serial communication operation.															
1	Enables output by serial communication operation.															

**Caution** Be sure to clear bits 15 to 4 of the SOE0 register, bits 15 to 2 of the SOE1 register for 30 to 64-pin products and bits 15 to 4 of the SOE1 register for 80 to 100-pin products to “0”.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.12 Serial output register m (SOm)

The SOm register is a buffer register for serial output of each channel.

The value of the SOmn bit of this register is output from the serial data output pin of channel n.

The value of the CKOmn bit of this register is output from the serial clock output pin of channel n.

The SOmn bit of this register can be rewritten by software only when serial output is disabled (SOEmn = 0).

When serial output is enabled (SOEmn = 1), rewriting by software is ignored, and the value of the register can be changed only by a serial communication operation.

The CKOmn bit of this register can be rewritten by software only when the channel operation is stopped (SEmn = 0). While channel operation is enabled (SEmn = 1), rewriting by software is ignored, and the value of the CKOmn bit can be changed only by a serial communication operation.

To use a pin for the serial interface as a port function pin other than a serial interface function pin, set the corresponding the CKOmn and SOmn bits to 1.

The SOm register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears the SOm register to 0F0FH.

**Figure 17 - 19 Format of Serial output register m (SOm)**

																Address: F0128H, F0129H	After reset: 0F0FH				R/W
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
SO0	0	0	0	0	CKO 03	CKO 02	CKO 01	CKO 00	0	0	0	0	SO 03	SO 02	SO 01	SO 00					

																Address: F0168H, F0169H	After reset: 0F0FH Note				R/W
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0					
SO1	0	0	0	0	CKO 13	CKO 12	CKO 11	CKO 10	0	0	0	0	SO 13	SO 12	SO 11	SO 10					

CKO mn	Serial clock output of channel n
0	Serial clock output value is "0".
1	Serial clock output value is "1".

SO mn	Serial data output of channel n
0	Serial data output value is "0".
1	Serial data output value is "1".

**Note** The register value becomes 0303H after a reset for the 30 to 64-pin products.

**Caution** Be sure to clear bits 15 to 12 and 7 to 4 of the SO0 register to "0".

Be sure to clear bits 15 to 10 and 7 to 2 of the SO1 register for 30 to 64-pin products and bits 15 to 12 and 7 to 4 of the SO1 register for 80 to 100-pin products to "0".

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

### 17.3.13 Serial output level register m (SOLm)

The SOLm register is a register that is used to set inversion of the data output level of each channel.

This register can be set only in the UART mode. Be sure to set 0 for corresponding bit in the CSI mode and simplifies I<sup>2</sup>C mode.

Inverting channel n by using this register is reflected on pin output only when serial output is enabled (SOEmn = 1). When serial output is disabled (SOEmn = 0), the value of the SOmn bit is output as is.

Rewriting the SOLm register is prohibited when the register is in operation (when SEMn = 1).

The SOLm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SOLm register can be set with an 8-bit memory manipulation instruction with SOLmL.

Reset signal generation clears the SOLm register to 0000H.

**Figure 17 - 20 Format of Serial output level register m (SOLm)**

Address: F0134H, F0135H (SOL0)

After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOL0	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL 02	0	SOL 00

Address: F0174H, F0175H (SOL1)

After reset: 0000H R/W

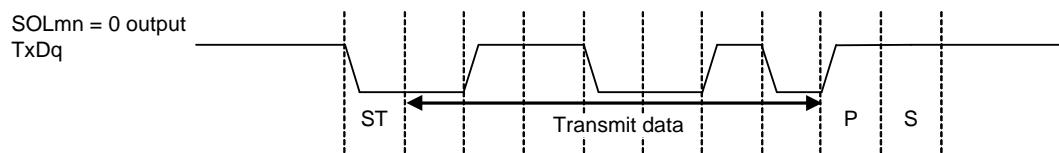
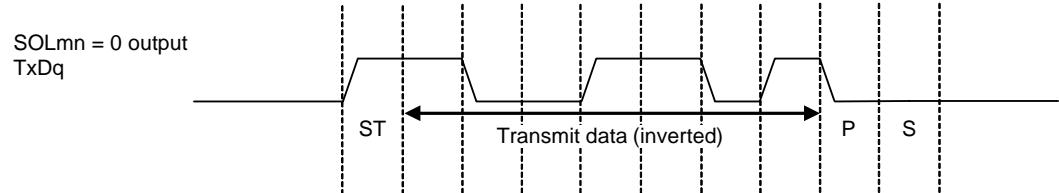
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOL1	0	0	0	0	0	0	0	0	0	0	0	0	0	SOL 12	0	SOL 10

SOL mn	Selects inversion of the level of the transmit data of channel n in UART mode
0	Communication data is output as is.
1	Communication data is inverted and output.

**Caution** Be sure to clear bits 15 to 3, and 1 of the SOL0 register, bits 15 to 1 of the SOL1 register for 30 to 64-pin products, and bits 15 to 3, and 1 of the SOL1 register for 80 to 100-pin products to "0".

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2)

Figure 17 - 21 shows examples in which the level of transmit data is reversed during UART transmission.

**Figure 17 - 21 Examples of Reverse Transmit Data**(a) Non-reverse Output ( $SOLmn = 0$ )(b) Reverse Output ( $SOLmn = 1$ )

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ )

### 17.3.14 Serial standby control register m (SSCm)

The SSC0 register is used to control the startup of reception (the SNOOZE mode) while in the STOP mode when receiving CSI00 or UART0 serial data.

The SSC1 register is used to control the startup of reception (the SNOOZE mode) while in the STOP mode when receiving CSI20 or UART2 serial data.

The SSCm register can be set by a 16-bit memory manipulation instruction.

The lower 8 bits of the SSCm register can be set with an 8-bit memory manipulation instruction with SSCmL.

Reset signal generation clears the SSCm register to 0000H.

**Caution The maximum transfer rate in the SNOOZE mode is as follows.**

- When using CSI00, CSI20: Up to 1 Mbps
- When using UART0, UART2: 4800 bps only  
(Can be used when FRQSEL4 in the option byte (000C2H) is set to 0.)

**Figure 17 - 22 Format of Serial standby control register m (SSCm)**

Address: F0138H (SSC0), F0178H (SSC1) Note After reset: 0000H R/W

Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSCm	0	0	0	0	0	0	0	0	0	0	0	0	0	0	SSECm	SWCm

SSECm	Selection of whether to enable or disable the generation of communication error interrupts in the SNOOZE mode
0	Enable the generation of error interrupts (INTSRE0/INTSRE2).
1	Disable the generation of error interrupts (INTSRE0/INTSRE2).
<ul style="list-style-type: none"> <li>• The SSECm bit can be set to 1 or 0 only when both the SWCm and EOCmn bits are set to 1 during UART reception in the SNOOZE mode. In other cases, clear the SSECm bit to 0.</li> <li>• Setting SSECm, SWCm = 1, 0 is prohibited.</li> </ul>	

SWCm	Setting of the SNOOZE mode
0	Do not use the SNOOZE mode function.
1	Use the SNOOZE mode function.
<ul style="list-style-type: none"> <li>• When there is a hardware trigger signal in the STOP mode, the STOP mode is exited, and A/D conversion is performed without operating the CPU (the SNOOZE mode).</li> <li>• The SNOOZE mode function can only be specified when the high-speed on-chip oscillator clock is selected for the CPU/peripheral hardware clock (fCLK). If any other clock is selected, specifying this mode is prohibited. Specifying this mode is also prohibited when using UART while FRQSEL4 in the option byte (000C2H) is set to 1.</li> <li>• Even when using SNOOZE mode, be sure to set the SWCm bit to 0 in normal operation mode and change it to 1 just before shifting to STOP mode.</li> </ul> <p>Also, be sure to change the SWCm bit to 0 after returning from STOP mode to normal operation mode.</p>	

**Note** 80 to 100-pin products only.

**Caution Setting SSECm, SWCm = 1, 0 is prohibited.**

**Figure 17 - 23 Interrupt in UART Reception Operation in SNOOZE Mode**

EOCmn Bit	SSECm Bit	Reception Ended Successfully	Reception Ended in an Error
0	0	INTSRx is generated.	INTSRx is generated.
0	1	INTSRx is generated.	INTSRx is generated.
1	0	INTSRx is generated.	INTSREx is generated.
1	1	INTSRx is generated.	No interrupt is generated.

### 17.3.15 Input switch control register (ISC)

The ISC1 and ISC0 bits of the ISC register are used to realize a LIN-bus communication operation by UART0 in coordination with an external interrupt and the timer array unit.

When bit 0 is set to 1, the input signal of the serial data input (RxD0) pin is selected as an external interrupt (INTP0) that can be used to detect a wakeup signal.

When bit 1 is set to 1, the input signal of the serial data input (RxD0) pin is selected as a timer input, so that wake up signal can be detected, the low width of the break field, and the pulse width of the sync field can be measured by the timer.

The SSIE0 bit controls the SSI00 pin input of channel 0 during CSI00 communication and in slave mode.

While a high level is being input to the SSI00 pin, no transmission/reception operation is performed even if a serial clock is input. While a low level is being input to the SSI00 pin, a transmission/reception operation is performed according to each mode setting if a serial clock is input.

The ISC register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears the ISC register to 00H.

**Figure 17 - 24 Format of Input switch control register (ISC)**

Address: F0073H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ISC	SSIE00	0	0	0	0	0	ISC1	ISC0
SSIE00	Channel 0 <u>SSI00</u> input setting in CSI communication and slave mode							
0	Disables <u>SSI00</u> pin input.							
1	Enables <u>SSI00</u> pin input.							
ISC1	Switching channel 3 input of timer array unit 0							
0	Uses the input signal of the TI03 pin as a timer input (normal operation).							
1	Input signal of the RxD0 pin is used as timer input (detects the wakeup signal and measures the low width of the break field and the pulse width of the sync field).							
ISC0	Switching external interrupt (INTP0) input							
0	Uses the input signal of the INTP0 pin as an external interrupt (normal operation).							
1	Uses the input signal of the RxD0 pin as an external interrupt (wakeup signal detection).							

**Caution   Be sure to clear bits 6 to 2 to “0”.**

### 17.3.16 Noise filter enable register 0 (NFENO)

The NFENO register is used to set whether the noise filter can be used for the input signal from the serial data input pin to each channel.

Disable the noise filter of the pin used for CSI or simplified I<sup>2</sup>C communication, by clearing the corresponding bit of this register to 0.

Enable the noise filter of the pin used for UART communication, by setting the corresponding bit of this register to 1.

When the noise filter is enabled, after synchronization is performed with the operation clock (fmck) of the target channel, 2-clock match detection is performed. When the noise filter is OFF, only synchronization is performed with the Operation clock of target channel (fmck).

The NFENO register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears the NFENO register to 00H.

**Figure 17 - 25 Format of Noise filter enable register 0 (NFENO)**

Address: F0070H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

NFENO	0	SNFEN30	0	SNFEN20	0	SNFEN10	0	SNFEN00
-------	---	---------	---	---------	---	---------	---	---------

SNFEN30	Use of noise filter of RxD3 pin
---------	---------------------------------

0	Noise filter OFF
---	------------------

1	Noise filter ON
---	-----------------

Set SNFEN30 to 1 to use the RxD3 pin.

Clear SNFEN30 to 0 to use the other than RxD3 pin.

SNFEN20	Use of noise filter of RxD2 pin
---------	---------------------------------

0	Noise filter OFF
---	------------------

1	Noise filter ON
---	-----------------

Set SNFEN20 to 1 to use the RxD2 pin.

Clear SNFEN20 to 0 to use the other than RxD2 pin.

SNFEN10	Use of noise filter of RxD1 pin
---------	---------------------------------

0	Noise filter OFF
---	------------------

1	Noise filter ON
---	-----------------

Set the SNFEN10 bit to 1 to use the RxD1 pin.

Clear the SNFEN10 bit to 0 to use the other than RxD1 pin.

SNFEN00	Use of noise filter of RxD0 pin
---------	---------------------------------

0	Noise filter OFF
---	------------------

1	Noise filter ON
---	-----------------

Set the SNFEN00 bit to 1 to use the RxD0 pin.

Clear the SNFEN00 bit to 0 to use the other than RxD0 pin.

**Caution** Be sure to clear bits 7 to 5, 3, and 1 for 30 to 64-pin products, bits 7, 5, 3, and 1 for 80 to 100-pin products to “0”.

### 17.3.17 Registers controlling port functions of serial input/output pins

Using the serial array unit requires setting of the registers that control the port functions multiplexed on the target channel (port mode register (PMxx), port register (Pxx), port input mode register (PIMxx), port output mode register (POMxx), port mode control register (PMCxx)).

For details, see [4.3.1 Port mode registers \(PMxx\)](#), [4.3.2 Port registers \(Pxx\)](#), [4.3.4 Port input mode registers \(PMxx\)](#), [4.3.5 Port output mode registers \(POMxx\)](#), and [4.3.6 Port mode control registers \(PMCxx\)](#).

Specifically, using a port pin with a multiplexed serial data or serial clock output function (e.g. P02/ANI17/SO10/TxD1) for serial data or serial clock output, requires setting the corresponding bits in the port mode control register (PMCxx) and port mode register (PMxx) to 0, and the corresponding bit in the port register (Pxx) to 1.

When using the port pin in N-ch open-drain output ( $V_{DD}$  tolerance Note 1/ $EV_{DD}$  tolerance Note 2) mode, set the corresponding bit in the port output mode register (POMxx) to 1. When connecting an external device operating on a different potential (1.8 V, 2.5 V or 3 V), see [4.4.5 Handling different potential \(1.8 V, 2.5 V, 3 V\) by using I/O buffers](#).

Example When P02/ANI17/SO10/TxD1 is to be used for serial data output

- Set the PMC02 bit of port mode control register 0 to 0.
- Set the PM02 bit of port mode register 0 to 0.
- Set the P02 bit of port register 0 to 1.

Specifically, using a port pin with a multiplexed serial data or serial clock input function (e.g. P03/ANI16/SI10/RxD1/SDA10) for serial data or serial clock input, requires setting the corresponding bit in the port mode register (PMxx) to 1, and the corresponding bit in the port mode control register (PMCxx) to 0. In this case, the corresponding bit in the port register (Pxx) can be set to 0 or 1.

When the TTL input buffer is selected, set the corresponding bit in the port input mode register (PIMxx) to 1.

When connecting an external device operating on a different potential (1.8 V, 2.5 V or 3 V), see [4.4.5 Handling different potential \(1.8 V, 2.5 V, 3 V\) by using I/O buffers](#).

Example When P03/ANI16/SI10/RxD1/SDA10 is to be used for serial data input

- Set the PMC03 bit of port mode control register 0 to 0.
- Set the PM03 bit of port mode register 0 to 1.
- Set the P03 bit of port register 0 to 0 or 1.

**Note 1.** 30 to 52-pin products

**Note 2.** 64 to 100-pin products

## 17.4 Operation Stop Mode

Each serial interface of serial array unit has the operation stop mode.

In this mode, serial communication cannot be executed, thus reducing the power consumption.

In addition, the pin for serial interface can be used as port function pins in this mode.

### 17.4.1 Stopping the operation by units

The stopping of the operation by units is set by using peripheral enable register 0 (PER0).

The PER0 register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

To stop the operation of serial array unit 0, set bit 2 (SAU0EN) to 0.

To stop the operation of serial array unit 1, set bit 3 (SAU1EN) to 0.

**Figure 17 - 26 Peripheral Enable Register 0 (PER0) Setting When Stopping the Operation by Units**

- (a) Peripheral enable register 0 (PER0)... Set only the bit of SAUm to be stopped to 0.

	7	6	5	4	3	2	1	0
PER0	RTCEN x	IICA1EN Note x	ADCEN x	IICA0EN x	SAU1EN 0/1	SAU0EN 0/1	TAU1EN Note x	TAU0EN x
Control of SAUm input clock 0: Stops supply of input clock 1: Supplies input clock								

**Note** 80 and 100-pin products only.

**Caution 1.** If SAUmEN = 0, writing to a control register of serial array unit m is ignored, and, even if the register is read, only the default value is read

Note that this does not apply to the following registers.

- Input switch control register (ISC)
- Noise filter enable register 0 (NFENO0)
- Port input mode registers 0, 1, 3, 5 (PIM0, PIM1, PIM3, PIM5)
- Port output mode registers 0, 1, 3, 5, 7 (POM0, POM1, POM3, POM5, POM7)
- Port mode registers 0, 1, 3, 5 to 7 (PM0, PM1, PM3, PM5 to PM7)
- Port registers 0, 1, 3, 5 to 7 (P0, P1, P3, P5 to P7)

**Caution 2.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

**Remark** x: Bits not used with serial array units (depending on the settings of other peripheral functions)

0/1: Set to 0 or 1 depending on the usage of the user

### 17.4.2 Stopping the operation by channels

The stopping of the operation by channels is set using each of the following registers.

**Figure 17 - 27 Each Register Setting When Stopping the Operation by Channels**

- (a) Serial channel stop register m (STm)... This register is a trigger register that is used to enable stopping communication/count by each channel.

STm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	STm3 Note 0/1	STm2 Note 0/1	STm1 0/1	STm0 0/1

1: Clears the SEmn bit to 0 and stops the communication operation.

\* Because the STmn bit is a trigger bit, it is cleared immediately when SEmn = 0.

- (b) Serial Channel Enable Status Register m (SEm)... This register indicates whether data transmission/reception operation of each channel is enabled or stopped.

SEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SEm3 Note 0/1	SEm2 Note 0/1	SEm1 0/1	SEm0 0/1

0: Operation stops.

\* The SEm register is a read-only status register, whose operation is stopped by using the STm register.

With a channel whose operation is stopped, the value of the CKOm bit of the SOm register can be set by software.

- (c) Serial output enable register m (SOEm)... This register is a register that is used to enable or stop output of the serial communication operation of each channel.

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 Note 0/1	SOEm2 Note 0/1	SOEm1 0/1	SOEm0 0/1

0: Stops output by serial communication operation.

\* For channel n, whose serial output is stopped, the SOmn bit value of the SOm register can be set by software.

- (d) Serial output register m (SOm)... This register is a buffer register for serial output of each channel.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 0/1	CKOm2 0/1	CKOm1 0/1	CKOm0 0/1	0	0	0	0	SOm3 Note 0/1	SOm2 Note 0/1	SOm1 0/1	SOm0 0/1

1: Serial clock output value is "1".

1: Serial data output value is "1".

\* When using pins corresponding to each channel as port function pins, set the corresponding CKOm, SOmn bits to "1".

**Note** For serial array unit 1, 80 to 100-pin products only.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3)

**Remark 2.** Setting disabled (set to the initial value)

0/1: Set to 0 or 1 depending on the usage of the user

## 17.5 Operation of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) Communication

This is a clocked communication function that uses three lines: serial clock (SCK) and serial data (SI and SO) lines.

[Data transmission/reception]

- Data length of 7 or 8 bits
- Phase control of transmit/receive data
- MSB/LSB first selectable

[Clock control]

- Master/slave selection
- Phase control of I/O clock
- Setting of transfer period by prescaler and internal counter of each channel
- Maximum transfer rate Note

During master communication: Max. fCLK/2 (CSI00 only)

Max. fCLK/4

During slave communication: Max. fmck/6

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt

[Error detection flag]

- Overrun error

CSIs of following channels supports the SNOOZE mode. When SCK input is detected while in the STOP mode, the SNOOZE mode makes data reception that does not require the CPU possible. Only following CSIs can be specified.

- 30 to 64-pin products: CSI00
- 80, 100-pin products: CSI00 and CSI2

**Note** Use the clocks within a range satisfying the SCK cycle time ( $t_{kcy}$ ) characteristics. For details, see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**.

The channels supporting 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) are channels 0 to 3 of SAU0 and channels 0 to 3 of SAU1.

- 30, 32-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	—		—

- 36, 40, 44-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 48, 52-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 64-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 80, 100-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21
	2	CSI30	UART3	IIC30
	3	CSI31		IIC31

3-wire serial I/O (CSI00, CSI01, CIS10, CIS11, CIS20, CIS21, CIS30, CIS31) performs the following seven types of communication operations.

- Master transmission (See 17.5.1.)
- Master reception (See 17.5.2.)
- Master transmission/reception (See 17.5.3.)
- Slave transmission (See 17.5.4.)
- Slave reception (See 17.5.5.)
- Slave transmission/reception (See 17.5.6.)
- SNOOZE mode function (CSI00 only) (See 17.5.7.)

### 17.5.1 Master transmission

Master transmission is that the RL78 microcontroller outputs a transfer clock and transmits data to another device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SO00	SCK01, SO01	SCK10, SO10	SCK11, SO11	SCK20, SO20	SCK21, SO21	SCK30, SO30	SCK31, SO31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.							
Error detection flag	None							
Transfer data length	7 or 8 bits							
Transfer rate <small>Note</small>	Max. fCLK/2 [Hz] (CSI00 only), fCLK/4 [Hz] Min. fCLK/(2 × 2 <sup>15</sup> × 128) [Hz] fCLK: System clock frequency							
Data phase	Selectable by the DAPmn bit of the SCRmn register • DAPmn = 0: Data output starts from the start of the operation of the serial clock. • DAPmn = 1: Data output starts half a clock before the start of the serial clock operation.							
Clock phase	Selectable by the CKPmn bit of the SCRmn register • CKPmn = 0: Non-reverse • CKPmn = 1: Reverse							
Data direction	MSB or LSB first							

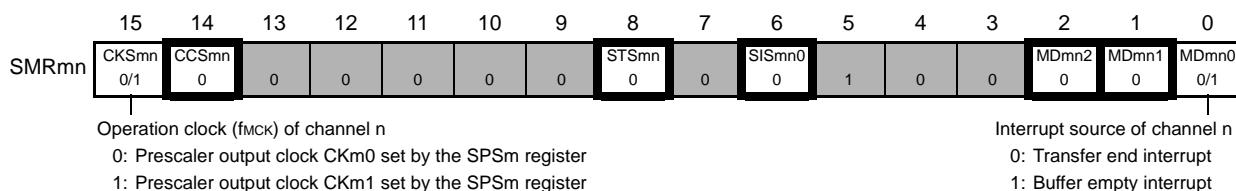
**Note** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

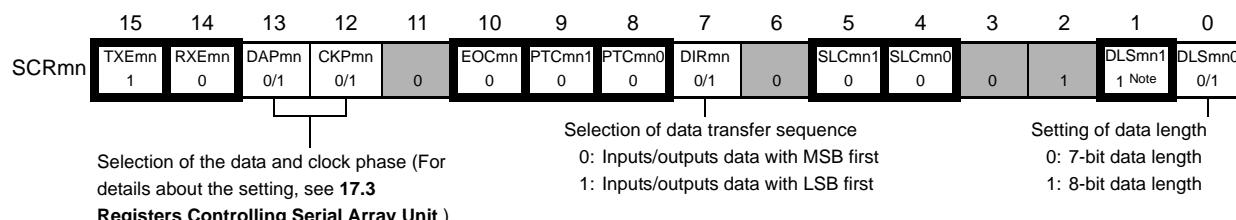
## (1) Register setting

**Figure 17 - 28 Example of Contents of Registers for Master Transmission of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)**

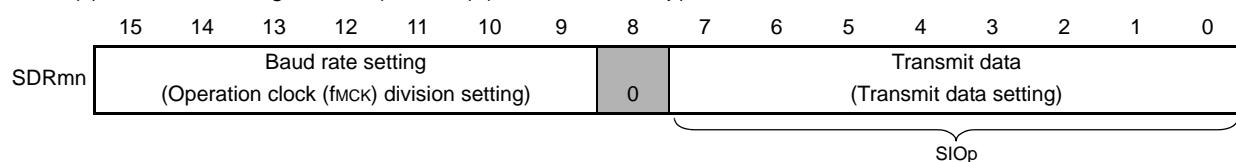
## (a) Serial mode register mn (SMRmn)



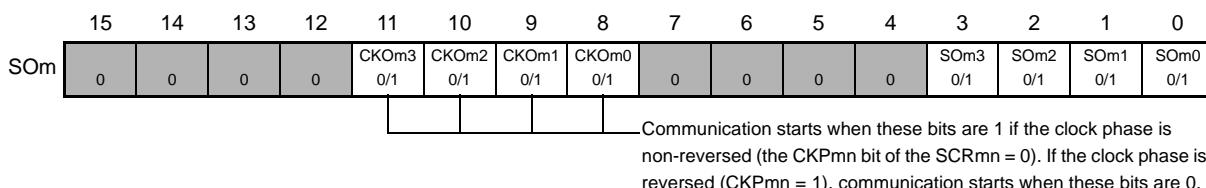
## (b) Serial communication operation setting register mn (SCRmn)



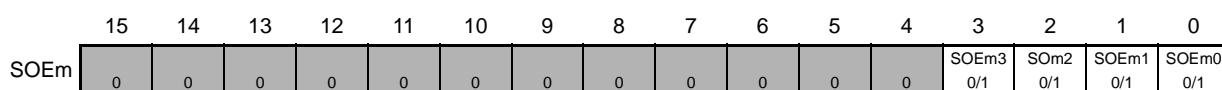
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



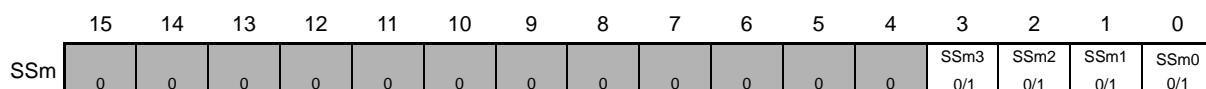
## (d) Serial output register m (SOm)... Sets only the bits of the target channel.



## (e) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.



## (f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.



**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

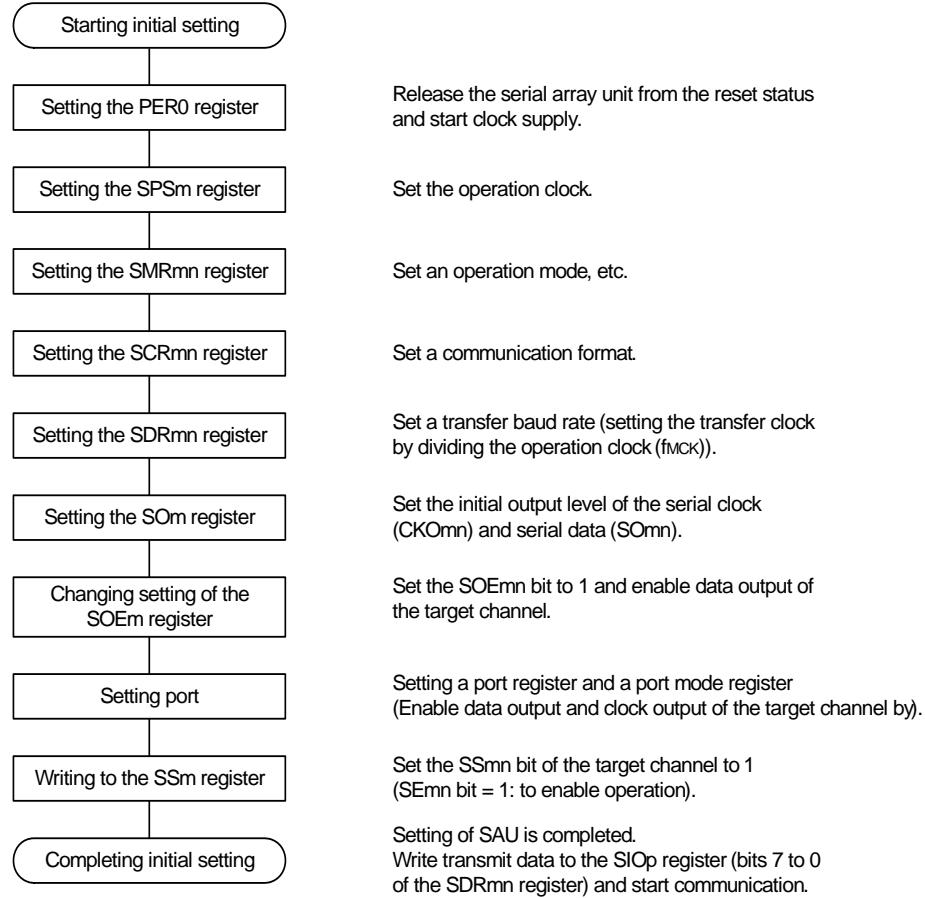
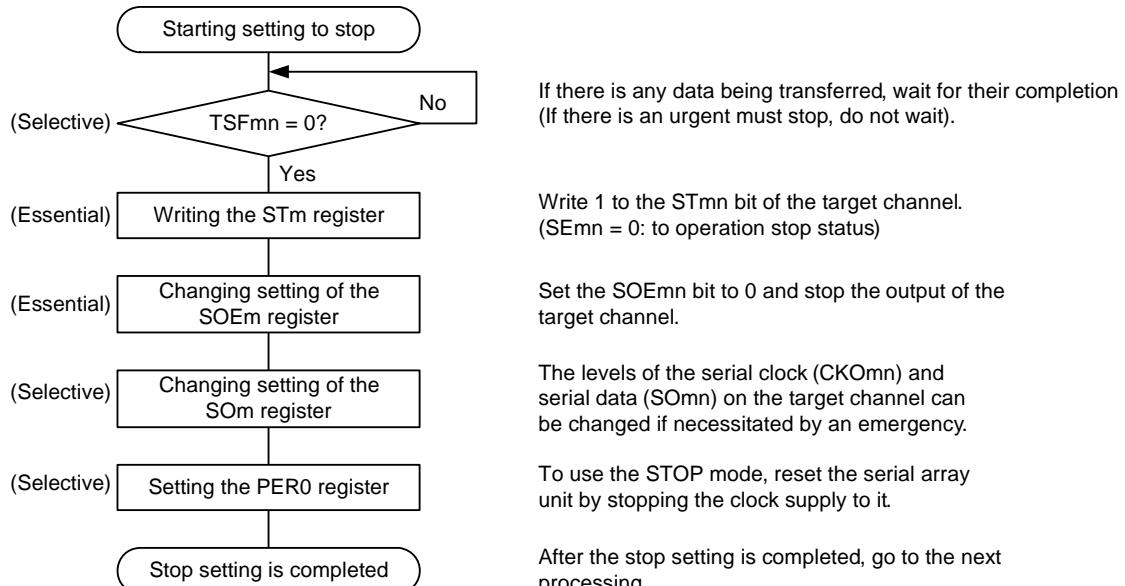
**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

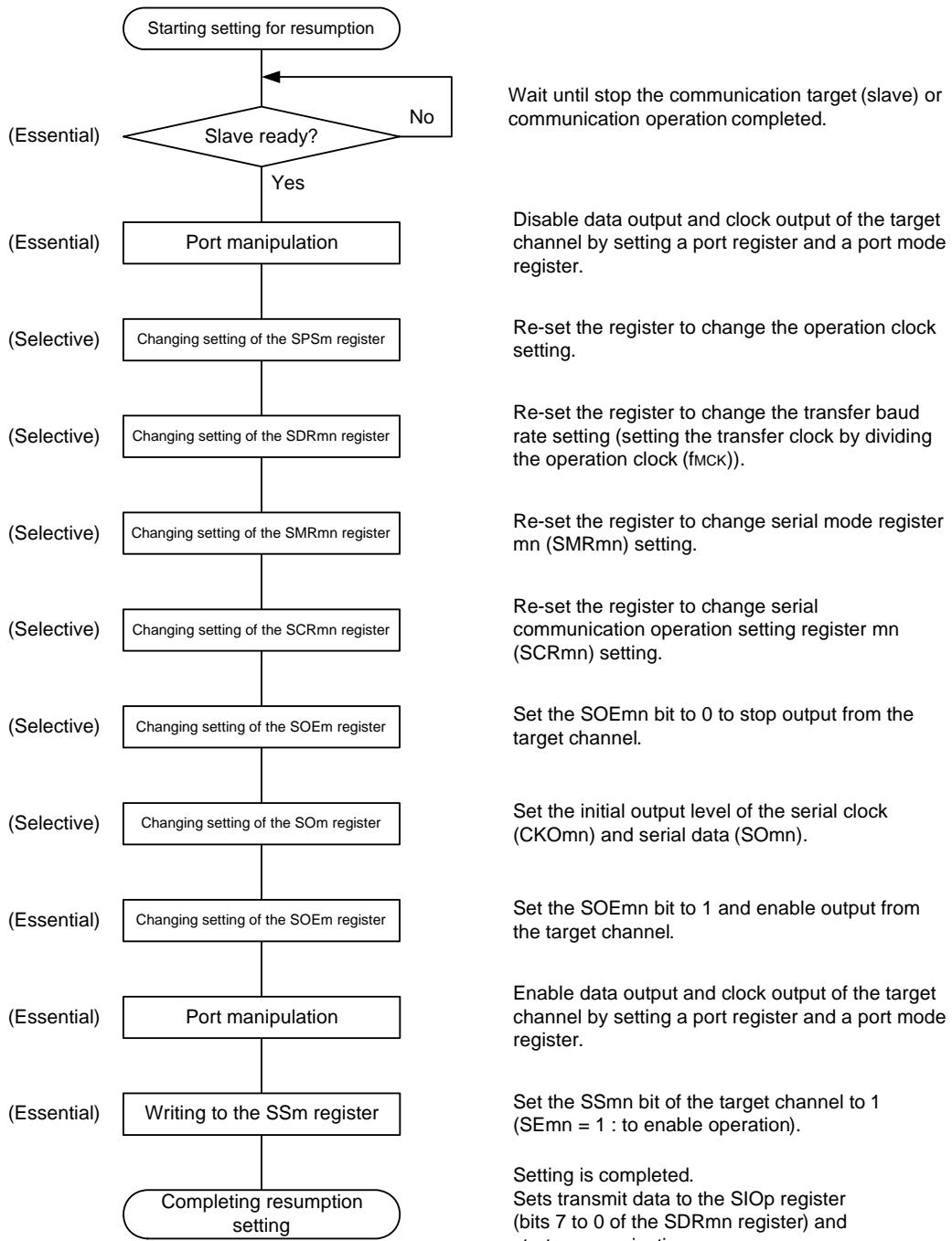
**Remark 2.** : Setting is fixed in the CSI master transmission mode,

: Setting disabled (set to the initial value)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

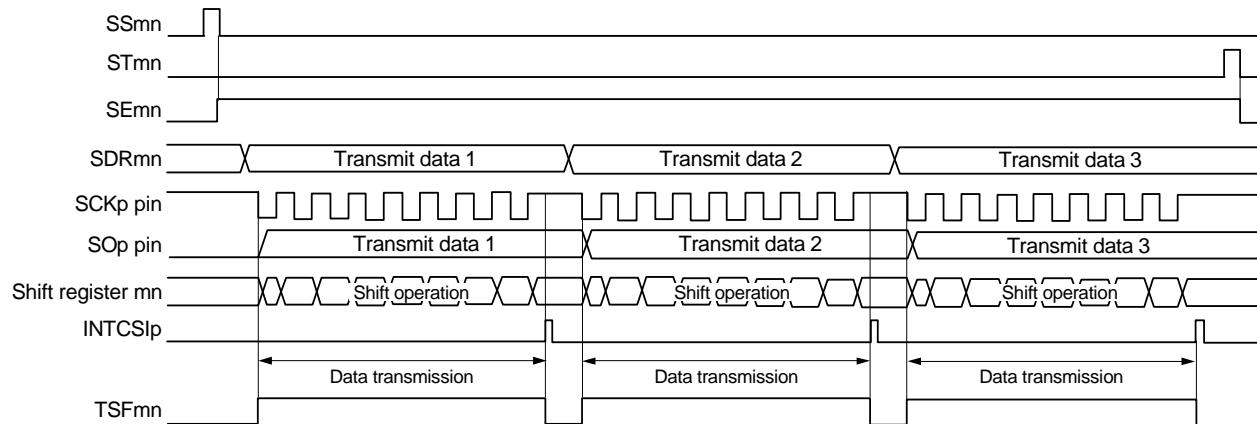
**Figure 17 - 29 Initial Setting Procedure for Master Transmission****Figure 17 - 30 Procedure for Stopping Master Transmission**

**Figure 17 - 31 Procedure for Resuming Master Transmission**

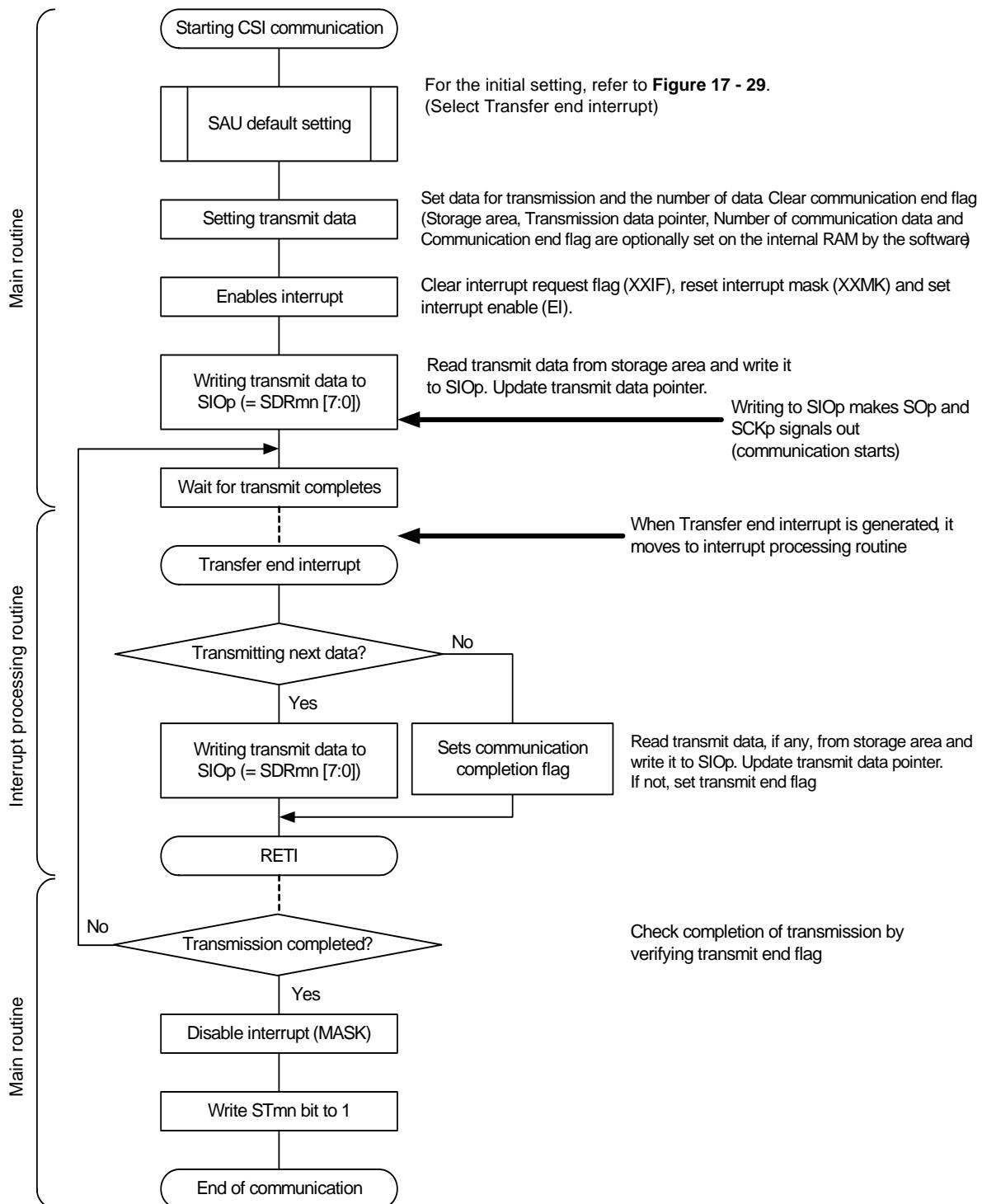
**Remark** If PERO0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (slave) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-transmission mode)

**Figure 17 - 32 Timing Chart of Master Transmission (in Single-Transmission Mode)**  
 (Type 1: DAP<sub>mn</sub> = 0, CKP<sub>mn</sub> = 0)

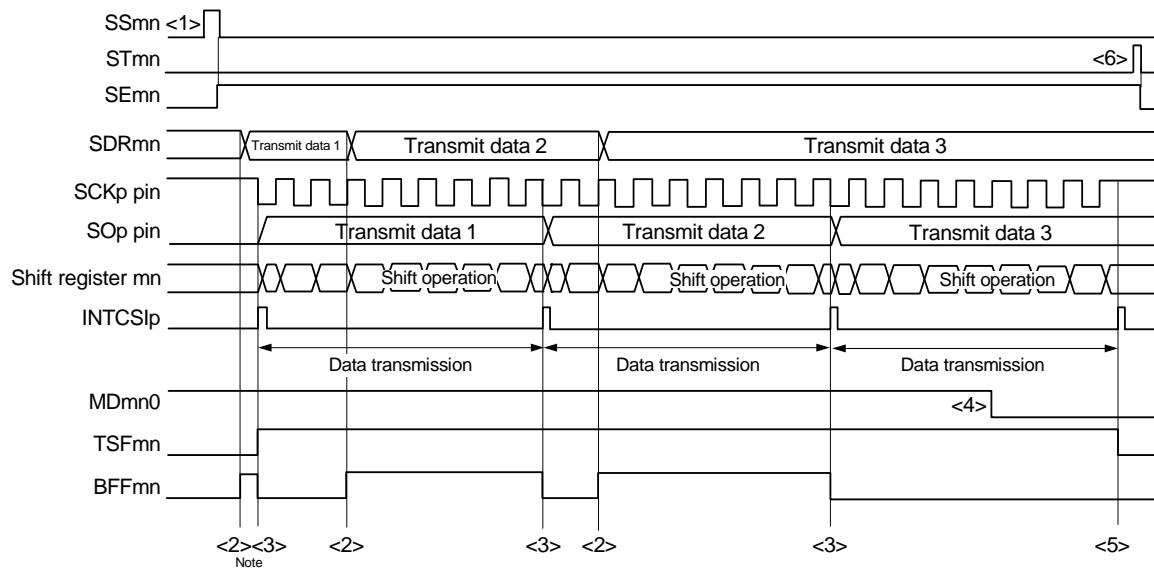


**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03, 10$  to  $13$

**Figure 17 - 33 Flowchart of Master Transmission (in Single-Transmission Mode)**

## (4) Processing flow (in continuous transmission mode)

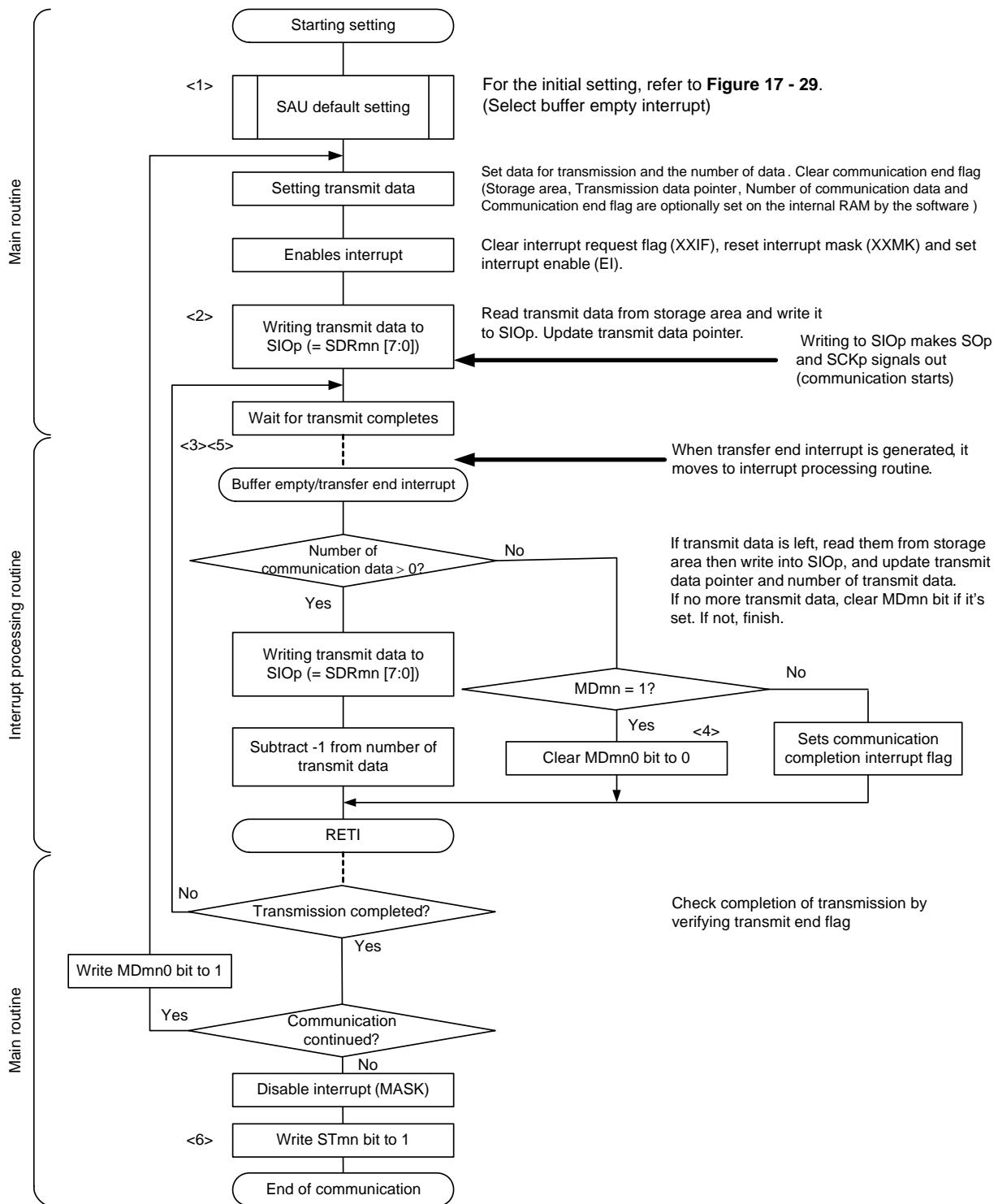
**Figure 17 - 34 Timing Chart of Master Transmission (in Continuous Transmission Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Note** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation. However, rewrite it before transfer of the last bit is started, so that it will be rewritten before the transfer end interrupt of the last transmit data.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
 mn = 00 to 03, 10 to 13

**Figure 17 - 35 Flowchart of Master Transmission (in Continuous Transmission Mode)**

**Remark** <1> to <6> in the figure correspond to <1> to <6> in Figure 17 - 34 Timing Chart of Master Transmission (in Continuous Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0).

### 17.5.2 Master reception

Master reception is that the RL78 microcontroller outputs a transfer clock and receives data from other device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SI00	SCK01, SI01	SCK10, SI10	SCK11, SI11	SCK20, SI20	SCK21, SI21	SCK30, SI30	SCK31, SI31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	7 or 8 bits							
Transfer rate Note	Max. fCLK/2 [Hz] (CSI00 only), fCLK/4 [Hz] Min. fCLK/(2 × 2 <sup>15</sup> × 128) [Hz] fCLK: System clock frequency							
Data phase	Selectable by the DAPmn bit of the SCRmn register • DAPmn = 0: Data input starts from the start of the operation of the serial clock. • DAPmn = 1: Data input starts half a clock before the start of the serial clock operation.							
Clock phase	Selectable by the CKPmn bit of the SCRmn register • CKPmn = 0: Non-reverse • CKPmn = 1: Reverse							
Data direction	MSB or LSB first							

**Note** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), mn = 00 to 03, 10 to 13

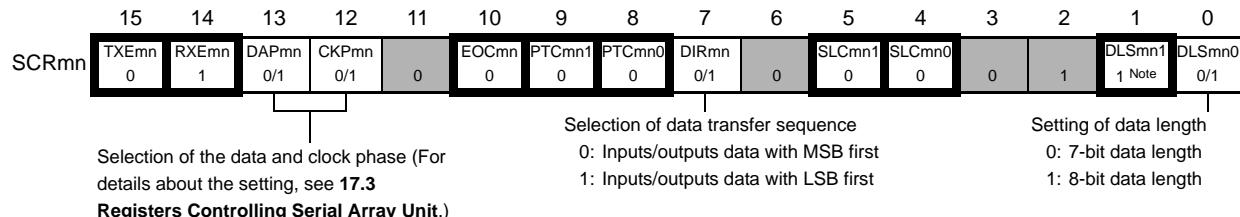
## (1) Register setting

**Figure 17 - 36 Example of Contents of Registers for Master Reception of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)**

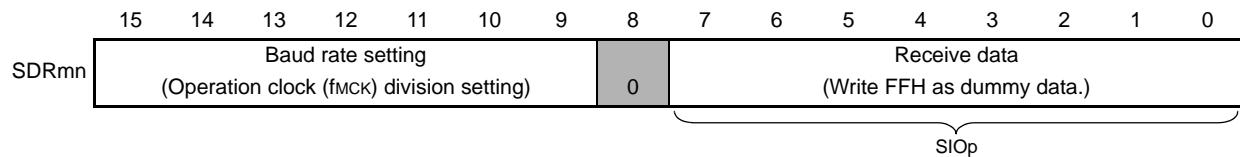
## (a) Serial mode register mn (SMRmn)



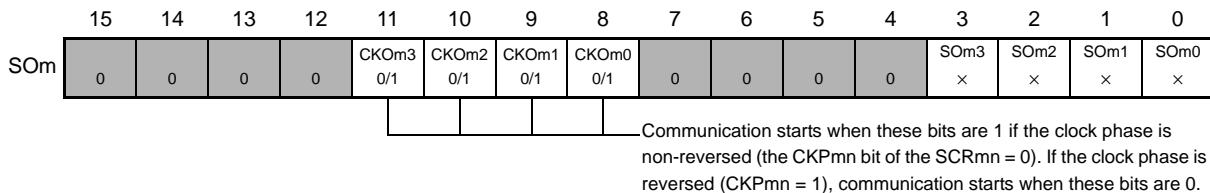
## (b) Serial communication operation setting register mn (SCRmn)



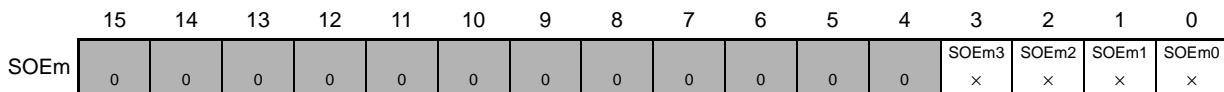
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



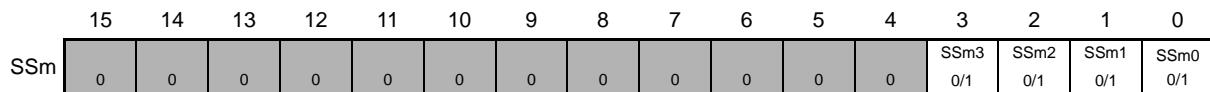
## (d) Serial output register m (SOm)... Sets only the bits of the target channel.



## (e) Serial output enable register m (SOEm)... The register that not used in this mode.



## (f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.



**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

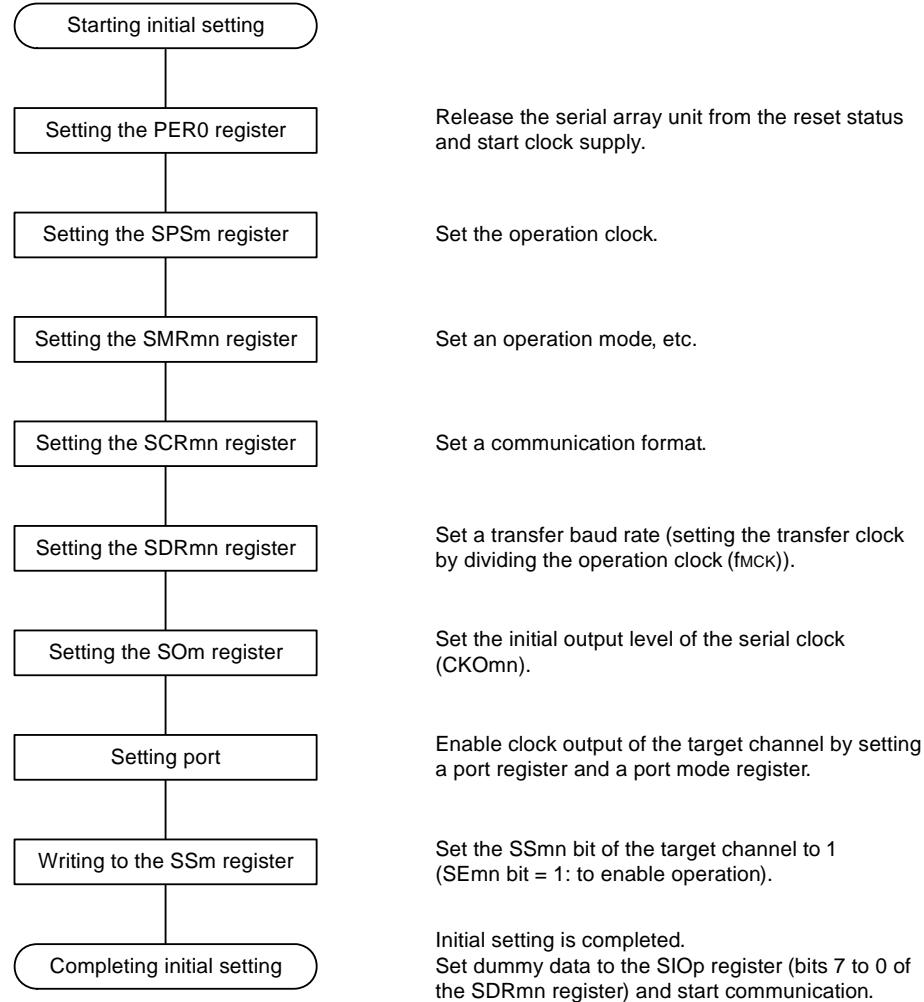
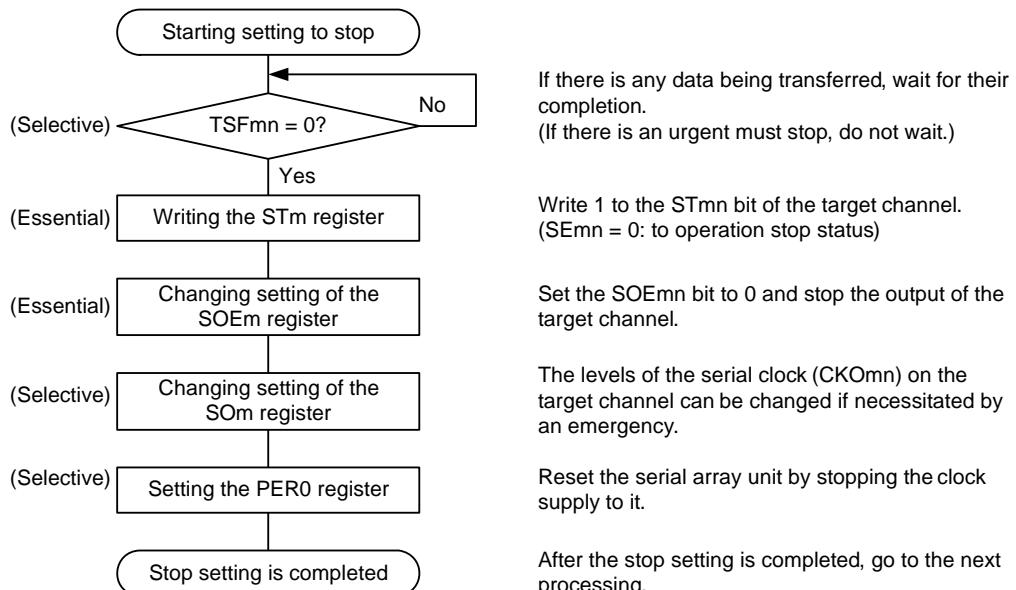
**Remark 2.** : Setting is fixed in the CSI master reception mode,

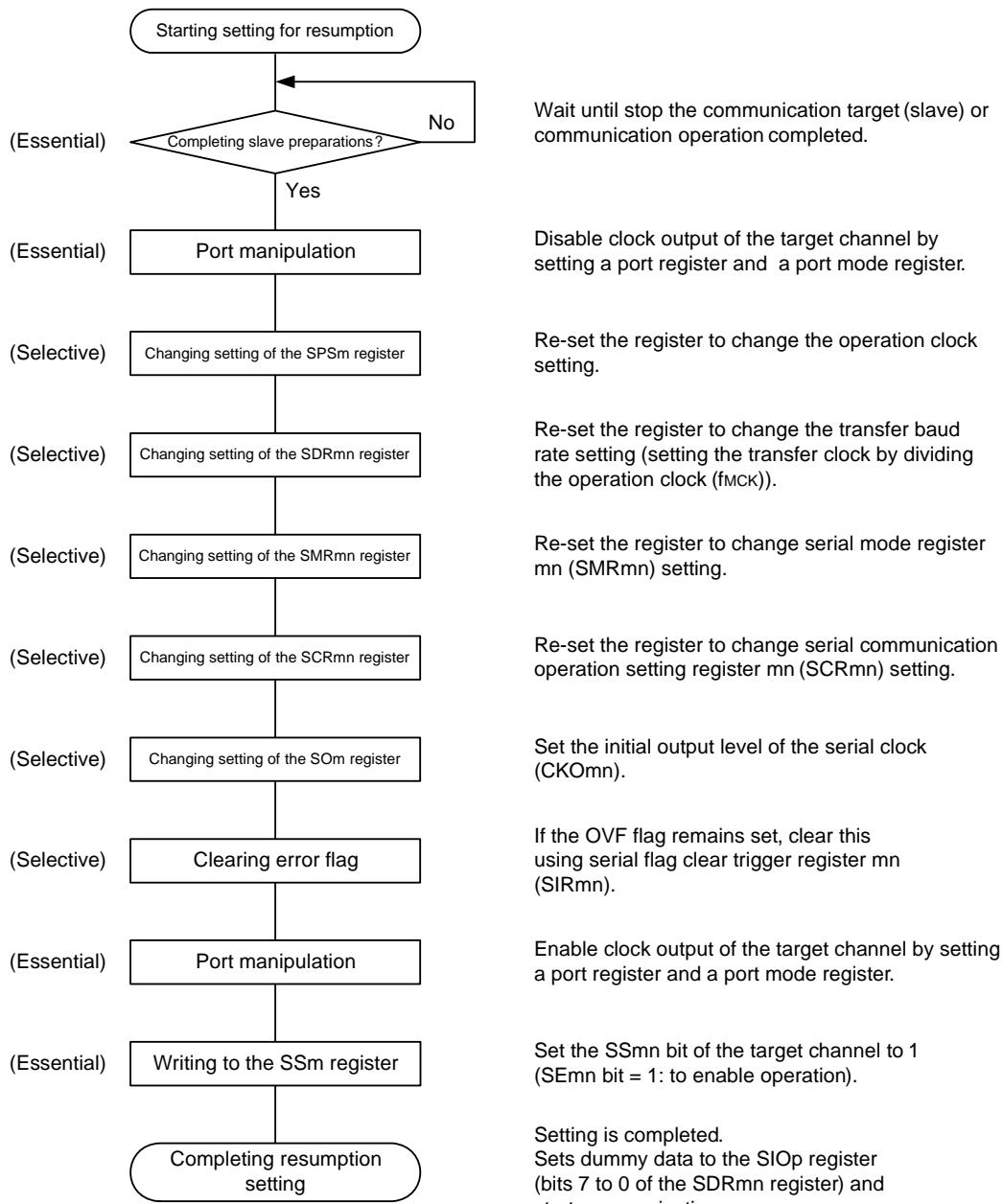
: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

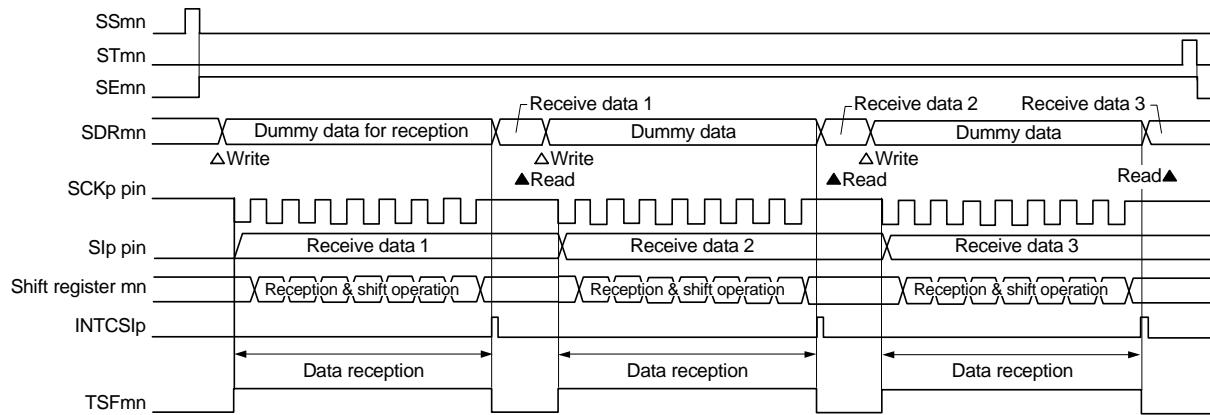
**Figure 17 - 37 Initial Setting Procedure for Master Reception****Figure 17 - 38 Procedure for Stopping Master Reception**

**Figure 17 - 39 Procedure for Resuming Master Reception**

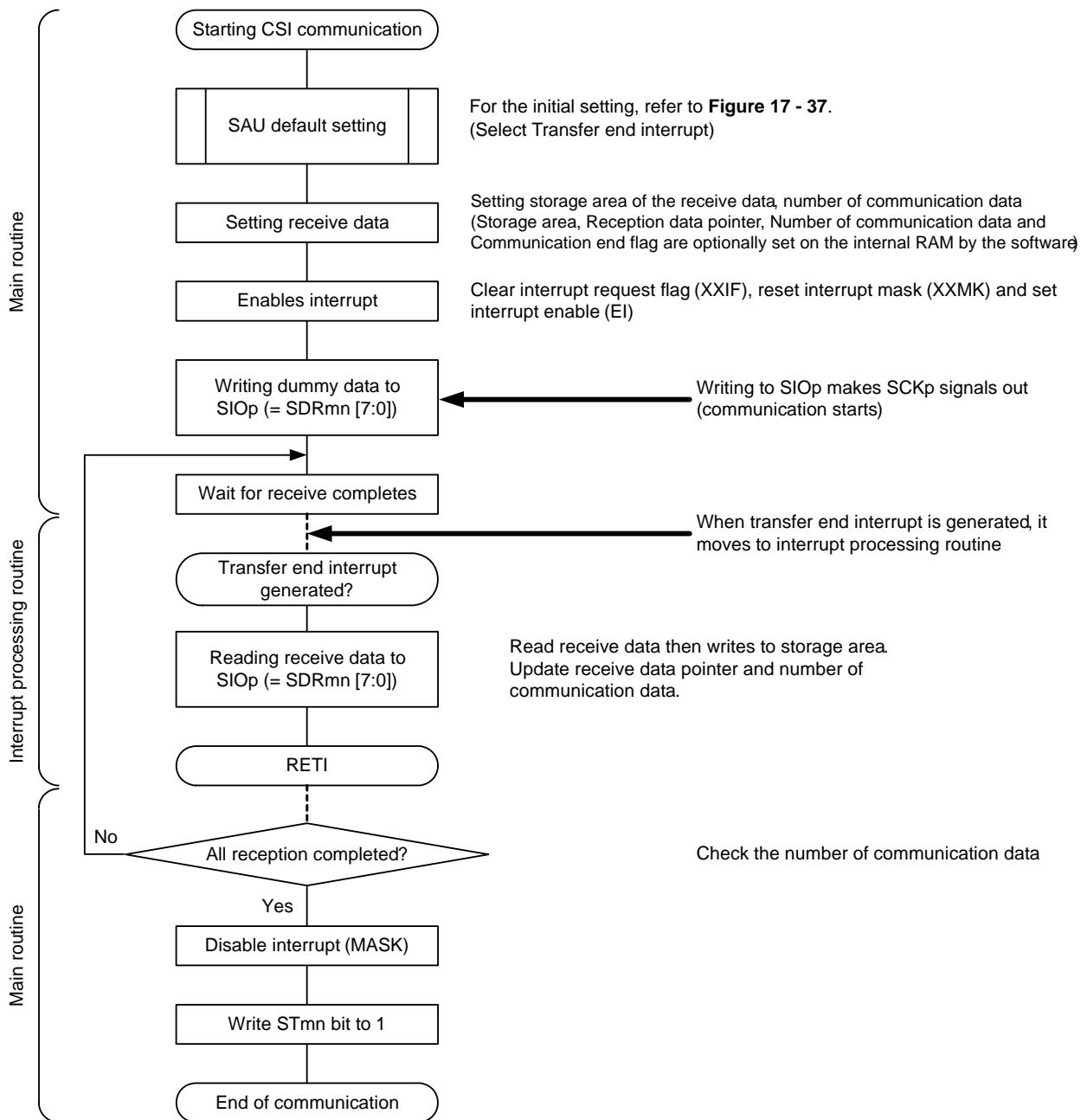
**Remark** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (slave) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-reception mode)

**Figure 17 - 40 Timing Chart of Master Reception (in Single-Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**

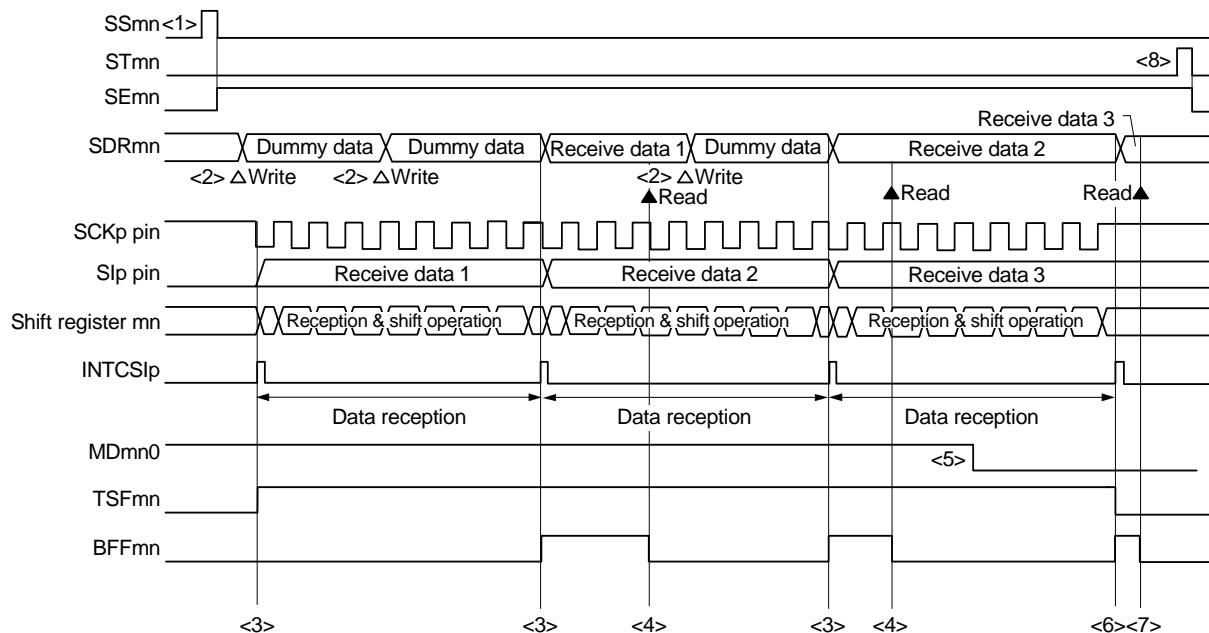


**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

**Figure 17 - 41 Flowchart of Master Reception (in Single-Reception Mode)**

## (4) Processing flow (in continuous reception mode)

**Figure 17 - 42 Timing Chart of Master Reception (in Continuous Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



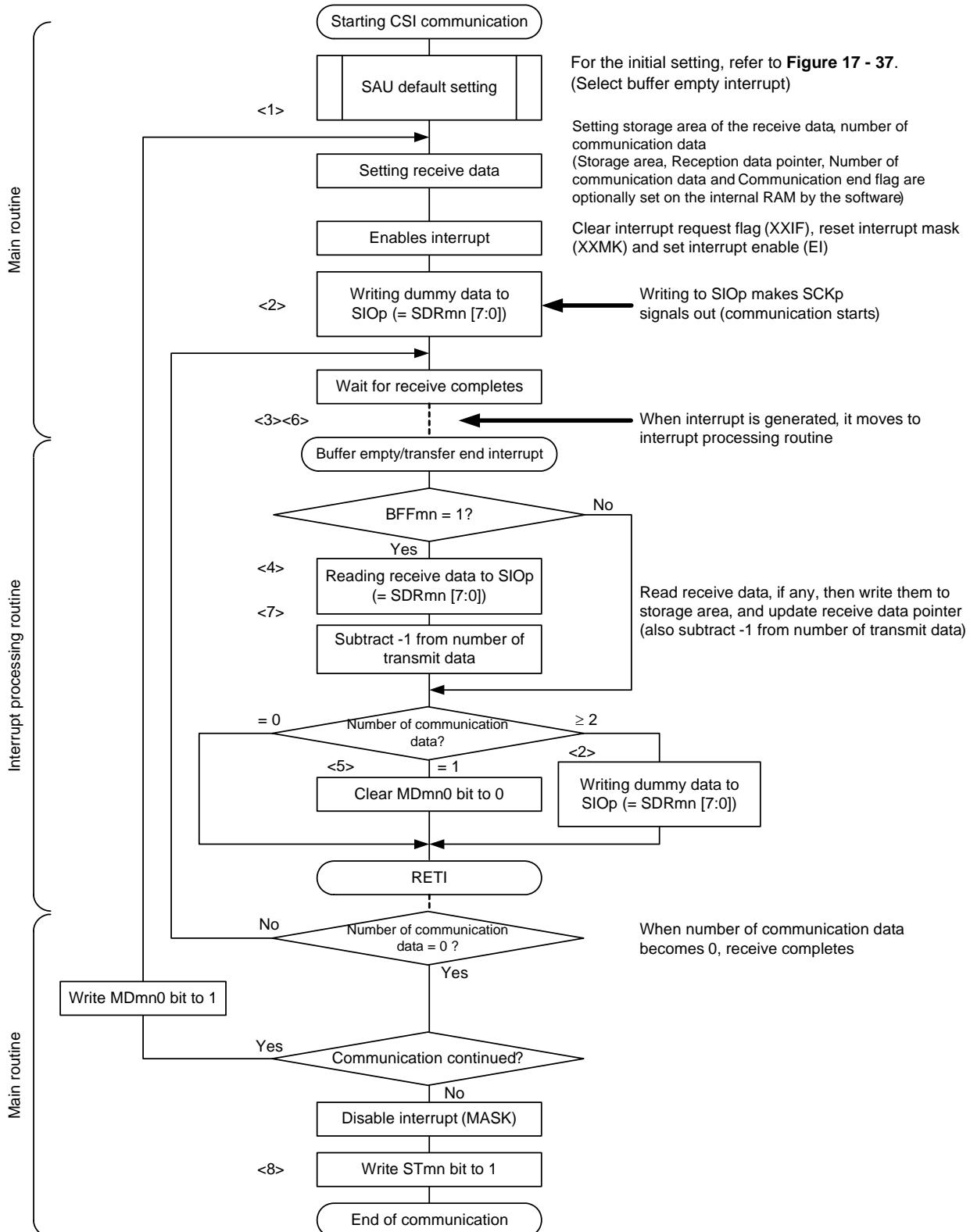
**Caution** The MDmn0 bit can be rewritten even during operation.

However, rewrite it before receive of the last bit is started, so that it has been rewritten before the transfer end interrupt of the last receive data.

**Remark 1.** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 43 Flowchart of Master Reception (in Continuous Reception Mode).

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
 mn = 00 to 03, 10 to 13

Figure 17 - 43 Flowchart of Master Reception (in Continuous Reception Mode)



**Remark** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 42 Timing Chart of Master Reception (in Continuous Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0).

### 17.5.3 Master transmission/reception

Master transmission/reception is that the RL78 microcontroller outputs a transfer clock and transmits/receives data to/from other device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SI00, SO00	SCK01, SI01, SO01	SCK10, SI10, SO10	SCK11, SI11, SO11	SCK20, SI20, SO20	SCK21, SI21, SO21	SCK30, SI30, SO30	SCK31, SI31, SO31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	7 or 8 bits							
Transfer rate <small>Note</small>	Max. fCLK/2 [Hz] (CSI00 only), fCLK/4 [Hz] Min. fCLK/(2 × 2 <sup>15</sup> × 128) [Hz] fCLK: System clock frequency							
Data phase	Selectable by the DAPmn bit of the SCRmn register • DAPmn = 0: Data I/O starts at the start of the operation of the serial clock. • DAPmn = 1: Data I/O starts half a clock before the start of the serial clock operation.							
Clock phase	Selectable by the CKPmn bit of the SCRmn register • CKPmn = 0: Non-reverse • CKPmn = 1: Reverse							
Data direction	MSB or LSB first							

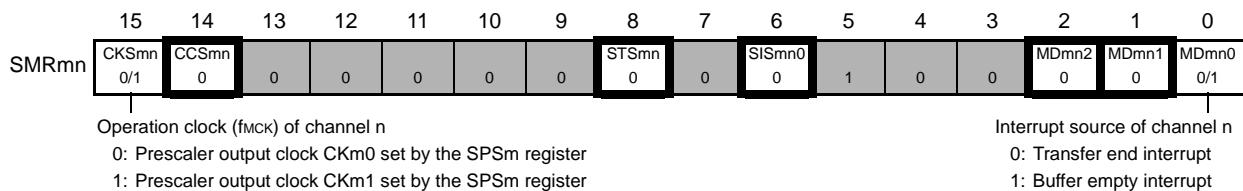
**Note** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

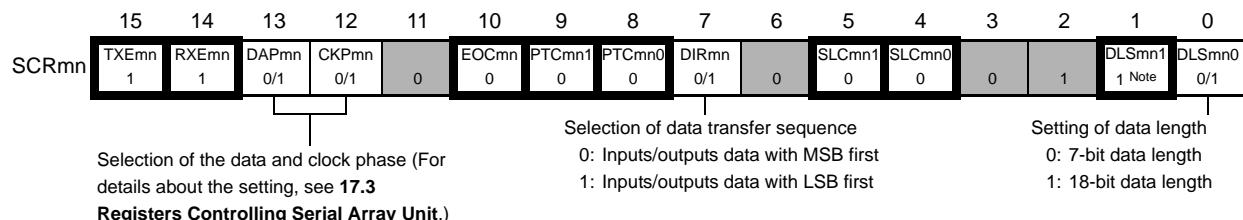
## (1) Register setting

**Figure 17 - 44 Example of Contents of Registers for Master Transmission/Reception of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CS31)**

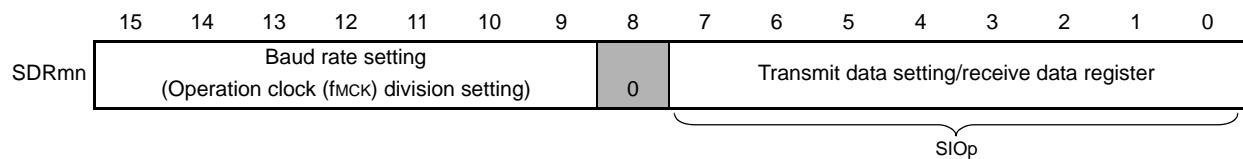
## (a) Serial mode register mn (SMRmn)



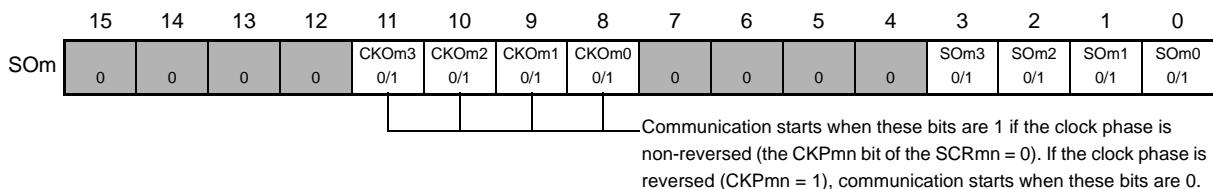
## (b) Serial communication operation setting register mn (SCRmn)



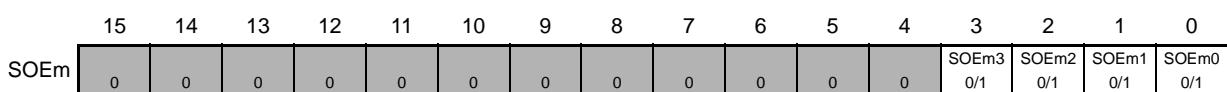
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



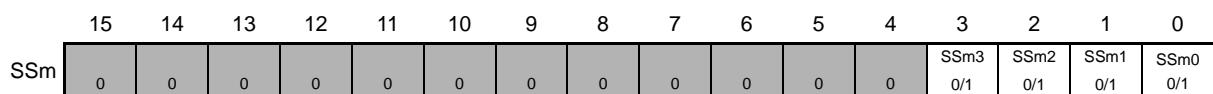
## (d) Serial output register m (SOm)... Sets only the bits of the target channel.



## (e) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.



## (f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.



**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

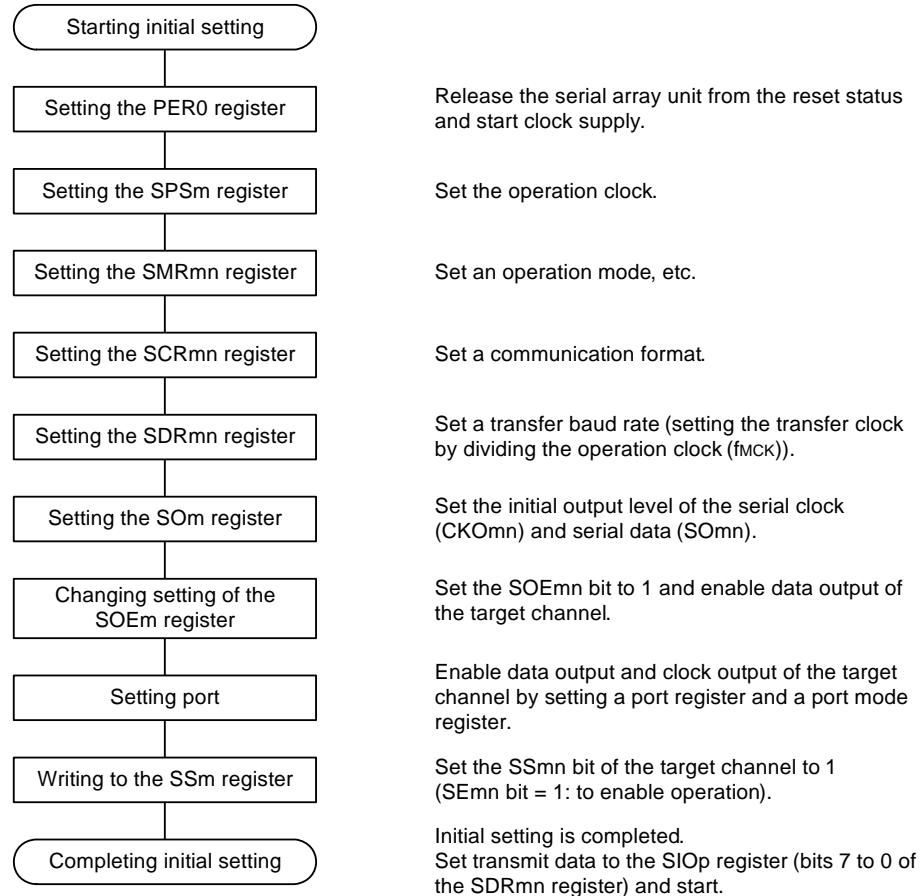
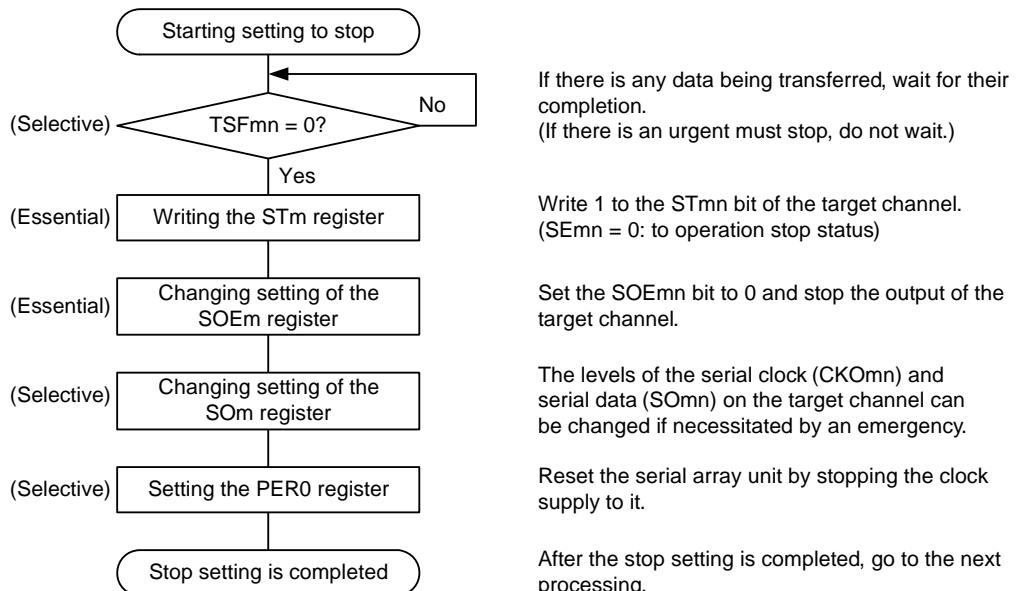
**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

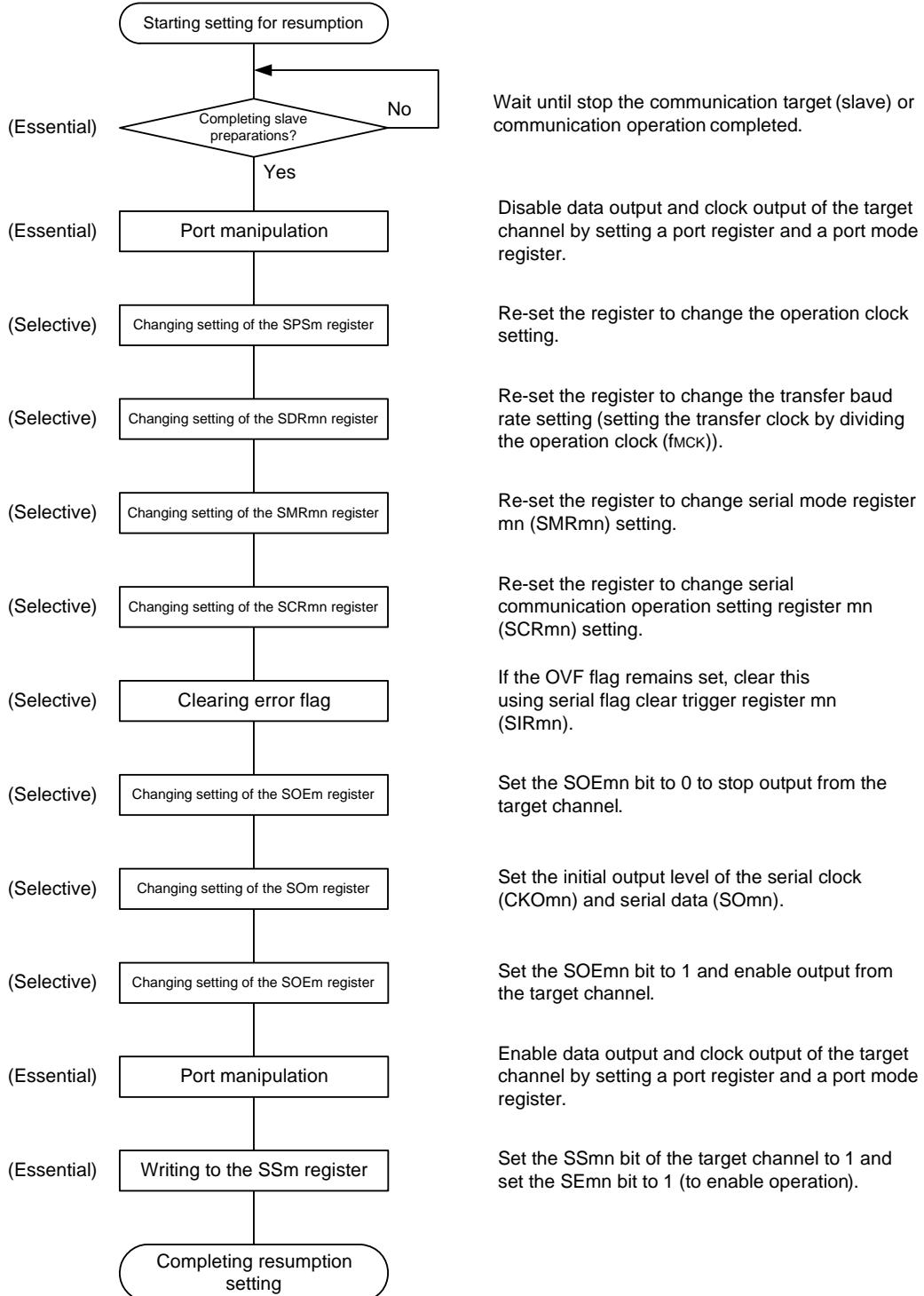
**Remark 2.** : Setting is fixed in the CSI master transmission/reception mode,

: Setting disabled (set to the initial value)

0/1: Set to 0 or 1 depending on the usage of the user

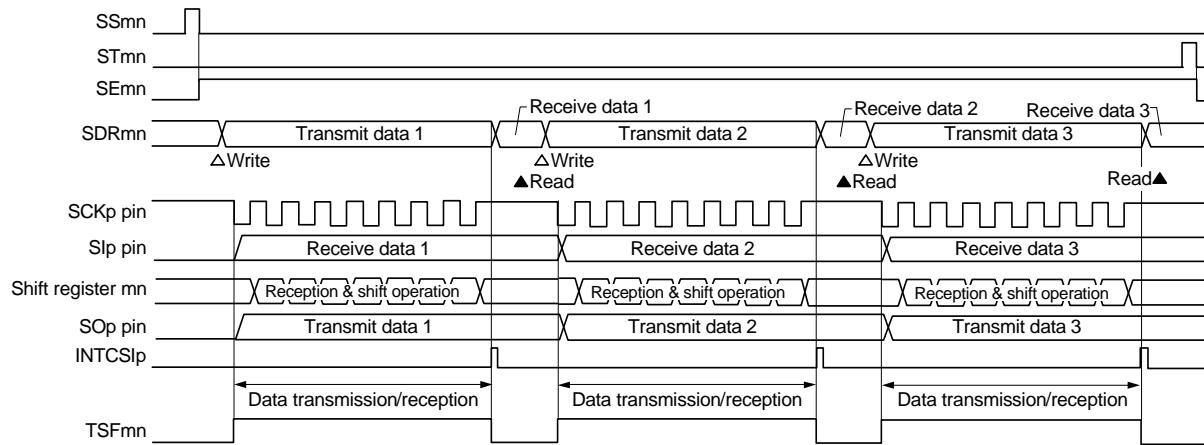
## (2) Operation procedure

**Figure 17 - 45 Initial Setting Procedure for Master Transmission/Reception****Figure 17 - 46 Procedure for Stopping Master Transmission/Reception**

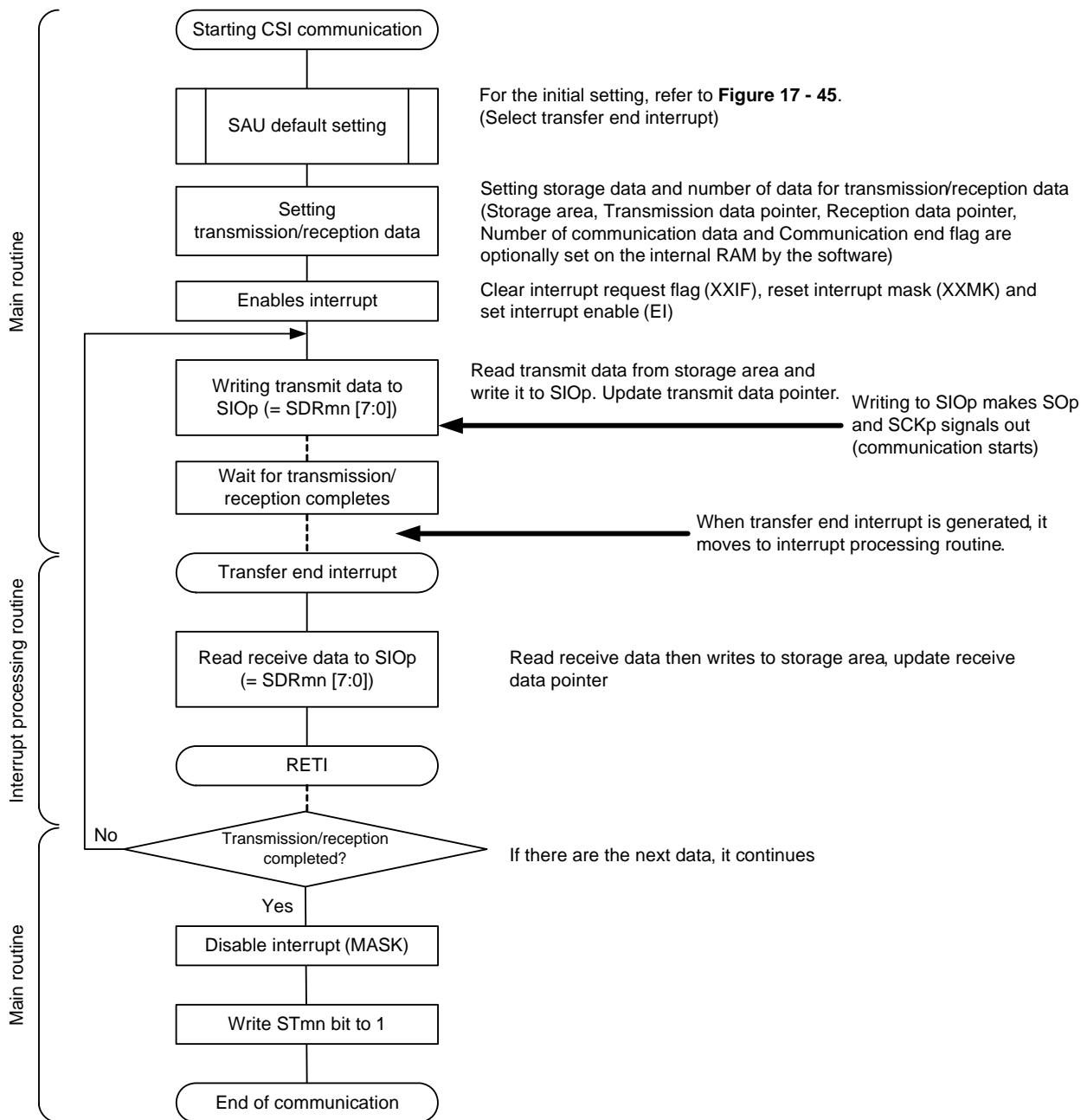
**Figure 17 - 47 Procedure for Resuming Master Transmission/Reception**

## (3) Processing flow (in single-transmission/reception mode)

**Figure 17 - 48 Timing Chart of Master Transmission/Reception (in Single-Transmission/Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**

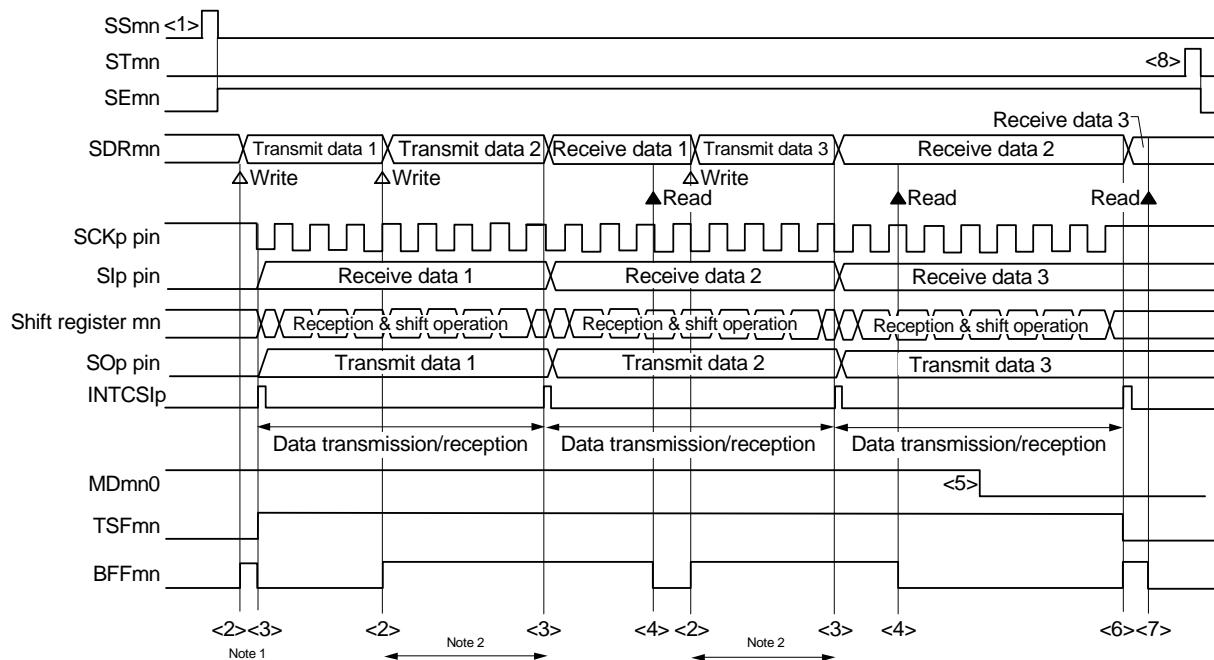


**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

**Figure 17 - 49 Flowchart of Master Transmission/Reception (in Single- Transmission/Reception Mode)**

## (4) Processing flow (in continuous transmission/reception mode)

**Figure 17 - 50 Timing Chart of Master Transmission/Reception (in Continuous Transmission/Reception Mode)  
(Type 1: DAPmn = 0, CKPmn = 0)**



**Note 1.** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

**Note 2.** The transmit data can be read by reading the SDRmn register during this period. At this time, the transfer operation is not affected.

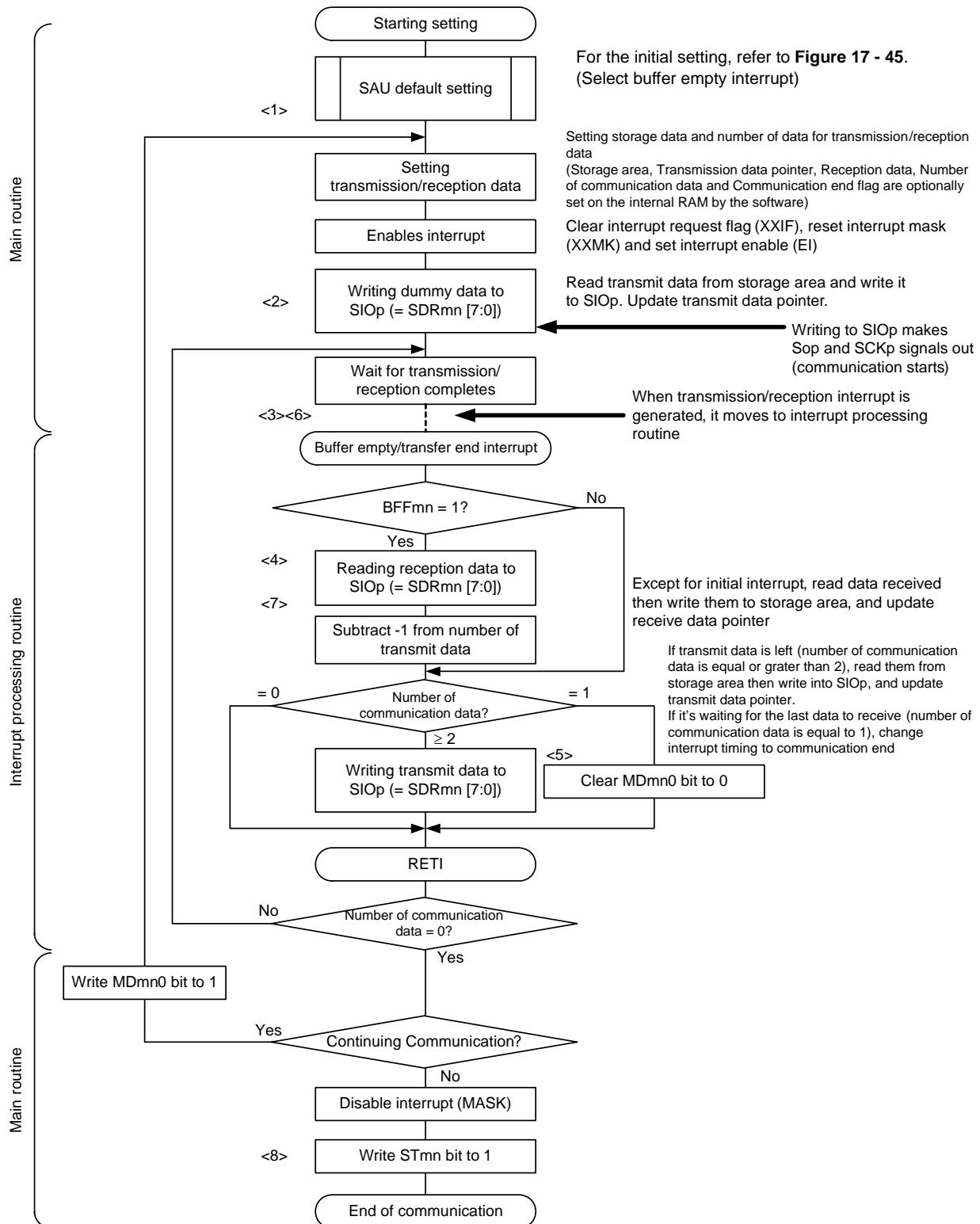
**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation.

However, rewrite it before transfer of the last bit is started, so that it has been rewritten before the transfer end interrupt of the last transmit data.

**Remark 1.** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 51 Flowchart of Master Transmission/Reception (in Continuous Transmission/Reception Mode).

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

Figure 17 - 51 Flowchart of Master Transmission/Reception (in Continuous Transmission/Reception Mode)



**Remark** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 50 Timing Chart of Master Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0).

### 17.5.4 Slave transmission

Slave transmission is that the RL78 microcontroller transmits data to another device in the state of a transfer clock being input from another device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SO00	SCK01, SO01	SCK10, SO10	SCK11, SO11	SCK20, SO20	SCK21, SO21	SCK30, SO30	SCK31, SO31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	7 or 8 bits							
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2.							
Data phase	Selectable by the DAPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• DAPmn = 0: Data output starts from the start of the operation of the serial clock.</li> <li>• DAPmn = 1: Data output starts half a clock before the start of the serial clock operation.</li> </ul>							
Clock phase	Selectable by the CKPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• CKPmn = 0: Non-reverse</li> <li>• CKPmn = 1: Reverse</li> </ul>							
Data direction	MSB or LSB first							

**Note 1.** Because the external serial clock input to the SCK00, SCK01, SCK10, SCK11, SCK20, SCK21, SCK30, and SCK31 pins is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## (1) Register setting

**Figure 17 - 52 Example of Contents of Registers for Slave Transmission of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)**

## (a) Serial mode register mn (SMRmn)

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 1	0	0	0	0	0	STSmn 0	0	SISmn0 0	1	0	0	MDmn2 0	MDmn1 0	MDmn0 0/1

Operation clock (fmcx) of channel n  
0: Prescaler output clock CKm0 set by the SPSm register  
1: Prescaler output clock CKm1 set by the SPSm register

Interrupt source of channel n  
0: Transfer end interrupt  
1: Buffer empty interrupt

## (b) Serial communication operation setting register mn (SCRmn)

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 1	RXEmn 0	DAPmn 0/1	CKPmn 0/1	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0/1	0	SLCmn1 0	SLCmn0 0	0	1	DLSmn1 1 Note	DLSmn0 0/1

Selection of the data and clock phase (For details about the setting, see 17.3 Registers Controlling Serial Array Unit.)

Selection of data transfer sequence  
0: Inputs/outputs data with MSB first  
1: Inputs/outputs data with LSB first

Setting of data length  
0: 7-bit data length  
1: 8-bit data length

## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	00000000 Baud rate setting										Transmit data setting					

SIOp

## (d) Serial output register m (SOm)... Sets only the bits of the target channel.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 x	CKOm2 x	CKOm1 x	CKOm0 x	0	0	0	0	SOm3 0/1	SOm2 0/1	SOm1 0/1	SOm0 0/1

## (e) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 0/1	SOEm2 0/1	SOEm1 0/1	SOEm0 0/1

**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

**Remark 2.** : Setting is fixed in the CSI slave transmission mode,

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

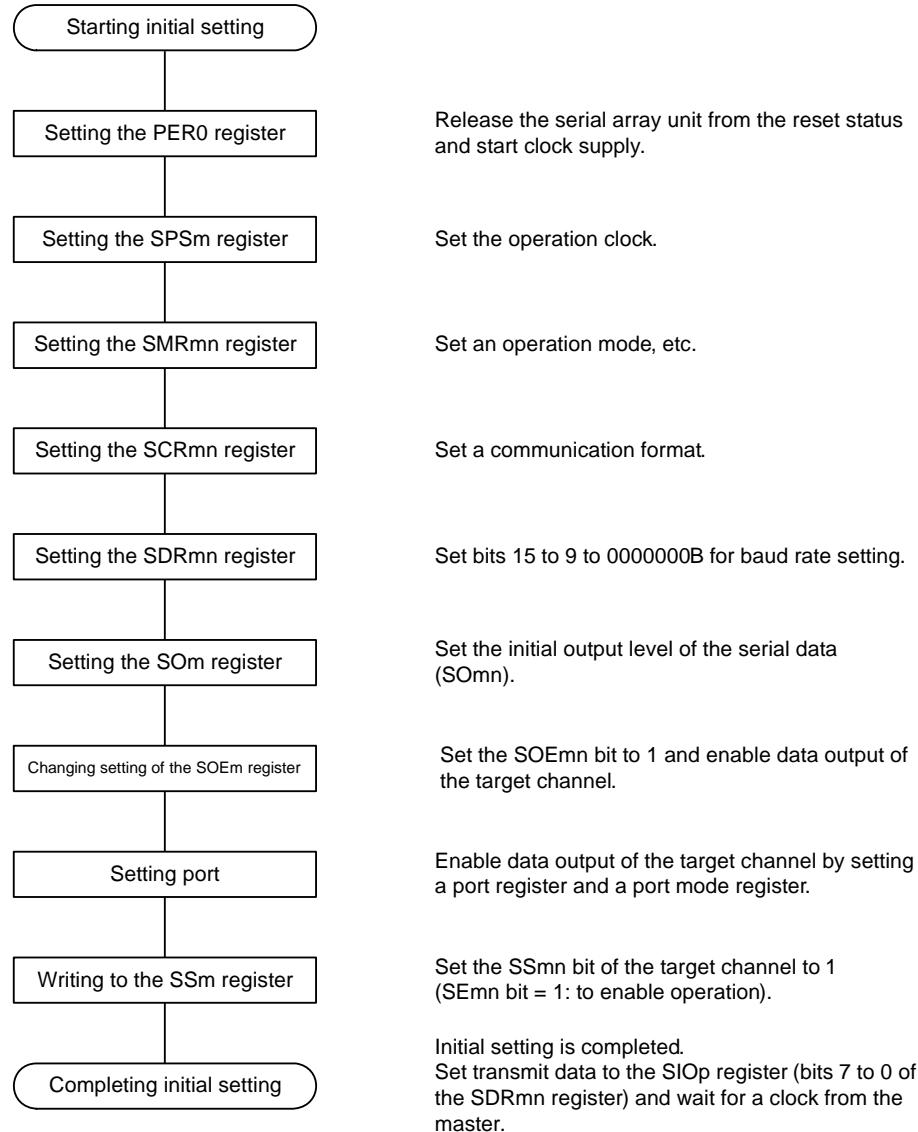
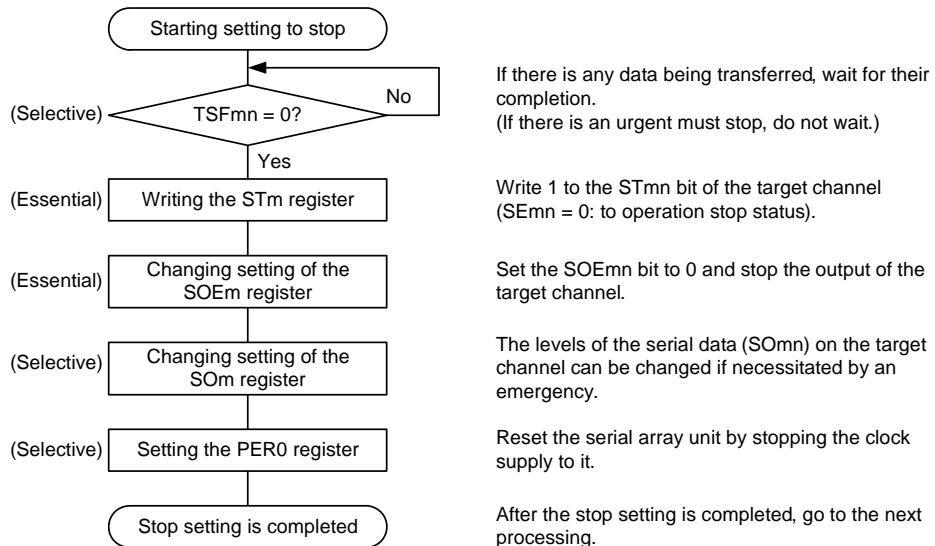
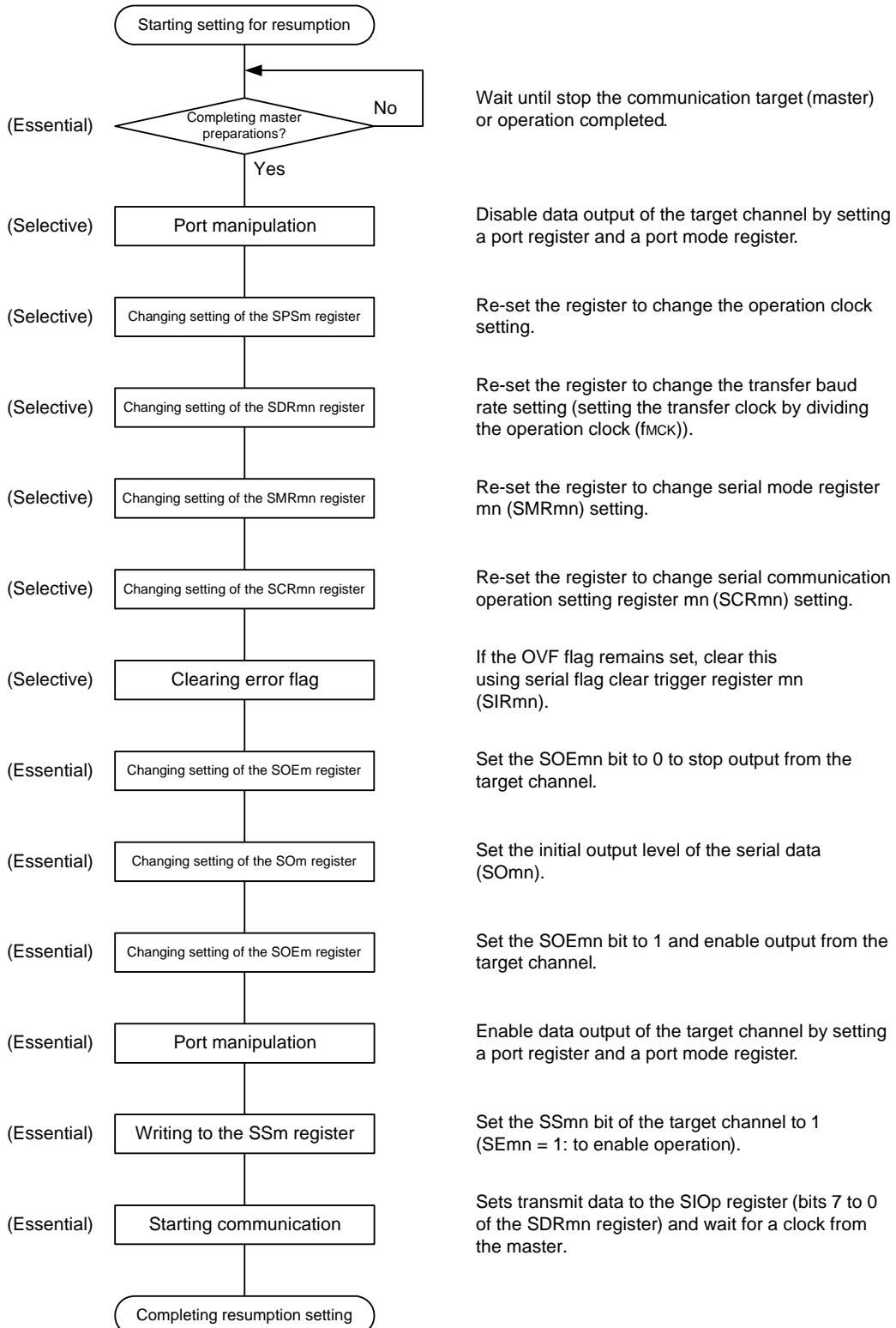
**Figure 17 - 53 Initial Setting Procedure for Slave Transmission****Figure 17 - 54 Procedure for Stopping Slave Transmission**

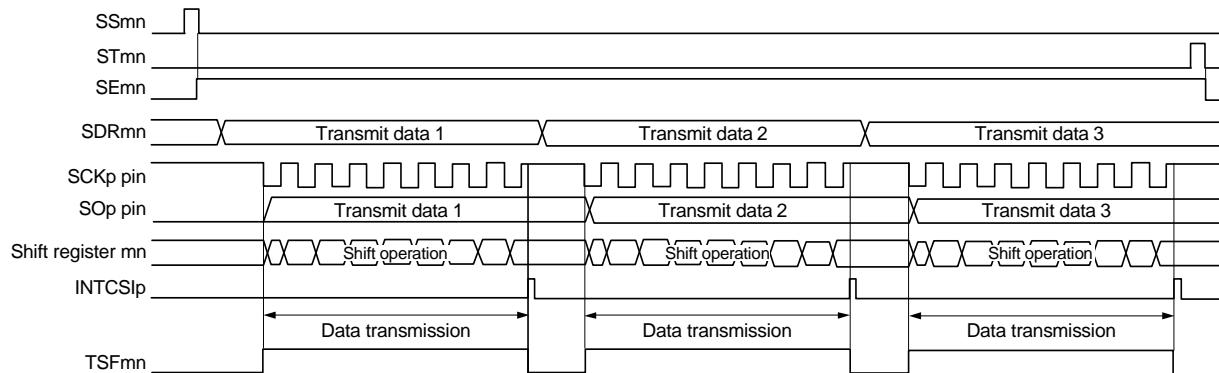
Figure 17 - 55 Procedure for Resuming Slave Transmission



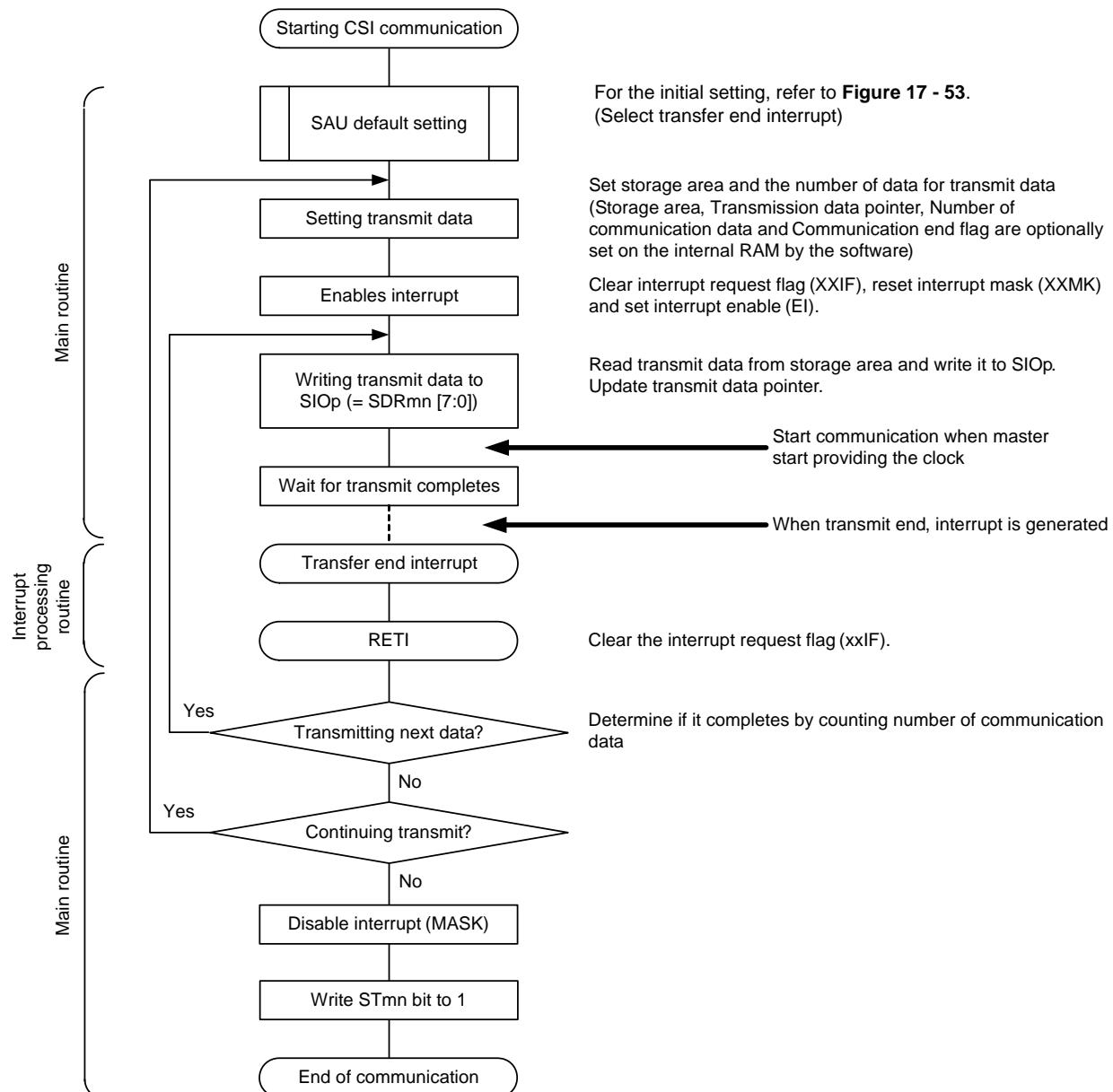
**Remark** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (master) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-transmission mode)

**Figure 17 - 56 Timing Chart of Slave Transmission (in Single-Transmission Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**

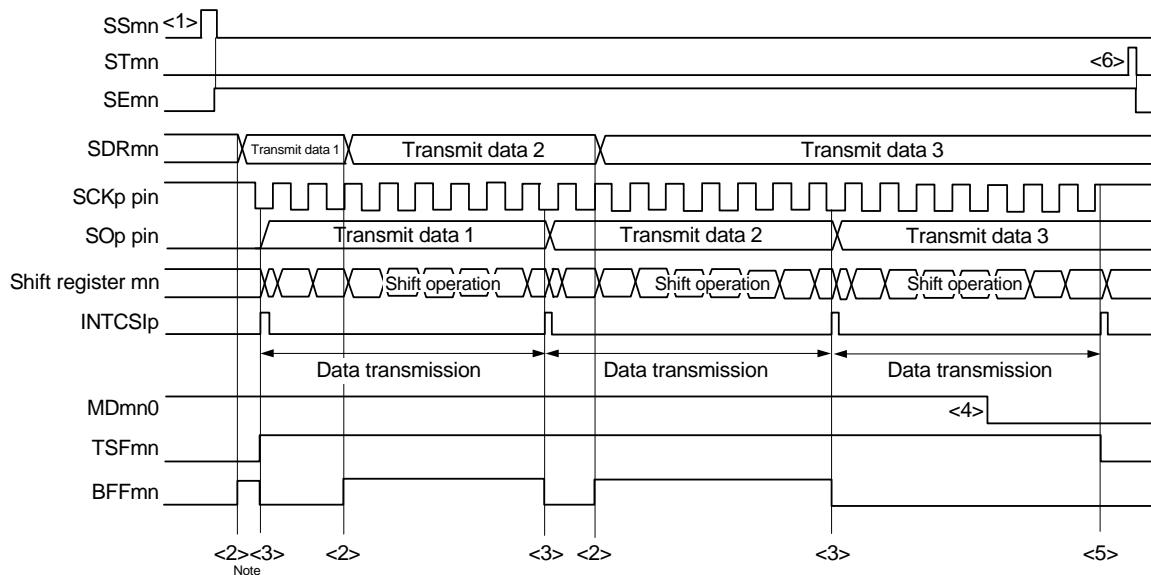


**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03, 10$  to  $13$

**Figure 17 - 57 Flowchart of Slave Transmission (in Single-Transmission Mode)**

## (4) Processing flow (in continuous transmission mode)

**Figure 17 - 58 Timing Chart of Slave Transmission (in Continuous Transmission Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**

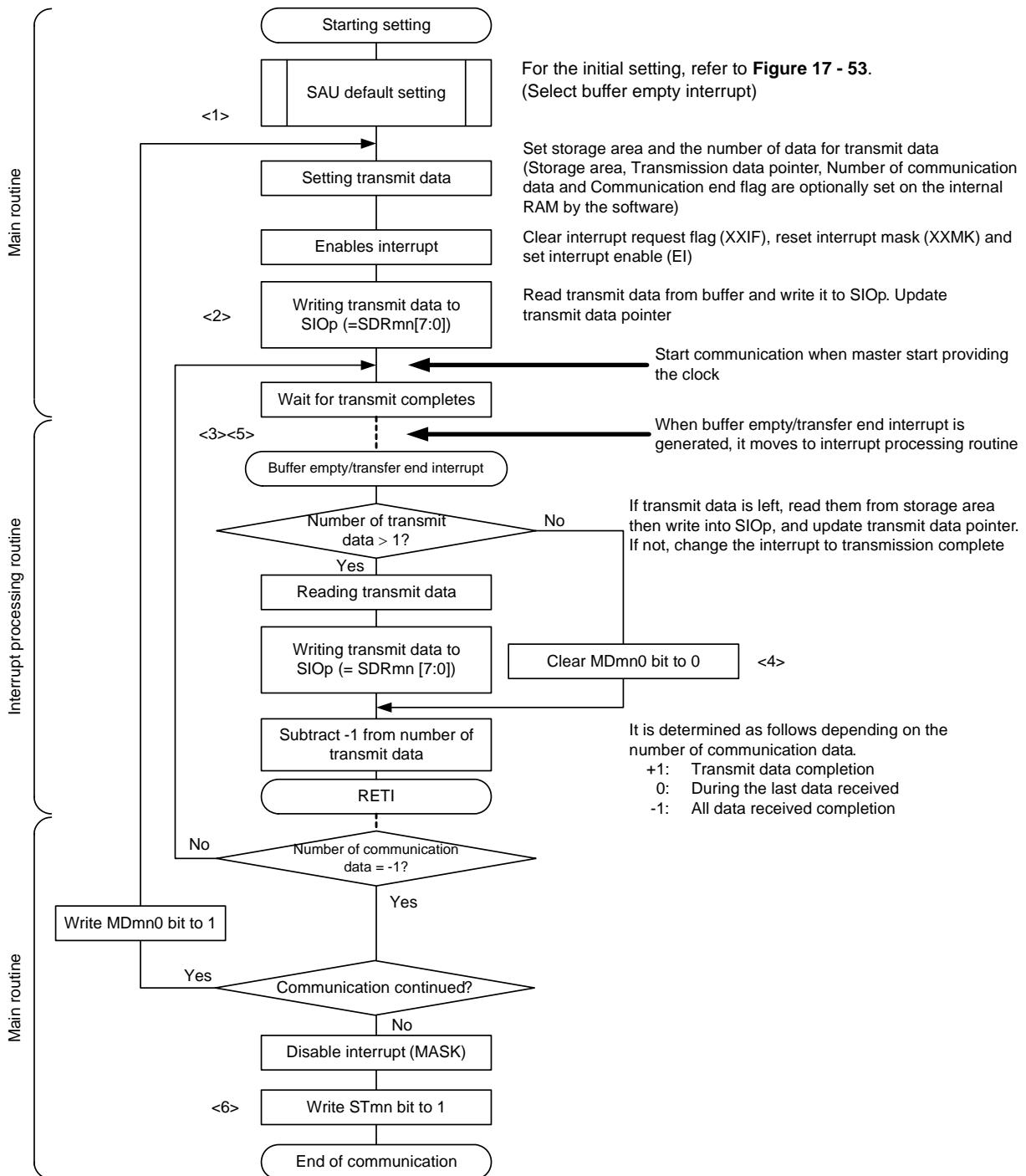


**Note** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation. However, rewrite it before transfer of the last bit is started.

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

Figure 17 - 59 Flowchart of Slave Transmission (in Continuous Transmission Mode)



**Remark** <1> to <6> in the figure correspond to <1> to <6> in Figure 17 - 58 Timing Chart of Slave Transmission (in Continuous Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0).

### 17.5.5 Slave reception

Slave reception is that the RL78 microcontroller receives data from another device in the state of a transfer clock being input from another device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SI00	SCK01, SI01	SCK10, SI10	SCK11, SI11	SCK20, SI20	SCK21, SI21	SCK30, SI30	SCK31, SI31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	7 or 8 bits							
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2							
Data phase	Selectable by the DAPmn bit of the SCRmn register • DAPmn = 0: Data input starts from the start of the operation of the serial clock. • DAPmn = 1: Data input starts half a clock before the start of the serial clock operation.							
Clock phase	Selectable by the CKPmn bit of the SCRmn register • CKPmn = 0: Non-reverse • CKPmn = 1: Reverse							
Data direction	MSB or LSB first							

**Note 1.** Because the external serial clock input to the SCK00, SCK01, SCK10, SCK11, SCK20, SCK21, SCK30, and SCK31 pins is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

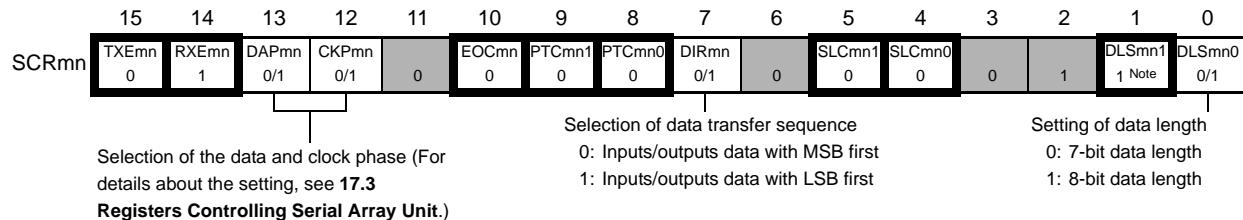
## (1) Register setting

**Figure 17 - 60 Example of Contents of Registers for Slave Reception of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)**

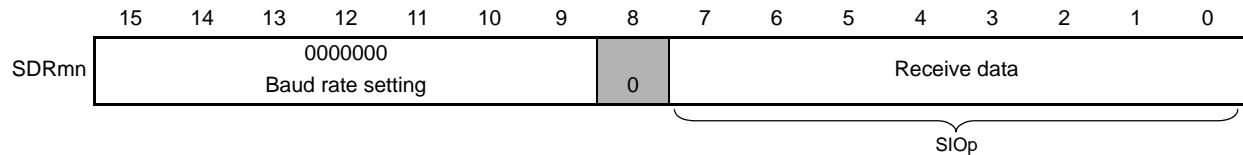
## (a) Serial mode register mn (SMRmn)



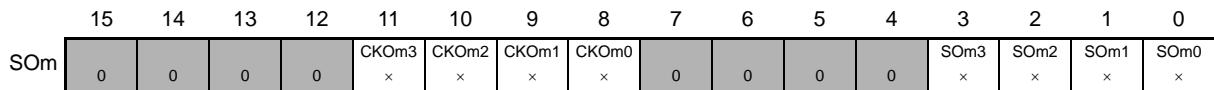
## (b) Serial communication operation setting register mn (SCRmn)



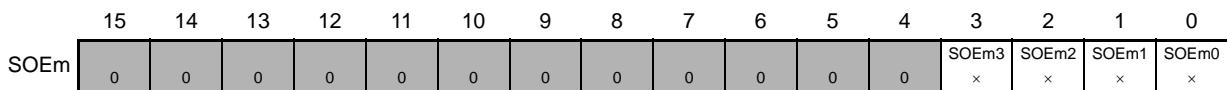
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



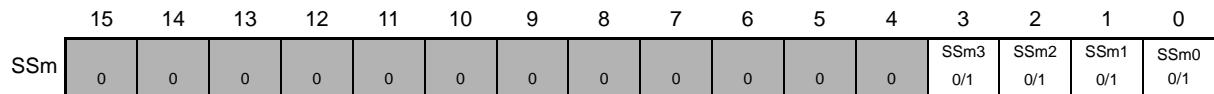
## (d) Serial output register m (SOm)... The Register that not used in this mode.



## (e) Serial output enable register m (SOEm)... The Register that not used in this mode.



## (f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.



**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

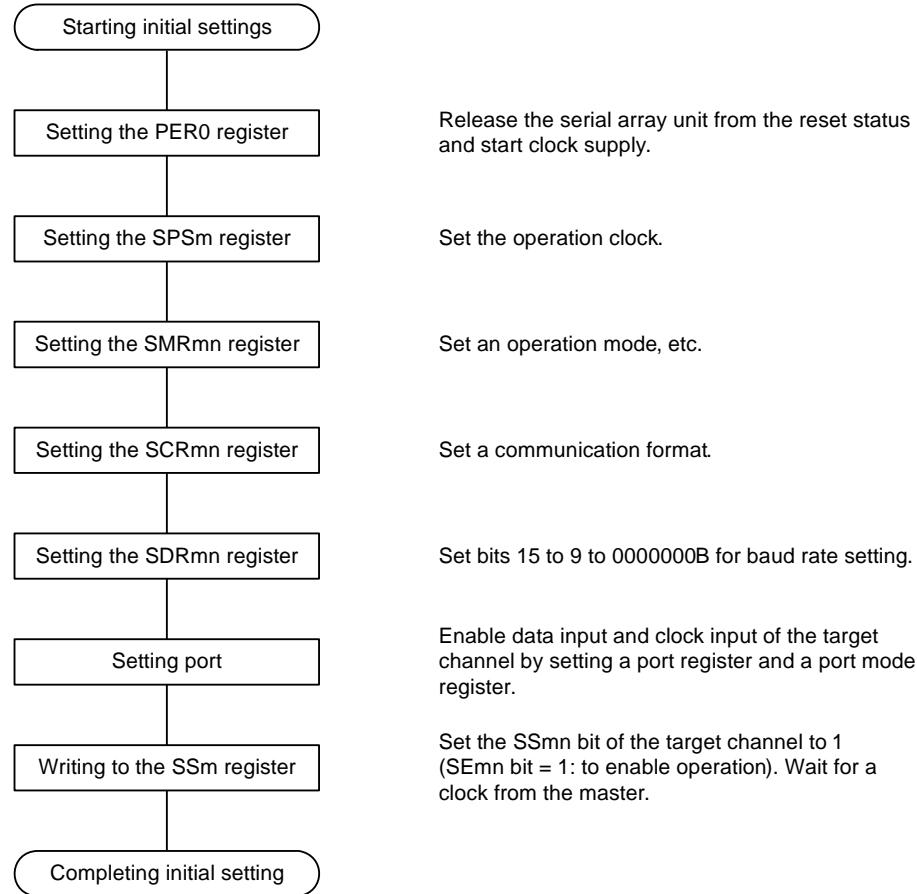
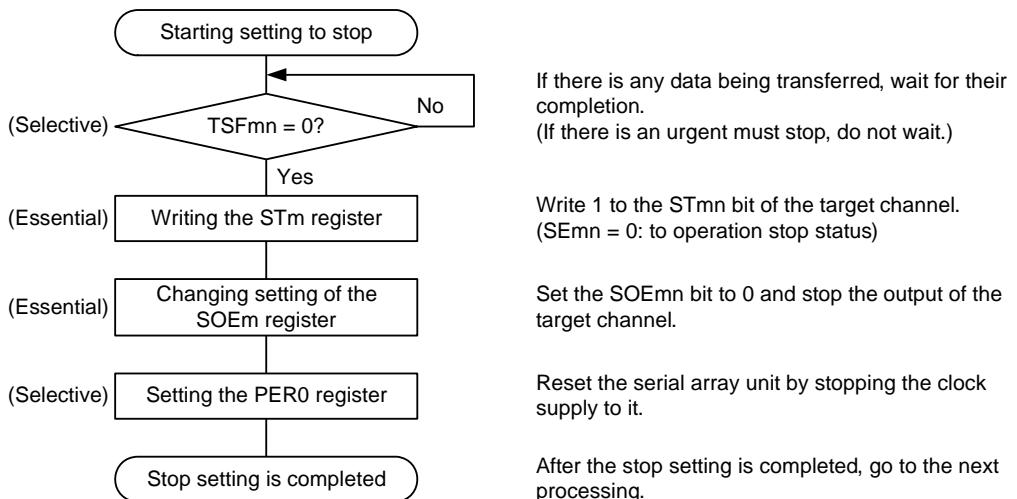
**Remark 2.** : Setting is fixed in the CSI slave reception mode,

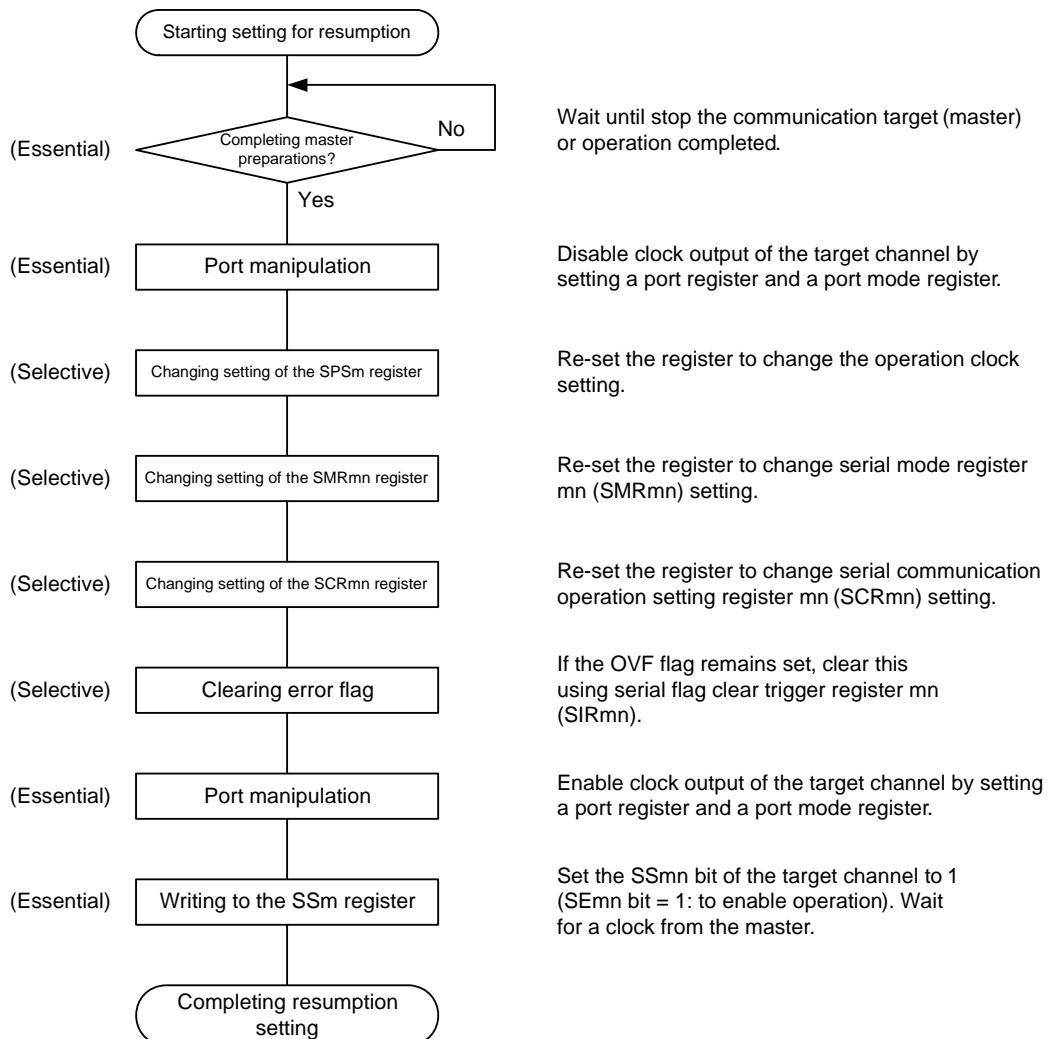
: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

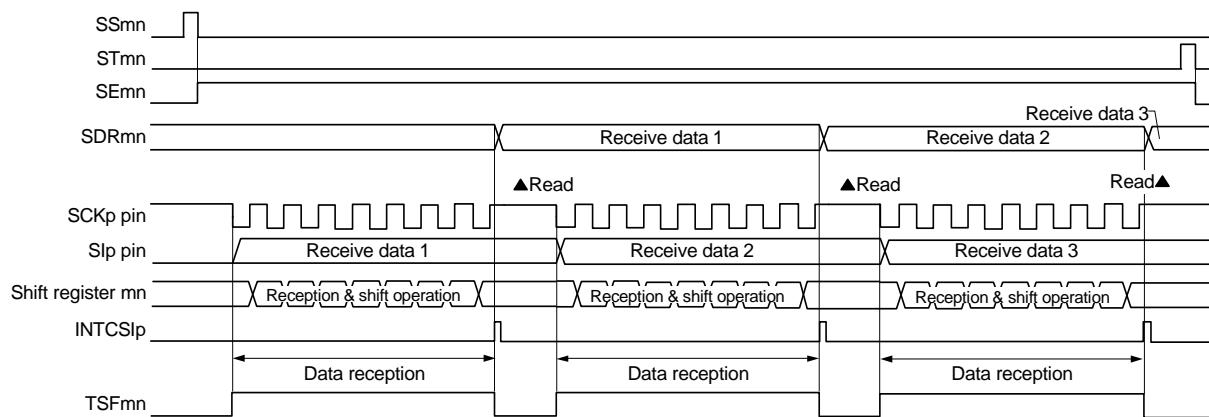
**Figure 17 - 61 Initial Setting Procedure for Slave Reception****Figure 17 - 62 Procedure for Stopping Slave Reception**

**Figure 17 - 63 Procedure for Resuming Slave Reception**

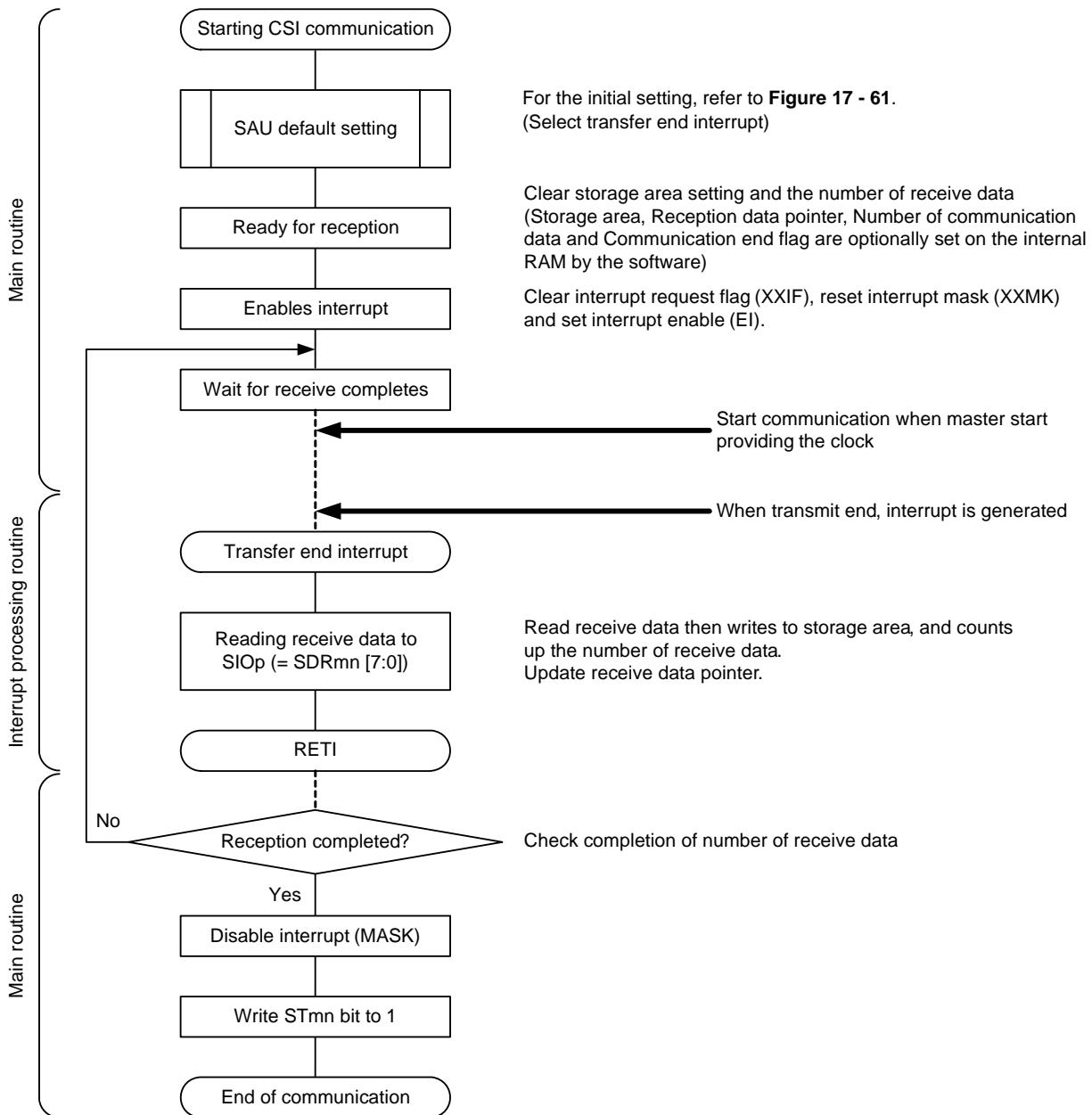
**Remark** If PERO is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (master) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-reception mode)

**Figure 17 - 64 Timing Chart of Slave Reception (in Single-Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

**Figure 17 - 65 Flowchart of Slave Reception (in Single-Reception Mode)**

### 17.5.6 Slave transmission/reception

Slave transmission/reception is that the RL78 microcontroller transmits/receives data to/from another device in the state of a transfer clock being input from another device.

3-Wire Serial I/O	CSI00	CSI01	CSI10	CSI11	CSI20	CSI21	CSI30	CSI31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCK00, SI00, SO00	SCK01, SI01, SO01	SCK10, SI10, SO10	SCK11, SI11, SO11	SCK20, SI20, SO20	SCK21, SI21, SO21	SCK30, SI30, SO30	SCK31, SI31, SO31
Interrupt	INTCSI00	INTCSI01	INTCSI10	INTCSI11	INTCSI20	INTCSI21	INTCSI30	INTCSI31
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	7 or 8 bits							
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2.							
Data phase	Selectable by the DAPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• DAPmn = 0: Data I/O starts from the start of the operation of the serial clock.</li> <li>• DAPmn = 1: Data I/O starts half a clock before the start of the serial clock operation.</li> </ul>							
Clock phase	Selectable by the CKPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• CKPmn = 0: Non-reverse</li> <li>• CKPmn = 1: Reverse</li> </ul>							
Data direction	MSB or LSB first							

**Note 1.** Because the external serial clock input to the SCK00, SCK01, SCK10, SCK11, SCK20, SCK21, SCK30, and SCK31 pins is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

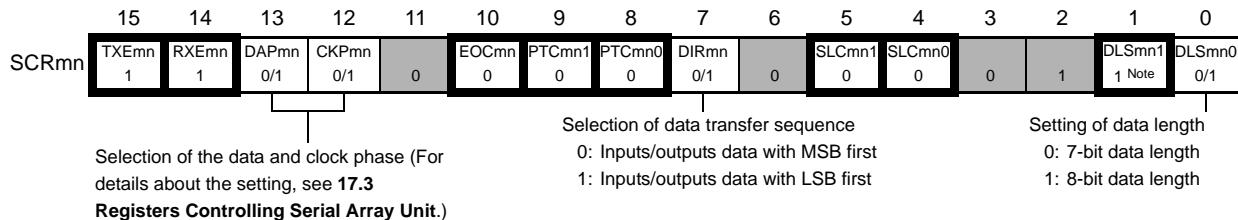
## (1) Register setting

**Figure 17 - 66 Example of Contents of Registers for Slave Transmission/Reception of 3-Wire Serial I/O  
(CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31)**

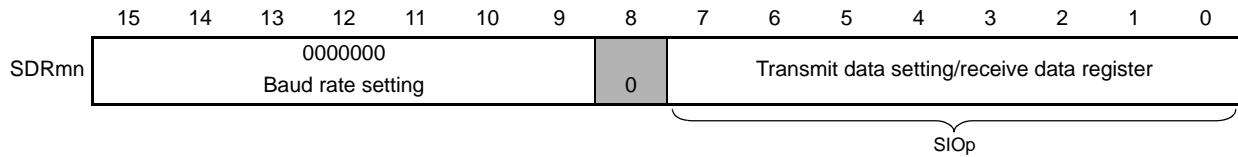
## (a) Serial mode register mn (SMRmn)



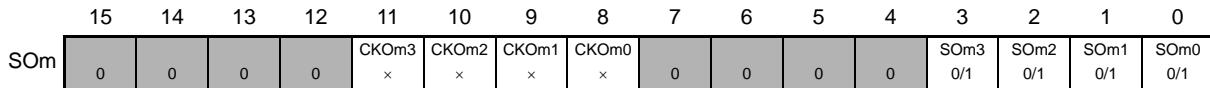
## (b) Serial communication operation setting register mn (SCRmn)



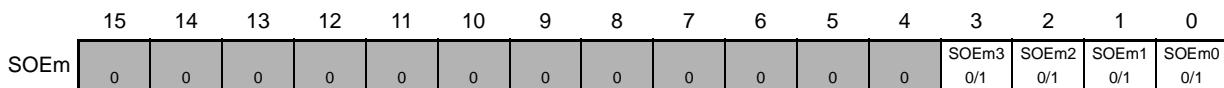
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



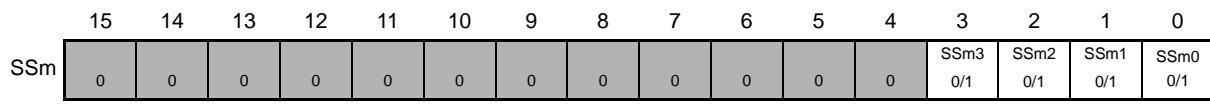
## (d) Serial output register m (SOM)... Sets only the bits of the target channel.



## (e) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.



## (f) Serial channel start register m (SSM)... Sets only the bits of the target channel to 1.



**Note** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

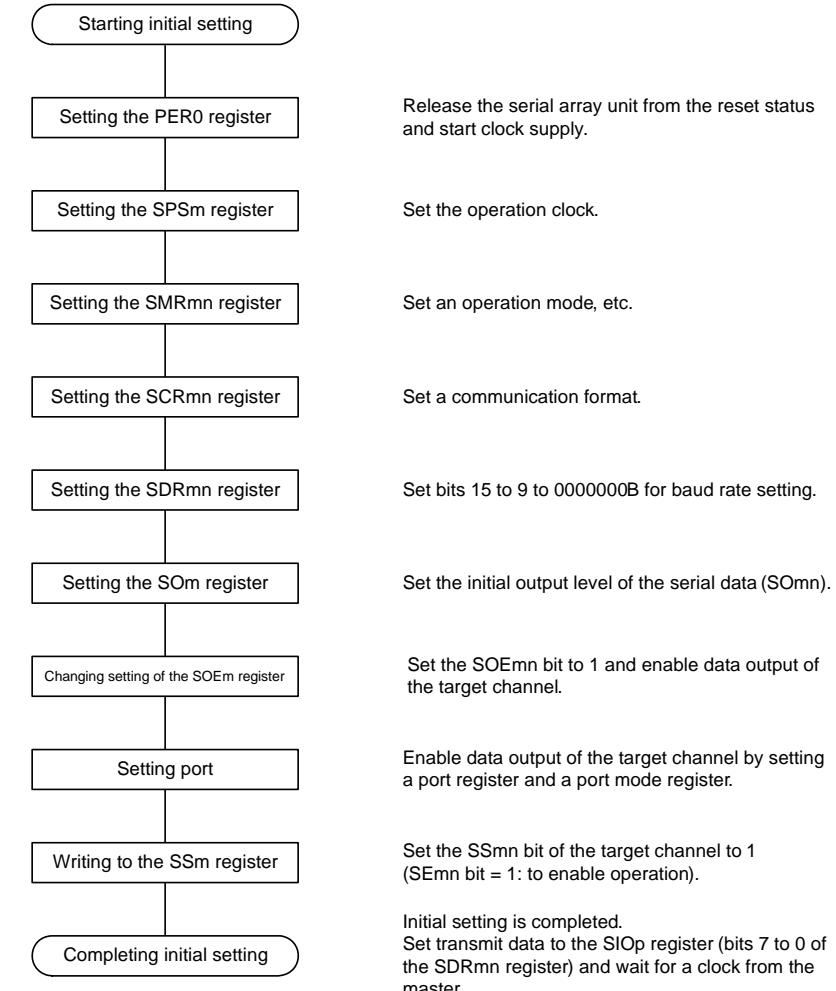
**Remark 2.** : Setting is fixed in the CSI master transmission/reception mode

: Setting disabled (set to the initial value)

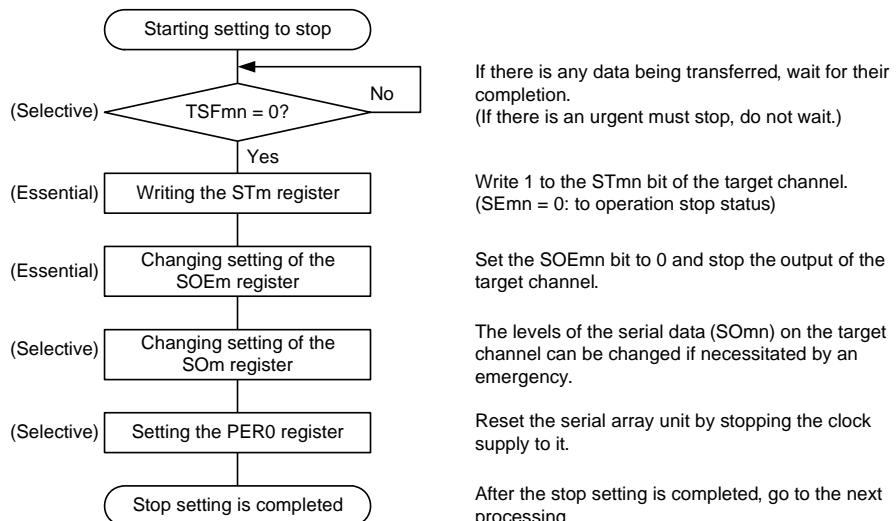
x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

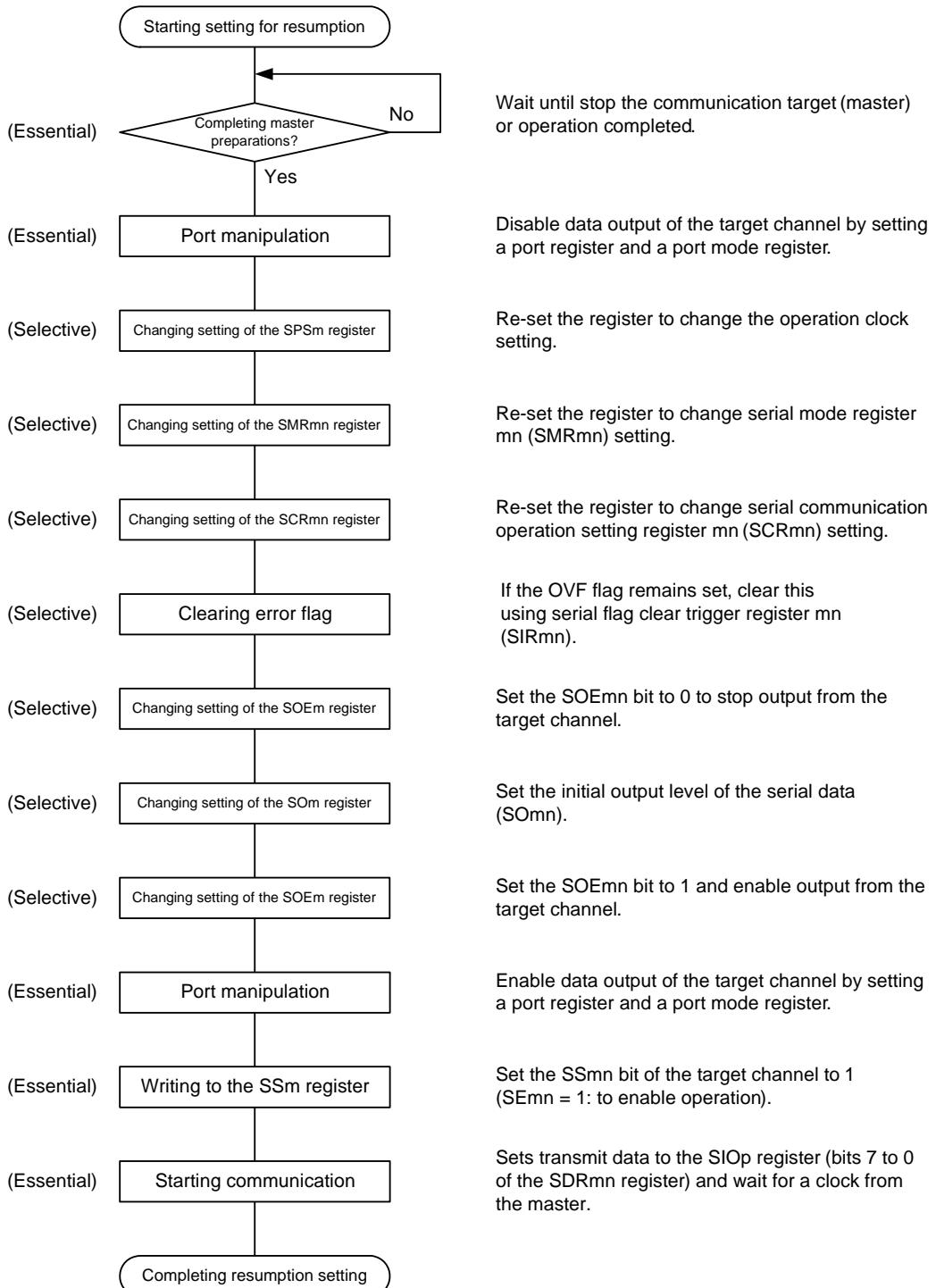
0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

**Figure 17 - 67 Initial Setting Procedure for Slave Transmission/Reception**

**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Figure 17 - 68 Procedure for Stopping Slave Transmission/Reception**

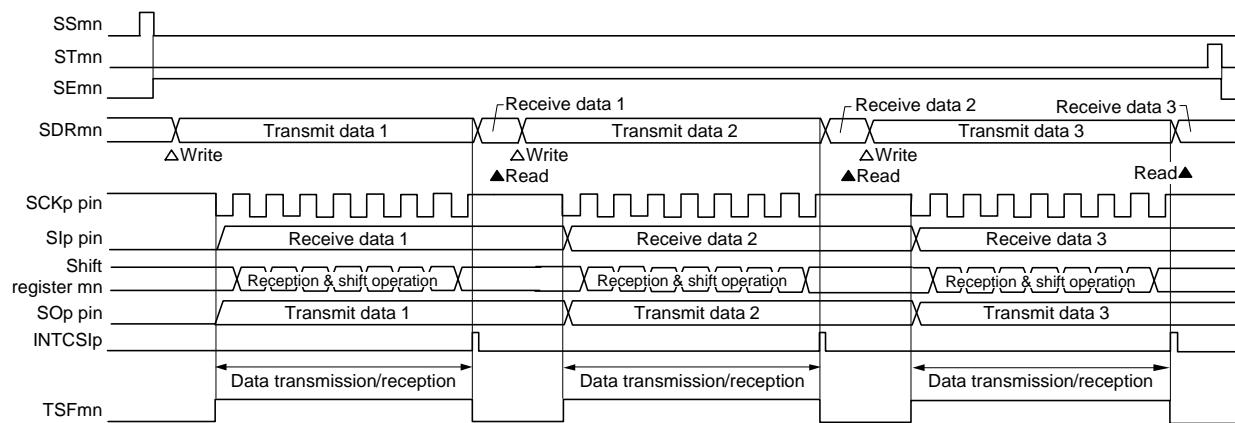
**Figure 17 - 69 Procedure for Resuming Slave Transmission/Reception**

**Caution 1.** Be sure to set transmit data to the SIOp register before the clock from the master is started.

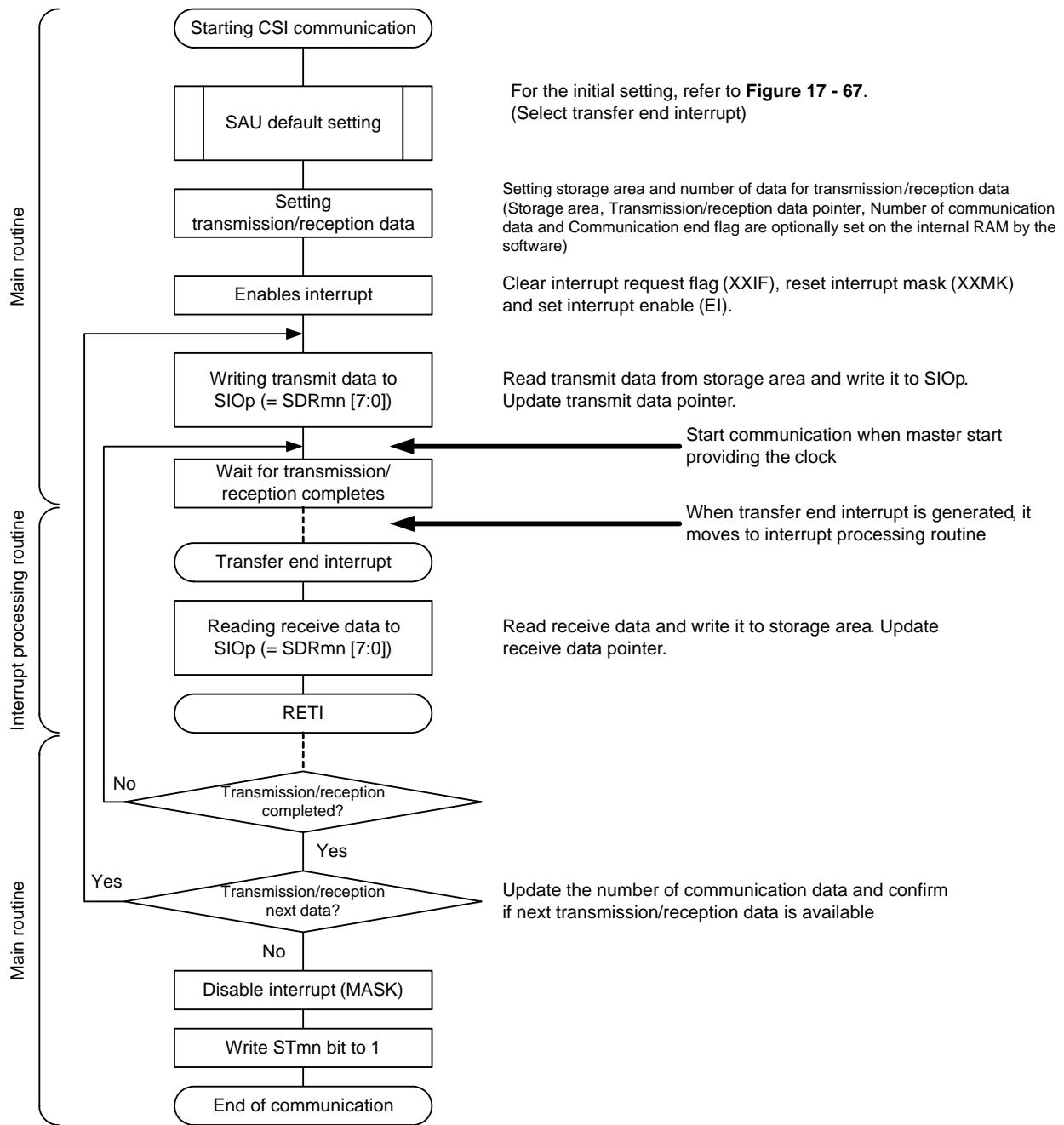
**Caution 2.** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (master) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-transmission/reception mode)

**Figure 17 - 70 Timing Chart of Slave Transmission/Reception (in Single-Transmission/Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



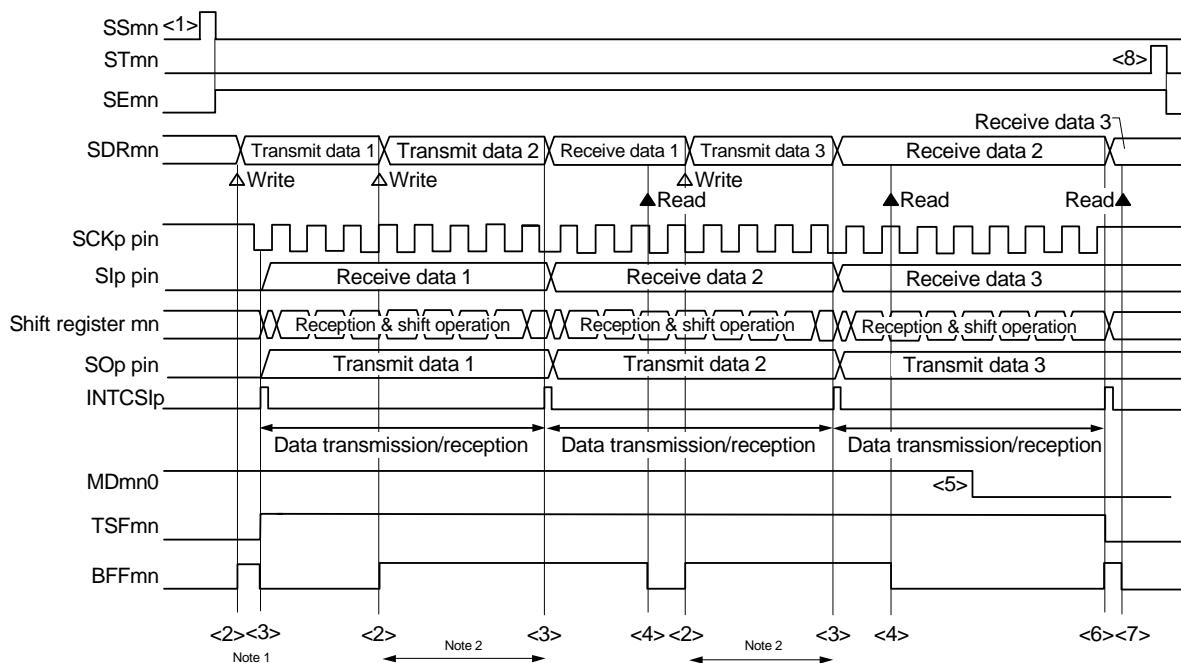
**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), p: CSI number ( $p = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

**Figure 17 - 71 Flowchart of Slave Transmission/Reception (in Single- Transmission/Reception Mode)**

**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

## (4) Processing flow (in continuous transmission/reception mode)

**Figure 17 - 72 Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Note 1.** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

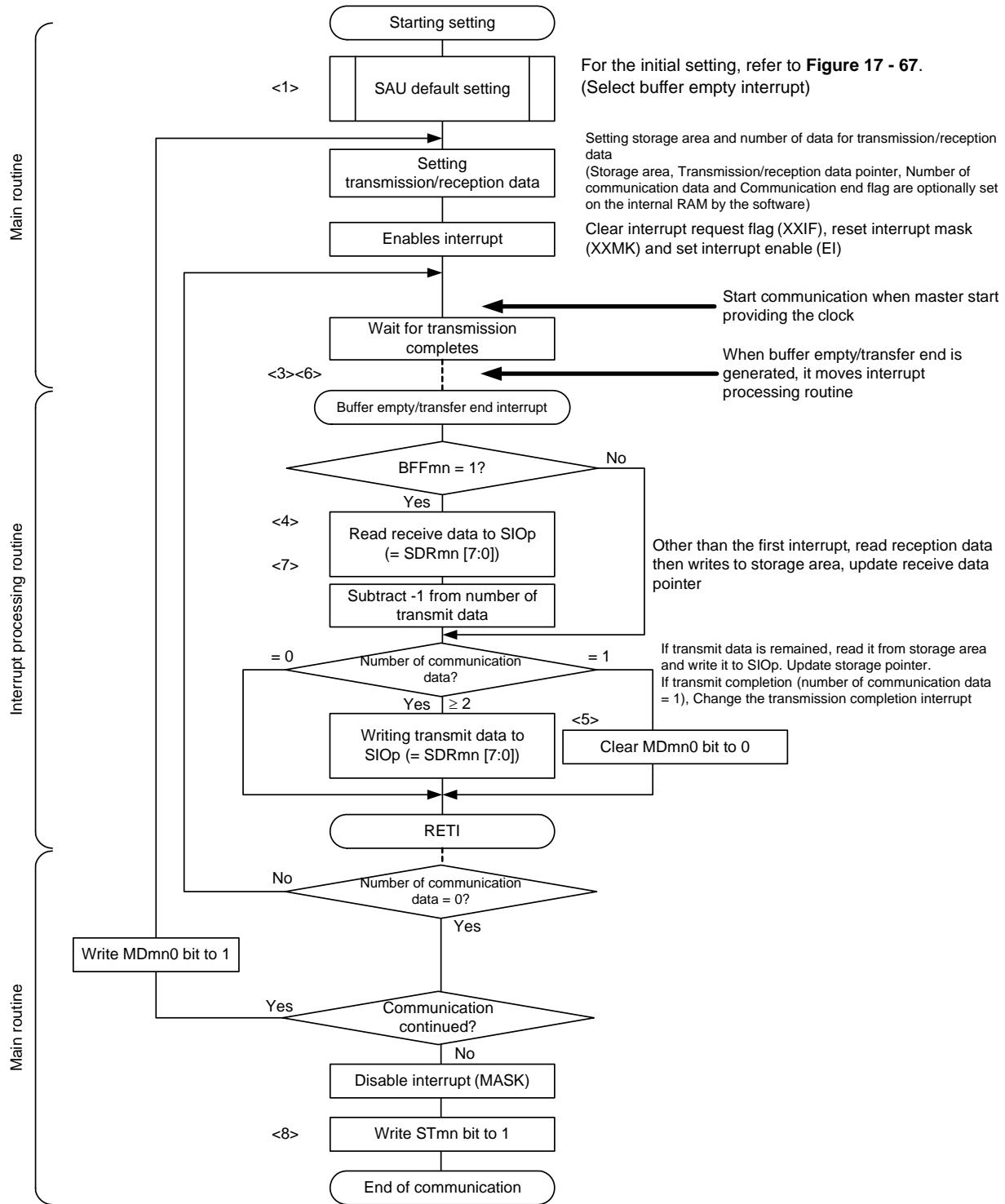
**Note 2.** The transmit data can be read by reading the SDRmn register during this period. At this time, the transfer operation is not affected.

**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation.  
 However, rewrite it before transfer of the last bit is started, so that it has been rewritten before the transfer end interrupt of the last transmit data.

**Remark 1.** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 73 Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode).

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)  
 mn = 00 to 03, 10 to 13

Figure 17 - 73 Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)



**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Remark** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 72 Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0).

### 17.5.7 SNOOZE mode function

SNOOZE mode makes CSI operate reception by SCKp pin input detection while the STOP mode. Normally CSI stops communication in the STOP mode. But, using the SNOOZE mode makes reception CSI operate unless the CPU operation by detecting SCKp pin input. Only following channels can be set to the SNOOZE mode.

- 30 to 64-pin products: CSI00
- 80, 100-pin products: CSI00 and CSI20

When using the CSI in SNOOZE mode, make the following setting before switching to the STOP mode (see **Figure 17 - 75** and **Figure 17 - 77 Flowchart of SNOOZE Mode Operation**).

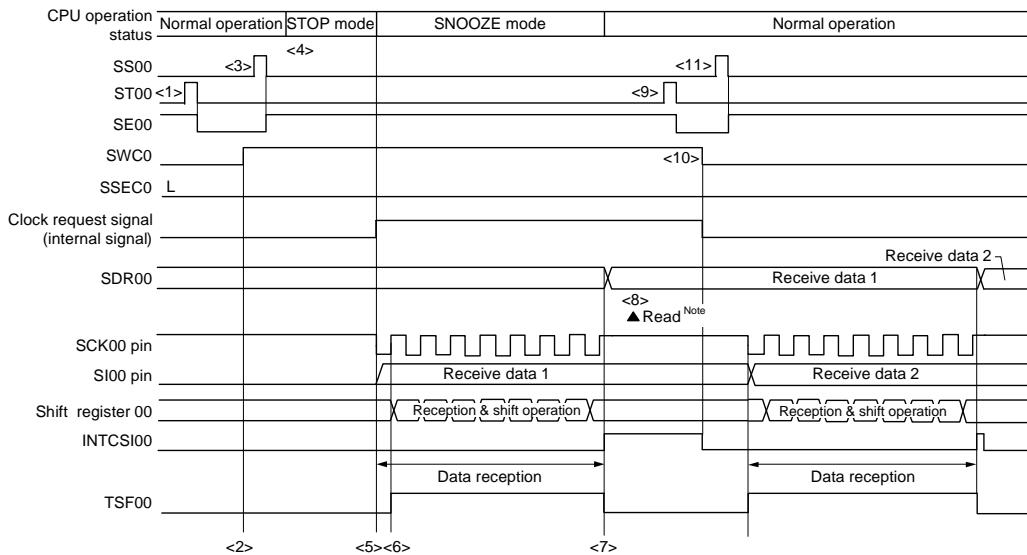
- When using the SNOOZE mode function, set the SWCm bit of serial standby control register m (SSCm) to 1 just before switching to the STOP mode. After the initial setting has been completed, set the SSm0 bit of serial channel start register m (SSm) to 1.
- The CPU shifts to the SNOOZE mode on detecting the valid edge of the SCKp signal following a transition to the STOP mode. A CSIp starts reception on detecting input of the serial clock on the SCKp pin.

**Caution 1. The SNOOZE mode can only be specified when the high-speed on-chip oscillator clock is selected for fCLK.**

**Caution 2. The maximum transfer rate when using CSIp in the SNOOZE mode is 1 Mbps.**

- (1) SNOOZE mode operation (once startup)

**Figure 17 - 74 Timing Chart of SNOOZE Mode Operation (once startup) (Type 1: DAPmn = 0, CKPmn = 0)**



**Note** Only read received data while SWCm = 1 and before the next edge of the SCKp pin input is detected.

**Caution 1. Before switching to the SNOOZE mode or after reception operation in the SNOOZE mode finishes, set the STM0 bit to 1 (clear the SEM0 bit, and stop the operation).**

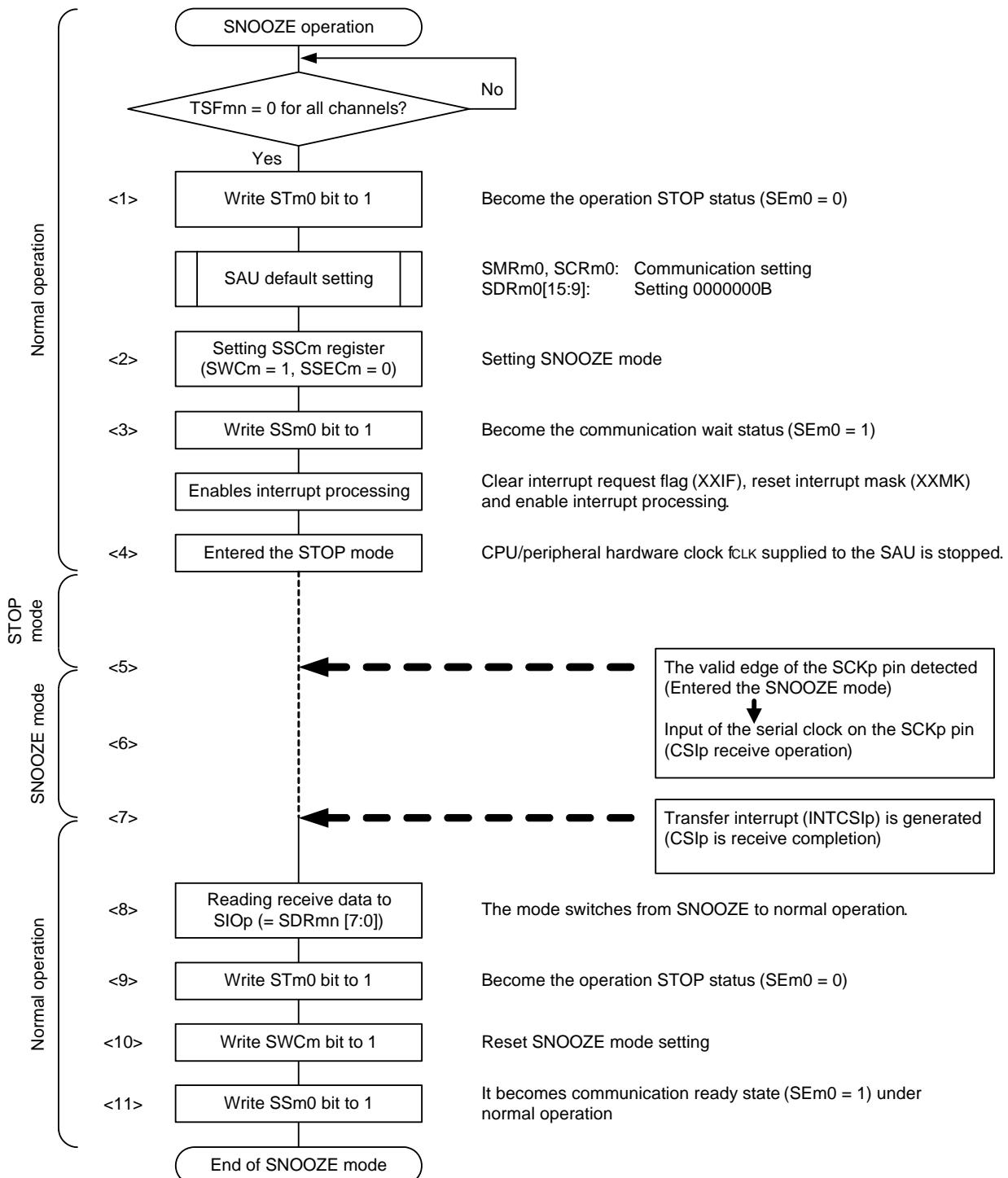
And after completion the receive operation, also clearing SWCm bit to 0 (SNOOZE mode release).

**Caution 2. When SWCm = 1, the BFFm1 and OVFm1 flags will not change.**

**Remark 1.** <1> to <11> in the figure correspond to <1> to <11> in Figure 17 - 75 Flowchart of SNOOZE Mode Operation (once startup).

**Remark 2.** 30 to 64-pin products: m = 0; p = 00  
80, 100-pin products: m = 0, 1; p = 00, 20

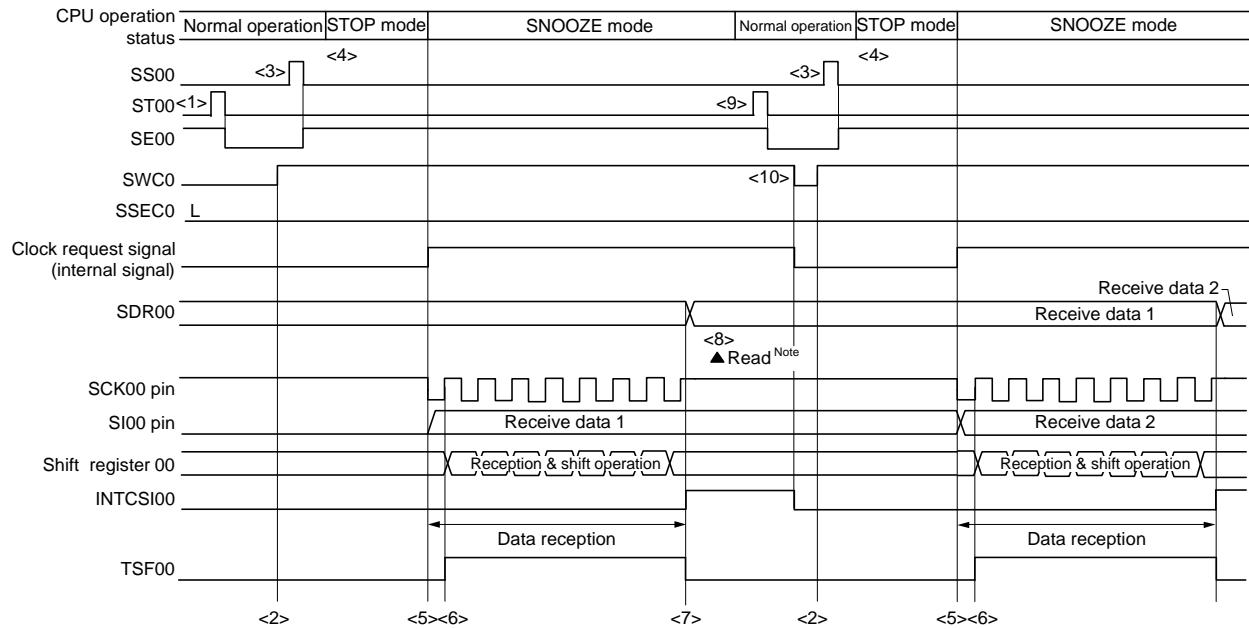
Figure 17 - 75 Flowchart of SNOOZE Mode Operation (once startup)



**Remark 1.** <1> to <11> in the figure correspond to <1> to <11> in Figure 17 - 74 Timing Chart of SNOOZE Mode Operation (once startup) (Type 1: DAPmn = 0, CKPmn = 0).

**Remark 2.** 30 to 64-pin products: m = 0; p = 00  
80, 100-pin products: m = 0, 1; p = 00, 20

## (2) SNOOZE mode operation (continuous startup)

**Figure 17 - 76 Timing Chart of SNOOZE Mode Operation (continuous startup) (Type 1: DAPmn = 0, CKPmn = 0)**

**Note** Only read received data while SWCm = 1 and before the next edge of the SCKp pin input is detected.

**Caution 1.** Before switching to the SNOOZE mode or after reception operation in the SNOOZE mode finishes, set the STm0 bit to 1 (clear the SEM0 bit, and stop the operation).

And after completion the receive operation, also clearing SWCm bit to 0 (SNOOZE release).

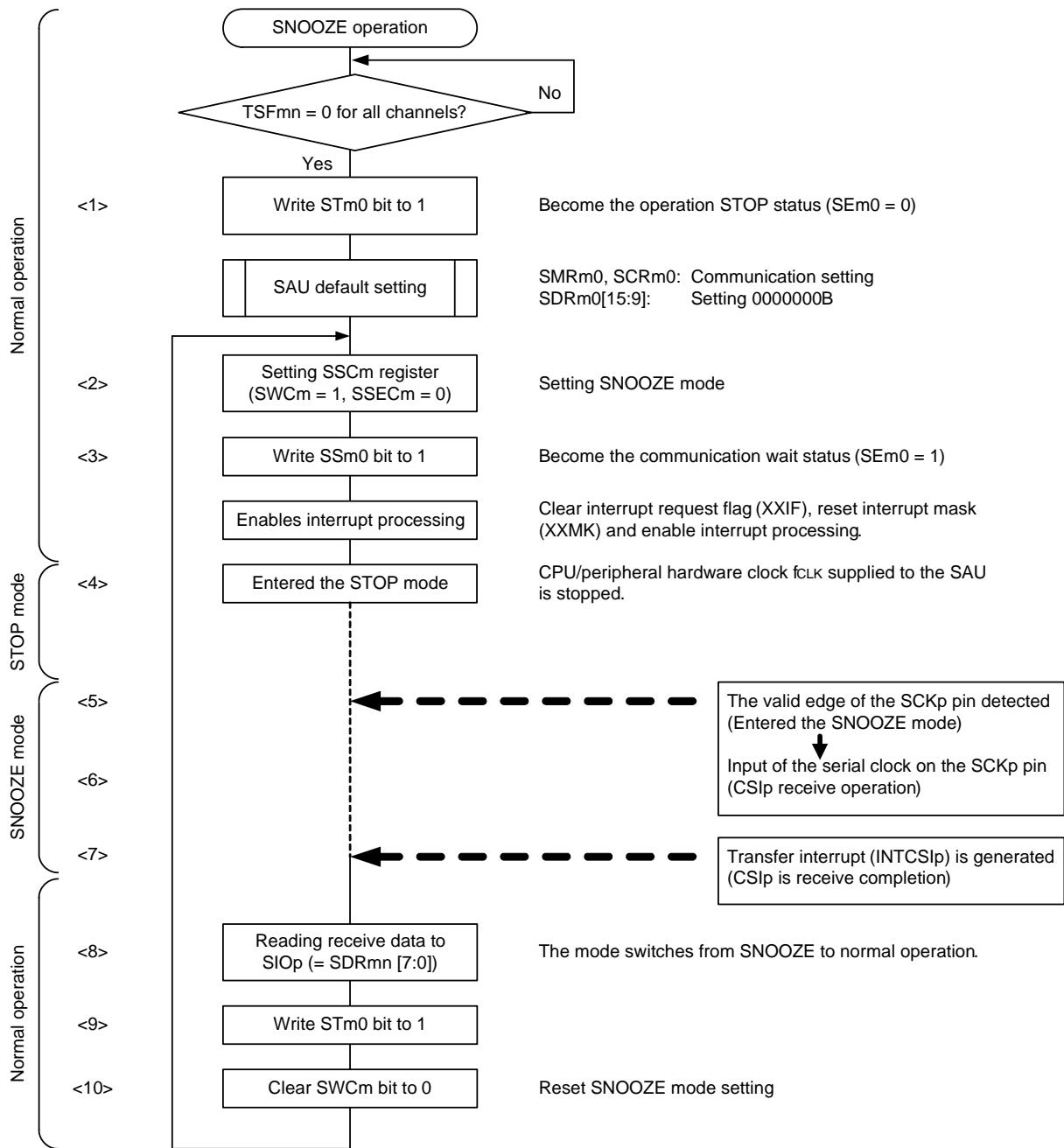
**Caution 2.** When SWCm = 1, the BFFm1 and OVFm1 flags will not change.

**Remark 1.** <1> to <10> in the figure correspond to <1> to <10> in Figure 17 - 77 Flowchart of SNOOZE Mode Operation (continuous startup).

**Remark 2.** 30 to 64-pin products: m = 0; p = 00

80, 100-pin products: m = 0, 1; p = 00, 20

Figure 17 - 77 Flowchart of SNOOZE Mode Operation (continuous startup)



**Remark 1.** <1> to <10> in the figure correspond to <1> to <10> in Figure 17 - 76 Timing Chart of SNOOZE Mode Operation (continuous startup) (Type 1: DAPmn = 0, CKPmn = 0).

**Remark 2.** 30 to 64-pin products: m = 0; p = 00  
80, 100-pin products: m = 0, 1; p = 00, 20

### 17.5.8 Calculating transfer clock frequency

The transfer clock frequency for 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) communication can be calculated by the following expressions.

(1) Master

$$\text{(Transfer clock frequency)} = \{\text{Operation clock (fmck) frequency of target channel}\} \div (\text{SDRmn[15:9]} + 1) \div 2 \text{ [Hz]}$$

(2) Slave

$$\text{(Transfer clock frequency)} = \{\text{Frequency of serial clock (SCK) supplied by master}\} \text{ Note [Hz]}$$

**Note** The permissible maximum transfer clock frequency is fmck/6.

**Remark** The value of SDRmn[15:9] is the value of bits 15 to 9 of serial data register mn (SDRmn) (0000000B to 1111111B) and therefore is 0 to 127.

The operation clock (fmck) is determined by serial clock select register m (SPSm) and bit 15 (CKSmn) of serial mode register mn (SMRmn).

Table 17 - 2 Selection of Operation Clock For 3-Wire Serial I/O

SMRmn Register	SPSm Register								Operation Clock (fMCK) Note	
CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00		
0	x	x	x	x	0	0	0	0	fCLK	32 MHz
	x	x	x	x	0	0	0	1	fCLK/2	16 MHz
	x	x	x	x	0	0	1	0	fCLK/2 <sup>2</sup>	8 MHz
	x	x	x	x	0	0	1	1	fCLK/2 <sup>3</sup>	4 MHz
	x	x	x	x	0	1	0	0	fCLK/2 <sup>4</sup>	2 MHz
	x	x	x	x	0	1	0	1	fCLK/2 <sup>5</sup>	1 MHz
	x	x	x	x	0	1	1	0	fCLK/2 <sup>6</sup>	500 kHz
	x	x	x	x	0	1	1	1	fCLK/2 <sup>7</sup>	250 kHz
	x	x	x	x	1	0	0	0	fCLK/2 <sup>8</sup>	125 kHz
	x	x	x	x	1	0	0	1	fCLK/2 <sup>9</sup>	62.5 kHz
	x	x	x	x	1	0	1	0	fCLK/2 <sup>10</sup>	31.25 kHz
	x	x	x	x	1	0	1	1	fCLK/2 <sup>11</sup>	15.63 kHz
	x	x	x	x	1	1	0	0	fCLK/2 <sup>12</sup>	7.81 kHz
	x	x	x	x	1	1	0	1	fCLK/2 <sup>13</sup>	3.91 kHz
	x	x	x	x	1	1	1	0	fCLK/2 <sup>14</sup>	1.95 kHz
	x	x	x	x	1	1	1	1	fCLK/2 <sup>15</sup>	977 Hz
1	0	0	0	0	x	x	x	x	fCLK	32 MHz
	0	0	0	1	x	x	x	x	fCLK/2	16 MHz
	0	0	1	0	x	x	x	x	fCLK/2 <sup>2</sup>	8 MHz
	0	0	1	1	x	x	x	x	fCLK/2 <sup>3</sup>	4 MHz
	0	1	0	0	x	x	x	x	fCLK/2 <sup>4</sup>	2 MHz
	0	1	0	1	x	x	x	x	fCLK/2 <sup>5</sup>	1 MHz
	0	1	1	0	x	x	x	x	fCLK/2 <sup>6</sup>	500 kHz
	0	1	1	1	x	x	x	x	fCLK/2 <sup>7</sup>	250 kHz
	1	0	0	0	x	x	x	x	fCLK/2 <sup>8</sup>	125 kHz
	1	0	0	1	x	x	x	x	fCLK/2 <sup>9</sup>	62.5 kHz
	1	0	1	0	x	x	x	x	fCLK/2 <sup>10</sup>	31.25 kHz
	1	0	1	1	x	x	x	x	fCLK/2 <sup>11</sup>	15.63 kHz
	1	1	0	0	x	x	x	x	fCLK/2 <sup>12</sup>	7.81 kHz
	1	1	0	1	x	x	x	x	fCLK/2 <sup>13</sup>	3.91 kHz
	1	1	1	0	x	x	x	x	fCLK/2 <sup>14</sup>	1.95 kHz
	1	1	1	1	x	x	x	x	fCLK/2 <sup>15</sup>	977 Hz

**Note** When changing the clock selected for fCLK (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (STM) = 000FH) the operation of the serial array unit (SAU).

**Remark 1.** x: Don't care

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

### 17.5.9 Procedure for processing errors that occurred during 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) communication

The procedure for processing errors that occurred during 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) communication is described in Figure 17 - 78.

**Figure 17 - 78 Processing Procedure in Case of Overrun Error**

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn).	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes 1 to serial flag clear trigger register mn (SIRmn).	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## 17.6 Clock Synchronous Serial Communication with Slave Select Input Function

Channel 0 of SAU0 correspond to the clock synchronous serial communication with slave select input function.

[Data transmission/reception]

- Data length of 7 or 8 bits
- Phase control of transmit/receive data
- MSB/LSB first selectable

[Clock control]

- Phase control of I/O clock
- Setting of transfer period by prescaler and internal counter of each channel
- Maximum transfer rate Note

During slave communication: Max. fmck/6

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt

[Error detection flag]

- Overrun error

**Note** Use the clocks within a range satisfying the SCK cycle time ( $t_{KCY}$ ) characteristics. For details, see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**.

- 30, 32-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	—		—

- 36, 40, 44-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 48, 52-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 64-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 80, 100-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21
	2	CSI30	UART3	IIC30
	3	CSI31		IIC31

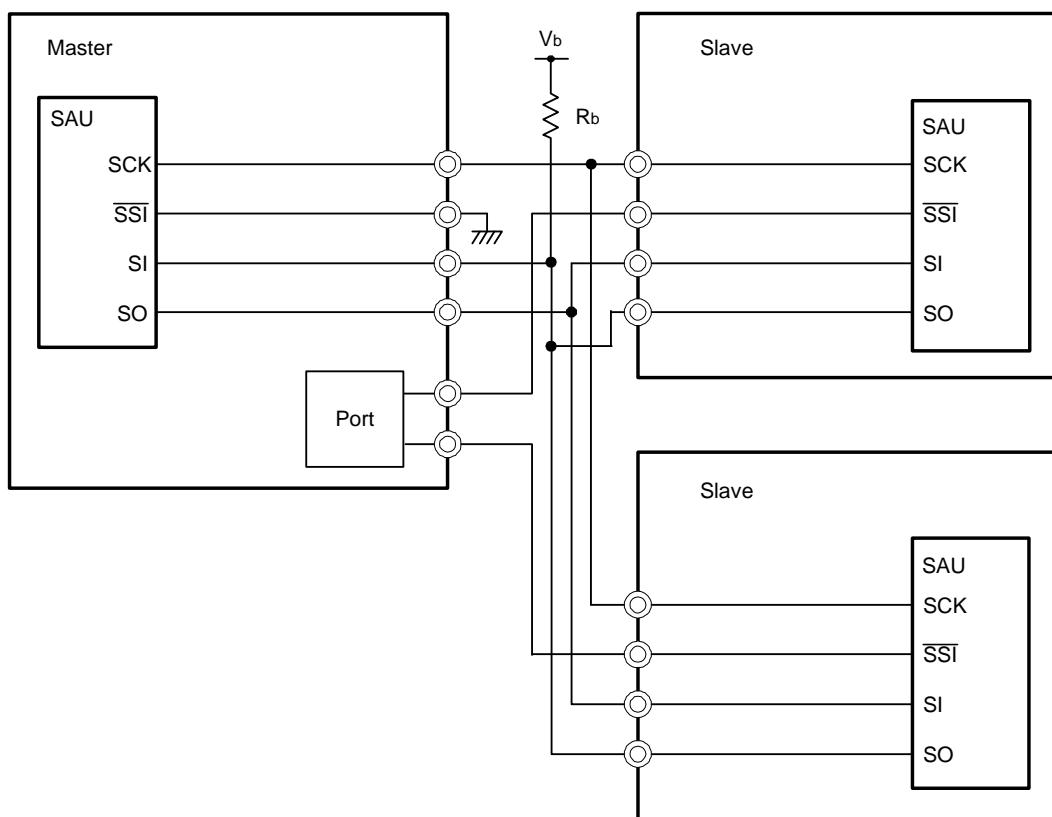
Slave select input function performs the following three types of communication operations.

- Slave transmission (See 17.6.1.)
- Slave reception (See 17.6.2.)
- Slave transmission/reception (See 17.6.3.)

Multiple slaves can be connected to a master and communication can be performed by using the slave select input function. The master outputs a slave select signal to the slave (one) that is the other party of communication, and each slave judges whether it has been selected as the other party of communication and controls the SO pin output. When a slave is selected, transmit data can be communicated from the SO pin to the master. When a slave is not selected, the SO pin is set to high-level output. Therefore, in an environment where multiple slaves are connected, it is necessary set the SO pin to N-ch open-drain and pull up the node. Furthermore, when a slave is not selected, no transmission/reception operation is performed even if a serial clock is input from the master.

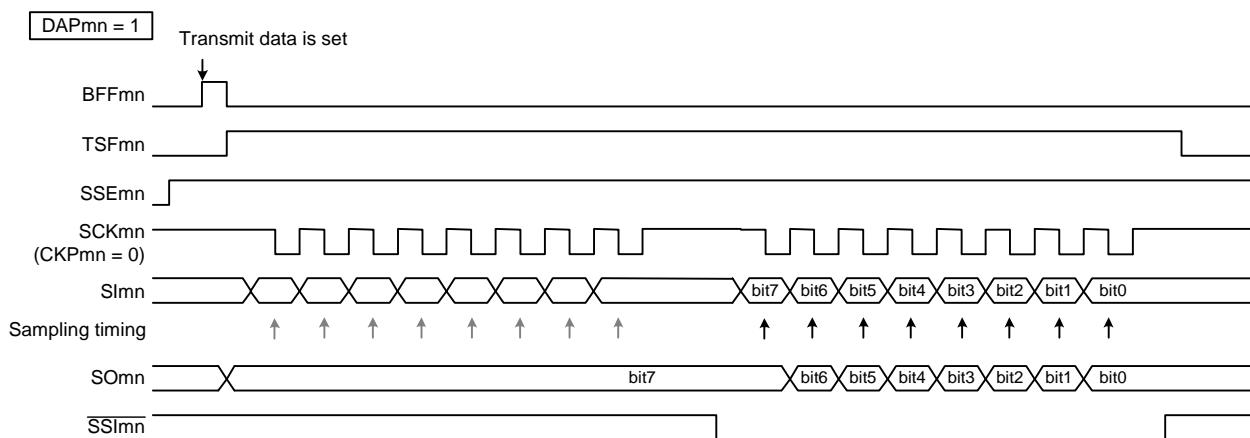
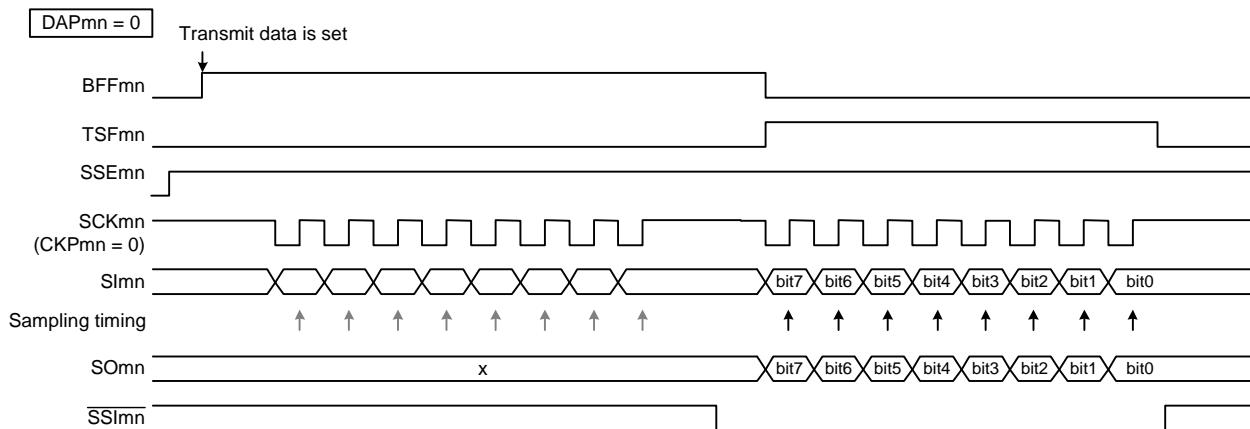
**Caution Output the slave select signal by port manipulation.**

Figure 17 - 79 Example of Slave Select Input Function Configuration



**Caution Make sure EV<sub>DD0</sub> ≥ V<sub>b</sub>.**

Select the N-ch open-drain output (EV<sub>DD</sub> tolerance) mode for the SO00 pin.

**Figure 17 - 80 Slave Select Input Function Timing Diagram**

**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ )

### 17.6.1 Slave transmission

Slave transmission is that the RL78 microcontroller transmits data to another device in the state of a transfer clock being input from another device.

Slave select Input function	CSI00
Target channel	Channel 0 of SAU0
Pins used	SCK00, SO00, SSI00
Interrupt	INTCSI00
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.
Error detection flag	Overrun error detection flag (OVFmn) only
Transfer data length	7 or 8 bits
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2
Data phase	Selectable by the DAPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• DAPmn = 0: Data output starts from the start of the operation of the serial clock.</li> <li>• DAPmn = 1: Data output starts half a clock before the start of the serial clock operation.</li> </ul>
Clock phase	Selectable by the CKPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• CKPmn = 0: Non-reverse</li> <li>• CKPmn = 1: Reverse</li> </ul>
Data direction	MSB or LSB first
Slave select Input function	Slave select input function operation selectable

**Note 1.** Because the external serial clock input to the SCK00 pin is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0)

## (1) Register setting

**Figure 17 - 81 Example of Contents of Registers for Slave Transmission of Slave Select Input Function (CSI00) (1/2)**

## (a) Serial mode register mn (SMRmn)

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 1	0	0	0	0	0	STSmn 0	0	SISmn0 0	1	0	0	MDmn2 0	MDmn1 0	MDmn0 0/1

Operation clock (fmck) of channel n

0: Prescaler output clock CKm0 set by the SPSm register  
1: Prescaler output clock CKm1 set by the SPSm registerInterrupt source of channel n  
0: Transfer end interrupt  
1: Buffer empty interrupt

## (b) Serial communication operation setting register mn (SCRmn)

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 1	RXEmn 0	DAPmn 0/1	CKPmn 0/1	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0/1	0	SLCmn1 0	SLCmn0 0	0	1	DLSmn1 1	DLSmn0 0/1

Selection of the data and clock phase (For details about the setting, see 17.3 Registers Controlling Serial Array Unit)

Selection of data transfer sequence  
0: Inputs/outputs data with MSB first  
1: Inputs/outputs data with LSB firstSetting of data length  
0: 7-bit data length  
1: 8-bit data length

## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	00000000 Baud rate setting								0	Transmit data setting						

SIOp

## (d) Serial output register m (SOm)... Sets only the bits of the target channel.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 x	CKOm2 x	CKOm1 x	CKOm0 x	0	0	0	0	SOm3 x	SOm2 x	SOm1 x	SOm0 0/1

**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)**Remark 2.** : Setting is fixed in the CSI slave transmission mode,: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

**Figure 17 - 82 Example of Contents of Registers for Slave Transmission of Slave Select Input Function (CSI00) (2/2)**

(f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	x	x	x	0/1

(g) Input switch control register (ISC)... SSI00 input setting in CSI00 slave channel (channel 0 of unit 0).

ISC	7	6	5	4	3	2	1	0
	SSI00 0/1	0	0	0	0	0	ISC1 0/1	ISC0 0/1

0: Disables the input value of the SSI00 pin  
1: Enables the input value of the SSI00 pin

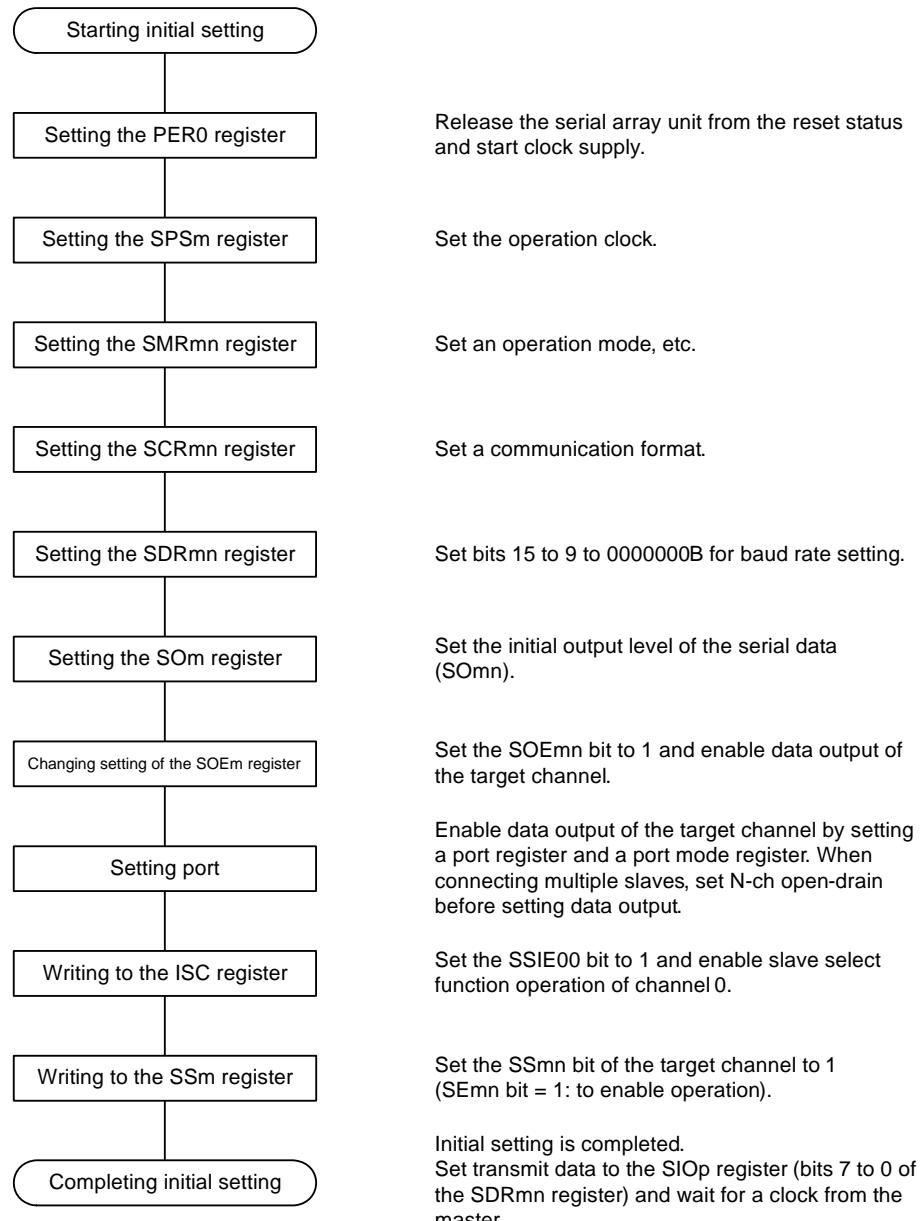
**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Remark 2.** : Setting disabled (set to the initial value)

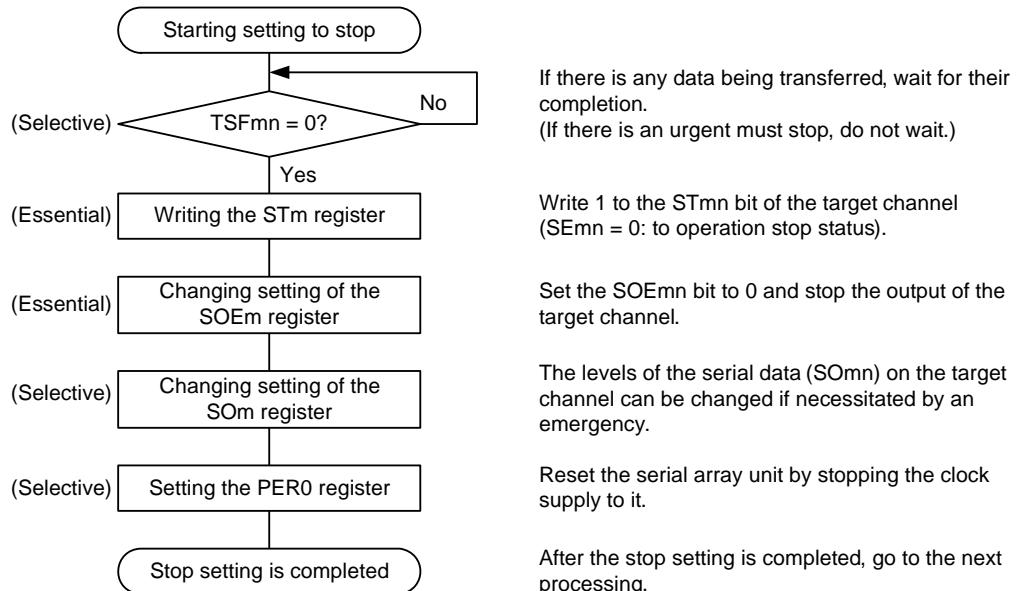
x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

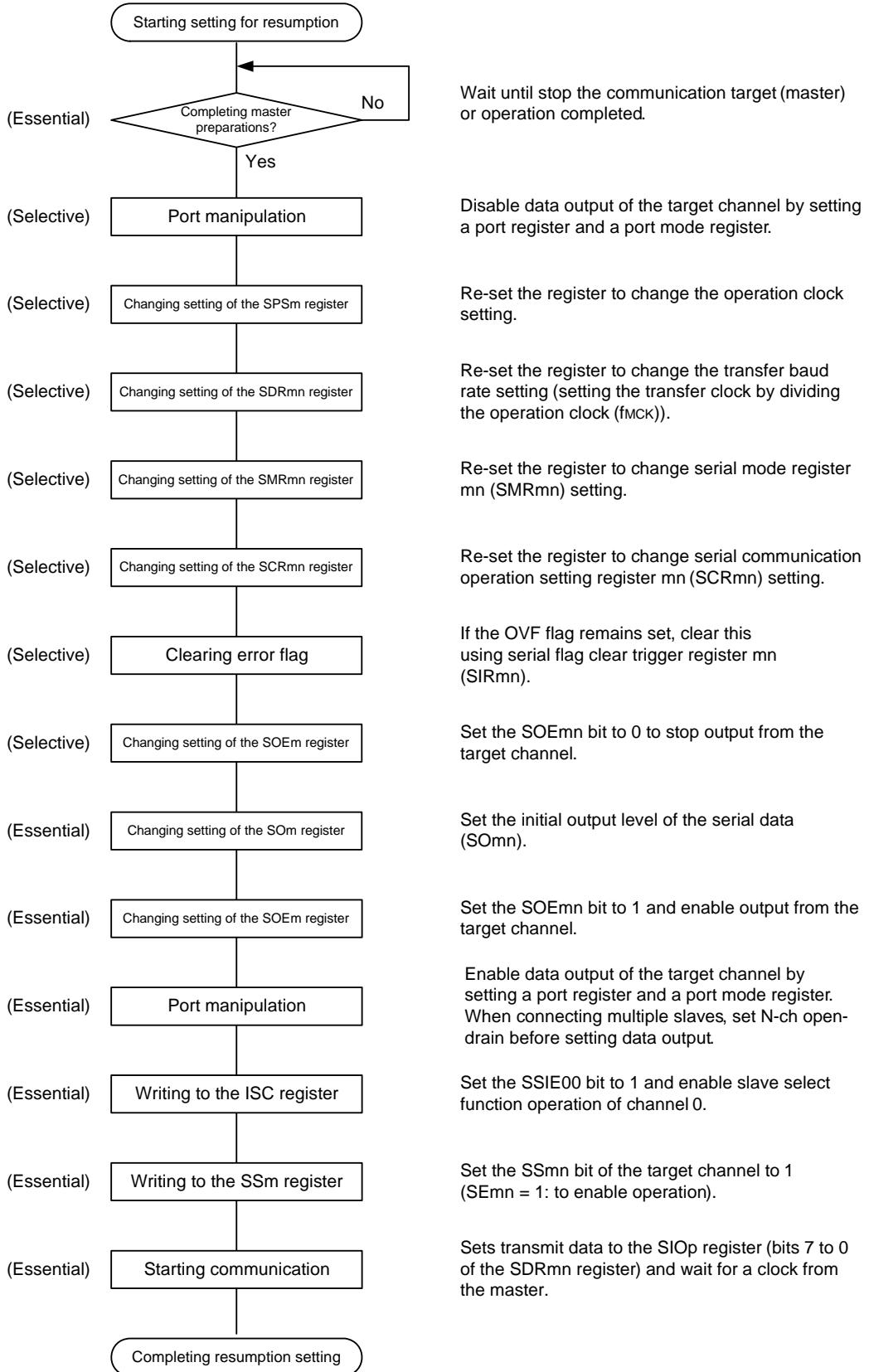
## (2) Operation procedure

**Figure 17 - 83 Initial Setting Procedure for Slave Transmission**

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Figure 17 - 84 Procedure for Stopping Slave Transmission**

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

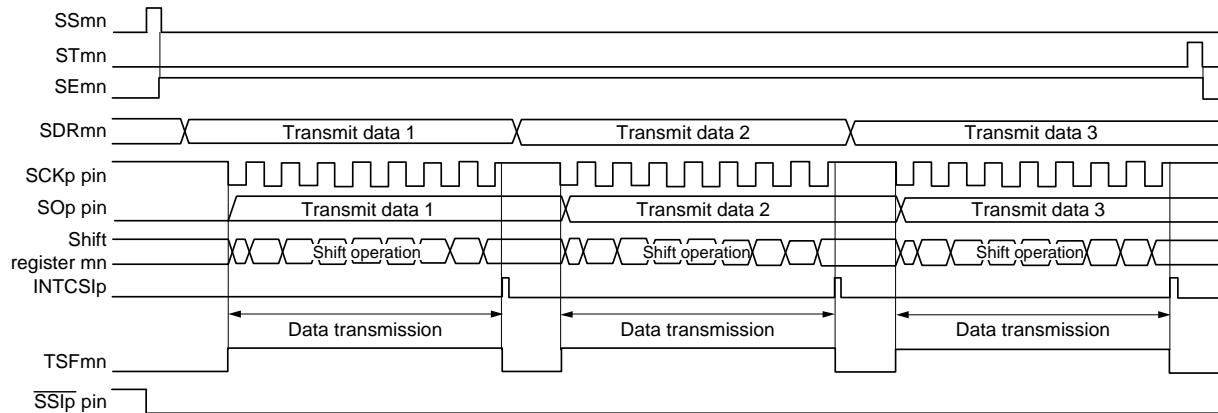
**Figure 17 - 85 Procedure for Resuming Slave Transmission**

**Remark 1.** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (master) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

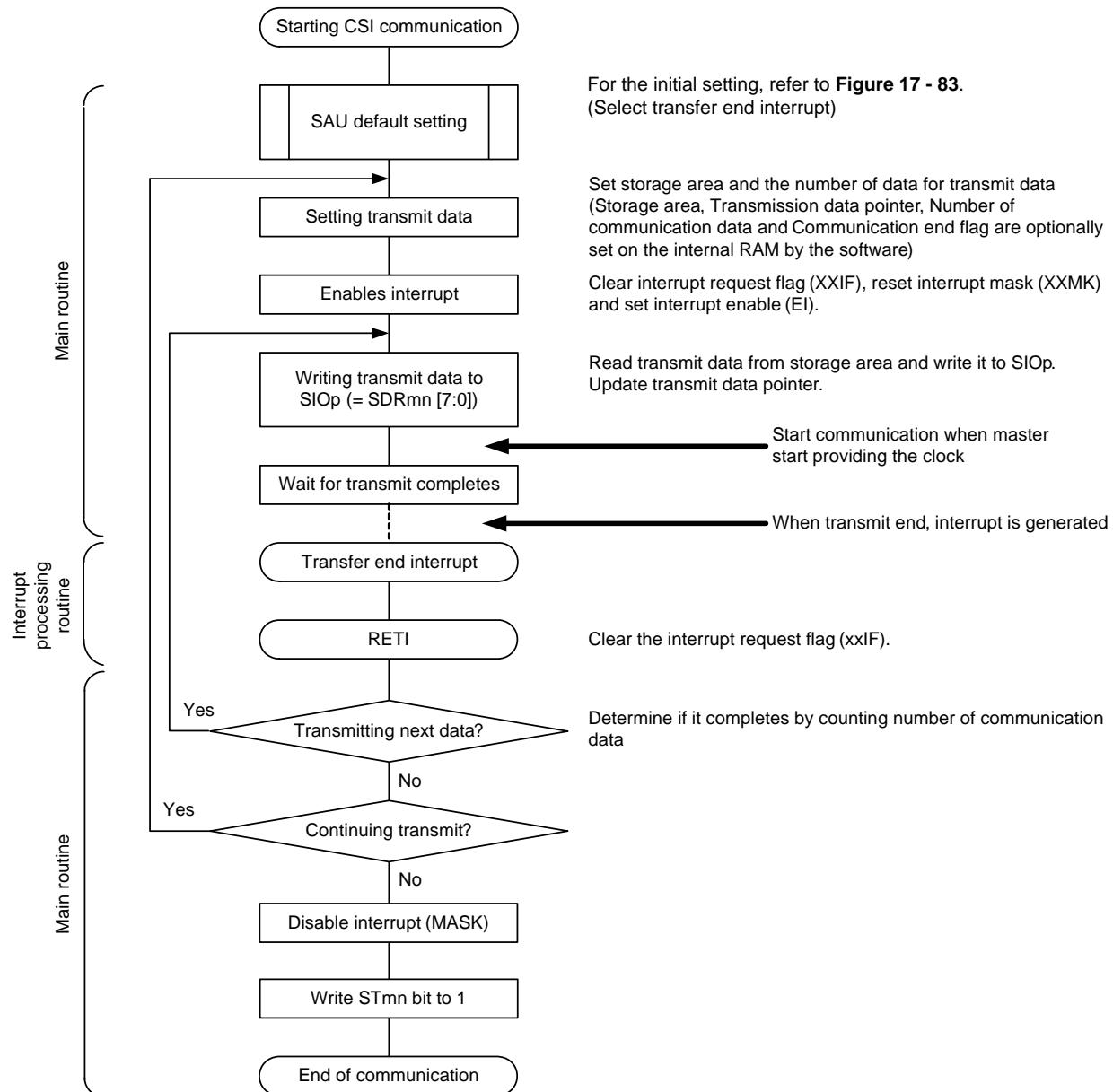
**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

## (3) Processing flow (in single-transmission mode)

**Figure 17 - 86 Timing Chart of Slave Transmission (in Single-Transmission Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



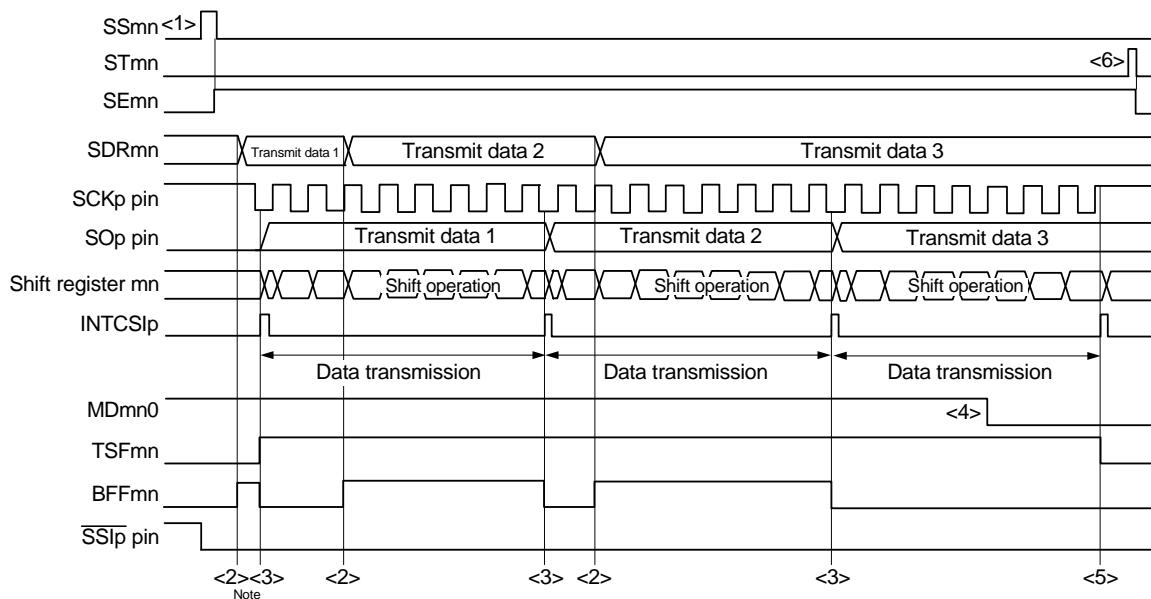
**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ ), p: CSI number ( $p = 00$ )

**Figure 17 - 87 Flowchart of Slave Transmission (in Single-Transmission Mode)**

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

## (4) Processing flow (in continuous transmission mode)

**Figure 17 - 88 Timing Chart of Slave Transmission (in Continuous Transmission Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**

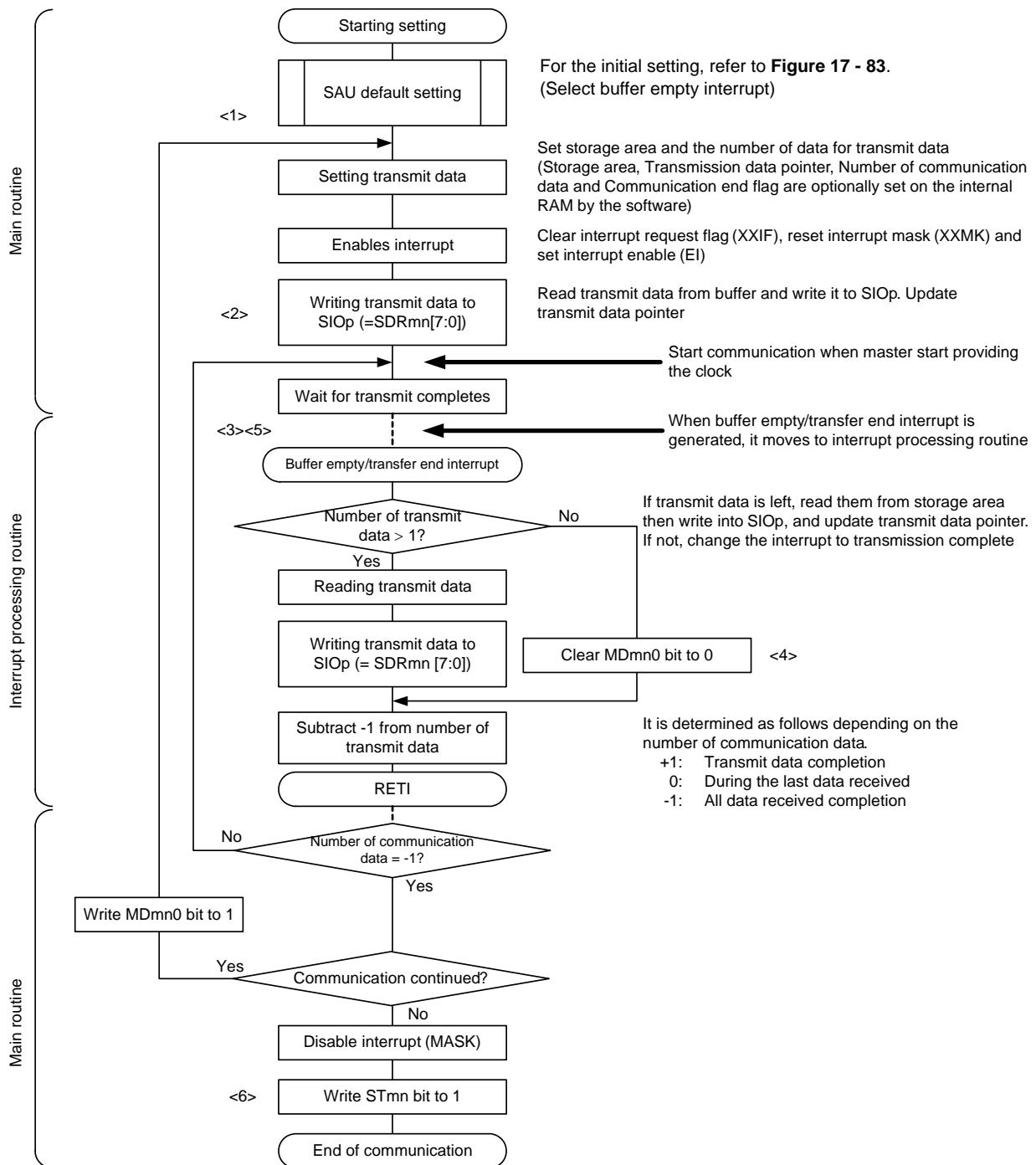


**Note** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation. However, rewrite it before transfer of the last bit is started.

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

Figure 17 - 89 Flowchart of Slave Transmission (in Continuous Transmission Mode)



**Remark 1.** <1> to <6> in the figure correspond to <1> to <6> in Figure 17 - 88 Timing Chart of Slave Transmission (in Continuous Transmission Mode) (Type 1: DAPmn = 0, CKPmn = 0).

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

## 17.6.2 Slave reception

Slave reception is that the RL78 microcontroller receives data from another device in the state of a transfer clock being input from another device.

Slave select input function	CSI00
Target channel	Channel 0 of SAU0
Pins used	SCK00, SI00, SSI00
Interrupt	INTCSI00
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)
Error detection flag	Overrun error detection flag (OVFmn) only
Transfer data length	7 or 8 bits
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2
Data phase	Selectable by the DAPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• DAPmn = 0: Data input starts from the start of the operation of the serial clock.</li> <li>• DAPmn = 1: Data input starts half a clock before the start of the serial clock operation.</li> </ul>
Clock phase	Selectable by the CKPmn bit of the SCRmn register <ul style="list-style-type: none"> <li>• CKPmn = 0: Non-reverse</li> <li>• CKPmn = 1: Reverse</li> </ul>
Data direction	MSB or LSB first
Slave select input function	Slave select input function operation selectable

**Note 1.** Because the external serial clock input to the SCK00 pin is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0)

## (1) Register setting

**Figure 17 - 90 Example of Contents of Registers for Slave Reception of Slave Select Input Function (CSI00) (1/2)**

## (a) Serial mode register mn (SMRmn)

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 1	0	0	0	0	0	STSmn 0	0	SI Smn0 0	1	0	0	MDmn2 0	MDmn1 0	MDmn0 0

Operation clock (fmck) of channel n  
0: Prescaler output clock CKm0 set by the SPSm register  
1: Prescaler output clock CKm1 set by the SPSm register

Interrupt source of channel n  
0: Transfer end interrupt

## (b) Serial communication operation setting register mn (SCRmn)

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 0	RXEmn 1	DAPmn 0/1	CKPmn 0/1	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0/1	SLCmn1 0	SLCmn0 0	0	1	DLSmn1 1	DLSmn0 0/1	

Selection of data transfer sequence  
0: Inputs/outputs data with MSB first  
1: Inputs/outputs data with LSB first

Setting of data length  
0: 7-bit data length  
1: 8-bit data length

## Registers Controlling Serial Array Unit

## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	00000000 Baud rate setting								0	Receive data						

SIOp

## (d) Serial output register m (SOm)... The Register that not used in this mode.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	0	0	0	0	x	CKOm3	CKOm2	CKOm1	CKOm0	0	0	0	0	SOm3	SOm2	SOm1	SOm0

**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Remark 2.** : Setting is fixed in the CSI slave reception mode,

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

**Figure 17 - 91 Example of Contents of Registers for Slave Reception of Slave Select Input Function (CSI00) (2/2)**

(f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 x	SSm2 x	SSm1 x	SSm0 0/1

(g) Input switch control register (ISC)...  $\overline{\text{SSI}00}$  input setting in CSI00 slave channel (channel 0 of unit 0).

ISC	7	6	5	4	3	2	1	0
	SSIE00 0/1	0	0	0	0	0	ISC1 0/1	ISC0 0/1

0: Disables the input value of the  $\overline{\text{SSI}00}$  pin  
1: Enables the input value of the  $\overline{\text{SSI}00}$  pin

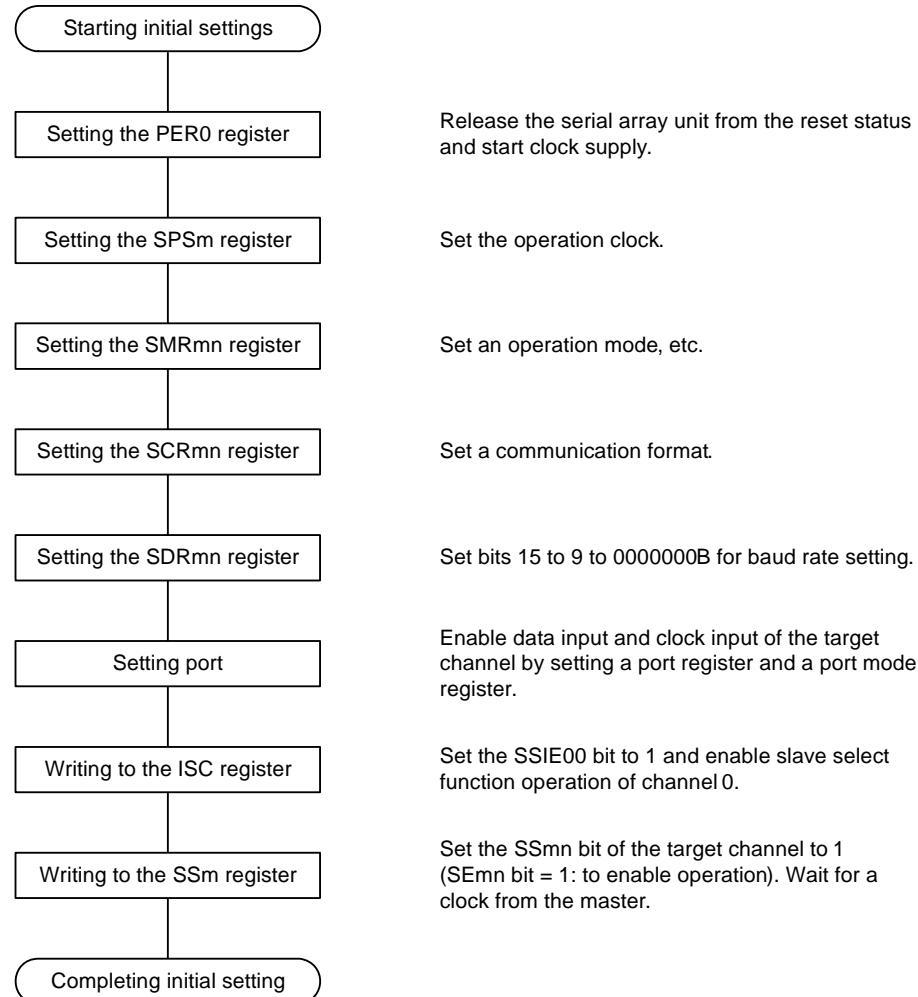
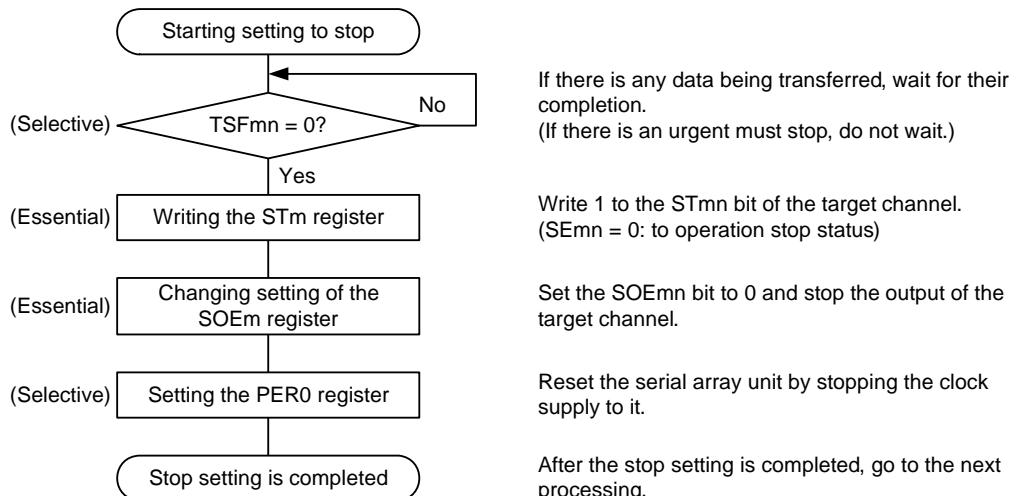
**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Remark 2.** Setting disabled (set to the initial value)

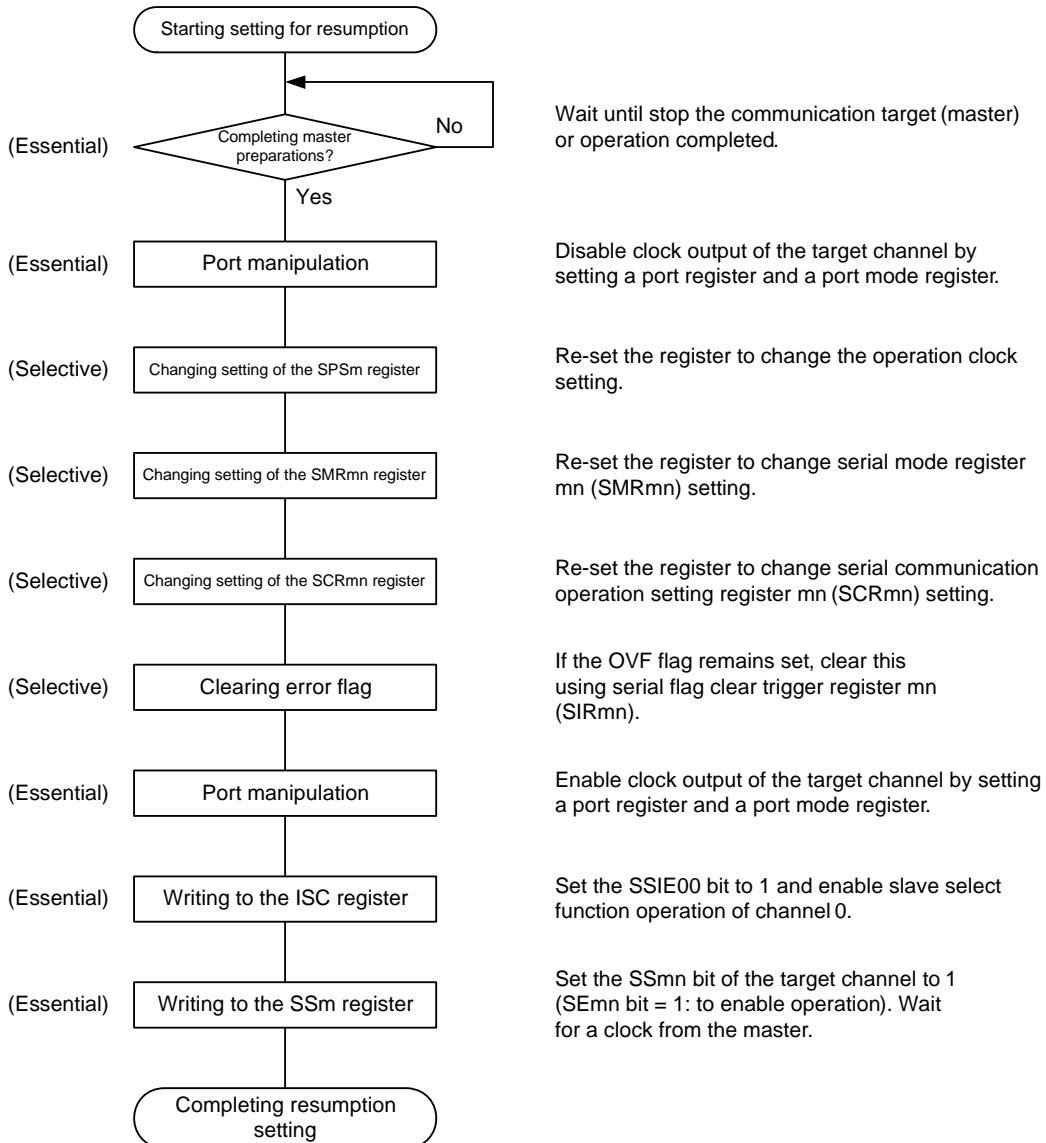
x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

**Figure 17 - 92 Initial Setting Procedure for Slave Reception****Figure 17 - 93 Procedure for Stopping Slave Reception**

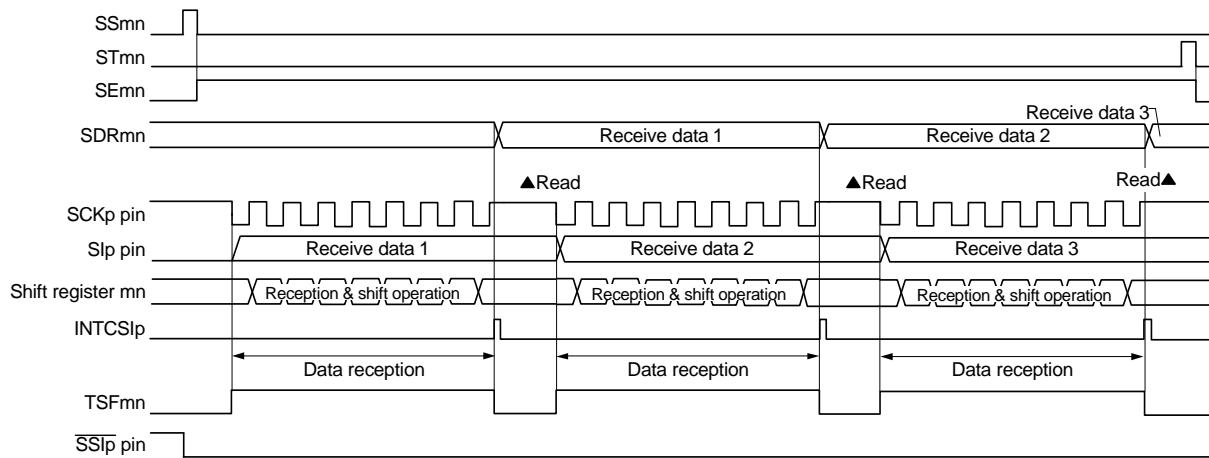
**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ ), p: CSI number ( $p = 00$ )

**Figure 17 - 94 Procedure for Resuming Slave Reception**

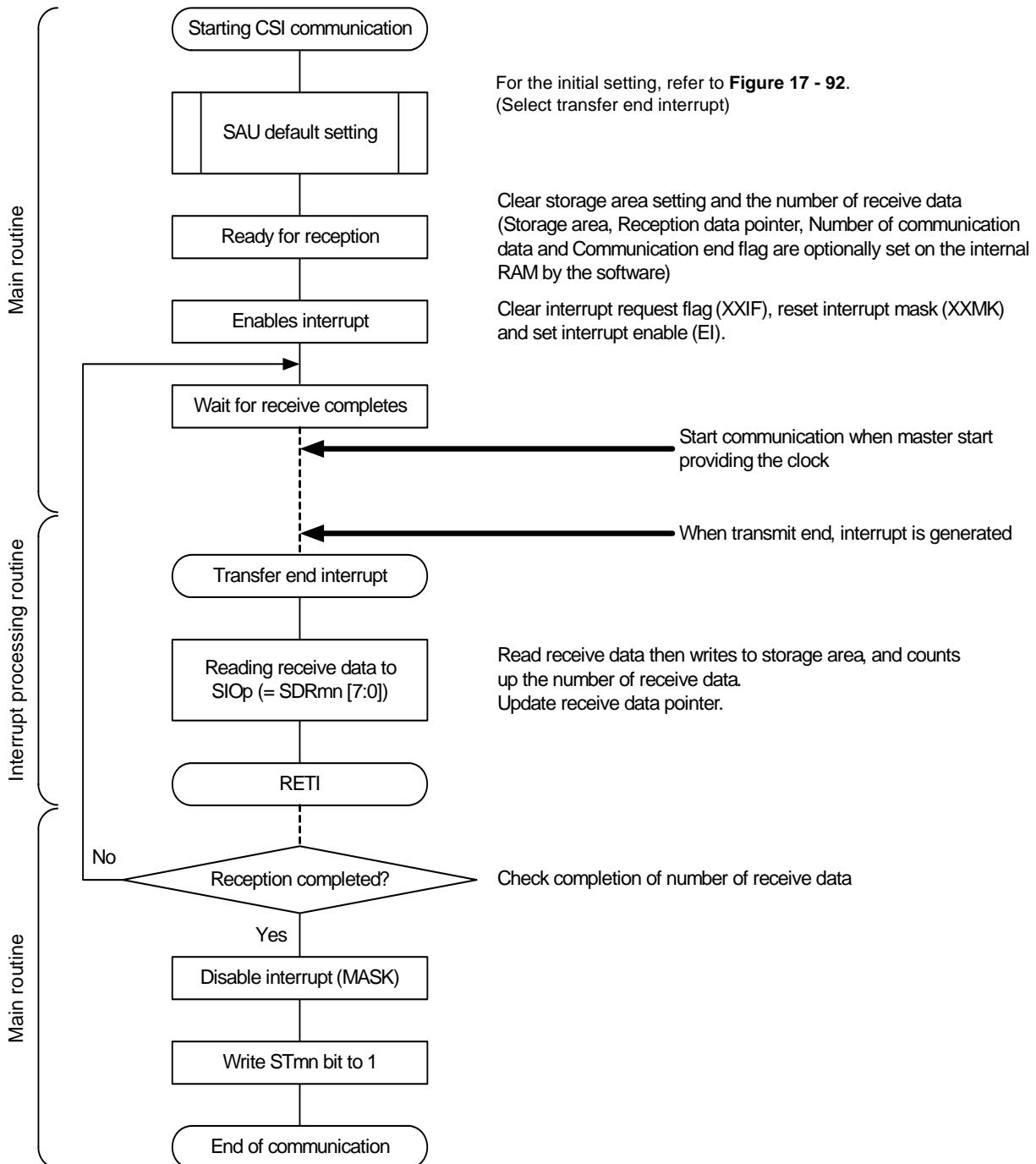
**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ ), p: CSI number ( $p = 00$ )

## (3) Processing flow (in single-reception mode)

**Figure 17 - 95 Timing Chart of Slave Reception (in Single-Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ ), p: CSI number ( $p = 00$ )

**Figure 17 - 96 Flowchart of Slave Reception (in Single-Reception Mode)**

### 17.6.3 Slave transmission/reception

Slave transmission/reception is that the RL78 microcontroller transmits/receives data to/from another device in the state of a transfer clock being input from another device.

Slave select input function	CSI00
Target channel	Channel 0 of SAU0
Pins used	SCK00, SI00, SO00, <u>SSI00</u>
Interrupt	INTCSI00
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.
Error detection flag	Overrun error detection flag (OVFmn) only
Transfer data length	7 or 8 bits
Transfer rate	Max. fmck/6 [Hz] Notes 1, 2
Data phase	Selectable by the DAPmn bit of the SCRmn register • DAPmn = 0: Data I/O starts from the start of the operation of the serial clock. • DAPmn = 1: Data I/O starts half a clock before the start of the serial clock operation.
Clock phase	Selectable by the CKPmn bit of the SCRmn register • CKPmn = 0: Non-reverse • CKPmn = 1: Reverse
Data direction	MSB or LSB first
Slave select input function	Slave select input function operation selectable

**Note 1.** Because the external serial clock input to the SCK00 pin is sampled internally and used, the fastest transfer rate is fmck/6 [Hz].

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

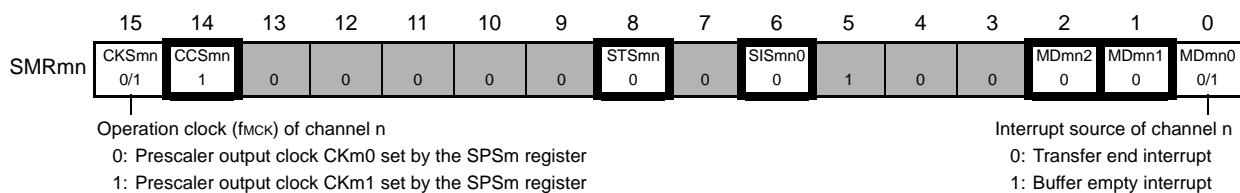
**Remark 1.** fmck: Operation clock frequency of target channel

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0)

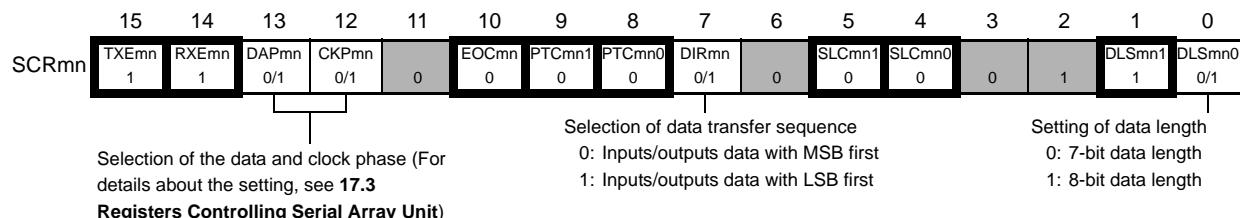
## (1) Register setting

**Figure 17 - 97 Example of Contents of Registers for Slave Transmission/Reception of Slave Select Input Function (CSI00) (1/2)**

## (a) Serial mode register mn (SMRmn)

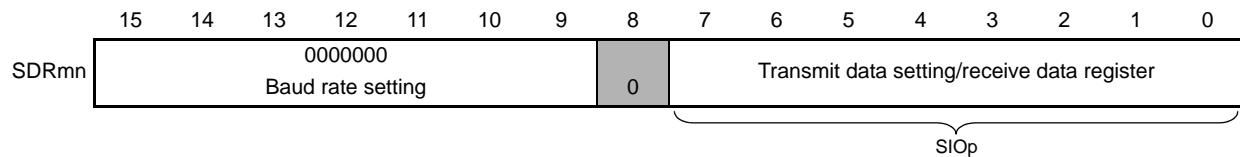


## (b) Serial communication operation setting register mn (SCRmn)

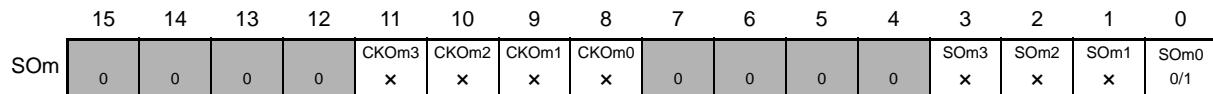


## Registers Controlling Serial Array Unit

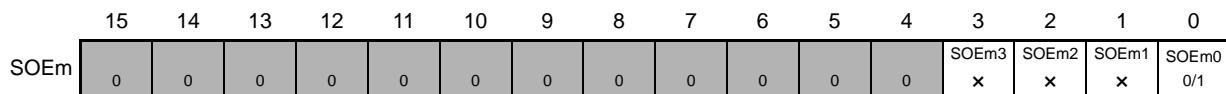
## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOp)



## (d) Serial output register m (SOm)... Sets only the bits of the target channel.



## (e) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.



**Caution Be sure to set transmit data to the SIOp register before the clock from the master is started.**

**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Remark 2.** : Setting is fixed in the CSI slave transmission/reception mode

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

**Figure 17 - 98 Example of Contents of Registers for Slave Transmission/Reception of Slave Select Input Function (CSI00) (2/2)**

(f) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	SSm3 x	SSm2 x	SSm1 x	SSm0 0/1

(g) Input switch control register (ISC)... SSI00 input setting in CSI00 slave channel (channel 0 of unit 0).

7	6	5	4	3	2	1	0
ISC	SSIE00 0/1	0	0	0	0	ISC1 0/1	ISC0 0/1

0: Disables the input value of the SSI00 pin  
1: Enables the input value of the SSI00 pin

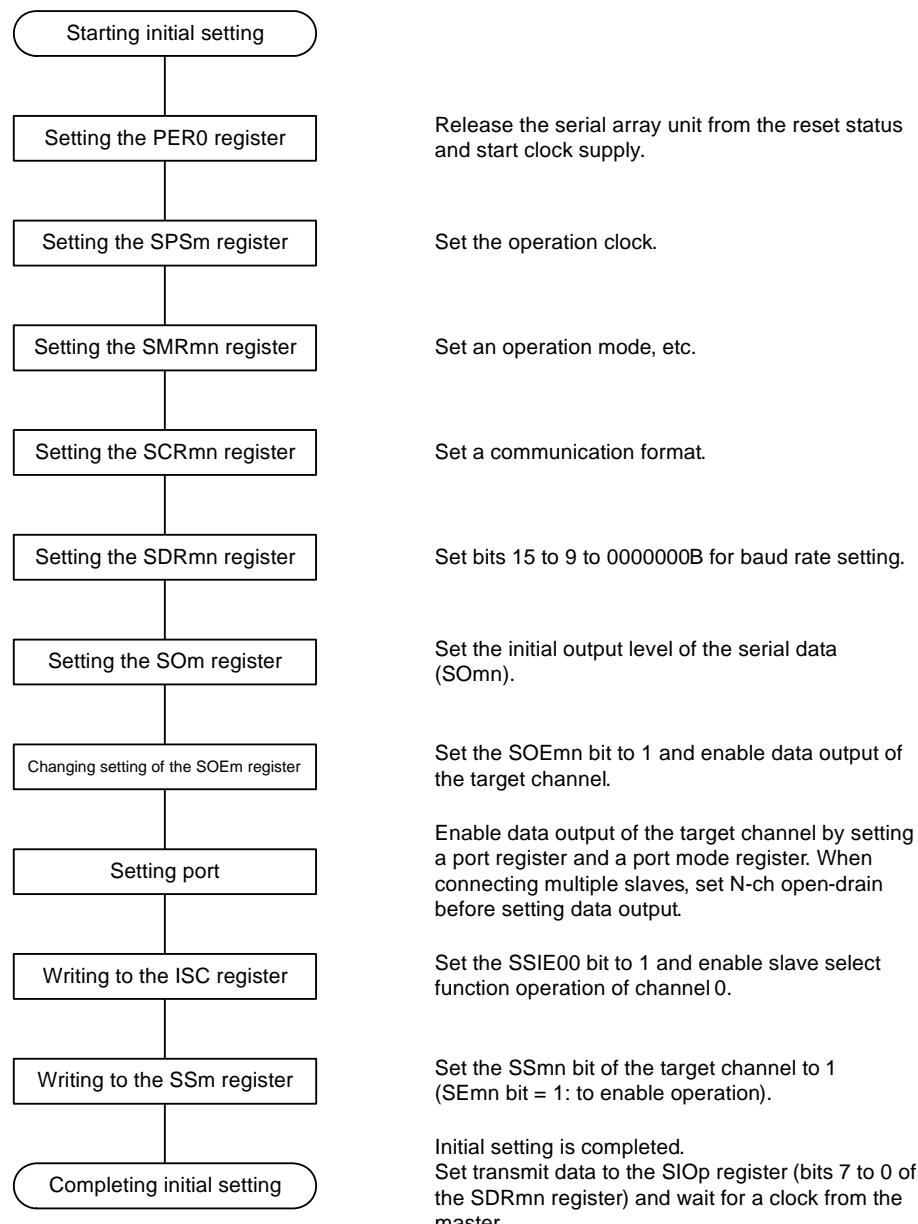
**Remark 1.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Remark 2.** Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

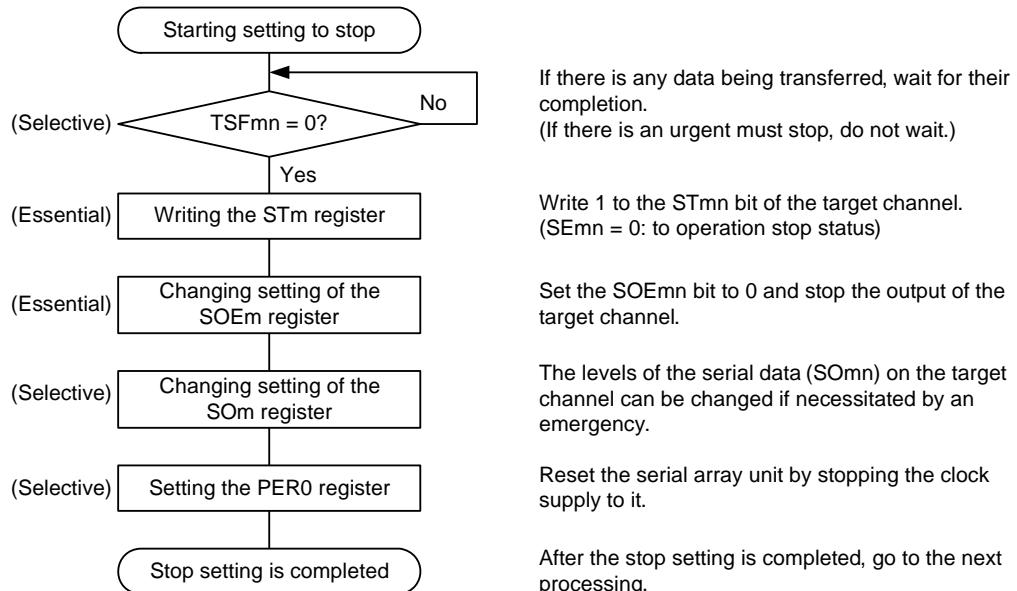
0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

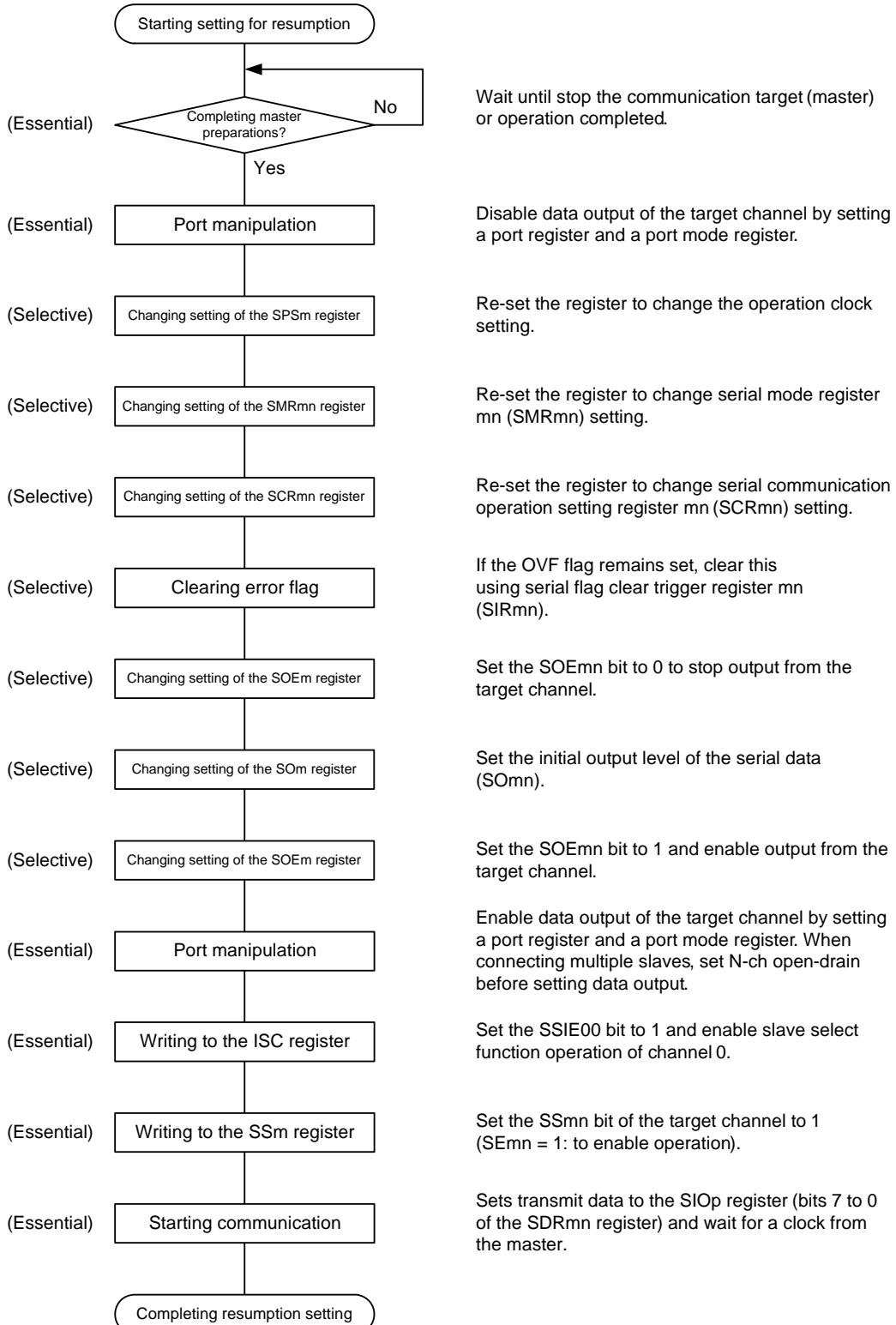
**Figure 17 - 99 Initial Setting Procedure for Slave Transmission/Reception**

**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Figure 17 - 100 Procedure for Stopping Slave Transmission/Reception**

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

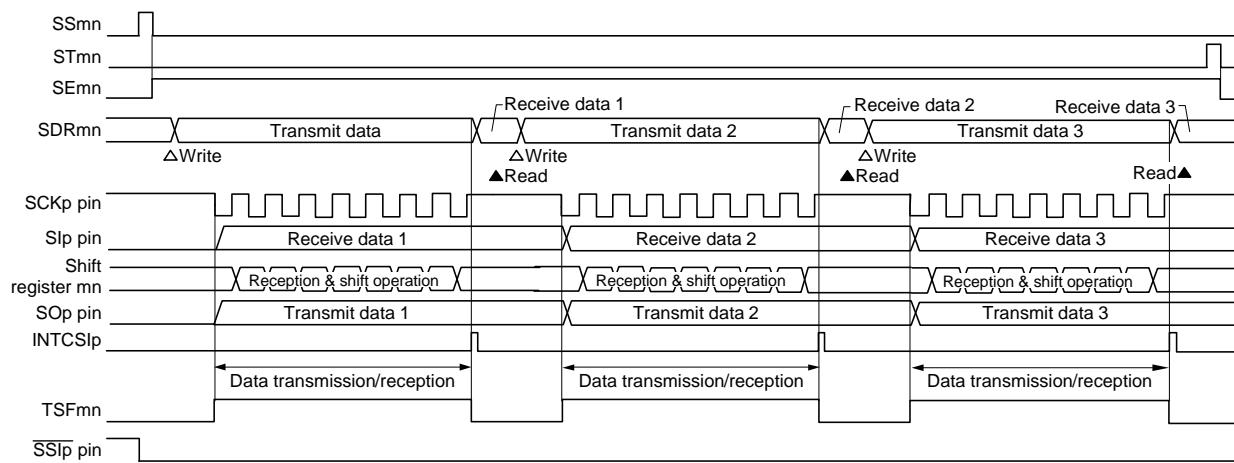
**Figure 17 - 101 Procedure for Resuming Slave Transmission/Reception**

**Caution 1.** Be sure to set transmit data to the SIOp register before the clock from the master is started.

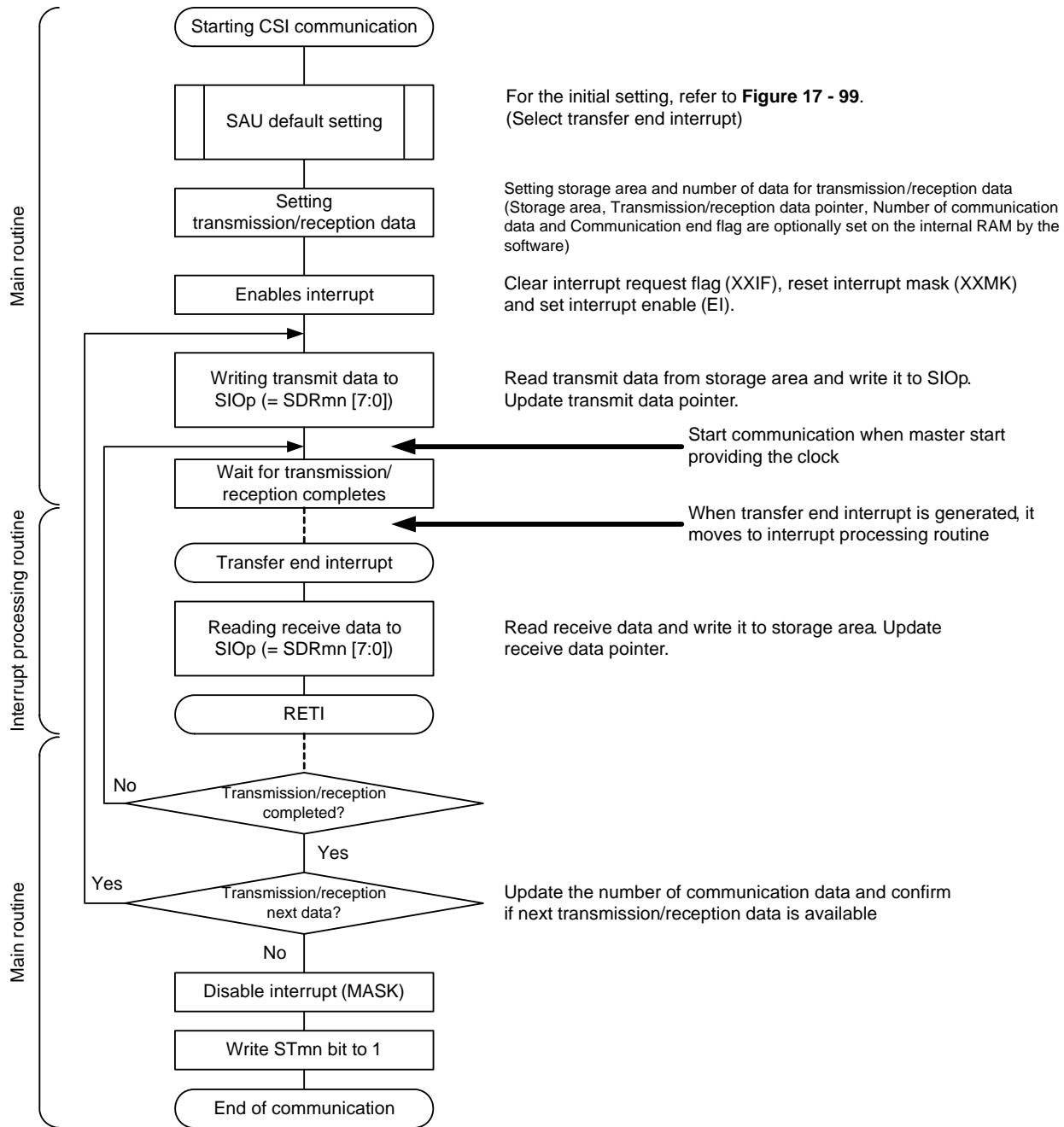
**Caution 2.** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target (master) stops or transmission finishes, and then perform initialization instead of restarting the transmission.

## (3) Processing flow (in single-transmission/reception mode)

**Figure 17 - 102 Timing Chart of Slave Transmission/Reception (in Single-Transmission/Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Remark** m: Unit number ( $m = 0$ ), n: Channel number ( $n = 0$ ), p: CSI number ( $p = 00$ )

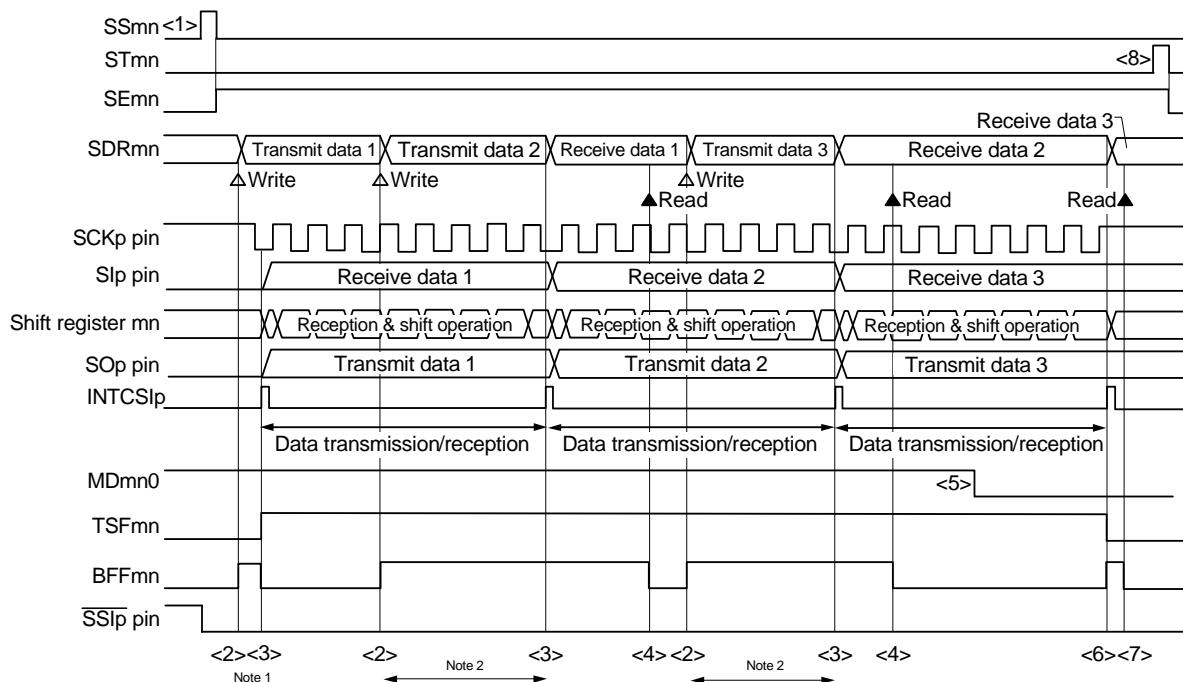
**Figure 17 - 103 Flowchart of Slave Transmission/Reception (in Single- Transmission/Reception Mode)**

**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

## (4) Processing flow (in continuous transmission/reception mode)

**Figure 17 - 104 Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)**  
**(Type 1: DAPmn = 0, CKPmn = 0)**



**Note 1.** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

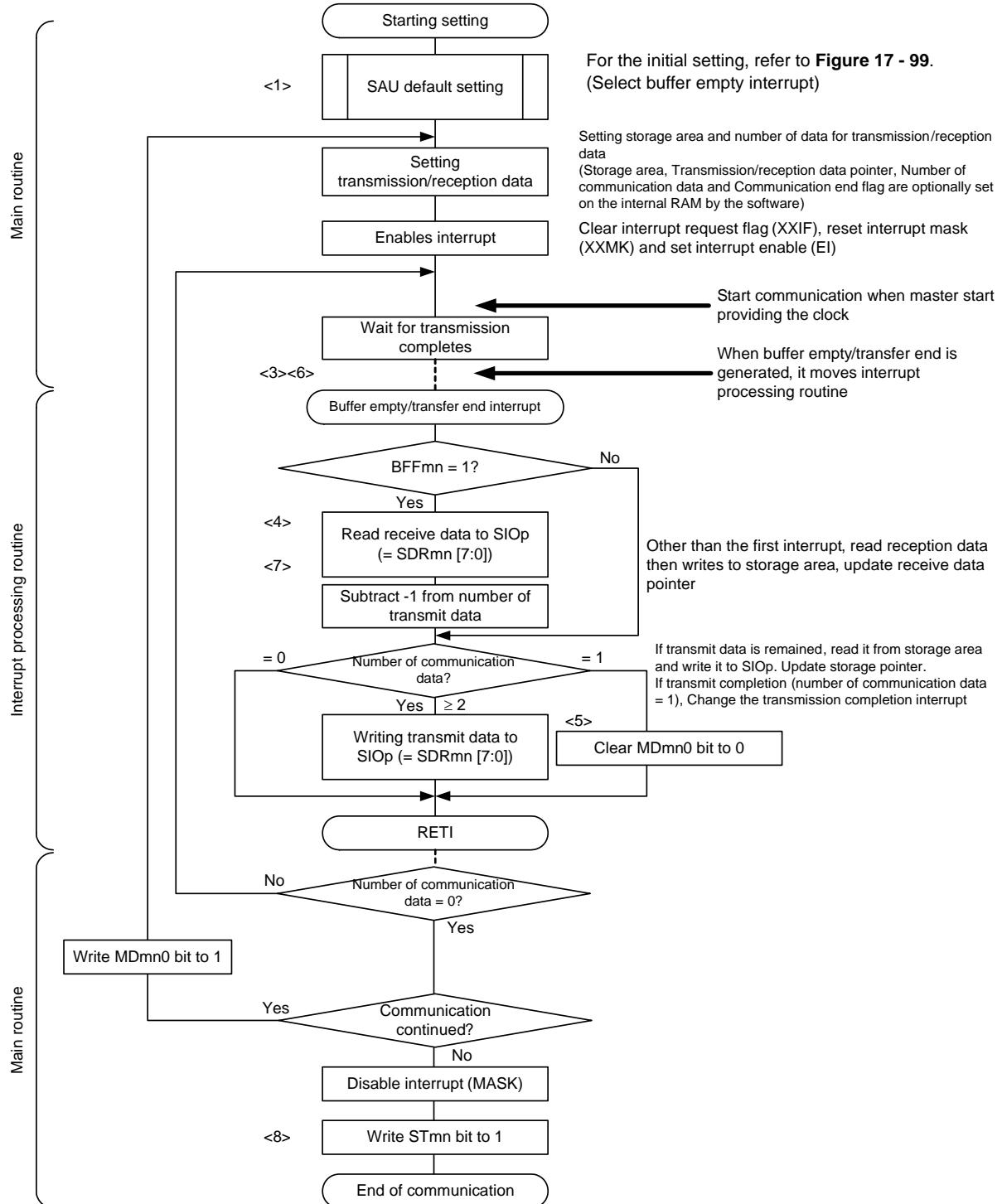
**Note 2.** The transmit data can be read by reading the SDRmn register during this period. At this time, the transfer operation is not affected.

**Caution** The MDmn0 bit of serial mode register mn (SMRmn) can be rewritten even during operation.  
 However, rewrite it before transfer of the last bit is started, so that it has been rewritten before the transfer end interrupt of the last transmit data.

**Remark 1.** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 105 Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode).

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

Figure 17 - 105 Flowchart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode)



**Caution** Be sure to set transmit data to the SIOp register before the clock from the master is started.

**Remark 1.** <1> to <8> in the figure correspond to <1> to <8> in Figure 17 - 104 Timing Chart of Slave Transmission/Reception (in Continuous Transmission/Reception Mode) (Type 1: DAPmn = 0, CKPmn = 0).

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

### 17.6.4 Calculating transfer clock frequency

The transfer clock frequency for slave select input function (CSI00) communication can be calculated by the following expressions.

(1) Slave

(Transfer clock frequency) = {Frequency of serial clock (SCK) supplied by master} Note [Hz]

**Note** The permissible maximum transfer clock frequency is  $f_{MCK}/6$ .

**Remark** m: Unit number (m = 0), n: Channel number (n = 0), p: CSI number (p = 00)

**Table 17 - 3 Selection of Operation Clock For Slave Select Input Function**

SMR <sub>m</sub> Register	SPSm Register								Operation Clock (f <sub>MCK</sub> ) <sup>Note</sup>	f <sub>CLK</sub> = 32 MHz
	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00		
0	x	x	x	x	0	0	0	0	f <sub>CLK</sub>	32 MHz
	x	x	x	x	0	0	0	1	f <sub>CLK</sub> /2	16 MHz
	x	x	x	x	0	0	1	0	f <sub>CLK</sub> /2 <sup>2</sup>	8 MHz
	x	x	x	x	0	0	1	1	f <sub>CLK</sub> /2 <sup>3</sup>	4 MHz
	x	x	x	x	0	1	0	0	f <sub>CLK</sub> /2 <sup>4</sup>	2 MHz
	x	x	x	x	0	1	0	1	f <sub>CLK</sub> /2 <sup>5</sup>	1 MHz
	x	x	x	x	0	1	1	0	f <sub>CLK</sub> /2 <sup>6</sup>	500 kHz
	x	x	x	x	0	1	1	1	f <sub>CLK</sub> /2 <sup>7</sup>	250 kHz
	x	x	x	x	1	0	0	0	f <sub>CLK</sub> /2 <sup>8</sup>	125 kHz
	x	x	x	x	1	0	0	1	f <sub>CLK</sub> /2 <sup>9</sup>	62.5 kHz
	x	x	x	x	1	0	1	0	f <sub>CLK</sub> /2 <sup>10</sup>	31.25 kHz
	x	x	x	x	1	0	1	1	f <sub>CLK</sub> /2 <sup>11</sup>	15.63 kHz
	x	x	x	x	1	1	0	0	f <sub>CLK</sub> /2 <sup>12</sup>	7.81 kHz
	x	x	x	x	1	1	0	1	f <sub>CLK</sub> /2 <sup>13</sup>	3.91 kHz
	x	x	x	x	1	1	1	0	f <sub>CLK</sub> /2 <sup>14</sup>	1.95 kHz
	x	x	x	x	1	1	1	1	f <sub>CLK</sub> /2 <sup>15</sup>	977 Hz

**Note** When changing the clock selected for f<sub>CLK</sub> (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (ST<sub>m</sub>) = 000FH) the operation of the serial array unit (SAU).

**Remark 1.** x: Don't care

**Remark 2.** m: Unit number (m = 0), n: Channel number (n = 0)

### 17.6.5 Procedure for processing errors that occurred during slave select input function communication

The procedure for processing errors that occurred during slave select input function communication is described in Figure 17 - 106.

**Figure 17 - 106 Processing Procedure in Case of Overrun Error**

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn). →	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes 1 to serial flag clear trigger register mn (SIRmn). →	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.

**Remark** m: Unit number (m = 0), n: Channel number (n = 0)

## 17.7 Operation of UART (UART0 to UART3) Communication

This is a start-stop synchronization function using two lines: serial data transmission (TxD) and serial data reception (RxD) lines. By using these two communication lines, each data frame, which consist of a start bit, data, parity bit, and stop bit, is transferred asynchronously (using the internal baud rate) between the microcontroller and the other communication party. Full-duplex UART communication can be performed by using a channel dedicated to transmission (even-numbered channel) and a channel dedicated to reception (odd-numbered channel). The LIN-bus can be implemented by using UART0, timer array unit 0 (channel 3), and an external interrupt (INTP0).

[Data transmission/reception]

- Data length of 7, 8, or 9 bits Note
- Select the MSB/LSB first
- Level setting of transmit/receive data (selecting whether to reverse the level)
- Parity bit appending and parity check functions
- Stop bit appending, stop bit check function

[Interrupt function]

- Transfer end interrupt/buffer empty interrupt
- Error interrupt in case of framing error, parity error, or overrun error

[Error detection flag]

- Framing error, parity error, or overrun error

In addition, UARTs of following channels supports the SNOOZE mode. When RxD input is detected while in the STOP mode, the SNOOZE mode makes data reception that does not require the CPU possible. Only the following UARTs can be specified when FRQSEL4 in the option byte (000C2H) = 0 in the SNOOZE mode.

- 30 to 64-pin products:   UART0
- 80, 100-pin products:   UART0 and UART2

The LIN-bus is accepted in UART0 (channels 0 and 1 of unit 0).

[LIN-bus functions]

- Wakeup signal detection
- Break field (BF) detection
- Sync field measurement, baud rate calculation

} Using the external interrupt (INTP0) and  
timer array unit 0 (channel 3)

**Note**      Only following UARTs can be specified for the 9-bit data length.

- 30 to 64-pin products:   UART0
- 80, 100-pin products:   UART0 and UART2

- UART0 uses channels 0 and 1 of SAU0.
- UART1 uses channels 2 and 3 of SAU0.
- UART2 uses channels 0 and 1 of SAU1.
- UART3 uses channels 2 and 3 of SAU1.

- 30, 32-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	—		—

- 36, 40, 44-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	—		—
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 48, 52-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	—		—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 64-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus) UART1	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 80, 100-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input function)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	CSI10	UART1	IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21
	2	CSI30	UART3	IIC30
	3	CSI31		IIC31

Select any function for each channel. Only the selected function is possible. If UART0 is selected for channels 0 and 1 of unit 0, for example, these channels cannot be used for CSI00 and IIC00.

At this time, however, channel 0, 1, or other channels of the same unit can be used for a function other than UART1, such as CSI00, UART0, and IIC00.

**Caution** When using a serial array unit for UART, both the transmitter side (even-numbered channel) and the receiver side (odd-numbered channel) can only be used for UART.

UART performs the following four types of communication operations.

- UART transmission (See 17.7.1.)
- UART reception (See 17.7.2.)
- LIN transmission (UART0 only) (See 17.8.1.)
- LIN reception (UART0 only) (See 17.8.2.)

### 17.7.1 UART transmission

UART transmission is an operation to transmit data from the RL78 microcontroller to another device asynchronously (start-stop synchronization).

Of two channels used for UART, the even channel is used for UART transmission.

UART	UART0	UART1	UART2	UART3
Target channel	Channel 0 of SAU0	Channel 2 of SAU0	Channel 0 of SAU1	Channel 2 of SAU1
Pins used	TxD0	TxD1	TxD2	TxD3
Interrupt	INTST0	INTST1	INTST2	INTST3
Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.				
Error detection flag	None			
Transfer data length	7, 8, or 9 bits Note 1			
Transfer rate	Max. fmck/6 [bps] (SDRmn [15:9] = 2 or more), Min. fclk/(2 × 2 <sup>15</sup> × 128) [bps] Note 2			
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)			
Parity bit	The following selectable <ul style="list-style-type: none"><li>• No parity bit</li><li>• Appending 0 parity</li><li>• Appending even parity</li><li>• Appending odd parity</li></ul>			
Stop bit	The following selectable <ul style="list-style-type: none"><li>• Appending 1 bit</li><li>• Appending 2 bits</li></ul>			
Data direction	MSB or LSB first			

**Note 1.** Only following UARTs can be specified for the 9-bit data length.

- 30 to 64-pin products:   UART0
- 80, 100-pin products:   UART0 and UART2

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

fclk: System clock frequency

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), mn = 00, 02, 10, 12

## (1) Register setting

**Figure 17 - 107 Example of Contents of Registers for UART Transmission of UART (UART0 to UART3) (1/2)**

## (a) Serial mode register mn (SMRmn)

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 0	0	0	0	0	0	0	0	0	1	0	0	MDmn2 0	MDmn1 1	MDmn0 0/1

Operation clock (fmck) of channel n  
0: Prescaler output clock CKm0 set by the SPSm register  
1: Prescaler output clock CKm1 set by the SPSm register

Interrupt source of channel n  
0: Transfer end interrupt  
1: Buffer empty interrupt

## (b) Serial communication operation setting register mn (SCRmn)

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 1	RXEmn 0	DAPmn 0	CKPmn 0	EOCmn 0	PTCmn1 0/1	PTCmn0 0/1	DIRmn 0/1		SLCmn1 0/1	SLCmn0 0/1	0	1	DLSmn1 0/1 Note 1	DLSmn0 0/1	

Setting of parity bit  
00B: No parity  
01B: Appending 0 parity  
10B: Appending Even parity  
11B: Appending Odd parity

Selection of data transfer sequence  
0: Inputs/outputs data with MSB first  
1: Inputs/outputs data with LSB first

Setting of stop bit  
01B: Appending 1 bit  
10B: Appending 2 bits

## (c) Serial data register mn (SDRmn) (lower 8 bits: TXDq)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	Baud rate setting								0 Note 2	Transmit data setting							

TXDq

## (d) Serial output level register m (SOLm)... Sets only the bits of the target channel.

SOLm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	SOLm2 0/1	0	SOLm0 0/1

0: Non-reverse (normal) transmission  
1: Reverse transmission

**Note 1.** Only provided for the SCR00 register and the SCR10 register of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Note 2.** When performing 9-bit communication, bits 0 to 8 of the SDRm0 register are used to specify the transmission data.

- 30 to 64-pin products: UART0
- 80, 100-pin products: UART0 and UART2

**Remark 1.** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ ), q: UART number ( $q = 0$  to  $3$ ),  
 $mn = 00, 02, 10, 12$

**Remark 2.** : Setting is fixed in the UART transmission mode,

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

**Figure 17 - 108 Example of Contents of Registers for UART Transmission of UART  
(UART0 to UART3) (2/2)**

(e) Serial output register m (SOm)... Sets only the bits of the target channel.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 x	CKOm2 x	CKOm1 x	CKOm0 x	0	0	0	0	SOm3 x	SOm2 0/1 Note	SOm1 x	SOm0 0/1 Note

0: Serial data output value is "0"  
1: Serial data output value is "1"

(f) Serial output enable register m (SOEm)... Sets only the bits of the target channel to 1.

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 x	SOEm2 0/1	SOEm1 x	SOEm0 0/1

(g) Serial channel start register m (SSm)... Sets only the bits of the target channel to 1.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 x	SSm2 0/1	SSm1 x	SSm0 0/1

**Note** Before transmission is started, be sure to set to 1 when the SOLmn bit of the target channel is set to 0, and set to 0 when the SOLmn bit of the target channel is set to 1. The value varies depending on the communication data during communication operation.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), q: UART number (q = 0 to 3)

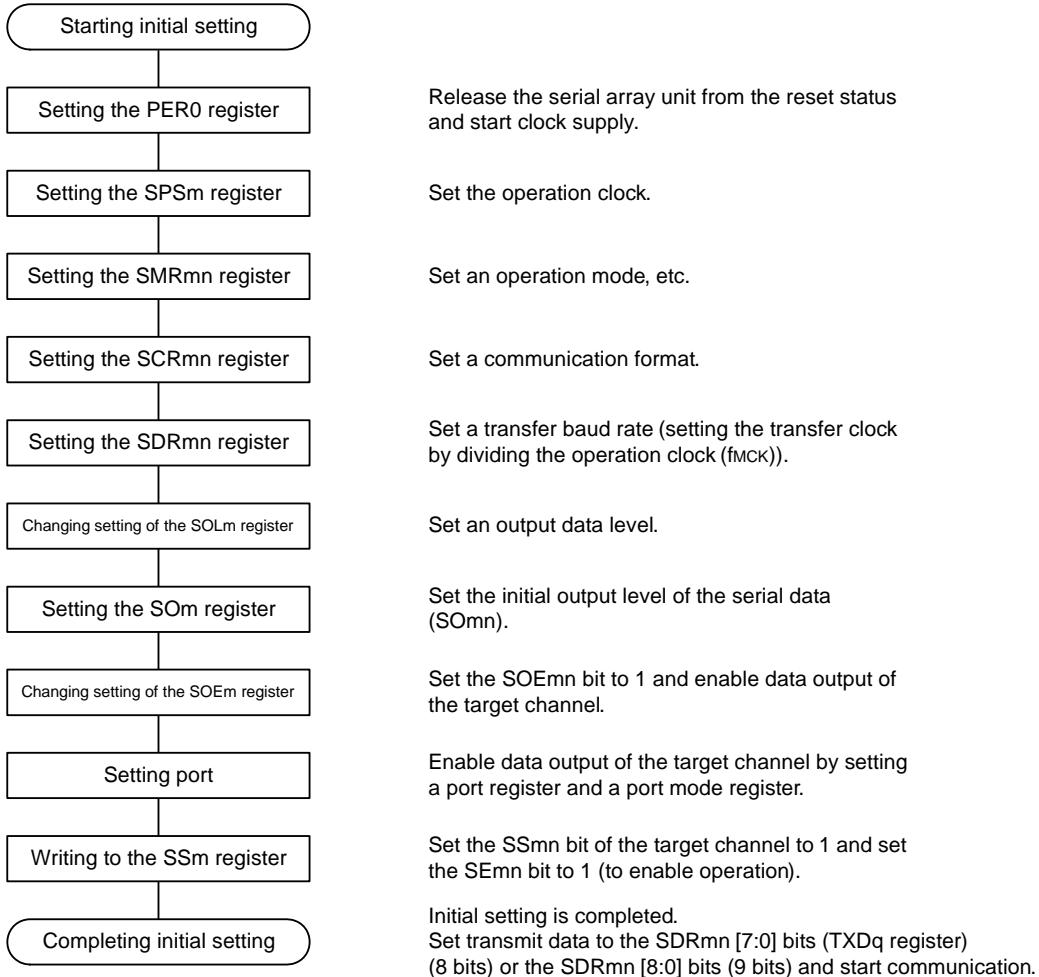
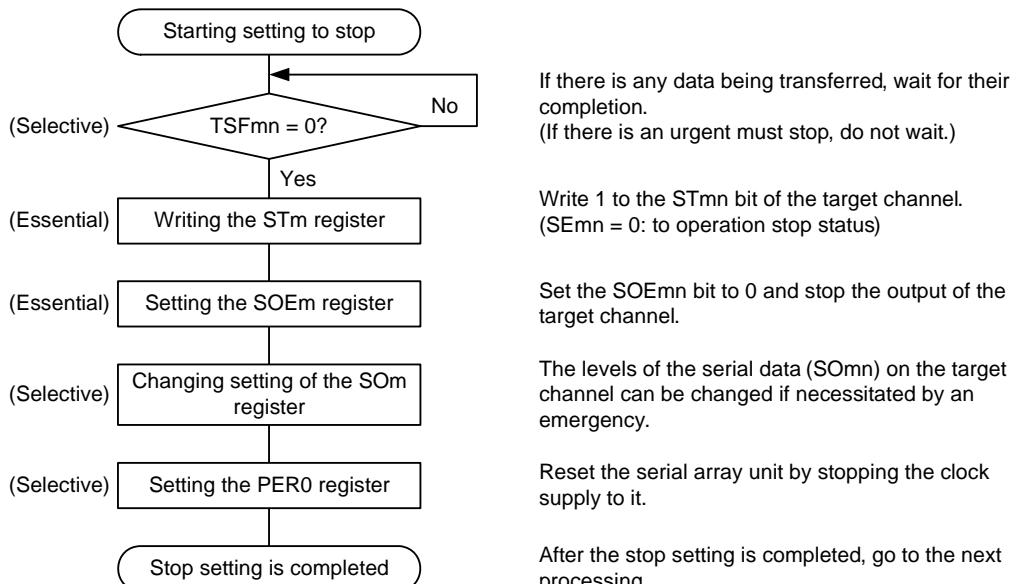
mn = 00, 02, 10, 12

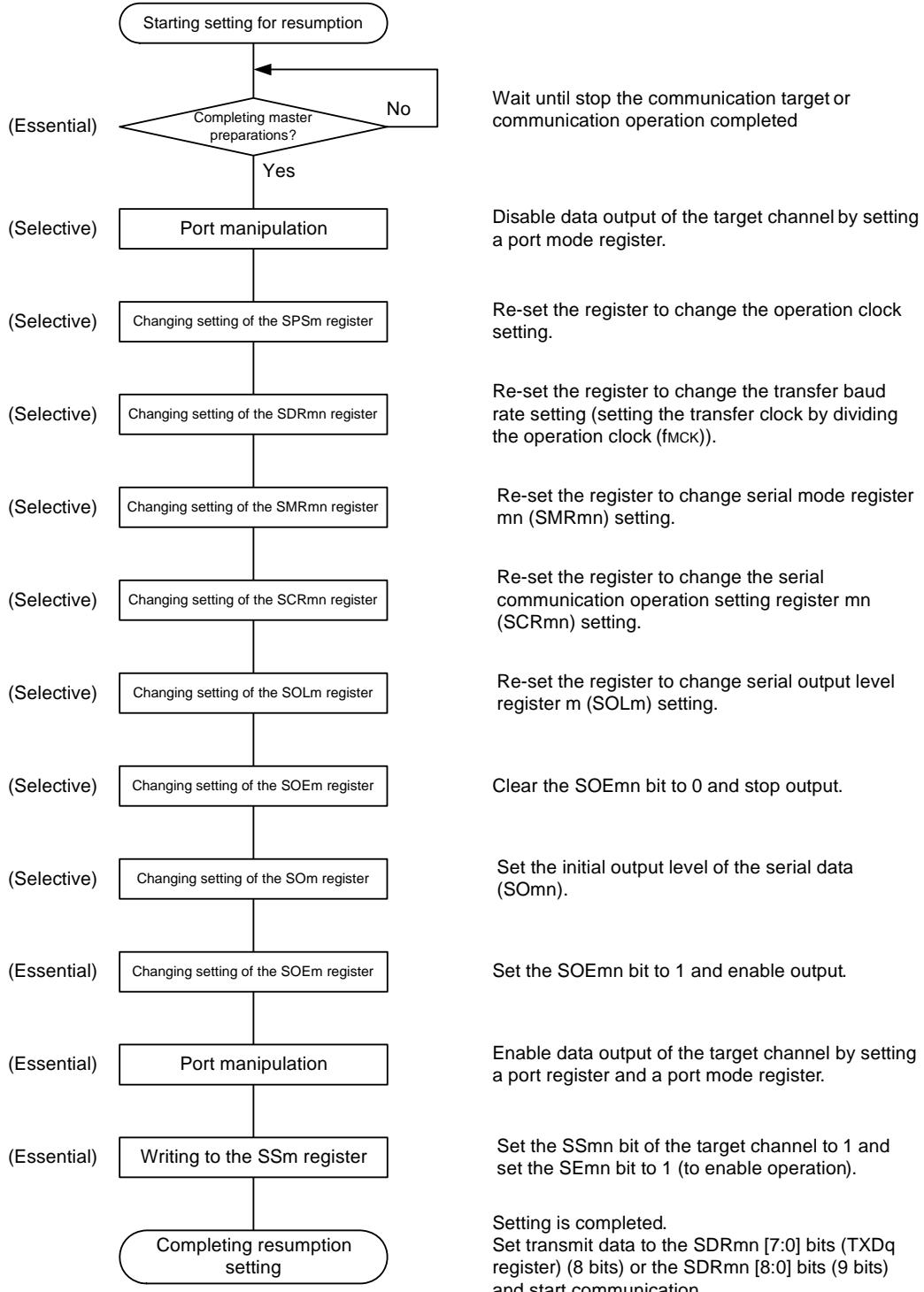
**Remark 2.** : Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

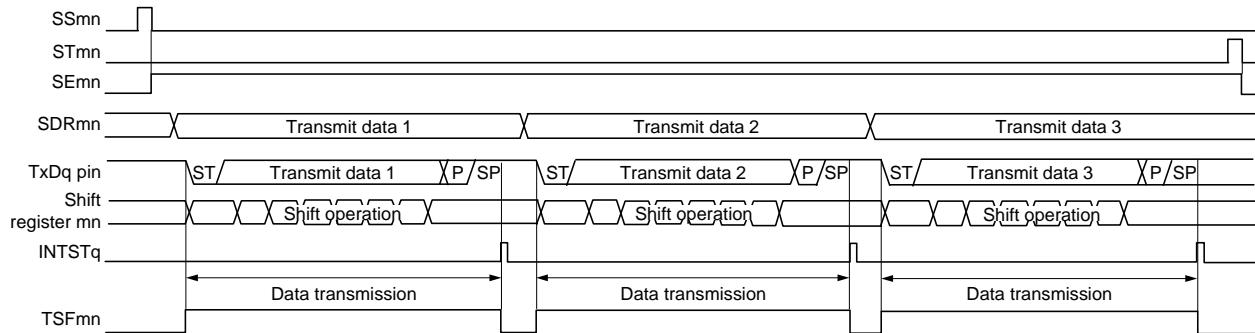
## (2) Operation procedure

**Figure 17 - 109 Initial Setting Procedure for UART Transmission****Figure 17 - 110 Procedure for Stopping UART Transmission**

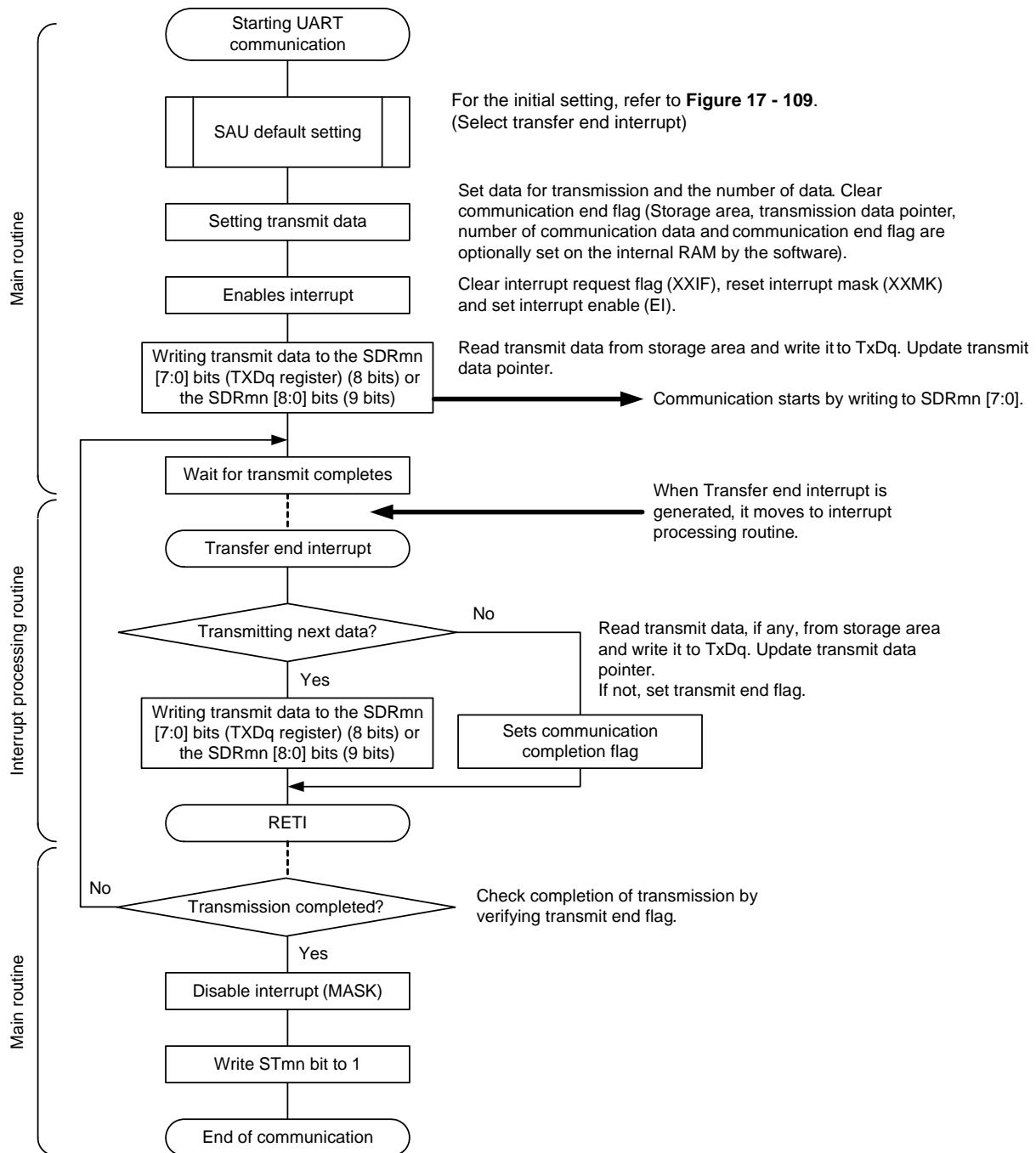
**Figure 17 - 111 Procedure for Resuming UART Transmission**

**Remark** If PER0 is rewritten while stopping the master transmission and the clock supply is stopped, wait until the transmission target stops or transmission finishes, and then perform initialization instead of restarting the transmission.

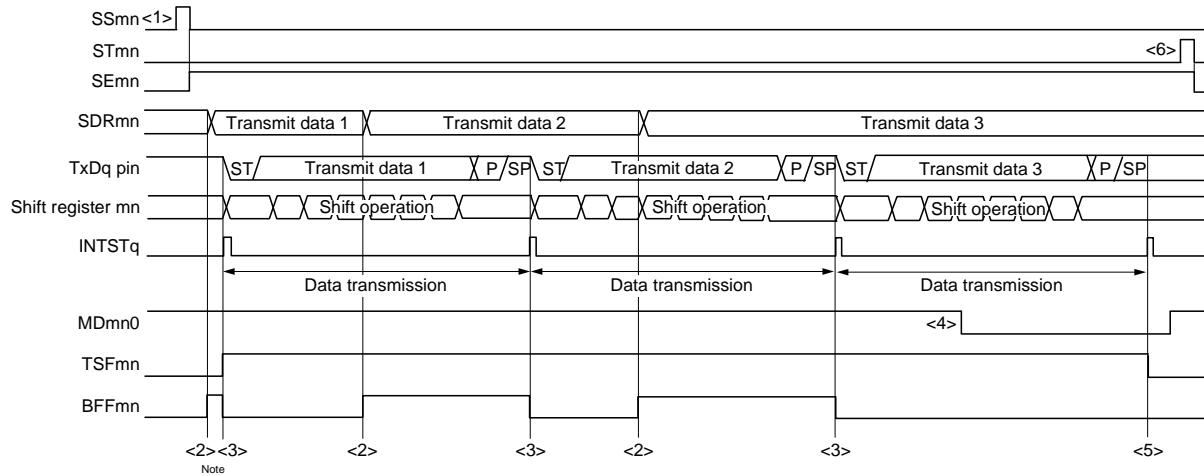
## (3) Processing flow (in single-transmission mode)

**Figure 17 - 112 Timing Chart of UART Transmission (in Single-Transmission Mode)**

**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0, 2$ ), q: UART number ( $q = 0$  to  $3$ )  
 $mn = 00, 02, 10, 12$

**Figure 17 - 113 Flowchart of UART Transmission (in Single-Transmission Mode)**

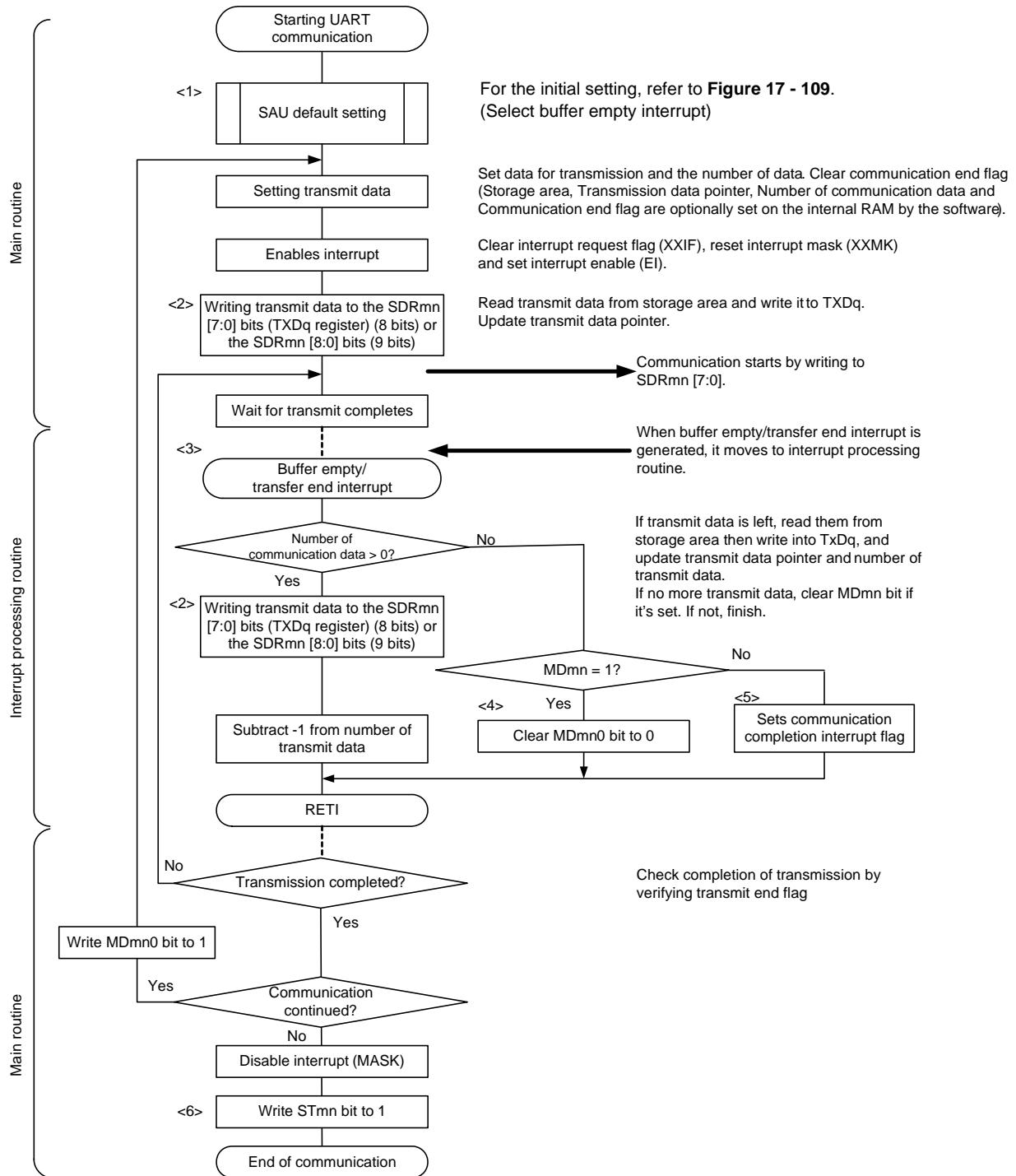
## (4) Processing flow (in continuous transmission mode)

**Figure 17 - 114 Timing Chart of UART Transmission (in Continuous Transmission Mode)**

**Note** If transmit data is written to the SDRmn register while the BFFmn bit of serial status register mn (SSRmn) is 1 (valid data is stored in serial data register mn (SDRmn)), the transmit data is overwritten.

**Caution** The MDmn0 bit of serial mode register mn (SSRmn) can be rewritten even during operation. However, rewrite it before transfer of the last bit is started, so that it will be rewritten before the transfer end interrupt of the last transmit data.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), q: UART number (q = 0 to 3)  
 mn = 00, 02, 10, 12

**Figure 17 - 115 Flowchart of UART Transmission (in Continuous Transmission Mode)**

**Remark** <1> to <6> in the figure correspond to <1> to <6> in **Figure 17 - 114 Timing Chart of UART Transmission (in Continuous Transmission Mode).**

## 17.7.2 UART reception

UART reception is an operation wherein the RL78 microcontroller asynchronously receives data from another device (start-stop synchronization).

For UART reception, the odd-number channel of the two channels used for UART is used. The SMR register of both the odd- and even-numbered channels must be set.

UART	UART0	UART1	UART2	UART3
Target channel	Channel 1 of SAU0	Channel 3 of SAU0	Channel 1 of SAU1	Channel 3 of SAU1
Pins used	RxD0	RxD1	RxD2	RxD3
Interrupt	INTSR0	INTSR1	INTSR2	INTSR3
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)			
Error interrupt	INTSRE0	INTSRE1	INTSRE2	INTSRE3
Error detection flag	<ul style="list-style-type: none"> <li>• Framing error detection flag (FEFmn)</li> <li>• Parity error detection flag (PEFmn)</li> <li>• Overrun error detection flag (OVFmn)</li> </ul>			
Transfer data length	7, 8 or 9 bits Note 1			
Transfer rate Note 2	Max. fmck/6 [bps] (SDRmn [15:9] = 2 or more), Min. fclk/(2 × 2 <sup>15</sup> × 128) [bps]			
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)			
Parity bit	The following selectable <ul style="list-style-type: none"> <li>• No parity bit (no parity check)</li> <li>• Appending 0 parity (no parity check)</li> <li>• Appending even parity</li> <li>• Appending odd parity</li> </ul>			
Stop bit	Appending 1 bit			
Data direction	MSB or LSB first			

**Note 1.** Only following UARTs can be specified for the 9-bit data length.

- 30 to 64-pin products:   UART0
- 80, 100-pin products:   UART0 and UART2

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark 1.** fmck: Operation clock frequency of target channel

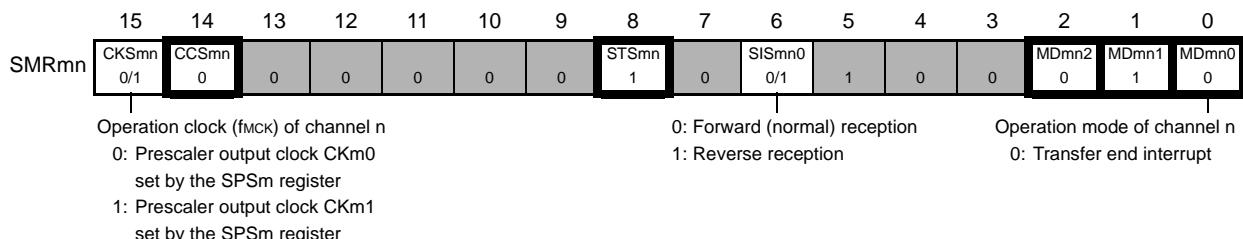
fclk: System clock frequency

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 1, 3), mn = 01, 03, 11, 13

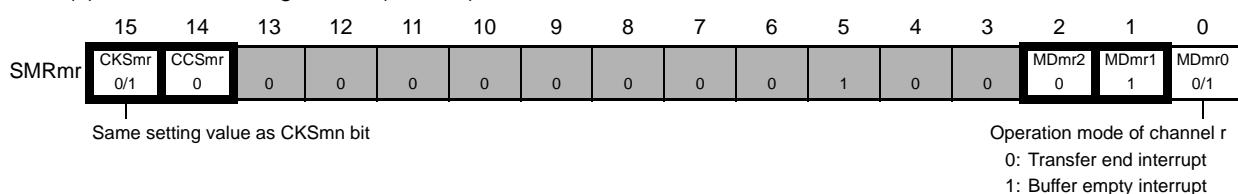
## (1) Register setting

**Figure 17 - 116 Example of Contents of Registers for UART Reception of UART (UART0 to UART3) (1/2)**

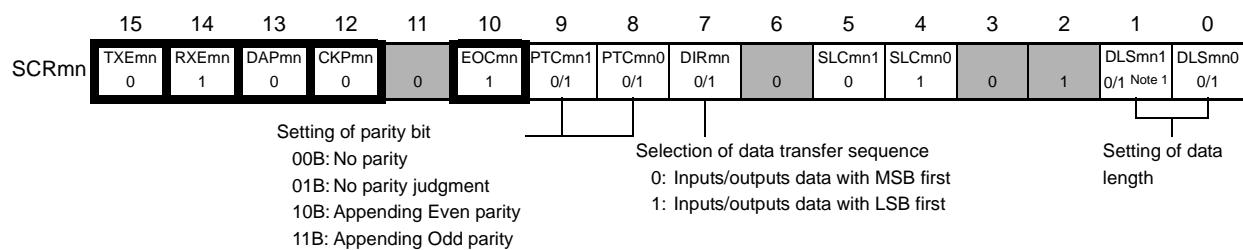
## (a) Serial mode register mn (SMRmn)



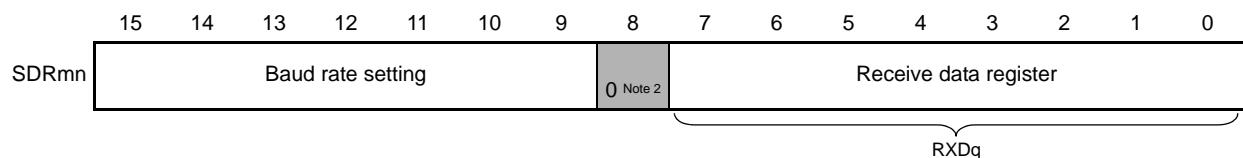
## (b) Serial mode register mr (SMRmr)



## (c) Serial communication operation setting register mn (SCRmn)



## (d) Serial data register mn (SDRmn) (lower 8 bits: RXDq)



**Note 1.** Only provided for the SCR01 register and the SCR11 register of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Note 2.** When performing 9-bit communication, bits 0 to 8 of the SDRm1 register are used to specify the transmission data.

- 30 to 64-pin products: UART0
- 80, 100-pin products: UART0 and UART2

**Caution** For the UART reception, be sure to set the SMRmr register of channel r that is to be paired with channel n.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 1, 3), mn = 01, 03, 11, 13  
r: Channel number (r = n - 1), q: UART number (q = 0 to 3)

**Remark 2.** : Setting is fixed in the UART reception mode,

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

**Figure 17 - 117 Example of Contents of Registers for UART Reception of UART  
(UART0 to UART3) (2/2)**

(e) Serial output register m (SOm)... The register that not used in this mode.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOm	0	0	0	0	CKOm3 x	CKOm2 x	CKOm1 x	CKOm0 x	0	0	0	0	SOm3 x	SOm2 x	SOm1 x	SOm0 x

(f) Serial output enable register m (SOEm)... The register that not used in this mode.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SOEm	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 x	SOEm2 x	SOEm1 x	SOEm0 x

(g) Serial channel start register m (SSm)... Sets only the bits of the target channel is 1.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SSm	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 0/1	SSm2 x	SSm1 0/1	SSm0 x

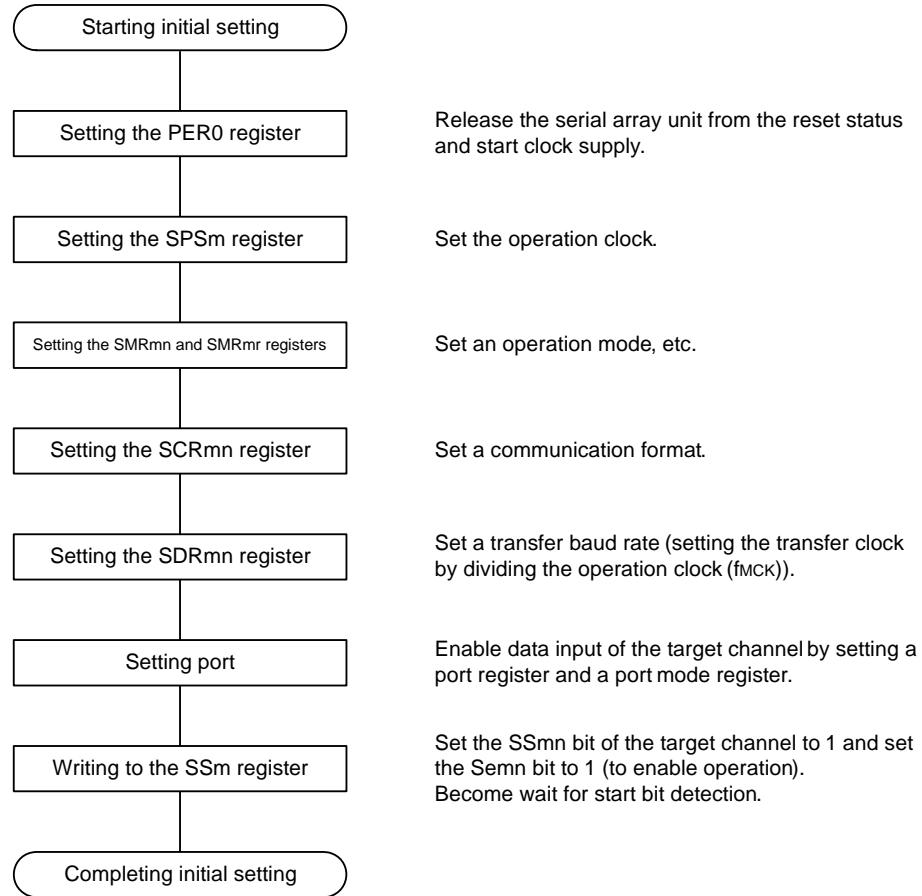
**Remark 1.** m: Unit number (m = 0, 1)

**Remark 2.** : Setting disabled (set to the initial value)

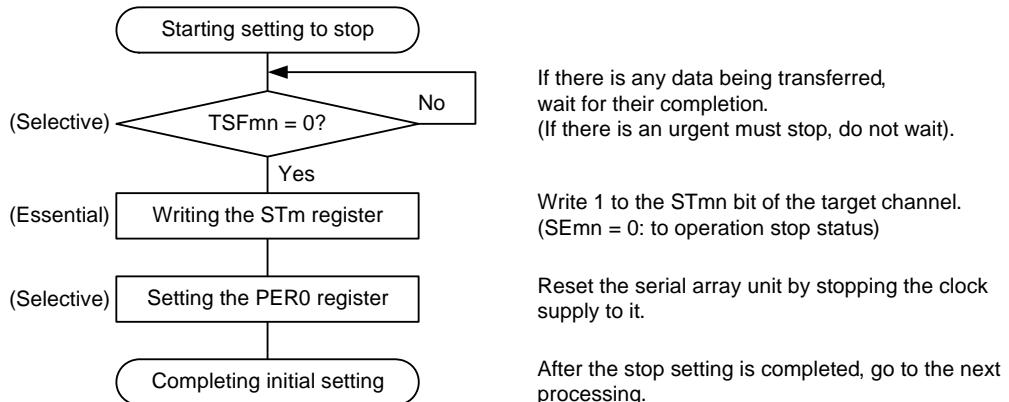
x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

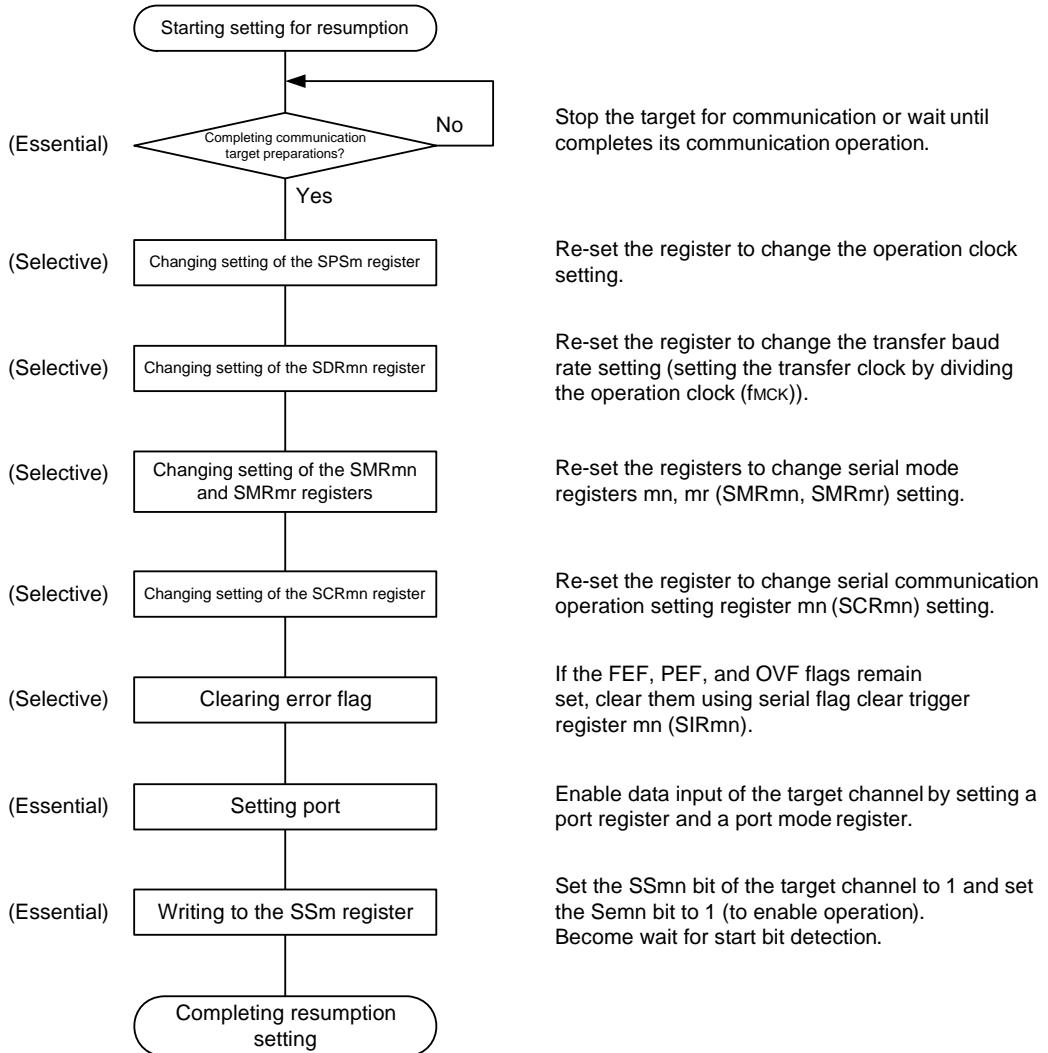
0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

**Figure 17 - 118 Initial Setting Procedure for UART Reception**

**Caution** Set the RXEmn bit of SCRmn register to 1, and then be sure to set SSmn to 1 after 4 or more fmck clocks have elapsed.

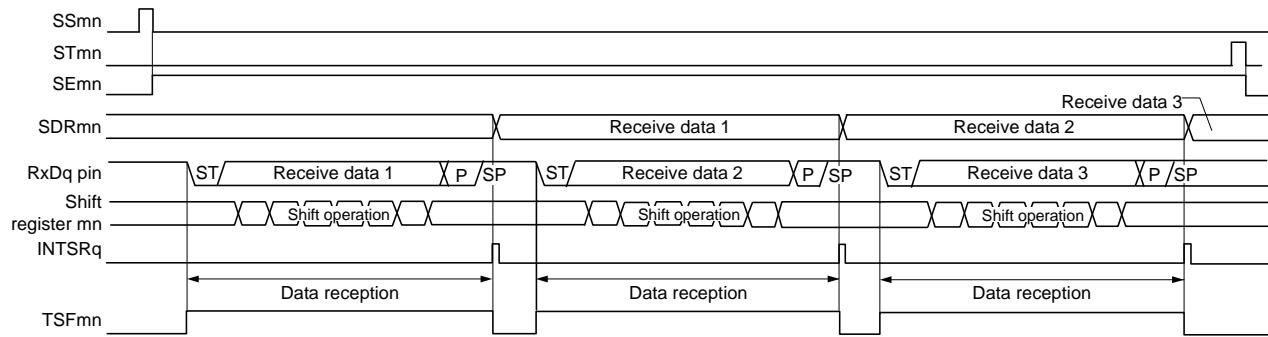
**Figure 17 - 119 Procedure for Stopping UART Reception**

**Figure 17 - 120 Procedure for Resuming UART Reception**

**Caution** After is set RXEmn bit to 1 of SCRmn register, set the SSmn = 1 from an interval of at least four clocks of fmck.

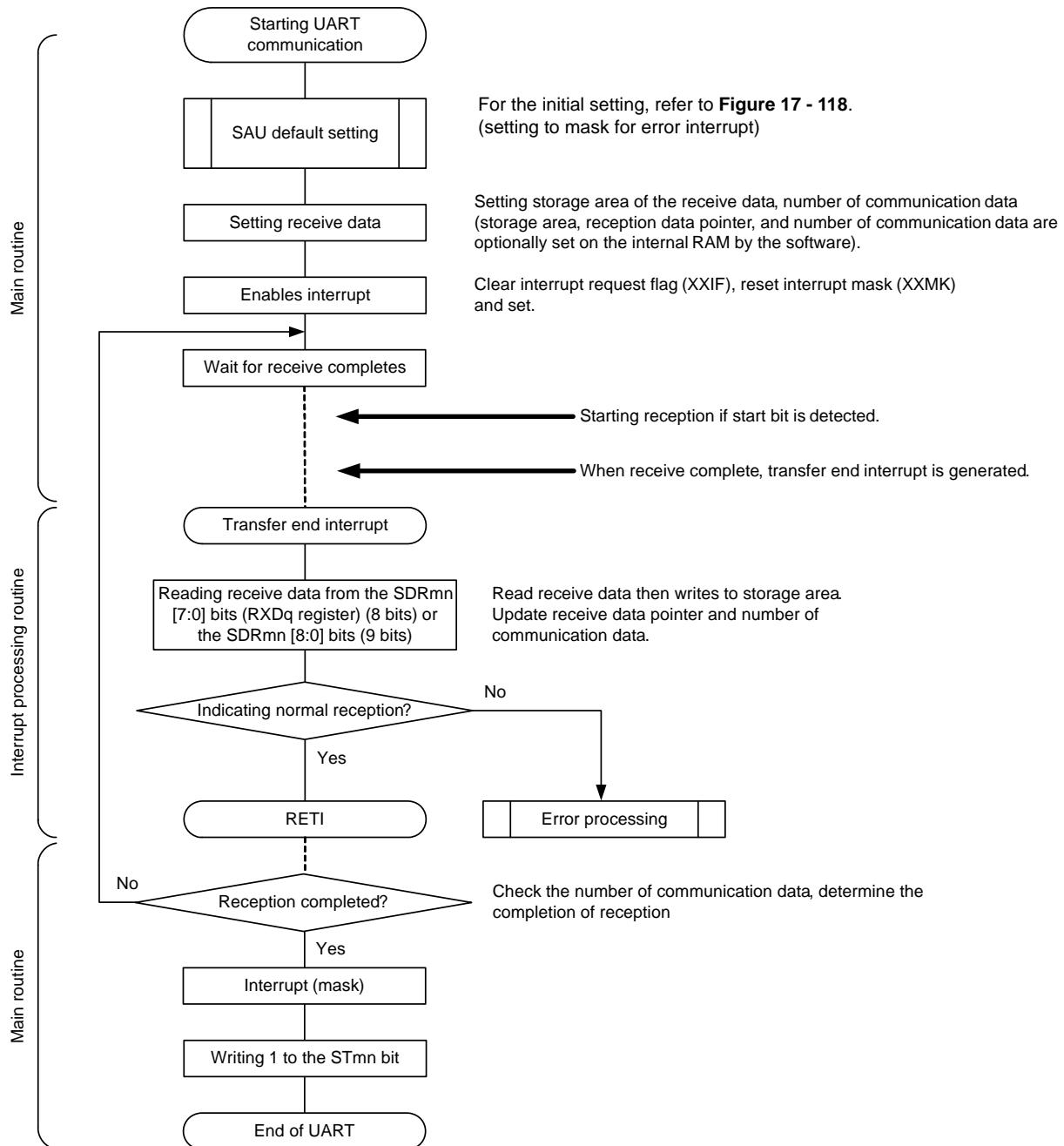
**Remark** If PER0 is rewritten while stopping the communication target and the clock supply is stopped, wait until the communication target stops or communication finishes, and then perform initialization instead of restarting the communication.

## (3) Processing flow

**Figure 17 - 121 Timing Chart of UART Reception**

**Remark**    m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 1, 3$ ), mn = 01, 03, 11, 13  
                  r: Channel number ( $r = n - 1$ ), q: UART number ( $q = 0$  to 3)

Figure 17 - 122 Flowchart of UART Reception



### 17.7.3 SNOOZE mode function

The SNOOZE mode makes the UART perform reception operations upon RxDq pin input detection while in the STOP mode. Normally the UART stops communication in the STOP mode. However, using the SNOOZE mode enables the UART to perform reception operations without CPU operation. Only the following UARTs can be specified when FRQSEL4 in the option byte (000C2H) = 0 in the SNOOZE mode.

- 30 to 64-pin products:   UART0
- 80, 100-pin products:   UART0 and UART2

When using UARTq in the SNOOZE mode, make the following settings before entering the STOP mode (See **Figures 17 - 125 and 17 - 127 Flowchart of SNOOZE Mode Operation**).

- In the SNOOZE mode, the baud rate setting for UART reception needs to be changed to a value different from that in normal operation. Set the SPSm register and bits 15 to 9 of the SDRmn register with reference to Table 17 - 4.
- Set the EOCmn and SSECmn bits. This is for enabling or stopping generation of an error interrupt (INTSRE0) when a communication error occurs.
- When using the SNOOZE mode function, set the SWCm bit of serial standby control register m (SSCm) to 1 just before switching to the STOP mode. After the initial setting has completed, set the SSm1 bit of serial channel start register m (SSm) to 1.
- A UARTq starts reception in SNOOZE mode on detecting input of the start bit on the RxDq pin following a transition of the CPU to the STOP mode.

**Caution 1. The SNOOZE mode can only be specified when the high-speed on-chip oscillator clock ( $f_{IH}$ ) is selected for fCLK.**

**Caution 2. The transfer rate in the SNOOZE mode is only 4800 bps.**

**Caution 3. When SWCm = 1, UARTq can be used only when the reception operation is started in the STOP mode. When used simultaneously with another SNOOZE mode function or interrupt, if the reception operation is started in a state other than the STOP mode, such as those given below, data may not be received correctly and a framing error or parity error may be generated.**

- When after the SWCm bit has been set to 1, the reception operation is started before the STOP mode is entered
- When the reception operation is started while another function is in the SNOOZE mode
- When after returning from the STOP mode to normal operation due to an interrupt or other cause, the reception operation is started before the SWCm bit is returned to 0

**Caution 4. If a parity error, framing error, or overrun error occurs while the SSECm bit is set to 1, the PEFmn, FEFmn, or OVFmn flag is not set and an error interrupt (INTSREq) is not generated. Therefore, when the setting of SSECm = 1 is made, clear the PEFmn, FEFmn, or OVFmn flag before setting the SWC0 bit to 1 and read the value in bits 7 to 0 (RxDq register) of the SDRm1 register.**

**Caution 5. The CPU shifts from the STOP mode to the SNOOZE mode on detecting the valid edge of the RxDq signal. Note, however, that transfer through the UART channel may not start and the CPU may remain in the SNOOZE mode if an input pulse on the RxDq pin is too short to be detected as a start bit.**

**In such cases, data may not be received correctly, and this may lead to a framing error or parity error in the next UART transfer.**

**Table 17 - 4 Baud Rate Setting for UART Reception in SNOOZE Mode**

High-speed On-chip Oscillator ( $f_{IH}$ )	Baud Rate for UART Reception in SNOOZE Mode			
	Baud Rate of 4800 bps			
	Operation Clock ( $f_{MCK}$ )	SDRmn [15:9]	Maximum Permissible Value	Minimum Permissible Value
32 MHz $\pm$ 1.0% Note	$f_{CLK}/2^5$	105	2.27%	-1.53%
24 MHz $\pm$ 1.0% Note	$f_{CLK}/2^5$	79	1.60%	-2.18%
16 MHz $\pm$ 1.0% Note	$f_{CLK}/2^4$	105	2.27%	-1.53%
12 MHz $\pm$ 1.0% Note	$f_{CLK}/2^4$	79	1.60%	-2.19%
8 MHz $\pm$ 1.0% Note	$f_{CLK}/2^3$	105	2.27%	-1.53%
6 MHz $\pm$ 1.0% Note	$f_{CLK}/2^3$	79	1.60%	-2.19%
4 MHz $\pm$ 1.0% Note	$f_{CLK}/2^2$	105	2.27%	-1.53%
3 MHz $\pm$ 1.0% Note	$f_{CLK}/2^2$	79	1.60%	-2.19%
2 MHz $\pm$ 1.0% Note	$f_{CLK}/2$	105	2.27%	-1.54%
1 MHz $\pm$ 1.0% Note	$f_{CLK}$	105	2.27%	-1.57%

**Note** When the accuracy of the clock frequency of the high-speed on-chip oscillator is  $\pm 1.5\%$  or  $\pm 2.0\%$ , the permissible range becomes smaller as shown below.

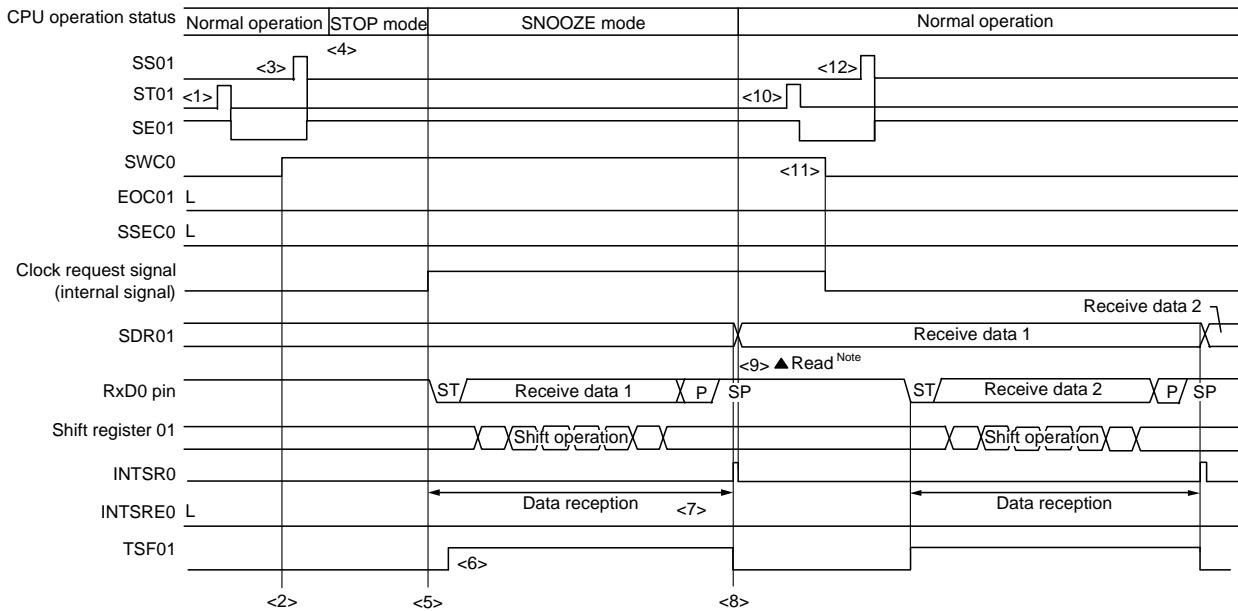
- In the case of  $f_{IH} \pm 1.5\%$ , perform (Maximum permissible value - 0.5%) and (Minimum permissible value + 0.5%) to the values in the above table.
- In the case of  $f_{IH} \pm 2.0\%$ , perform (Maximum permissible value - 1.0%) and (Minimum permissible value + 1.0%) to the values in the above table.

**Remark** The maximum permissible value and minimum permissible value are permissible values for the baud rate in UART reception. The baud rate on the transmitting side should be set to fall inside this range.

(1) SNOOZE mode operation ( $\text{EOCm1} = 0$ ,  $\text{SSECm} = 0/1$ )

Because of the setting of  $\text{EOCm1} = 0$ , even though a communication error occurs, an error interrupt ( $\text{INTSREq}$ ) is not generated, regardless of the setting of the  $\text{SSECm}$  bit. A transfer end interrupt ( $\text{INTSRq}$ ) will be generated.

**Figure 17 - 123 Timing Chart of SNOOZE Mode Operation ( $\text{EOCm1} = 0$ ,  $\text{SSECm} = 0/1$ )**



**Note** Read the received data when  $\text{SWCm}$  is 1.

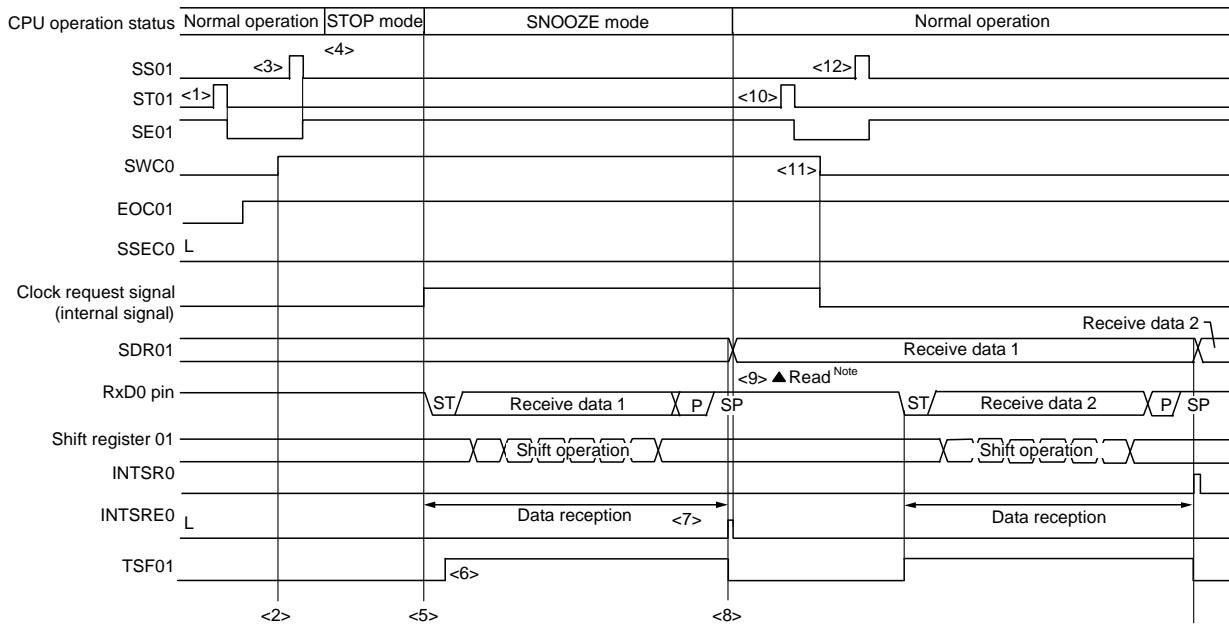
**Caution** Before switching to the SNOOZE mode or after reception operation in the SNOOZE mode finishes, set the  $\text{STM1}$  bit to 1 (clear the  $\text{SEm1}$  bit, and stop the operation).  
And after completion the receive operation, also clearing  $\text{SWCm}$  bit to 0 (SNOOZE mode release).

**Remark 1.** <1> to <12> in the figure correspond to <1> to <12> in Figure 17 - 125 Flowchart of SNOOZE Mode Operation ( $\text{EOCm1} = 0$ ,  $\text{SSECm} = 0/1$  or  $\text{EOCm1} = 1$ ,  $\text{SSECm} = 0$ ).

**Remark 2.** 30 to 64-pin products:  $m = 0$ ;  $q = 0$   
80, 100-pin products:  $m = 0, 1$ ;  $q = 0, 2$

- (2) SNOOZE mode operation (EOCm1 = 1, SSECm = 0: Error interrupt (INTSReq) generation is enabled)  
Because EOCm1 = 1 and SSECm = 0, an error interrupt (INTSReq) is generated when a communication error occurs.

**Figure 17 - 124 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 0)**



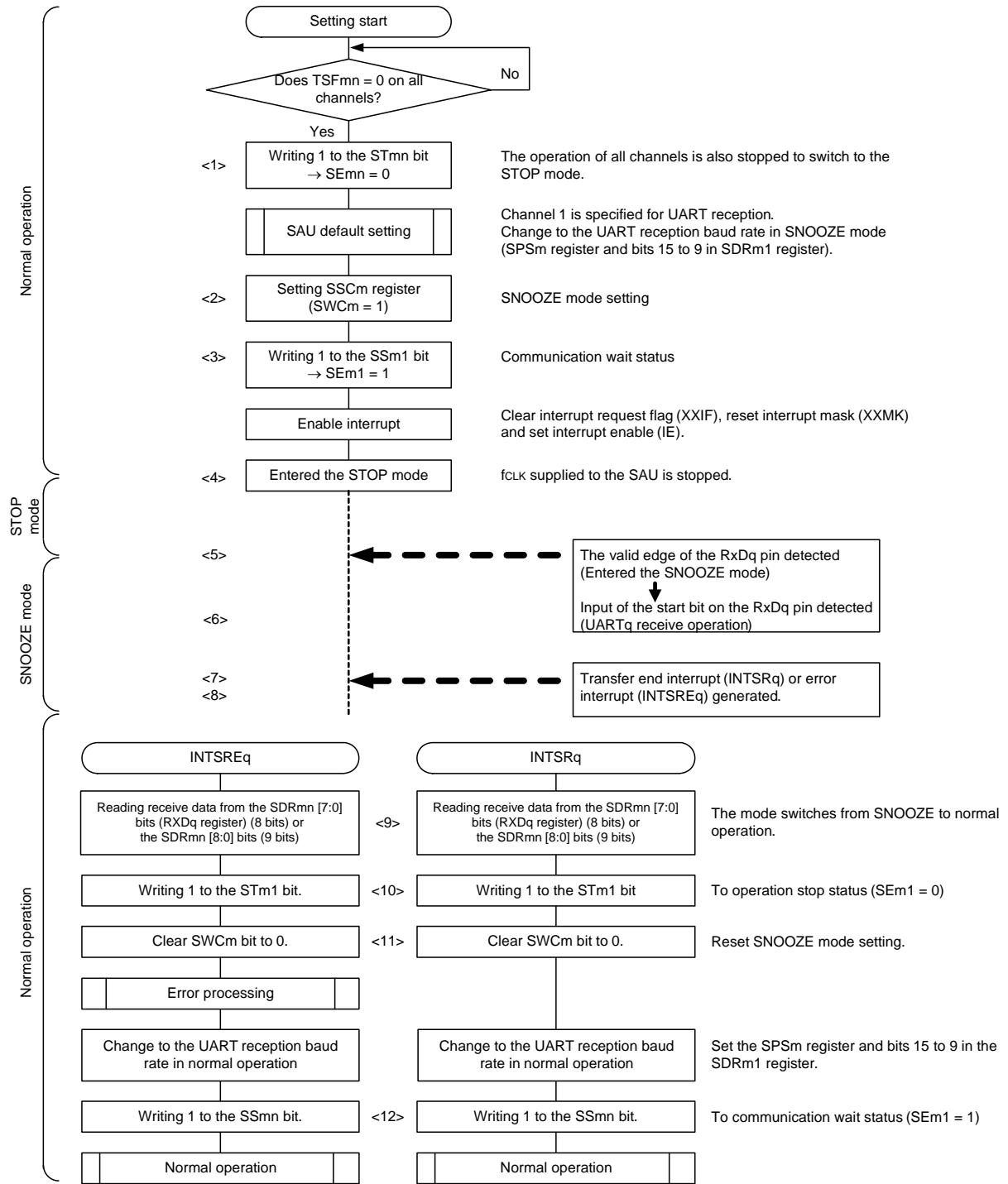
**Note** Read the received data when SWCm = 1.

**Caution** Before switching to the SNOOZE mode or after reception operation in the SNOOZE mode finishes, set the STM1 bit to 1 (clear the SEM1 bit, and stop the operation).  
And after completion the receive operation, also clearing SWCm bit to 0 (SNOOZE mode release).

**Remark 1.** <1> to <12> in the figure correspond to <1> to <12> in Figure 17 - 125 Flowchart of SNOOZE Mode Operation (EOCm1 = 0, SSECm = 0/1 or EOCm1 = 1, SSECm = 0).

**Remark 2.** 30 to 64-pin products: m = 0; q = 0  
80, 100-pin products: m = 0, 1; q = 0, 2

Figure 17 - 125 Flowchart of SNOOZE Mode Operation (EOCm1 = 0, SSECm = 0/1 or EOCm1 = 1, SSECm = 0)

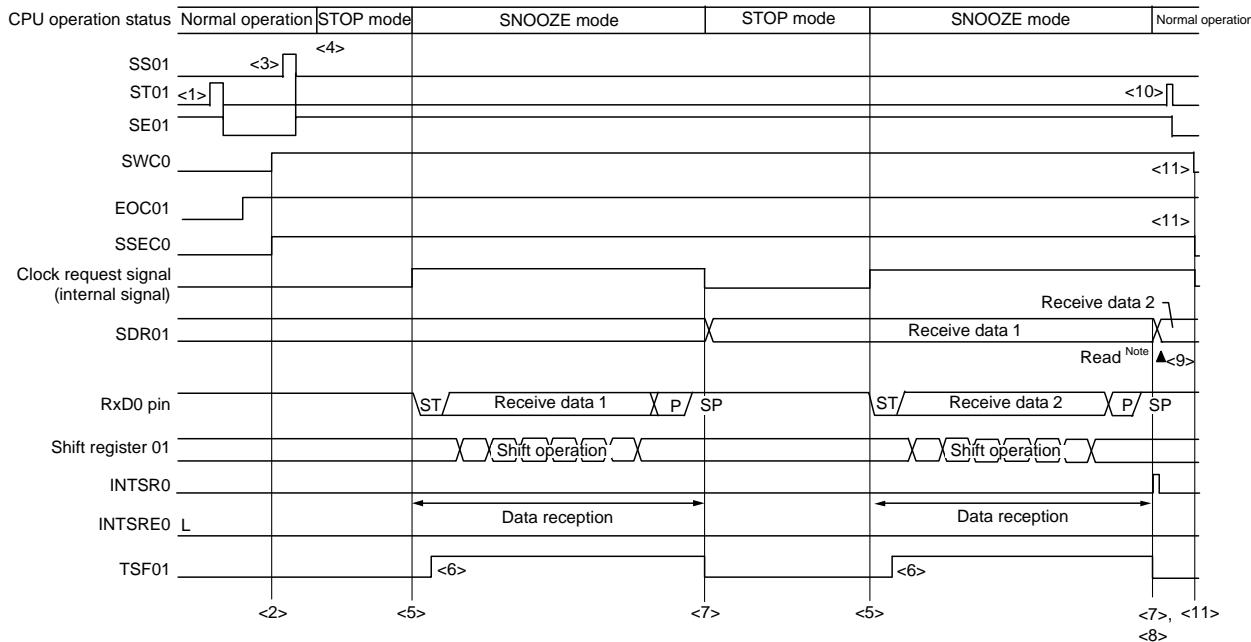


**Remark 1.** <1> to <12> in the figure correspond to <1> to <12> in Figure 17 - 123 Timing Chart of SNOOZE Mode Operation (EOCm1 = 0, SSECm = 0/1) and Figure 17 - 124 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 0).

**Remark 2.** 30 to 64-pin products: m = 0; q = 0  
80, 100-pin products: m = 0, 1; q = 0, 2

- (3) SNOOZE mode operation ( $\text{EOCm1} = 1$ ,  $\text{SSECm} = 1$ : Error interrupt (INTSREQ) generation is stopped)  
 Because  $\text{EOCm1} = 1$  and  $\text{SSECm} = 1$ , an error interrupt (INTSREQ) is not generated when a communication error occurs.

**Figure 17 - 126 Timing Chart of SNOOZE Mode Operation ( $\text{EOCm1} = 1$ ,  $\text{SSECm} = 1$ )**



**Note** Only read received data while  $\text{SWCm} = 1$ .

**Caution 1.** Before switching to the SNOOZE mode or after reception operation in the SNOOZE mode finishes, set the  $\text{STM1}$  bit to 1 (clear the  $\text{SEm1}$  bit, and stop the operation).

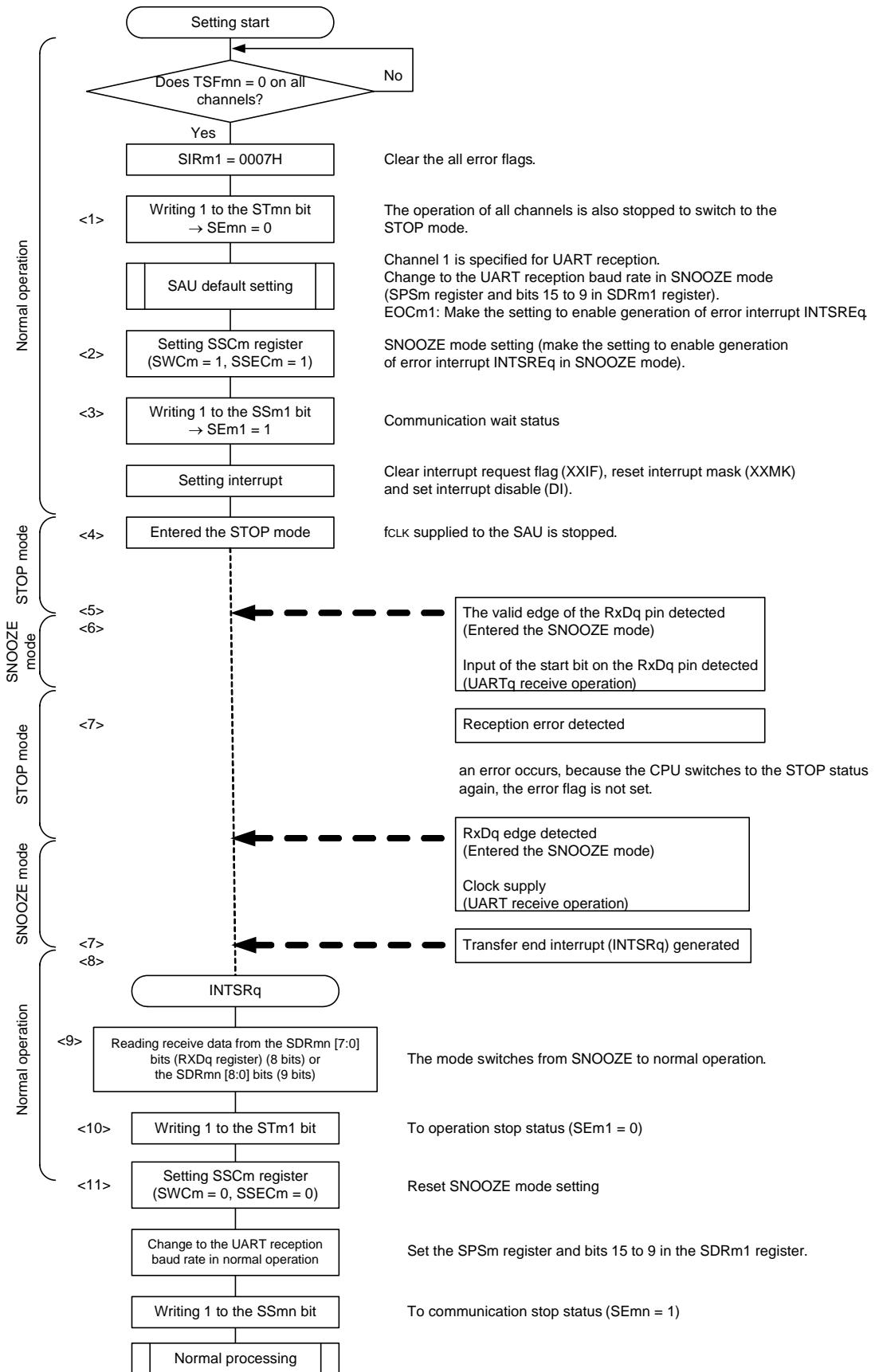
And after completion the receive operation, also clearing  $\text{SWCm}$  bit to 0 (SNOOZE mode release).

**Caution 2.** If a parity error, framing error, or overrun error occurs while the  $\text{SSECm}$  bit is set to 1, the  $\text{PEFm1}$ ,  $\text{FEFm1}$ , or  $\text{OVFm1}$  flag is not set and an error interrupt (INTSREQ) is not generated. Therefore, when the setting of  $\text{SSECm} = 1$  is made, clear the  $\text{PEFm1}$ ,  $\text{FEFm1}$ , or  $\text{OVFm1}$  flag before setting the  $\text{SWCm}$  bit to 1 and read the value in  $\text{SDRm1}[7:0]$  (RxDq register) (8 bits) or  $\text{SDRm1}[8:0]$  (9 bits).

**Remark 1.** <1> to <11> in the figure correspond to <1> to <11> in Figure 17 - 127 Flowchart of SNOOZE Mode Operation ( $\text{EOCm1} = 1$ ,  $\text{SSECm} = 1$ ).

**Remark 2.** 30 to 64-pin products:  $m = 0$ ;  $q = 0$   
 80, 100-pin products:  $m = 0, 1$ ;  $q = 0, 2$

Figure 17 - 127 Flowchart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 1)



(Caution and Remarks are listed on the next page.)

**Caution** If a parity error, framing error, or overrun error occurs while the SSECm bit is set to 1, the PEFm1, FEFm1, or OVFm1 flag is not set and an error interrupt (INTSREq) is not generated. Therefore, when the setting of SSECm = 1 is made, clear the PEFm1, FEFm1, or OVFm1 flag before setting the SWCm bit to 1 and read the value in SDRm1[7:0] (RxDq register) (8 bits) or SDRm1[8:0] (9 bits).

**Remark 1.** <1> to <11> in the figure correspond to <1> to <11> in Figure 17 - 126 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 1).

**Remark 2.** 30 to 64-pin products: m = 0; q = 0  
80, 100-pin products: m = 0, 1; q = 0, 2

### 17.7.4 Calculating baud rate

#### (1) Baud rate calculation expression

The baud rate for UART (UART0 to UART3) communication can be calculated by the following expressions.

$$\text{(Baud rate)} = \{\text{Operation clock (fmck) frequency of target channel}\} \div (\text{SDRmn}[15:9] + 1) \div 2 \text{ [bps]}$$

**Caution** Setting serial data register mn (SDRmn) SDRmn[15:9] = (0000000B, 0000001B) is prohibited.

**Remark 1.** When UART is used, the value of SDRmn[15:9] is the value of bits 15 to 9 of the SDRmn register (0000010B to 1111111B) and therefore is 2 to 127.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

The operation clock (fmck) is determined by serial clock select register m (SPSm) and bit 15 (CKSmn) of serial mode register mn (SMRmn).

Table 17 - 5 Selection of Operation Clock For UART

SMRmn Register	SPSm Register								Operation Clock (fMCK) Note	
CKSmn	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00		
0	x	x	x	x	0	0	0	0	fCLK	32 MHz
	x	x	x	x	0	0	0	1	fCLK/2	16 MHz
	x	x	x	x	0	0	1	0	fCLK/2 <sup>2</sup>	8 MHz
	x	x	x	x	0	0	1	1	fCLK/2 <sup>3</sup>	4 MHz
	x	x	x	x	0	1	0	0	fCLK/2 <sup>4</sup>	2 MHz
	x	x	x	x	0	1	0	1	fCLK/2 <sup>5</sup>	1 MHz
	x	x	x	x	0	1	1	0	fCLK/2 <sup>6</sup>	500 kHz
	x	x	x	x	0	1	1	1	fCLK/2 <sup>7</sup>	250 kHz
	x	x	x	x	1	0	0	0	fCLK/2 <sup>8</sup>	125 kHz
	x	x	x	x	1	0	0	1	fCLK/2 <sup>9</sup>	62.5 kHz
	x	x	x	x	1	0	1	0	fCLK/2 <sup>10</sup>	31.25 kHz
	x	x	x	x	1	0	1	1	fCLK/2 <sup>11</sup>	15.63 kHz
	x	x	x	x	1	1	0	0	fCLK/2 <sup>12</sup>	7.81 kHz
	x	x	x	x	1	1	0	1	fCLK/2 <sup>13</sup>	3.91 kHz
	x	x	x	x	1	1	1	0	fCLK/2 <sup>14</sup>	1.95 kHz
	x	x	x	x	1	1	1	1	fCLK/2 <sup>15</sup>	977 Hz
1	0	0	0	0	x	x	x	x	fCLK	32 MHz
	0	0	0	1	x	x	x	x	fCLK/2	16 MHz
	0	0	1	0	x	x	x	x	fCLK/2 <sup>2</sup>	8 MHz
	0	0	1	1	x	x	x	x	fCLK/2 <sup>3</sup>	4 MHz
	0	1	0	0	x	x	x	x	fCLK/2 <sup>4</sup>	2 MHz
	0	1	0	1	x	x	x	x	fCLK/2 <sup>5</sup>	1 MHz
	0	1	1	0	x	x	x	x	fCLK/2 <sup>6</sup>	500 kHz
	0	1	1	1	x	x	x	x	fCLK/2 <sup>7</sup>	250 kHz
	1	0	0	0	x	x	x	x	fCLK/2 <sup>8</sup>	125 kHz
	1	0	0	1	x	x	x	x	fCLK/2 <sup>9</sup>	62.5 kHz
	1	0	1	0	x	x	x	x	fCLK/2 <sup>10</sup>	31.25 kHz
	1	0	1	1	x	x	x	x	fCLK/2 <sup>11</sup>	15.63 kHz
	1	1	0	0	x	x	x	x	fCLK/2 <sup>12</sup>	7.81 kHz
	1	1	0	1	x	x	x	x	fCLK/2 <sup>13</sup>	3.91 kHz
	1	1	1	0	x	x	x	x	fCLK/2 <sup>14</sup>	1.95 kHz
	1	1	1	1	x	x	x	x	fCLK/2 <sup>15</sup>	977 Hz

**Note** When changing the clock selected for fCLK (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (STM) = 000FH) the operation of the serial array unit (SAU).

**Remark 1.** x: Don't care

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## (2) Baud rate error during transmission

The baud rate error of UART (UART0 to UART3) communication during transmission can be calculated by the following expression. Make sure that the baud rate at the transmission side is within the permissible baud rate range at the reception side.

$$\text{(Baud rate error)} = (\text{Calculated baud rate value}) \div (\text{Target baud rate}) \times 100 - 100 [\%]$$

Here is an example of setting a UART baud rate at fCLK = 32 MHz.

UART Baud Rate (Target Baud Rate)	fCLK = 32 MHz			
	Operation Clock (fmck)	SDRmn[15:9]	Calculated Baud Rate	Error from Target Baud Rate
300 bps	fCLK/2 <sup>9</sup>	103	300.48 bps	+0.16%
600 bps	fCLK/2 <sup>8</sup>	103	600.96 bps	+0.16%
1200 bps	fCLK/2 <sup>7</sup>	103	1201.92 bps	+0.16%
2400 bps	fCLK/2 <sup>6</sup>	103	2403.85 bps	+0.16%
4800 bps	fCLK/2 <sup>5</sup>	103	4807.69 bps	+0.16%
9600 bps	fCLK/2 <sup>4</sup>	103	9615.38 bps	+0.16%
19200 bps	fCLK/2 <sup>3</sup>	103	19230.8 bps	+0.16%
31250 bps	fCLK/2 <sup>3</sup>	63	31250.0 bps	±0.0%
38400 bps	fCLK/2 <sup>2</sup>	103	38461.5 bps	+0.16%
76800 bps	fCLK/2	103	76923.1 bps	+0.16%
153600 bps	fCLK	103	153846 bps	+0.16%
312500 bps	fCLK	50	313725 bps	+0.39%

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), mn = 00, 02, 10, 12

## (3) Permissible baud rate range for reception

The permissible baud rate range for reception during UART (UART0 to UART3) communication can be calculated by the following expression. Make sure that the baud rate at the transmission side is within the permissible baud rate range at the reception side.

$$\text{(Maximum receivable baud rate)} = \frac{2 \times k \times Nfr}{2 \times k \times Nfr - k + 2} \times \text{Brate}$$

$$\text{(Minimum receivable baud rate)} = \frac{2 \times k \times (Nfr - 1)}{2 \times k \times Nfr - k - 2} \times \text{Brate}$$

Brate: Calculated baud rate value at the reception side (See **17.7.4 (1) Baud rate calculation expression.**)

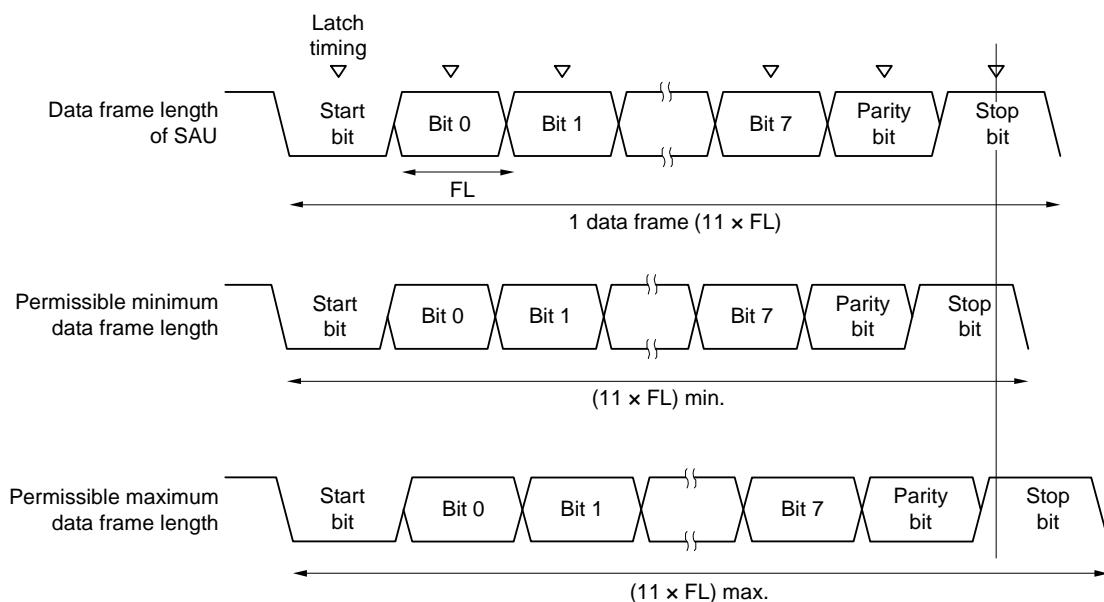
k: SDRmn[15:9] + 1

Nfr: 1 data frame length [bits]

$$= (\text{Start bit}) + (\text{Data length}) + (\text{Parity bit}) + (\text{Stop bit})$$

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 1, 3), mn = 01, 03, 11, 13

**Figure 17 - 128 Permissible Baud Rate Range for Reception (1 Data Frame Length = 11 Bits)**



As shown in Figure 17 - 128, the timing of latching receive data is determined by the division ratio set by bits 15 to 9 of serial data register mn (SDRmn) after the start bit is detected. If the last data (stop bit) is received before this latch timing, the data can be correctly received.

### 17.7.5 Procedure for processing errors that occurred during UART (UART0 to UART3) communication

The procedure for processing errors that occurred during UART (UART0 to UART3) communication is described in Figures 17 - 129 and 17 - 130.

**Figure 17 - 129 Processing Procedure in Case of Parity Error or Overrun Error**

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn)	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes 1 to serial flag clear trigger register mn (SIRmn).	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.

**Figure 17 - 130 Processing Procedure in Case of Framing Error**

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn).	The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes serial flag clear trigger register mn (SIRmn).	Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.
Sets the STmn bit of serial channel stop register m (STm) to 1.	The SEMn bit of serial channel enable status register m (SEM) is set to 0 and channel n stops operating.	
Synchronization with other party of communication		Synchronization with the other party of communication is re-established and communication is resumed because it is considered that a framing error has occurred because the start bit has been shifted.
Sets the SSmn bit of serial channel start register m (SSm) to 1.	The SEMn bit of serial channel enable status register m (SEM) is set to 1 and channel n is enabled to operate.	

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## 17.8 LIN Communication Operation

### 17.8.1 LIN transmission

Of UART transmission, UART0 support LIN communication.

For LIN transmission, channel 0 of unit 0 is used.

UART	UART0	UART1	UART2	UART3
Support of LIN communication	Supported	Not supported	Not supported	Not supported
Target channel	Channel 0 of SAU0	—	—	—
Pins used	TxD0	—	—	—
Interrupt	INTST0	—	—	—
	Transfer end interrupt (in single-transfer mode) or buffer empty interrupt (in continuous transfer mode) can be selected.			
Error detection flag	None			
Transfer data length	8 bits			
Transfer rate <small>Note</small>	Max. fmck/6 [bps] (SDR00 [15:9] = 2 or more), Min. fclk/(2 × 2 <sup>15</sup> × 128) [bps]			
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)			
Parity bit	No parity bit			
Stop bit	Appending 1 bit			
Data direction	MSB or LSB first			

**Note** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**). In addition, LIN communication is usually 2.4/9.6/19.2 kbps is often used.

**Remark** fmck: Operation clock frequency of target channel  
fclk: System clock frequency

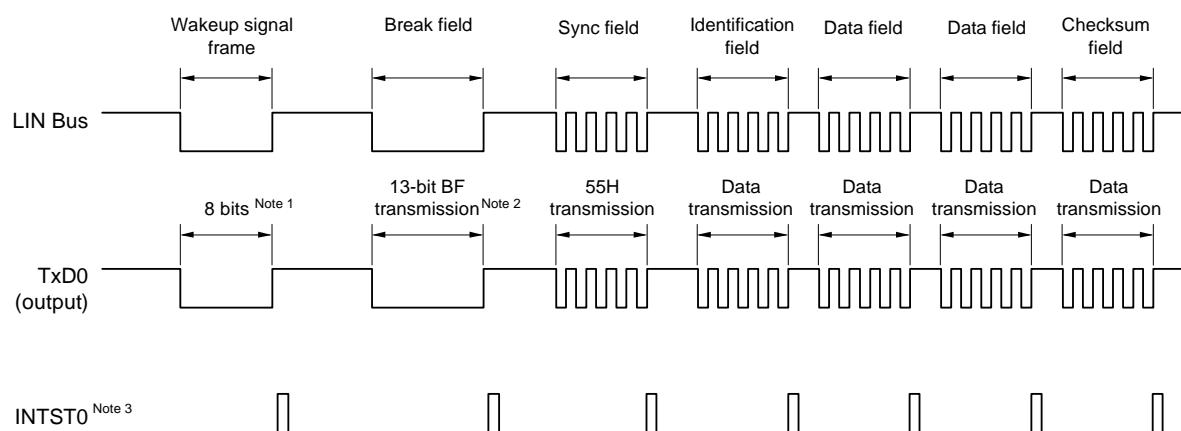
LIN stands for Local Interconnect Network and is a low-speed (1 to 20 kbps) serial communication protocol designed to reduce the cost of an automobile network. Communication of LIN is single-master communication and up to 15 slaves can be connected to one master. The slaves are used to control switches, actuators, and sensors, which are connected to the master via LIN.

Usually, the master is connected to a network such as CAN (Controller Area Network). A LIN bus is a single-wire bus to which nodes are connected via transceiver conforming to ISO9141.

According to the protocol of LIN, the master transmits a frame by attaching baud rate information to it. A slave receives this frame and corrects a baud rate error from the master. If the baud rate error of a slave is within  $\pm 15\%$ , communication can be established.

Figure 17 - 131 outlines a transmission operation of LIN.

**Figure 17 - 131 Transmission Operation of LIN**



**Note 1.** Set the baud rate in accordance with the wakeup signal regulations and transmit data of 80H.

**Note 2.** A break field is defined to have a width of 13 bits and output a low level. Where the baud rate for main transfer is N [bps], therefore, the baud rate of the break field is calculated as follows.

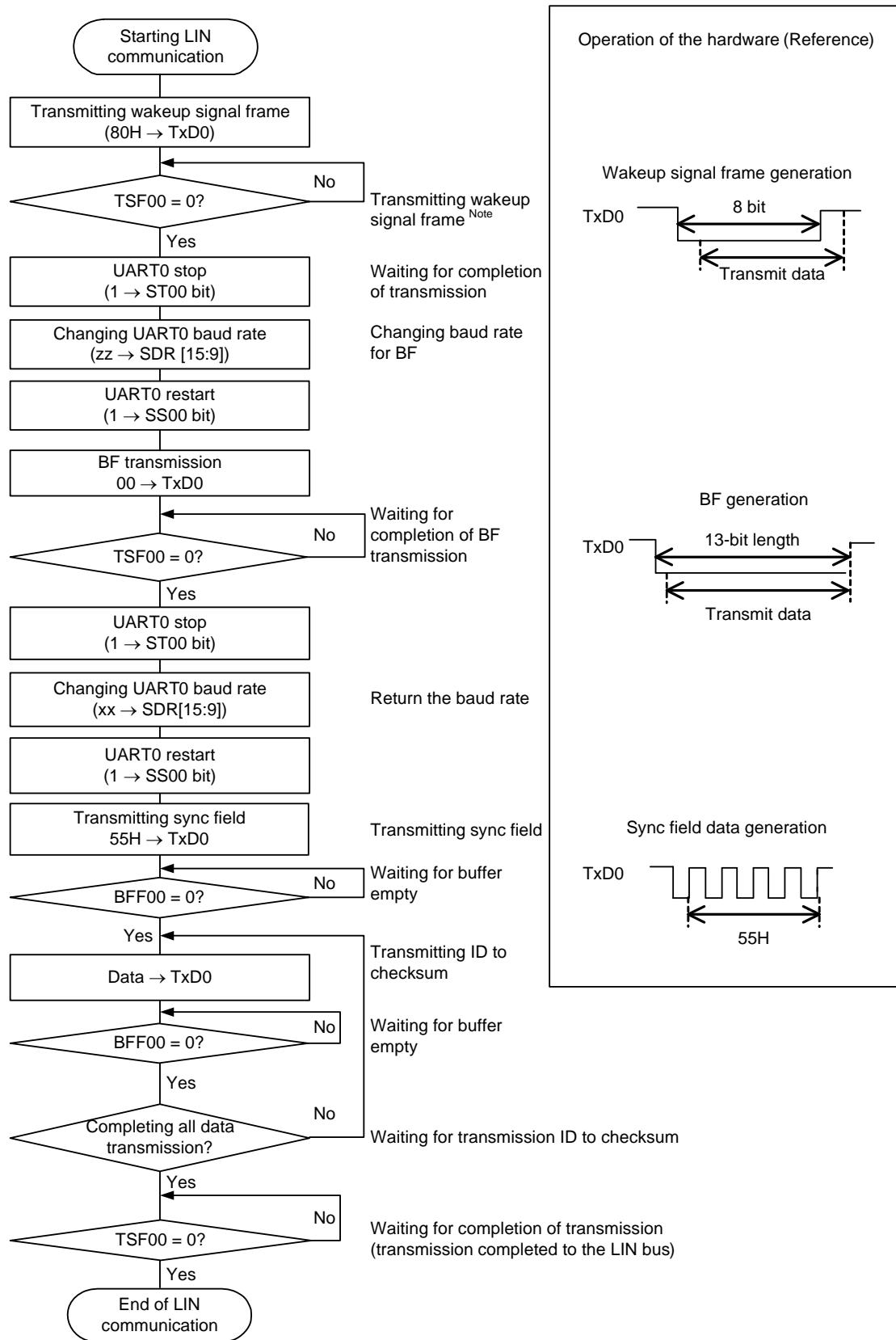
$$\text{(Baud rate of break field)} = 9/13 \times N$$

By transmitting data of 00H at this baud rate, a break field is generated.

**Note 3.** INTST0 is output upon completion of transmission. INTST0 is also output at BF transmission.

**Remark** The interval between fields is controlled by software.

**Figure 17 - 132 Flowchart for LIN Transmission**



**Note** When LIN-bus start from sleep status only.

**Remark** Default setting of the UART is complete, and the flow from the transmission enable status.

## 17.8.2 LIN reception

Of UART reception, UART0 support LIN communication.

For LIN reception, channel 1 of unit 0 is used.

UART	UART0	UART1	UART2	UART3			
Support of LIN communication	Supported	Not supported	Not supported	Not supported			
Target channel	Channel 1 of SAU0	—	—	—			
Pins used	RxD0	—	—	—			
Interrupt	INTSR0	—	—	—			
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)						
Error interrupt	INTSRE0	—	—	—			
Error detection flag	<ul style="list-style-type: none"> <li>• Framing error detection flag (FEF01)</li> <li>• Overrun error detection flag (OVF01)</li> </ul>						
Transfer data length	8 bits						
Transfer rate Note	Max. fmck/6 [bps] (SDR01 [15:9] = 2 or more), Min. fclk/(2 × 2 <sup>15</sup> × 128) [bps]						
Data phase	Non-reverse output (default: high level) Reverse output (default: low level)						
Parity bit	No parity bit (The parity bit is not checked.)						
Stop bit	Appending 1 bit						
Data direction	LSB first						

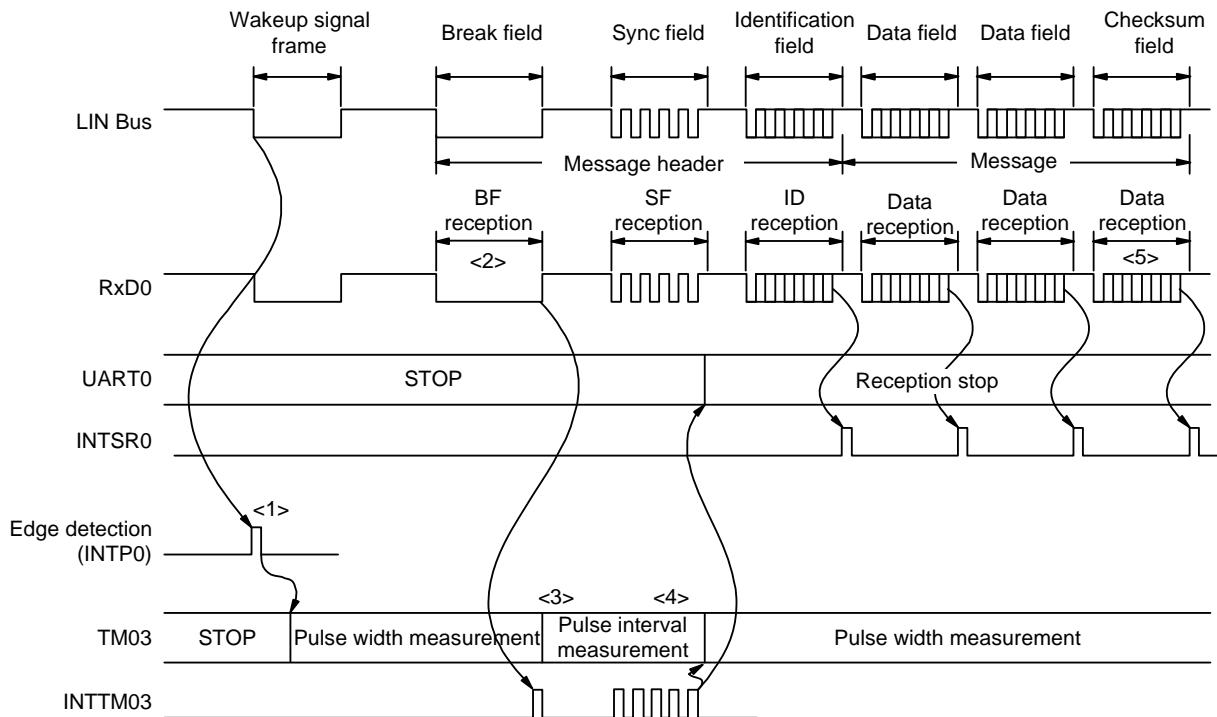
**Note** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** fmck: Operation clock frequency of target channel

fclk: System clock frequency

Figure 17 - 133 outlines a reception operation of LIN.

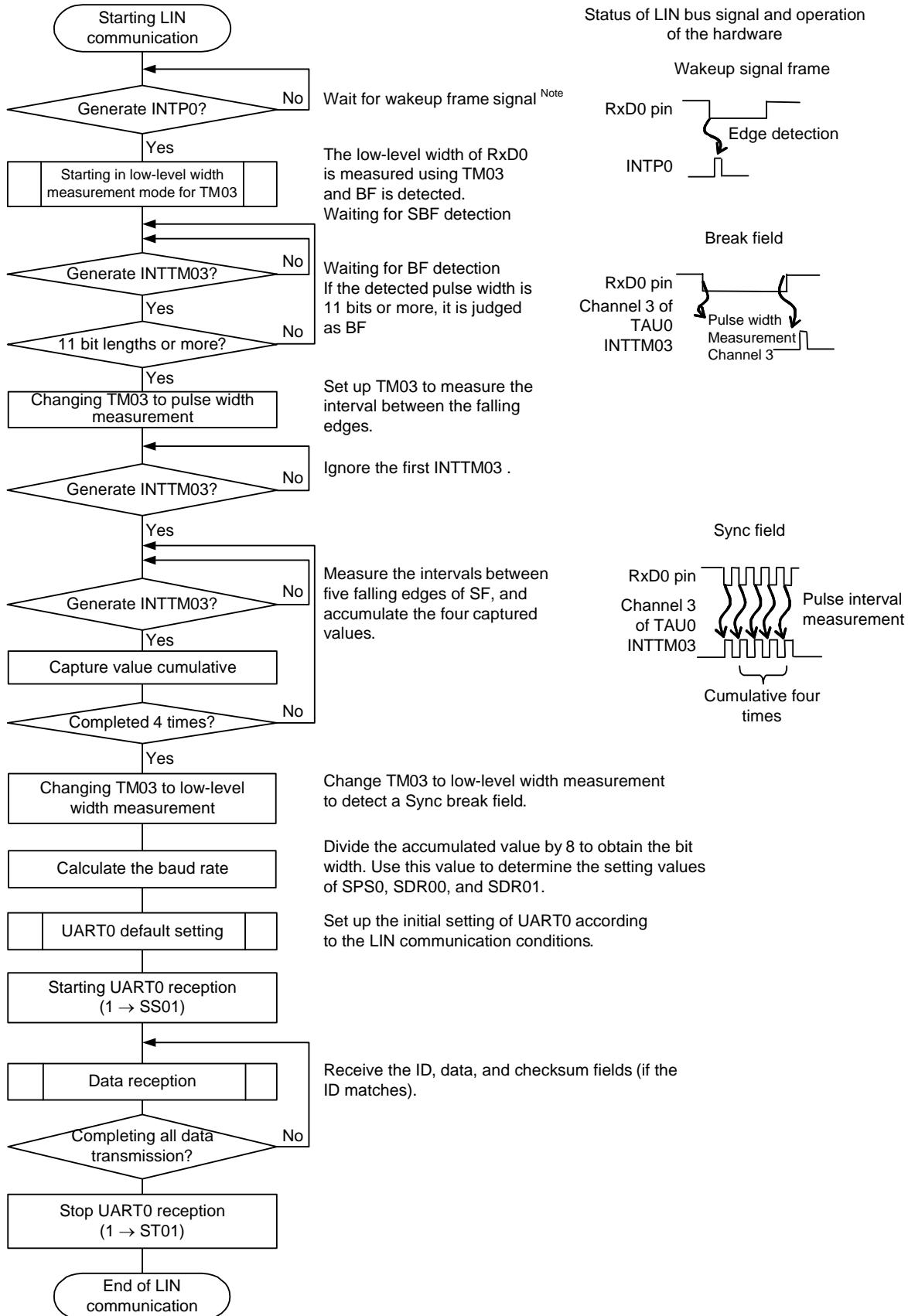
Figure 17 - 133 Reception Operation of LIN



Here is the flow of signal processing.

- <1> The wakeup signal is detected by detecting an interrupt edge (INTP0) on a pin. When the wakeup signal is detected, change TM03 to pulse width measurement upon detection of the wakeup signal to measure the low level width of the BF signal. Then wait for BF signal reception.
- <2> TM03 starts measuring the low-level width upon detection of the falling edge of the BF signal, and then captures the data upon detection of the rising edge of the BF signal. The captured data is used to judge whether it is the BF signal.
- <3> When the BF signal has been received normally, change TM03 to pulse interval measurement and measure the interval between the falling edges of the RxDO signal in the Sync field four times (see **6.8.4 Operation as input pulse interval measurement**).
- <4> Calculate a baud rate error from the bit interval of sync field (SF). Stop UART0 once and adjust (re-set) the baud rate.
- <5> The checksum field should be distinguished by software. In addition, processing to initialize UART0 after the checksum field is received and to wait for reception of BF should also be performed by software.

Figure 17 - 134 Flowchart for LIN Reception



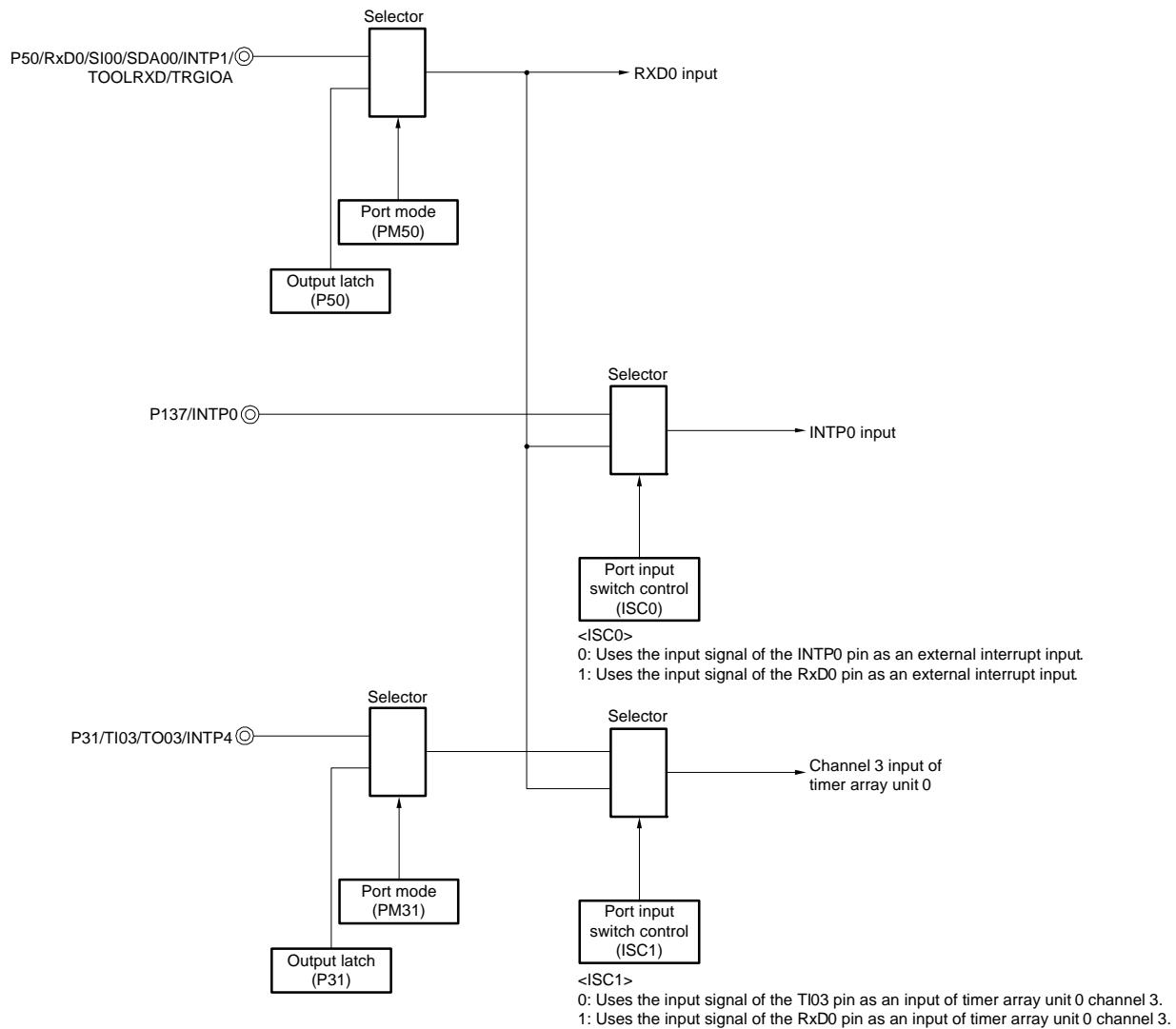
**Note** Required in the sleep status only.

Figure 17 - 135 shows the configuration of a port that manipulates reception of LIN.

The wakeup signal transmitted from the master of LIN is received by detecting an edge of an external interrupt (INTP0). The length of the sync field transmitted from the master can be measured by using the external event capture operation of the timer array unit 0 to calculate a baud-rate error.

By controlling switch of port input (ISC0/ISC1), the input source of port input (RxDO) for reception can be input to the external interrupt pin (INTP0) and timer array unit

**Figure 17 - 135 Port Configuration for Manipulating Reception of LIN**



**Remark** ISC0, ISC1: Bits 0 and 1 of the input switch control register (ISC) (See **Figure 17 - 24.**)

The peripheral functions used for the LIN communication operation are as follows.

<Peripheral functions used>

- External interrupt (INTP0); Wakeup signal detection  
Usage: To detect an edge of the wakeup signal and the start of communication
- Channel 3 of timer array unit; Baud rate error detection, break field (BF) detection.  
Usage: To detect the length of the sync field (SF) and divide it by the number of bits in order to detect an error  
(The interval of the edge input to RxD0 is measured in the capture mode.)  
Measured the low-level width, determine whether break field (BF).
- Channels 0 and 1 (UART0) of serial array unit 0 (SAU0)

## 17.9 Operation of Simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) Communication

This is a clocked communication function to communicate with two or more devices by using two lines: serial clock (SCL) and serial data (SDA). This communication function is designed to execute single communication with devices such as EEPROM, flash memory, and A/D converter, and therefore, can be used only by the master.

Make sure by using software, as well as operating the control registers, that the AC specifications of the start and stop conditions are observed.

[Data transmission/reception]

- Master transmission, master reception (only master function with a single master)
- ACK output function Note and ACK detection function
- Data length of 8 bits  
(When an address is transmitted, the address is specified by the higher 7 bits, and the least significant bit is used for R/W control.)
- Generation of start condition and stop condition for software

[Interrupt function]

- Transfer end interrupt

[Error detection flag]

- Overrun error
- ACK error

\* [Functions not supported by simplified I<sup>2</sup>C]

- Slave transmission, slave reception
- Multi-master function (arbitration loss detection function)
- Wait detection function

**Note** When receiving the last data, ACK will not be output if 0 is written to the SOEmn (SOEm register) bit and serial communication data output is stopped. See the processing flow in **17.9.3 (2)** for details.

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

The channel supporting simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) is channels 0 to 3 of SAU0 and channels 0 to 3 of SAU1.

- 30, 32-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	—		—

- 36, 40, 44-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input)	UART0 (supporting LIN-bus)	IIC00
	1	—		—
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 48, 52-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	—	UART1	—
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 64-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	CSI10	UART1	IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21

- 80, 100-pin products

Unit	Channel	Used as CSI	Used as UART	Used as Simplified I <sup>2</sup> C
0	0	CSI00 (supporting slave select input)	UART0 (supporting LIN-bus)	IIC00
	1	CSI01		IIC01
	2	CSI10		IIC10
	3	CSI11		IIC11
1	0	CSI20	UART2	IIC20
	1	CSI21		IIC21
	2	CSI30	UART3	IIC30
	3	CSI31		IIC31

Simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) performs the following four types of communication operations.

- Address field transmission (See [17.9.1.](#))
- Data transmission (See [17.9.2.](#))
- Data reception (See [17.9.3.](#))
- Stop condition generation (See [17.9.4.](#))

### 17.9.1 Address field transmission

Address field transmission is a transmission operation that first executes in I<sup>2</sup>C communication to identify the target for transfer (slave). After a start condition is generated, an address (7 bits) and a transfer direction (1 bit) are transmitted in one frame.

Simplified I <sup>2</sup> C	IIC00	IIC01	IIC10	IIC11	IIC20	IIC21	IIC30	IIC31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCL00, SDA00 Note 1	SCL01, SDA01 Note 1	SCL10, SDA10 Note 1	SCL11, SDA11 Note 1	SCL20, SDA20 Note 1	SCL21, SDA21 Note 1	SCL30, SDA30 Note 1	SCL31, SDA31 Note 1
Interrupt	INTIIC00	INTIIC01	INTIIC10	INTIIC11	INTIIC20	INTIIC21	INTIIC30	INTIIC31
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)							
Error detection flag	ACK error detection flag (PEFmn)							
Transfer data length	8 bits (transmitted with specifying the higher 7 bits as address and the least significant bit as R/W control)							
Transfer rate Note 2	Max. fmck/4 [Hz] (SDRmn[15:9] = 1 or more) fmck: Operation clock frequency of target channel However, the following condition must be satisfied in each mode of I <sup>2</sup> C. <ul style="list-style-type: none"><li>• Max. 1 MHz (fast mode plus)</li><li>• Max. 400 kHz (fast mode)</li><li>• Max. 100 kHz (standard mode)</li></ul>							
Data level	Non-reversed output (default: high level)							
Parity bit	No parity bit							
Stop bit	Appending 1 bit (for ACK reception timing)							
Data direction	MSB first							

**Note 1.** To perform communication via simplified I<sup>2</sup>C, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) with the port output mode register (POMxx). For details, see **4.3 Registers Controlling Port Function** and **4.5 Register Settings When Using Alternate Function**.

When IIC00, IIC10, IIC20, IIC30, IIC31 is communicating with an external device with a different potential, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) also for the clock input/output pins (SCL00, SCL10, SCL20, SCL30, SCL31).

For details, see **4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers**.

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## (1) Register setting

**Figure 17 - 136 Example of Contents of Registers for Address Field Transmission of Simplified I<sup>2</sup>C  
(IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31)**

## (a) Serial mode register mn (SMRmn)

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 0	0	0	0	0	0	STSmn 0 Note 1	0	SISmn0 0 Note 1	1	0	0	MDmn2 1	MDmn1 0	MDmn0 0

Operation clock (f<sub>mck</sub>) of channel n  
0: Prescaler output clock CKm0 set by the SPSm register  
1: Prescaler output clock CKm1 set by the SPSm register

Operation mode of channel n  
0: Transfer end interrupt

## (b) Serial communication operation setting register mn (SCRmn)

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 1	RXEmn 0	DAPmn 0	CKPmn 0	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0	0	SLCmn1 0 Note 2	SLCmn0 1	0	1	DLSmn1 1 Note 3	DLSmn0 1

Setting of parity bit      |  
00B: No parity      |  
Setting of stop bit  
01B: Appending 1 bit (ACK)

## (c) Serial data register mn (SDRmn) (lower 8 bits: SIOR)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Baud rate setting										Transmit data setting (address + R/W)					

SIOr

## (d) Serial output register m (SOM)

SOM	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 0/1	CKOm2 0/1	CKOm1 0/1	CKOm0 0/1	0	0	0	0	SOm3 0/1	SOm2 0/1	SOm1 0/1	SOm0 0/1

Start condition is generated by manipulating the SOMn bit.

## (e) Serial output enable register m (SOEm)

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 0/1	SOEm2 0/1	SOEm1 0/1	SOEm0 0/1

SOEmn = 0 until the start condition is generated, and SOEmn = 1 after generation.

## (f) Serial channel start register m (SSm)... Sets only the bits of the target channel is 1.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 0/1	SSm2 0/1	SSm1 0/1	SSm0 0/1

**Note 1.** Only provided for the SMR00, SMR03, SMR11, and SMR13 registers.

**Note 2.** Only provided for the SCR00, SCR02, SCR10, and SCR12 registers.

**Note 3.** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of 80- to 100-pin products. This bit is fixed to 1 for the other registers.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

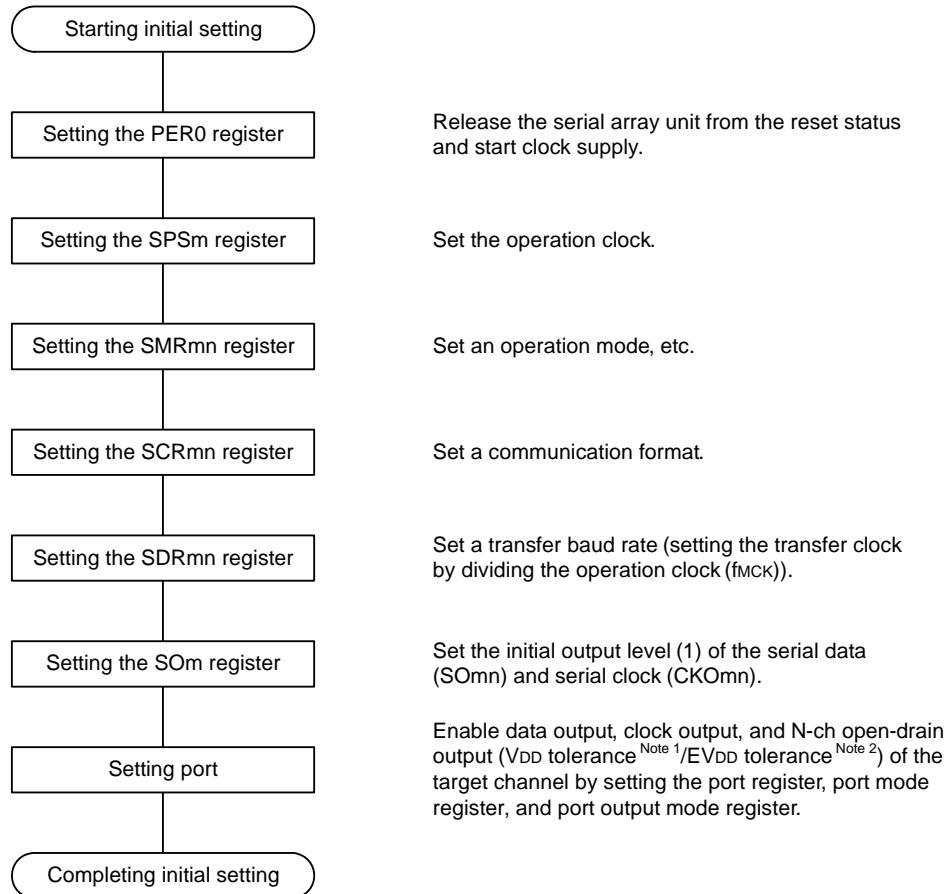
**Remark 2.** : Setting is fixed in the IIC mode,

: Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Operation procedure

**Figure 17 - 137 Initial Setting Procedure for Address Field Transmission**

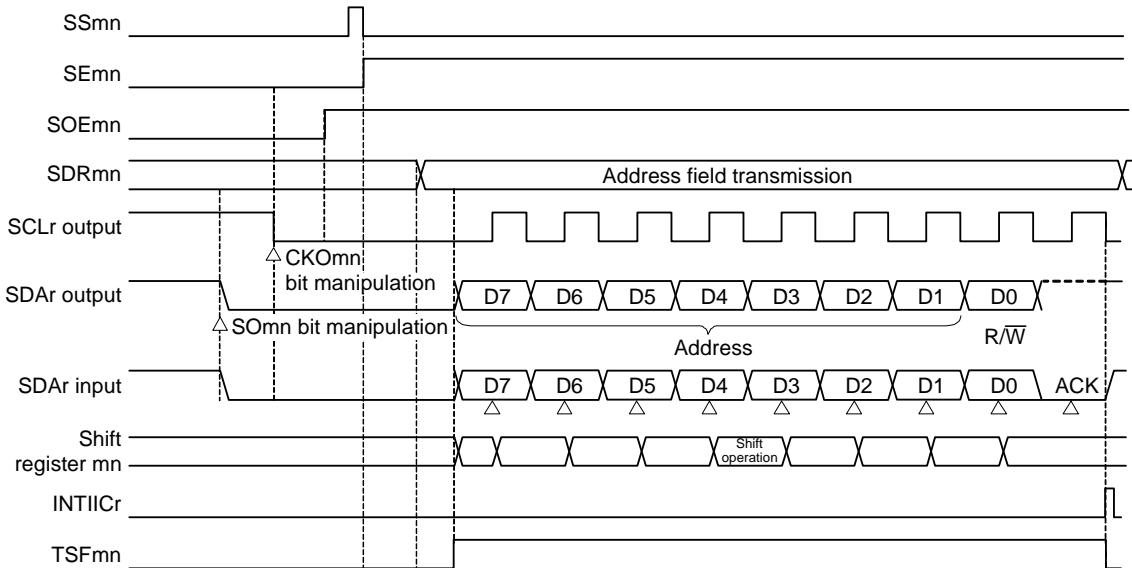
**Note 1.** 30 to 52-pin products

**Note 2.** 64 to 100-pin products

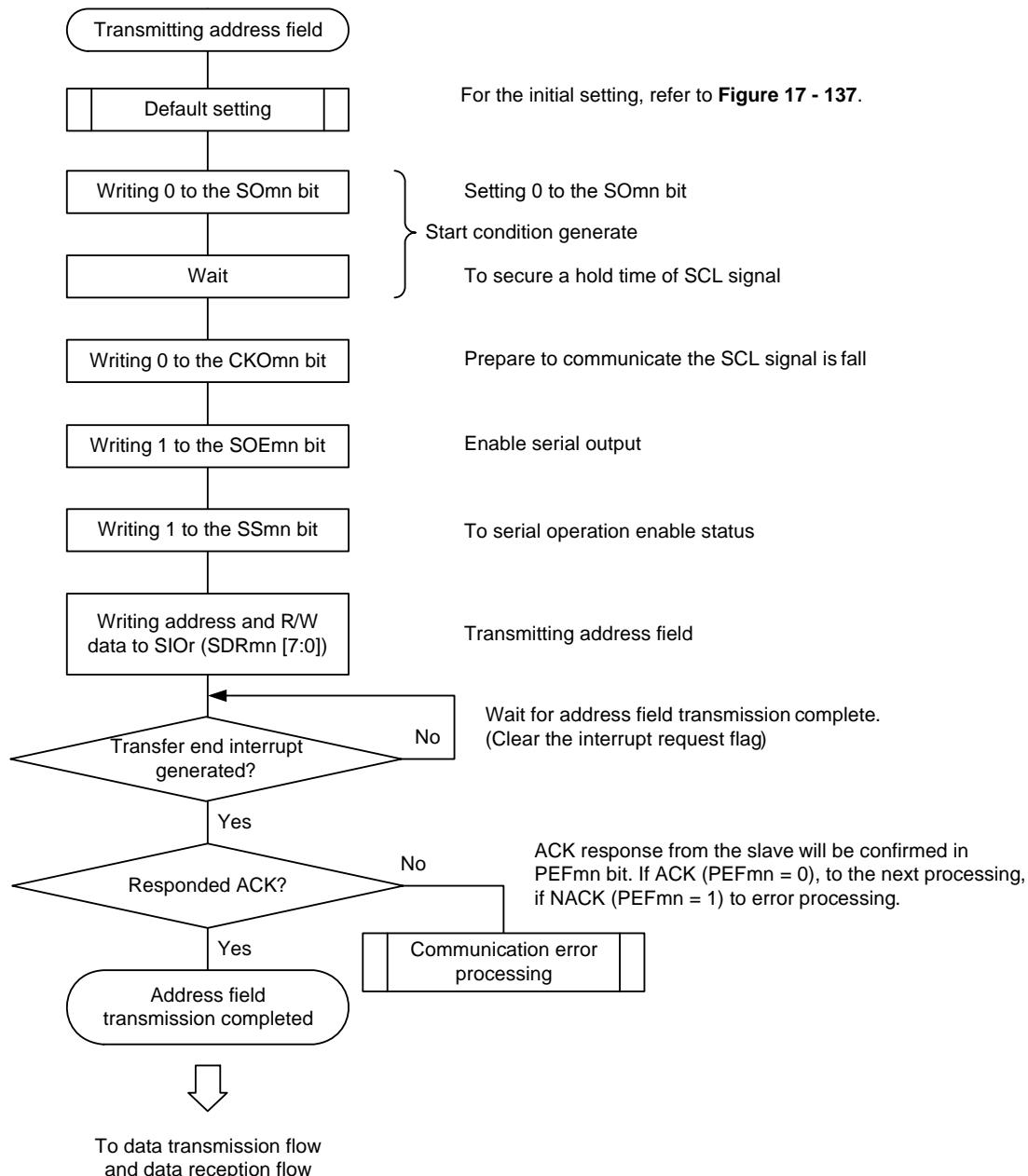
**Remark** At the end of the initial setting, the simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) must be set so that output is disabled and operations are stopped.

## (3) Processing flow

Figure 17 - 138 Timing Chart of Address Field Transmission



**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), r: IIC number ( $r = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

**Figure 17 - 139 Flowchart of Address Field Transmission**

### 17.9.2 Data transmission

Data transmission is an operation to transmit data to the target for transfer (slave) after transmission of an address field. After all data are transmitted to the slave, a stop condition is generated and the bus is released.

Simplified I <sup>2</sup> C	IIC00	IIC01	IIC10	IIC11	IIC20	IIC21	IIC30	IIC31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCL00, SDA00 Note 1	SCL01, SDA01 Note 1	SCL10, SDA10 Note 1	SCL11, SDA11 Note 1	SCL20, SDA20 Note 1	SCL21, SDA21 Note 1	SCL30, SDA30 Note 1	SCL31, SDA31 Note 1
Interrupt	INTIIC00	INTIIC01	INTIIC10	INTIIC11	INTIIC20	INTIIC21	INTIIC30	INTIIC31
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)							
Error detection flag	ACK error flag (PEFmn)							
Transfer data length	8 bits							
Transfer rate Note 2	Max. fmck/4 [Hz] (SDRmn[15:9] = 1 or more) fmck: Operation clock frequency of target channel However, the following condition must be satisfied in each mode of I <sup>2</sup> C. <ul style="list-style-type: none"><li>• Max. 1 MHz (fast mode plus)</li><li>• Max. 400 kHz (fast mode)</li><li>• Max. 100 kHz (standard mode)</li></ul>							
Data level	Non-reverse output (default: high level)							
Parity bit	No parity bit							
Stop bit	Appending 1 bit (for ACK reception timing)							
Data direction	MSB first							

**Note 1.** To perform communication via simplified I<sup>2</sup>C, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) with the port output mode register (POMxx). For details, see **4.3 Registers Controlling Port Function** and **4.5 Register Settings When Using Alternate Function**.

When IIC00, IIC10, IIC20, IIC30, IIC31 is communicating with an external device with a different potential, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) also for the clock input/output pins (SCL00, SCL10, SCL20, SCL30, SCL31).

For details, see **4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers**.

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## (1) Register setting

**Figure 17 - 140 Example of Contents of Registers for Data Transmission of Simplified I<sup>2</sup>C  
(IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31)**

(a) Serial mode register mn (SMRmn)... Do not manipulate this register during data transmission/reception.

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 0	0	0	0	0	0	STSmn 0 Note 1	0	SISmn0 0 Note 1	1	0	0	MDmn2 1	MDmn1 0	MDmn0 0

(b) Serial communication operation setting register mn (SCRmn)... Do not manipulate the bits of this register, except the TXEmn and RXEmn bits, during data transmission/reception.

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 1	RXEmn 0	DAPmn 0	CKPmn 0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0	SLCmn1 0 Note 2	SLCmn0 1	0	1	0	DLSmn1 1 Note 3	DLSmn0 1	

(c) Serial data register mn (SDRmn) (lower 8 bits: SIOR)... During data transmission/reception, valid only lower 8-bits (SIOR).

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	Baud rate setting Note 4								0	Transmit data setting							

SIOR

(d) Serial output register m (SOm)... Do not manipulate this register during data transmission/reception.

SOm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 0/1 Note 5	CKOm2 0/1 Note 5	CKOm1 0/1 Note 5	CKOm0 0/1 Note 5	0	0	0	0	SOm3 0/1 Note 5	SOm2 0/1 Note 5	SOm1 0/1 Note 5	SOm0 0/1 Note 5

(e) Serial output enable register m (SOEm)... Do not manipulate this register during data transmission/reception

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 1	SOEm2 1	SOEm1 1	SOEm0 1

(f) Serial channel start register m (SSm)... Do not manipulate this register during data transmission/reception.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 0/1	SSm2 0/1	SSm1 0/1	SSm0 0/1

**Note 1.** Only provided for the SMR01, SMR03, SMR11, and SMR13 registers.

**Note 2.** Only provided for the SCR00, SCR02, SCR10, and SCR12 registers.

**Note 3.** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Note 4.** Because the setting is completed by address field transmission, setting is not required.

**Note 5.** The value varies depending on the communication data during communication operation.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

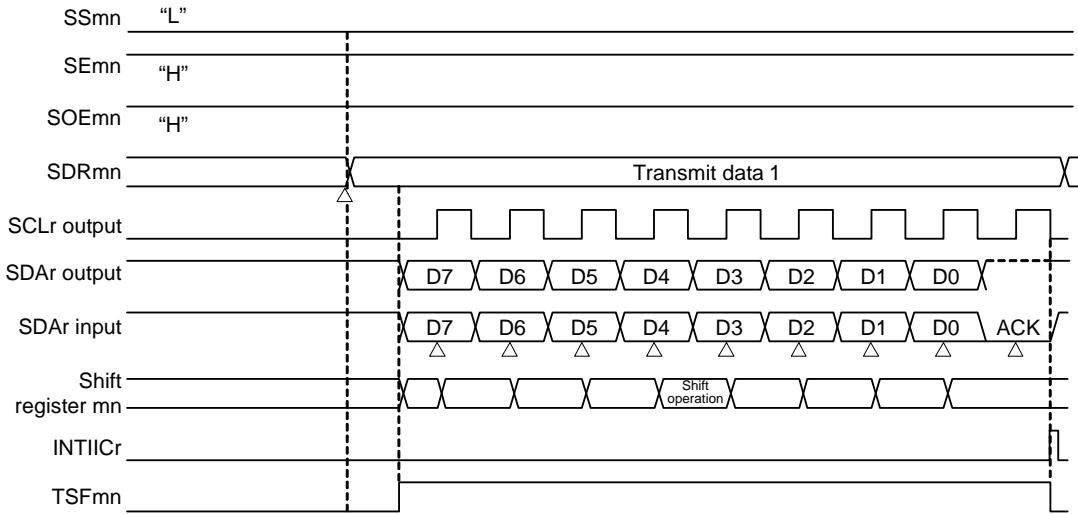
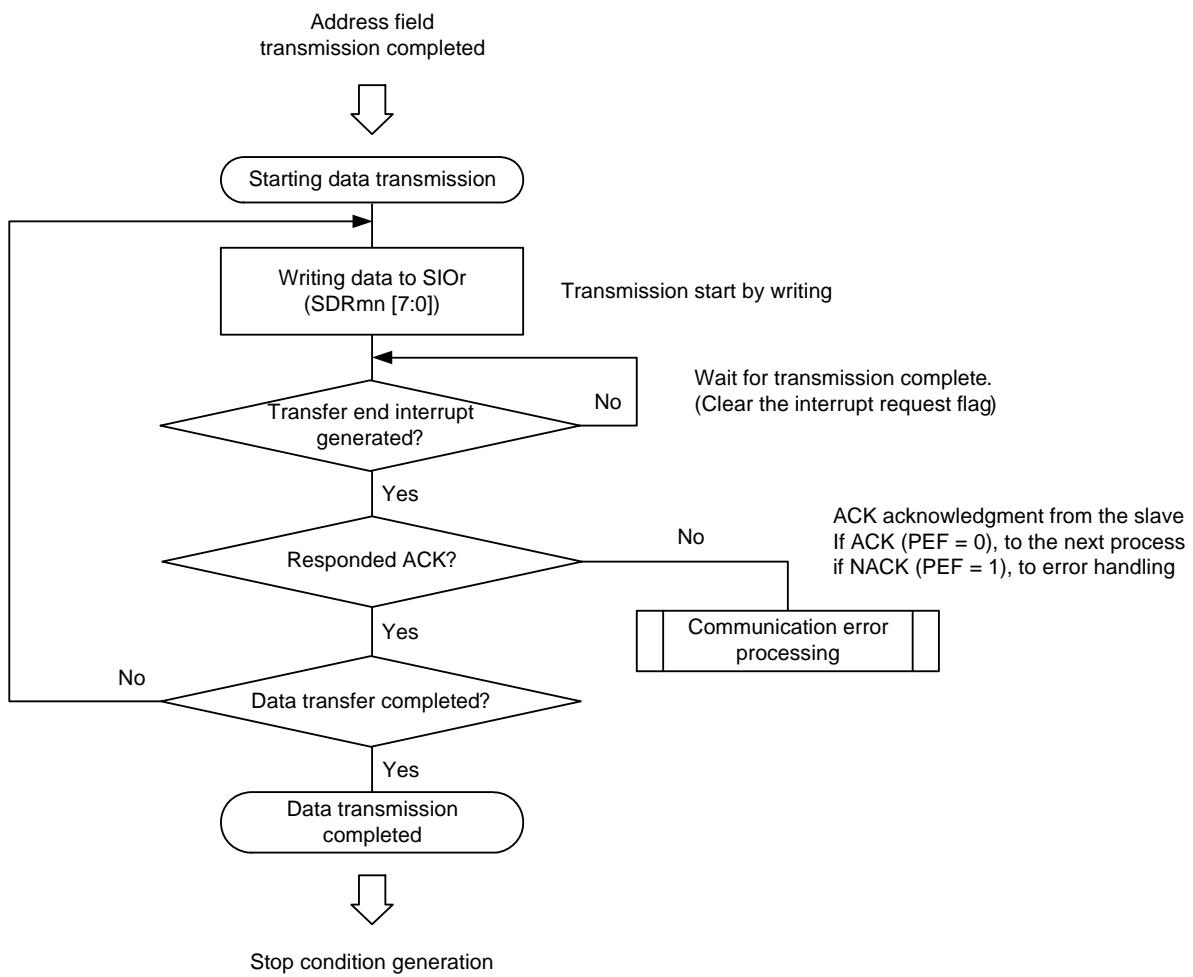
**Remark 2.**  Setting is fixed in the IIC mode,

Setting disabled (set to the initial value)

x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

## (2) Processing flow

**Figure 17 - 141 Timing Chart of Data Transmission****Figure 17 - 142 Flowchart of Data Transmission**

### 17.9.3 Data reception

Data reception is an operation to receive data to the target for transfer (slave) after transmission of an address field. After all data are received to the slave, a stop condition is generated and the bus is released.

Simplified I <sup>2</sup> C	IIC00	IIC01	IIC10	IIC11	IIC20	IIC21	IIC30	IIC31
Target channel	Channel 0 of SAU0	Channel 1 of SAU0	Channel 2 of SAU0	Channel 3 of SAU0	Channel 0 of SAU1	Channel 1 of SAU1	Channel 2 of SAU1	Channel 3 of SAU1
Pins used	SCL00, SDA00 Note 1	SCL01, SDA01 Note 1	SCL10, SDA10 Note 1	SCL11, SDA11 Note 1	SCL20, SDA20 Note 1	SCL21, SDA21 Note 1	SCL30, SDA30 Note 1	SCL31, SDA31 Note 1
Interrupt	INTIIC00	INTIIC01	INTIIC10	INTIIC11	INTIIC20	INTIIC21	INTIIC30	INTIIC31
	Transfer end interrupt only (Setting the buffer empty interrupt is prohibited.)							
Error detection flag	Overrun error detection flag (OVFmn) only							
Transfer data length	8 bits							
Transfer rate Note 2	Max. fmck/4 [Hz] (SDRmn[15:9] = 1 or more) fmck: Operation clock frequency of target channel However, the following condition must be satisfied in each mode of I <sup>2</sup> C. <ul style="list-style-type: none"><li>• Max. 1 MHz (fast mode plus)</li><li>• Max. 400 kHz (fast mode)</li><li>• Max. 100 kHz (standard mode)</li></ul>							
Data level	Non-reverse output (default: high level)							
Parity bit	No parity bit							
Stop bit	Appending 1 bit (ACK transmission)							
Data direction	MSB first							

**Note 1.** To perform communication via simplified I<sup>2</sup>C, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) with the port output mode register (POMxx). For details, see **4.3 Registers Controlling Port Function** and **4.5 Register Settings When Using Alternate Function**.

When IIC00, IIC10, IIC20, IIC30, IIC31 is communicating with an external device with a different potential, set the N-ch open-drain output (V<sub>DD</sub> tolerance (30 to 52-pin products)/EV<sub>DD</sub> tolerance (64 to 100-pin products)) mode (POMxx = 1) also for the clock input/output pins (SCL00, SCL10, SCL20, SCL30, SCL31).

For details, see **4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers**.

**Note 2.** Use this operation within a range that satisfies the conditions above and the peripheral functions characteristics in the electrical specifications (see **CHAPTER 34** or **CHAPTER 35 ELECTRICAL SPECIFICATIONS**).

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

## (1) Register setting

**Figure 17 - 143 Example of Contents of Registers for Data Reception of Simplified I<sup>2</sup>C  
(IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31)**

(a) Serial mode register mn (SMRmn)... Do not manipulate this register during data transmission/reception.

SMRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CKSmn 0/1	CCSmn 0	0	0	0	0	0	STSmn 0 Note 1	0	SISmn0 0 Note 1	1	0	0	MDmn2 1	MDmn1 0	MDmn0 0

(b) Serial communication operation setting register mn (SCRmn)... Do not manipulate the bits of this register, except the TXEmn and RXEmn bits, during data transmission/reception.

SCRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXEmn 0	RXEmn 1	DAPmn 0	CKPmn 0	0	EOCmn 0	PTCmn1 0	PTCmn0 0	DIRmn 0	0	SLCmn1 0 Note 2	SLCmn0 1	0	1	DLSmn1 Note 3 1	DLSmn0 1

(c) Serial data register mn (SDRmn) (lower 8 bits: SIOR)

SDRmn	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	Baud rate setting Note 4								0	Dummy transmit data setting (FFH)							

SIOr

(d) Serial output register m (SOM)... Do not manipulate this register during data transmission/reception.

SOM	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	CKOm3 0/1 Note 5	CKOm2 0/1 Note 5	CKOm1 0/1 Note 5	CKOm0 0/1 Note 5	0	0	0	0	SOM3 0/1 Note 5	SOM2 0/1 Note 5	SOM1 0/1 Note 5	SOM0 0/1 Note 5

(e) Serial output enable register m (SOEm)... Do not manipulate this register during data transmission/reception.

SOEm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SOEm3 0/1	SOEm2 0/1	SOEm1 0/1	SOEm0 0/1

(f) Serial channel start register m (SSm)... Do not manipulate this register during data transmission/reception.

SSm	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	SSm3 0/1	SSm2 0/1	SSm1 0/1	SSm0 0/1

**Note 1.** Only provided for the SMR01, SMR03, SMR11, and SMR13 registers.

**Note 2.** Only provided for the SCR00, SCR02, SCR10, and SCR12 registers.

**Note 3.** Only provided for the SCR00 and SCR01 registers and the SCR10 and SCR11 registers of an 80- to 100-pin product. This bit is fixed to 1 for the other registers.

**Note 4.** The baud rate setting is not required because the baud rate has already been set when the address field was transmitted.

**Note 5.** The value varies depending on the communication data during communication operation.

**Remark 1.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)  
mn = 00 to 03, 10 to 13

**Remark 2.** : Setting is fixed in the IIC mode,

: Setting disabled (set to the initial value)

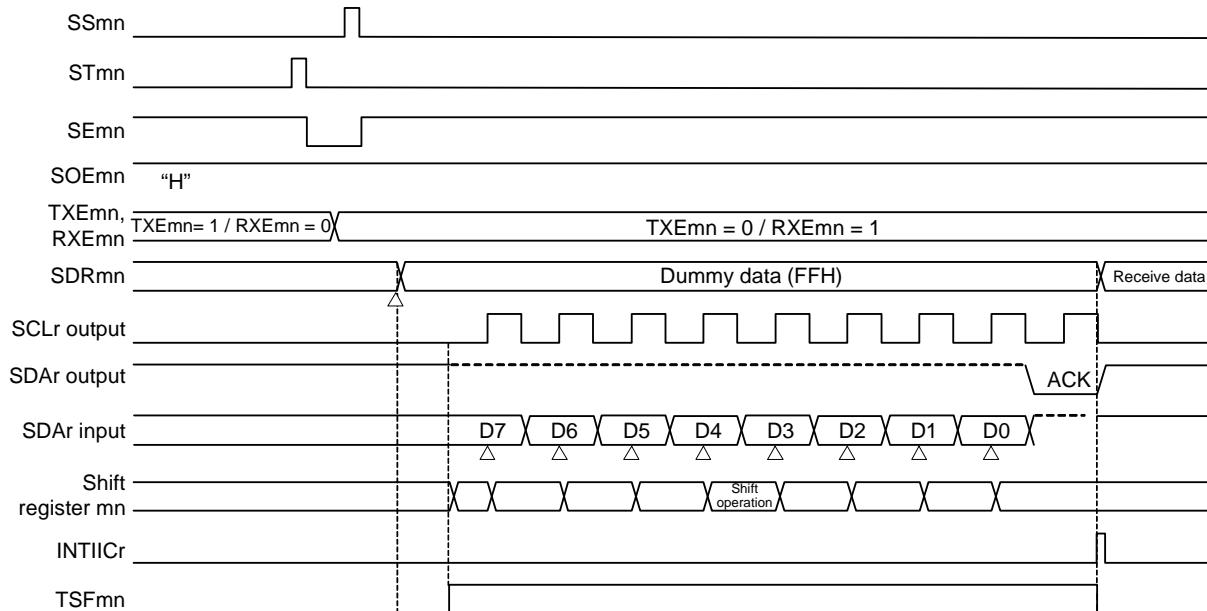
x: Bit that cannot be used in this mode (set to the initial value when not used in any mode)

0/1: Set to 0 or 1 depending on the usage of the user

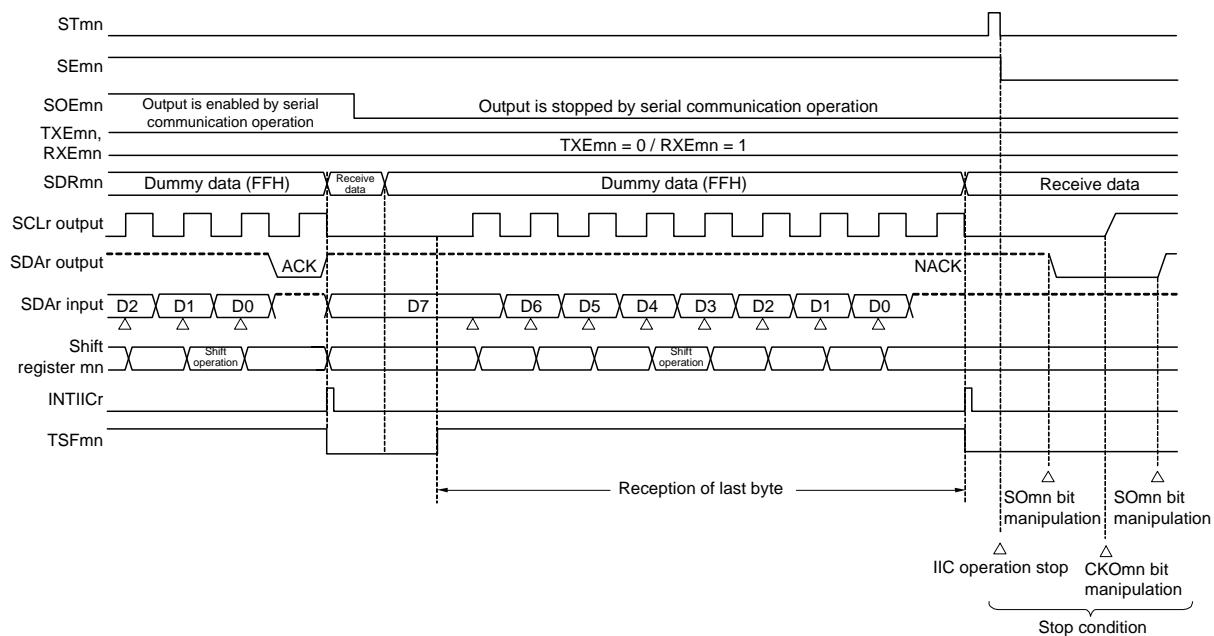
## (2) Processing flow

Figure 17 - 144 Timing Chart of Data Reception

## (a) When starting data reception

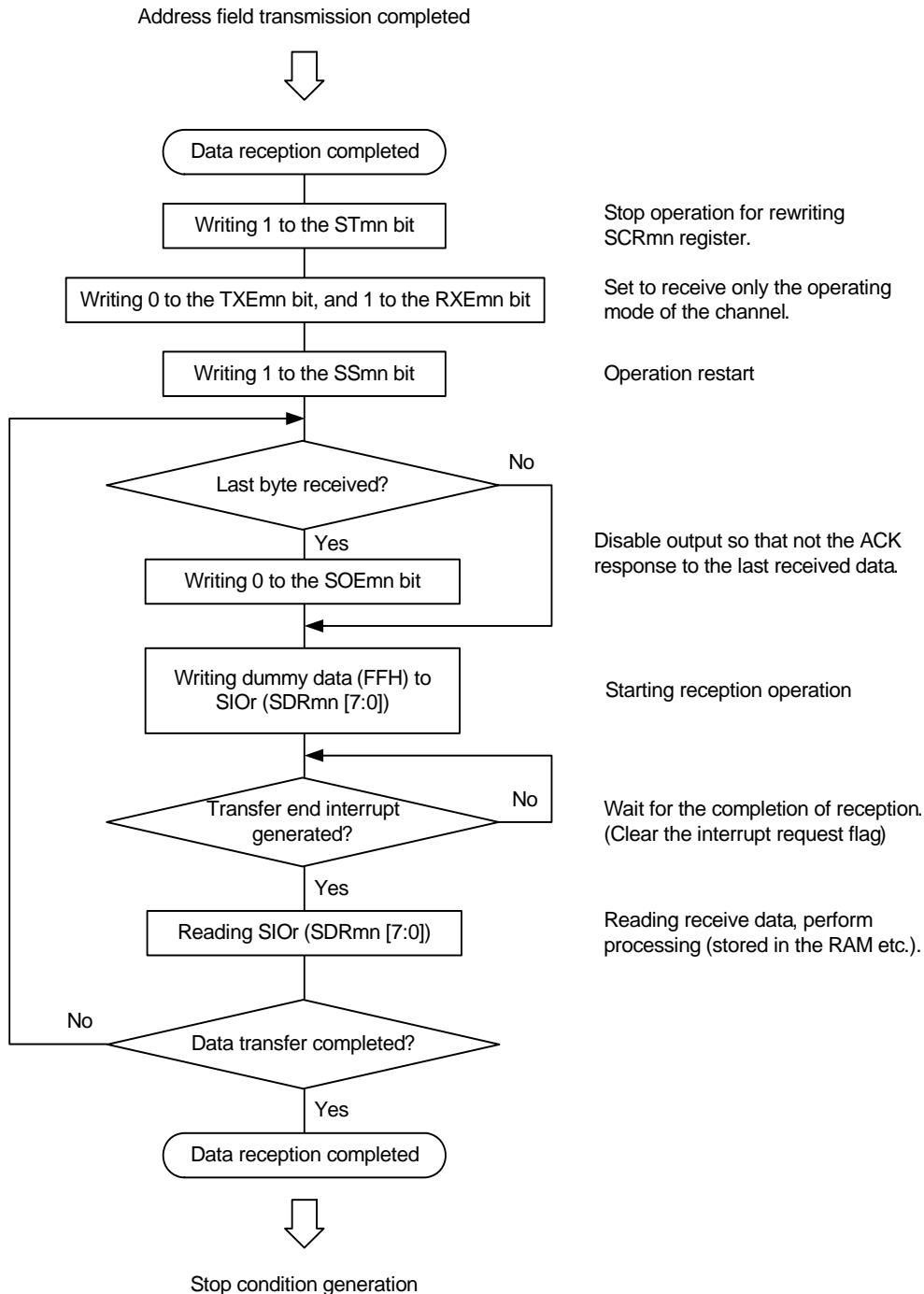


## (b) When receiving last data



**Remark** m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), r: IIC number ( $r = 00, 01, 10, 11, 20, 21, 30, 31$ )  
 $mn = 00$  to  $03$ ,  $10$  to  $13$

Figure 17 - 145 Flowchart of Data Reception



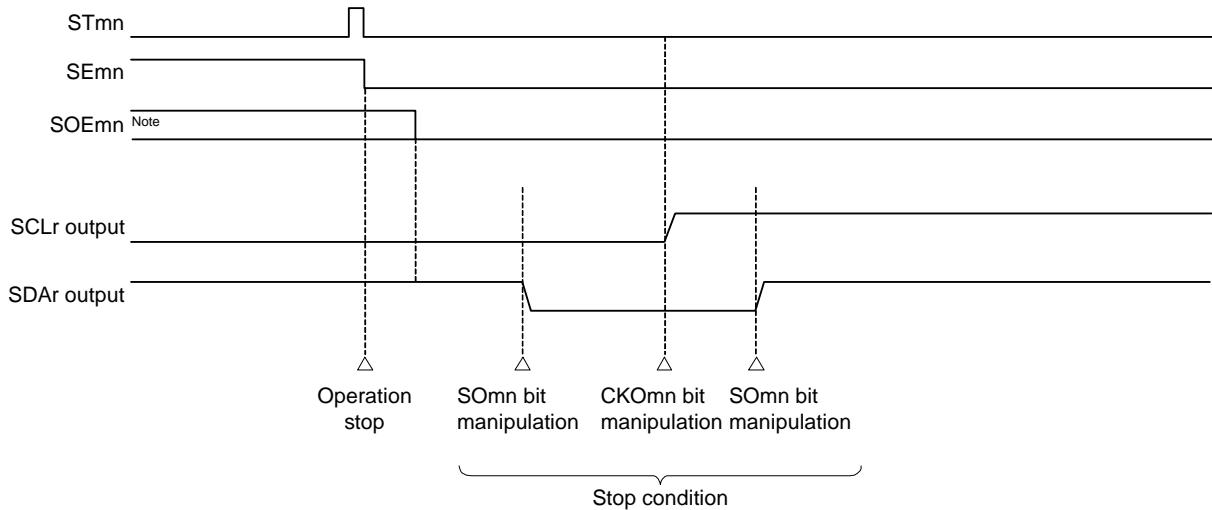
**Caution** ACK is not output when the last data is received (NACK). Communication is then completed by setting “1” to the STmn bit of serial channel stop register m (STm) to stop operation and generating a stop condition.

### 17.9.4 Stop condition generation

After all data are transmitted to or received from the target slave, a stop condition is generated and the bus is released.

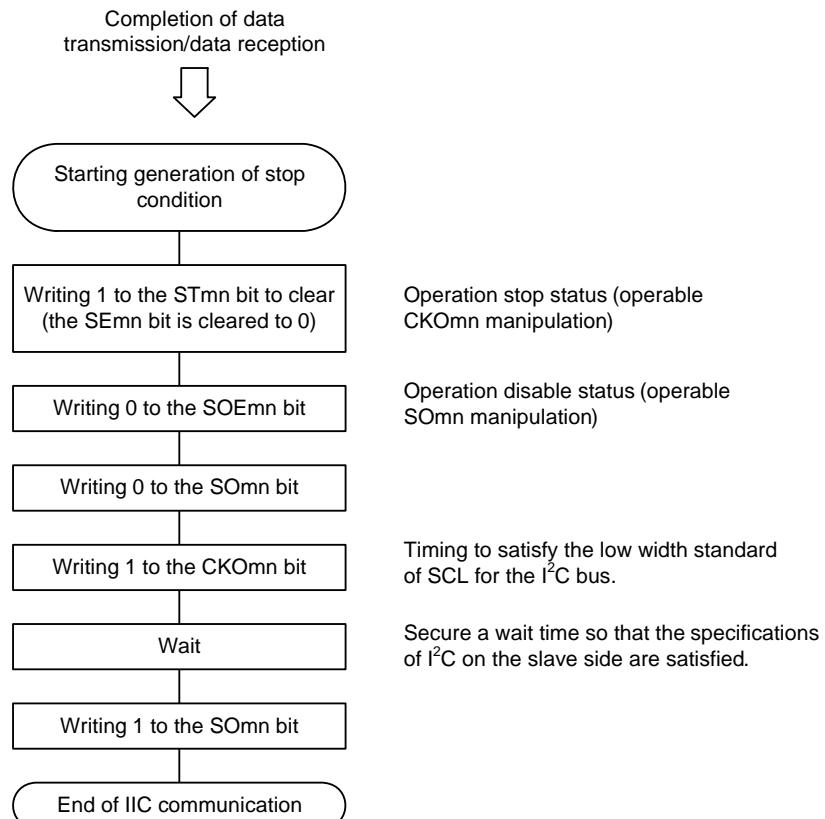
(1) Processing flow

**Figure 17 - 146 Timing Chart of Stop Condition Generation**



**Note** During a receive operation, the SOEmn bit of serial output enable register m (SOEm) is cleared to 0 before receiving the last data.

**Figure 17 - 147 Flowchart of Stop Condition Generation**



### 17.9.5 Calculating transfer rate

The transfer rate for simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) communication can be calculated by the following expressions.

$$\text{(Transfer rate)} = \{\text{Operation clock (fmck) frequency of target channel}\} \div (\text{SDRmn[15:9]} + 1) \div 2$$

**Caution** SDRmn[15:9] must not be set to 0000000B. Be sure to set a value of 0000001B or greater for SDRmn[15:9]. The duty ratio of the SCL signal output by the simplified I<sup>2</sup>C is 50%. The I<sup>2</sup>C bus specifications define that the low-level width of the SCL signal is longer than the high-level width. If 400 kbps (fast mode) or 1 Mbps (fast mode plus) is specified, therefore, the low-level width of the SCL output signal becomes shorter than the value specified in the I<sup>2</sup>C bus specifications. Make sure that the SDRmn[15:9] value satisfies the I<sup>2</sup>C bus specifications.

**Remark 1.** The value of SDRmn[15:9] is the value of bits 15 to 9 of the SDRmn register (0000001B to 1111111B) and therefore is 1 to 127.

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

The operation clock (fmck) is determined by serial clock select register m (SPSm) and bit 15 (CKS<sub>mn</sub>) of serial mode register mn (SMR<sub>mn</sub>).

**Table 17 - 6 Selection of Operation Clock For Simplified I<sup>2</sup>C**

SMR <sub>mnn</sub> Register	SPSm Register								Operation Clock (f <sub>MCK</sub> ) Note	
	CKS <sub>mnn</sub>	PRS m13	PRS m12	PRS m11	PRS m10	PRS m03	PRS m02	PRS m01	PRS m00	
0	x	x	x	x	0	0	0	0	f <sub>CLK</sub>	32 MHz
	x	x	x	x	0	0	0	1	f <sub>CLK</sub> /2	16 MHz
	x	x	x	x	0	0	1	0	f <sub>CLK</sub> /2 <sup>2</sup>	8 MHz
	x	x	x	x	0	0	1	1	f <sub>CLK</sub> /2 <sup>3</sup>	4 MHz
	x	x	x	x	0	1	0	0	f <sub>CLK</sub> /2 <sup>4</sup>	2 MHz
	x	x	x	x	0	1	0	1	f <sub>CLK</sub> /2 <sup>5</sup>	1 MHz
	x	x	x	x	0	1	1	0	f <sub>CLK</sub> /2 <sup>6</sup>	500 kHz
	x	x	x	x	0	1	1	1	f <sub>CLK</sub> /2 <sup>7</sup>	250 kHz
	x	x	x	x	1	0	0	0	f <sub>CLK</sub> /2 <sup>8</sup>	125 kHz
	x	x	x	x	1	0	0	1	f <sub>CLK</sub> /2 <sup>9</sup>	62.5 kHz
	x	x	x	x	1	0	1	0	f <sub>CLK</sub> /2 <sup>10</sup>	31.25 kHz
	x	x	x	x	1	0	1	1	f <sub>CLK</sub> /2 <sup>11</sup>	15.63 kHz
1	0	0	0	0	x	x	x	x	f <sub>CLK</sub>	32 MHz
	0	0	0	1	x	x	x	x	f <sub>CLK</sub> /2	16 MHz
	0	0	1	0	x	x	x	x	f <sub>CLK</sub> /2 <sup>2</sup>	8 MHz
	0	0	1	1	x	x	x	x	f <sub>CLK</sub> /2 <sup>3</sup>	4 MHz
	0	1	0	0	x	x	x	x	f <sub>CLK</sub> /2 <sup>4</sup>	2 MHz
	0	1	0	1	x	x	x	x	f <sub>CLK</sub> /2 <sup>5</sup>	1 MHz
	0	1	1	0	x	x	x	x	f <sub>CLK</sub> /2 <sup>6</sup>	500 kHz
	0	1	1	1	x	x	x	x	f <sub>CLK</sub> /2 <sup>7</sup>	250 kHz
	1	0	0	0	x	x	x	x	f <sub>CLK</sub> /2 <sup>8</sup>	125 kHz
	1	0	0	1	x	x	x	x	f <sub>CLK</sub> /2 <sup>9</sup>	62.5 kHz
	1	0	1	0	x	x	x	x	f <sub>CLK</sub> /2 <sup>10</sup>	31.25 kHz
	1	0	1	1	x	x	x	x	f <sub>CLK</sub> /2 <sup>11</sup>	15.63 kHz
Other than above									Setting prohibited	

**Note** When changing the clock selected for f<sub>CLK</sub> (by changing the system clock control register (CKC) value), do so after having stopped (serial channel stop register m (STm) = 000FH) the operation of the serial array unit (SAU).

**Remark 1.** x: Don't care

**Remark 2.** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), mn = 00 to 03, 10 to 13

Here is an example of setting an I<sup>2</sup>C transfer rate where fMCK = fCLK = 32 MHz.

I <sup>2</sup> C Transfer Mode (Desired Transfer Rate)	fCLK = 32 MHz			
	Operation Clock (fMCK)	SDRmn[15:9]	Calculated Transfer Rate	Error from Desired Transfer Rate
100 kHz	fCLK/2	79	100 kHz	0.0%
400 kHz	fCLK	41	380 kHz	5.0% Note
1 MHz	fCLK	18	0.84 MHz	16.0% Note

**Note** The error cannot be set to about 0% because the duty ratio of the SCL signal is 50%.

### 17.9.6 Procedure for processing errors that occurred during simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) communication

The procedure for processing errors that occurred during simplified I<sup>2</sup>C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) communication is described in **Figures 17 - 148** and **17 - 149**.

**Figure 17 - 148 Processing Procedure in Case of Overrun Error**

Software Manipulation	Hardware Status	Remark
Reads serial data register mn (SDRmn).	→ The BFFmn bit of the SSRmn register is set to 0 and channel n is enabled to receive data.	This is to prevent an overrun error if the next reception is completed during error processing.
Reads serial status register mn (SSRmn).		The error type is identified and the read value is used to clear the error flag.
Writes 1 to serial flag clear trigger register mn (SIRmn).	→ The error flag is cleared.	The error only during reading can be cleared, by writing the value read from the SSRmn register to the SIRmn register without modification.

**Figure 17 - 149 Processing Procedure in Case of ACK error in Simplified I<sup>2</sup>C Mode**

Software Manipulation	Hardware Status	Remark
Reads serial status register mn (SSRmn).		Error type is identified and the read value is used to clear error flag.
Writes serial flag clear trigger register mn (SIRmn).	→ Error flag is cleared.	Error can be cleared only during reading, by writing the value read from the SSRmn register to the SIRmn register without modification.
Sets the STmn bit of serial channel stop register m (STm) to 1.	→ The SEMn bit of serial channel enable status register m (SEm) is set to 0 and channel n stops operation.	Slave is not ready for reception because ACK is not returned. Therefore, a stop condition is created, the bus is released, and communication is started again from the start condition. Or, a restart condition is generated and transmission can be redone from address transmission.
Creates stop condition.		
Creates start condition.		
Sets the SSmn bit of serial channel start register m (SSm) to 1.	→ The SEMn bit of serial channel enable status register m (SEm) is set to 1 and channel n is enabled to operate.	

**Remark** m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), r: IIC number (r = 00, 01, 10, 11, 20, 21, 30, 31)  
 mn = 00 to 03, 10 to 13

## CHAPTER 18 SERIAL INTERFACE IICA

The number of channels of the serial Interface IICA differs, depending on the product.

	30, 32, 36, 40, 44, 48, 52, 56, 64-pin	80, 100-pin
Channels	1 ch	2 ch

### 18.1 Functions of Serial Interface IICA

Serial interface IICA has the following three modes.

(1) Operation stop mode

This mode is used when serial transfers are not performed. It can therefore be used to reduce power consumption.

(2) I<sup>2</sup>C bus mode (multimaster supported)

This mode is used for 8-bit data transfers with several devices via two lines: a serial clock (SCLAn) line and a serial data bus (SDAAn) line.

This mode complies with the I<sup>2</sup>C bus format and the master device can generate “start condition”, “address”, “transfer direction specification”, “data”, and “stop condition” data to the slave device, via the serial data bus. The slave device automatically detects these received status and data by hardware. This function can simplify the part of application program that controls the I<sup>2</sup>C bus.

Since the SCLAn and SDAAn pins are used for open drain outputs, serial interface IICA requires pull-up resistors for the serial clock line and the serial data bus line.

(3) Wakeup mode

The STOP mode can be released by generating an interrupt request signal (INTIICAn) when an extension code from the master device or a local address has been received while in STOP mode. This can be set by using the WUPn bit of IICA control register n1 (IICCTLn1).

Figure 18 - 1 shows a block diagram of serial interface IICA

**Remark** n = 0, 1

Figure 18 - 1 Block Diagram of Serial Interface IICA

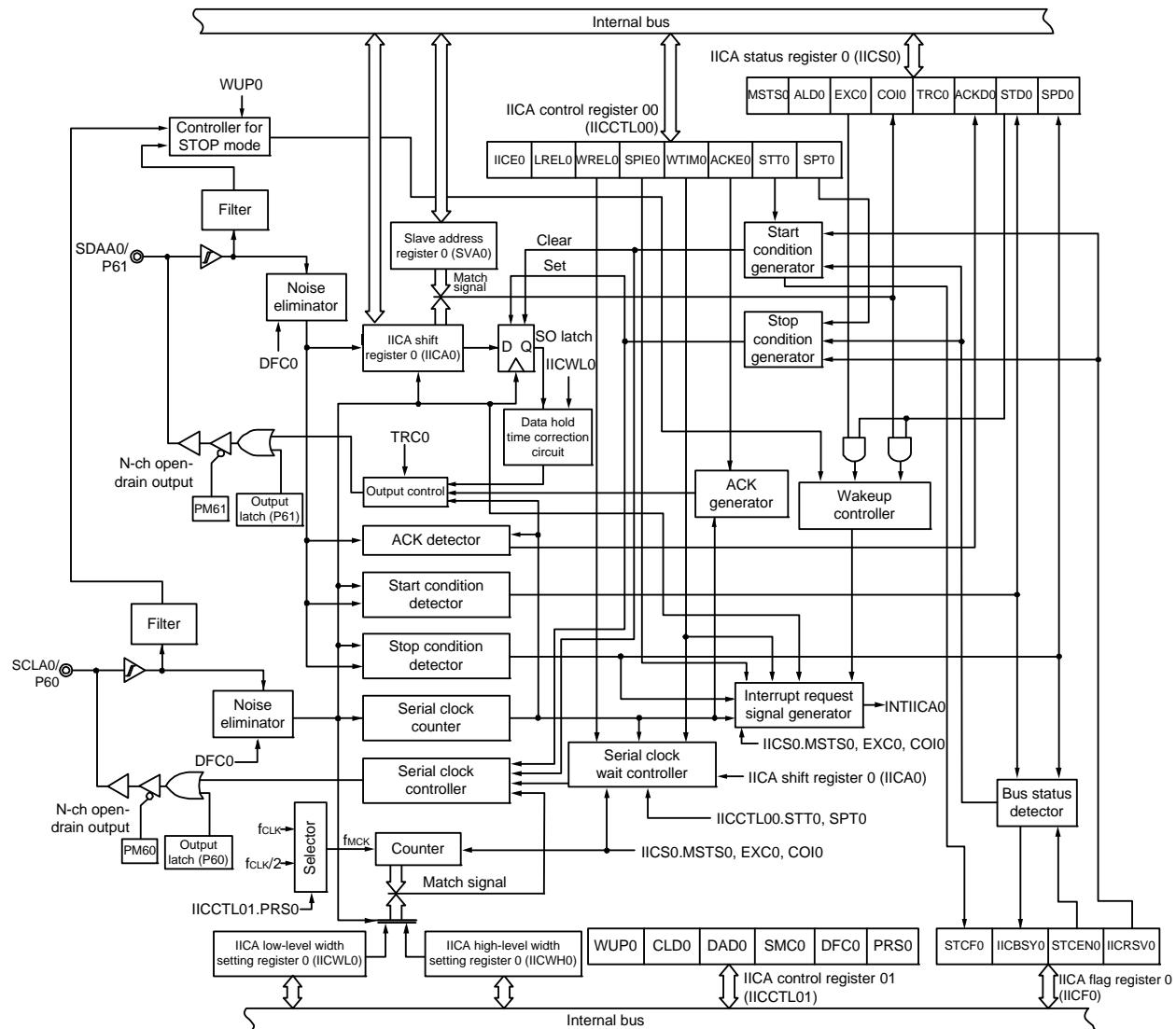
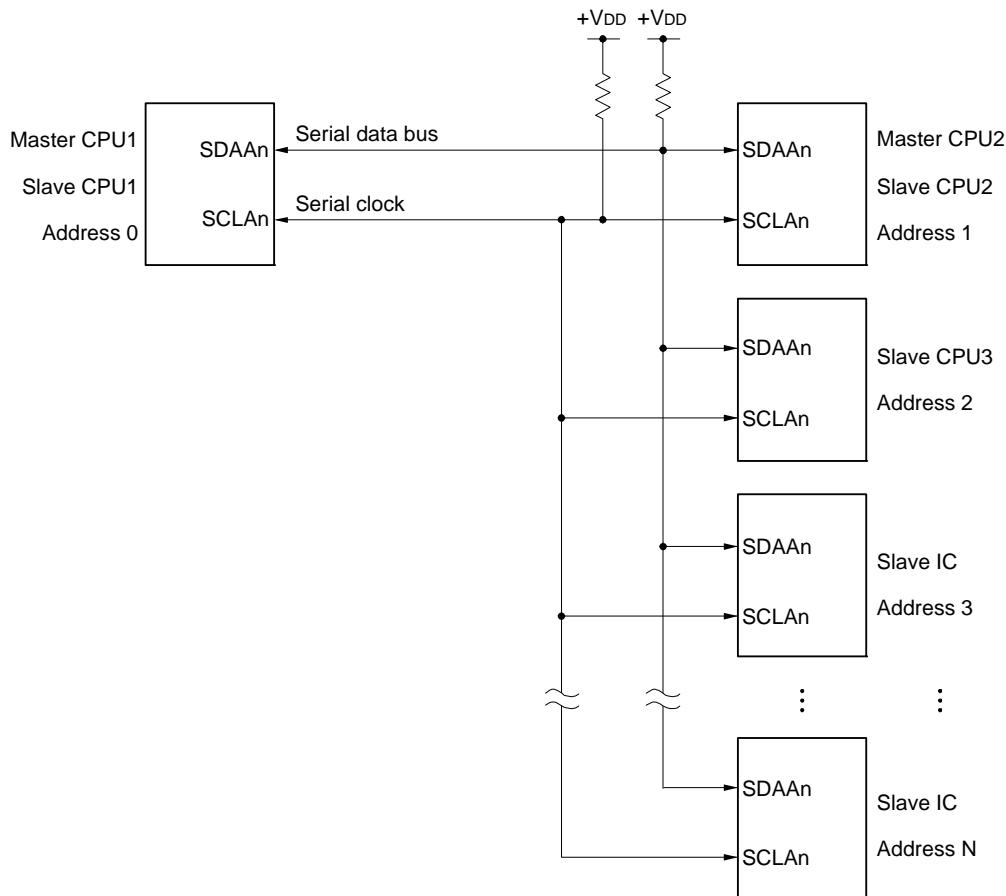


Figure 18 - 2 shows a serial bus configuration example.

**Figure 18 - 2 Serial Bus Configuration Example Using I<sup>2</sup>C Bus**



**Remark** n = 0, 1

## 18.2 Configuration of Serial Interface IICA

Serial interface IICA includes the following hardware.

**Table 18 - 1 Configuration of Serial Interface IICA**

Item	Configuration
Registers	IICA shift register n (IICAn) Slave address register n (SVAn)
Control registers	Peripheral enable register 0 (PER0) IICA control register n0 (IICCTLn0) IICA status register n (IICSn) IICA flag register n (IICFn) IICA control register n1 (IICCTLn1) IICA low-level width setting register n (IICWLn) IICA high-level width setting register n (IICWHn) Port mode register 6 (PM6) Port register 6 (P6)

**Remark** n = 0, 1

(1) IICA shift register n (IICAn)

The IICAn register is used to convert 8-bit serial data to 8-bit parallel data and vice versa in synchronization with the serial clock. The IICAn register can be used for both transmission and reception.

The actual transmit and receive operations can be controlled by writing and reading operations to the IICAn register.

Cancel the wait state and start data transfer by writing data to the IICAn register during the wait period.

The IICAn register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears IICAn to 00H.

**Figure 18 - 3 Format of IICA shift register n (IICAn)**

Address: FFF50H (IICA0), FFF54H (IICA1)								After reset: 00H	R/W
Symbol	7	6	5	4	3	2	1	0	
IICAn									

**Caution 1. Do not write data to the IICAn register during data transfer.**

**Caution 2. Write or read the IICAn register only during the wait period. Accessing the IICAn register in a communication state other than during the wait period is prohibited. When the device serves as the master, however, the IICAn register can be written only once after the communication trigger bit (STTn) is set to 1.**

**Caution 3. When communication is reserved, write data to the IICAn register after the interrupt triggered by a stop condition is detected.**

**Remark** n = 0, 1

(2) Slave address register n (SVA<sub>n</sub>)

This register stores seven bits of local addresses {A6, A5, A4, A3, A2, A1, A0} when in slave mode.

The SVA<sub>n</sub> register can be set by an 8-bit memory manipulation instruction.

However, rewriting to this register is prohibited while STD<sub>n</sub> = 1 (while the start condition is detected).

Reset signal generation clears the SVA<sub>n</sub> register to 00H.

**Figure 18 - 4 Format of Slave address register n (SVA<sub>n</sub>)**

Address: F0234H (SVA0), F023CH (SVA1)

After reset: 00H

R/W

Symbol	7	6	5	4	3	2	1	0
SVA <sub>n</sub>	A6	A5	A4	A3	A2	A1	A0	0 Note

**Note** Bit 0 is fixed to 0.

(3) SO latch

The SO latch is used to retain the SDAAn pin's output level.

(4) Wakeup controller

This circuit generates an interrupt request (INTIICAn) when the address received by this register matches the address value set to the slave address register n (SVA<sub>n</sub>) or when an extension code is received.

(5) Serial clock counter

This counter counts the serial clocks that are output or input during transmit/receive operations and is used to verify that 8-bit data was transmitted or received.

(6) Interrupt request signal generator

This circuit controls the generation of interrupt request signals (INTIICAn).

An I<sup>2</sup>C interrupt request is generated by the following two triggers.

- Falling edge of eighth or ninth clock of the serial clock (set by the WTIM<sub>n</sub> bit)
- Interrupt request generated when a stop condition is detected (set by the SPIEn bit)

**Remark** WTIM<sub>n</sub> bit: Bit 3 of IICA control register n0 (IICCTLn0)

SPIEn bit: Bit 4 of IICA control register n0 (IICCTLn0)

(7) Serial clock controller

In master mode, this circuit generates the clock output via the SCLAn pin from a sampling clock.

(8) Serial clock wait controller

This circuit controls the wait timing.

**Remark** n = 0, 1

## (9) ACK generator, stop condition detector, start condition detector, and ACK detector

These circuits generate and detect each status.

## (10) Data hold time correction circuit

This circuit generates the hold time for data corresponding to the falling edge of the serial clock.

## (11) Start condition generator

This circuit generates a start condition when the STTn bit is set to 1.

However, in the communication reservation disabled status (IICRSVn bit = 1), when the bus is not released (IICBSYn bit = 1), start condition requests are ignored and the STCFn bit is set to 1.

## (12) Stop condition generator

This circuit generates a stop condition when the SPTn bit is set to 1.

## (13) Bus status detector

This circuit detects whether or not the bus is released by detecting start conditions and stop conditions.

However, as the bus status cannot be detected immediately following operation, the initial status is set by the STCENn bit.

**Remark 1.** STTn bit: Bit 1 of IICA control register n0 (IICCTLn0)

SPTn bit: Bit 0 of IICA control register n0 (IICCTLn0)

IICRSVn bit: Bit 0 of IICA flag register n (IICFn)

IICBSYn bit: Bit 6 of IICA flag register n (IICFn)

STCFn bit: Bit 7 of IICA flag register n (IICFn)

STCENn bit: Bit 1 of IICA flag register n (IICFn)

**Remark 2.** n = 0, 1

### 18.3 Registers Controlling Serial Interface IICA

Serial interface IICA is controlled by the following eight registers.

- Peripheral enable register 0 (PER0)
- IICA control register n0 (IICCTLn0)
- IICA flag register n (IICFn)
- IICA status register n (IICSn)
- IICA control register n1 (IICCTLn1)
- IICA low-level width setting register n (IICWLn)
- IICA high-level width setting register n (IICWHn)
- Port mode register 6 (PM6)
- Port register 6 (P6)

**Remark** n = 0, 1

### 18.3.1 Peripheral enable register 0 (PER0)

This register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to a hardware macro that is not used is stopped in order to reduce the power consumption and noise.

When serial interface IICAn is used, be sure to set bits 6, 4 (IICA1EN, IICA0EN) of this register to 1.

The PER0 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 18 - 5 Format of Peripheral enable register 0 (PER0)**

Address: F00F0H      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

PER0	RTCWEN	IICA1EN Note	ADCEN	IICA0EN	SAU1EN	SAU0EN	TAU1EN Note	TAU0EN
IICAnEN	Control of serial interface IICAn input clock supply							
0	Stops input clock supply. • SFR used by serial interface IICAn cannot be written. • Serial interface IICAn is in the reset status.							
1	Enables input clock supply. • SFR used by serial interface IICAn can be read/written.							

**Note**      80, 100-pin products only.

**Caution 1.** When setting serial interface IICA, be sure to set the following registers first while the IICAnEN bit is set to 1. If IICAnEN = 0, the control registers of serial interface IICA are set to their initial values, and writing to them is ignored (except for port mode register 6 (PM6) and port register 6 (P6)).

- IICA control register n0 (IICCTLn0)
- IICA flag register n (IICFn)
- IICA status register n (IICSn)
- IICA control register n1 (IICCTLn1)
- IICA low-level width setting register n (IICWLn)
- IICA high-level width setting register n (IICWHn)

**Caution 2.** Be sure to clear the following bits to 0.

30, 32, 36, 40, 44, 48, 52, 64-pin products: bits 1, 6

**Remark**    n = 0, 1

### 18.3.2 IICA control register n0 (IICCTLn0)

This register is used to enable/stop I<sup>2</sup>C operations, set wait timing, and set other I<sup>2</sup>C operations.

The IICCTLn0 register can be set by a 1-bit or 8-bit memory manipulation instruction. However, set the SPIEn, WTIMn, and ACKEn bits while IICEn = 0 or during the wait period. These bits can be set at the same time when the IICEn bit is set from “0” to “1”.

Reset signal generation clears this register to 00H.

**Remark**    n = 0, 1

**Figure 18 - 6 Format of IICA control register n0 (IICCTLn0) (1/4)**

Address: F0230H (IICCTL00), F0238H (IICCTL10) After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
--------	-----	-----	-----	-----	-----	-----	-----	-----

IICCTLn0	IICEn	LRELn	WRELn	SPIEn	WTIMn	ACKEn	STTn	SPTn
----------	-------	-------	-------	-------	-------	-------	------	------

IICEn	I <sup>2</sup> C operation enable						
0	Stop operation. Reset the IICA status register n (IICSn) <sup>Note 1</sup> . Stop internal operation.						
1	Enable operation.						
Be sure to set this bit (1) while the SCLAn and SDAAn lines are at high level.							
Condition for clearing (IICEn = 0)				Condition for setting (IICEn = 1)			
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Reset</li> </ul>				<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>			

LRELn Notes 2, 3	Exit from communications													
0	Normal operation													
1	This exits from the current communications and sets standby mode. This setting is automatically cleared to 0 after being executed. Its uses include cases in which a locally irrelevant extension code has been received. The SCLAn and SDAAn lines are set to high impedance. The following flags of IICA control register n0 (IICCTLn0) and the IICA status register n (IICSn) are cleared to 0.													
<ul style="list-style-type: none"> <li>• STTn</li> <li>• SPTn</li> <li>• MSTSn</li> <li>• EXCn</li> <li>• COIn</li> <li>• TRCn</li> <li>• ACKDn</li> <li>• STDn</li> </ul>														
The standby mode following exit from communications remains in effect until the following communications entry conditions are met.														
<ul style="list-style-type: none"> <li>• After a stop condition is detected, restart is in master mode.</li> <li>• An address match or extension code reception occurs after the start condition.</li> </ul>														
Condition for clearing (LRELn = 0)				Condition for setting (LRELn = 1)										
<ul style="list-style-type: none"> <li>• Automatically cleared after execution</li> <li>• Reset</li> </ul>				<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>										

WRELn Notes 2, 3	Wait cancellation						
0	Do not cancel wait						
1	Cancel wait. This setting is automatically cleared after wait is canceled.						
When the WRELn bit is set (wait canceled) during the wait period at the ninth clock pulse in the transmission status (TRCn = 1), the SDAAn line goes into the high impedance state (TRCn = 0).							
Condition for clearing (WRELn = 0)				Condition for setting (WRELn = 1)			
<ul style="list-style-type: none"> <li>• Automatically cleared after execution</li> <li>• Reset</li> </ul>				<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>			

**Note 1.** The IICA shift register n (IICAn), the STCFn and IICBSYn bits of the IICA flag register n (IICFn), and the CLDn and DADn bits of IICA control register n1 (IICCTLn1) are reset.

**Note 2.** The signal of this bit is invalid while IICEn is 0.

**Note 3.** When the LRELn and WRELn bits are read, 0 is always read.

**Caution** If the operation of I<sup>2</sup>C is enabled (IICEn = 1) when the SCLAn line is high level, the SDAAn line is low level, and the digital filter is turned on (DFCn bit of IICCTLn1 register = 1), a start condition will be inadvertently detected immediately. In this case, set (1) the LRELn bit by using a 1-bit memory manipulation instruction immediately after enabling operation of I<sup>2</sup>C (IICEn = 1).

**Remark** n = 0, 1

**Figure 18 - 7 Format of IICA control register n0 (IICCTLn0) (2/4)**

SPIEn Note 1	Enable/disable generation of interrupt request when stop condition is detected
0	Disable
1	Enable
If the WUPn bit of IICA control register n1 (IICCTLn1) is 1, no stop condition interrupt will be generated even if SPIEn = 1.	
Condition for clearing (SPIEn = 0)	Condition for setting (SPIEn = 1)
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Reset</li> </ul>	<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>

WTIMn Note 1	Control of wait and interrupt request generation
0	<p>Interrupt request is generated at the eighth clock's falling edge.            Master mode: After output of eight clocks, clock output is set to low level and wait is set.            Slave mode: After input of eight clocks, the clock is set to low level and wait is set for master device.</p>
1	<p>Interrupt request is generated at the ninth clock's falling edge.            Master mode: After output of nine clocks, clock output is set to low level and wait is set.            Slave mode: After input of nine clocks, the clock is set to low level and wait is set for master device.</p>
An interrupt is generated at the falling edge of the ninth clock during address transfer independently of the setting of this bit. The setting of this bit is valid when the address transfer is completed. When in master mode, a wait is inserted at the falling edge of the ninth clock during address transfers. For a slave device that has received a local address, a wait is inserted at the falling edge of the ninth clock after an acknowledgement (ACK) is issued. However, when the slave device has received an extension code, a wait is inserted at the falling edge of the eighth clock.	
Condition for clearing (WTIMn = 0)	Condition for setting (WTIMn = 1)
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Reset</li> </ul>	<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>

ACKEn Notes 1, 2	Acknowledgment control
0	Disable acknowledgment.
1	Enable acknowledgment. During the ninth clock period, the SDAAn line is set to low level.
Condition for clearing (ACKEn = 0)	Condition for setting (ACKEn = 1)
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Reset</li> </ul>	<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>

**Note 1.** The signal of this bit is invalid while IICEn is 0. Set this bit during that period.

**Note 2.** The set value is invalid during address transfer and if the code is not an extension code.

When the device serves as a slave and the addresses match, an acknowledgement is generated regardless of the set value.

**Remark** n = 0, 1

**Figure 18 - 8 Format of IICA control register n0 (IICCTLn0) (3/4)**

STTn Notes 1, 2	Start condition trigger
0	Do not generate a start condition.
1	<p>When bus is released (in standby state, when IICBSYn = 0):  If this bit is set (1), a start condition is generated (startup as the master).</p> <p>When a third party is communicating:</p> <ul style="list-style-type: none"> <li>• When communication reservation function is enabled (IICRSVn = 0)  Functions as the start condition reservation flag. When set to 1, automatically generates a start condition after the bus is released.</li> <li>• When communication reservation function is disabled (IICRSVn = 1)  Even if this bit is set (1), the STTn bit is cleared and the STTn clear flag (STCFn) is set (1). No start condition is generated.</li> </ul> <p>In the wait state (when master device):  Generates a restart condition after releasing the wait.</p>
Cautions concerning set timing	
	<ul style="list-style-type: none"> <li>• For master reception: Cannot be set to 1 during transfer. Can be set to 1 only in the waiting period when the ACKEn bit has been cleared to 0 and slave has been notified of final reception.</li> <li>• For master transmission: A start condition cannot be generated normally during the acknowledge period. Set to 1 during the wait period that follows output of the ninth clock.</li> <li>• Cannot be set to 1 at the same time as stop condition trigger (SPTn).</li> <li>• Once STTn is set (1), setting it again (1) before the clear condition is met is not allowed.</li> </ul>
Condition for clearing (STTn = 0)	Condition for setting (STTn = 1)
<ul style="list-style-type: none"> <li>• Cleared by setting the STTn bit to 1 while communication reservation is prohibited.</li> <li>• Cleared by loss in arbitration</li> <li>• Cleared after start condition is generated by master device</li> <li>• Cleared by LRELn = 1 (exit from communications)</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>	<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>

**Note 1.** The signal of this bit is invalid while IICEn is 0.

**Note 2.** The STTn bit is always read as 0.

**Remark 1.** Bit 1 (STTn) becomes 0 when it is read after data setting.

**Remark 2.** IICRSVn: Bit 0 of IICA flag register n (IICFn)

STCFn: Bit 7 of IICA flag register n (IICFn)

**Remark 3.** n = 0, 1

**Figure 18 - 9 Format of IICA control register n0 (IICCTLn0) (4/4)**

SPTn Note	Stop condition trigger
0	Stop condition is not generated.
1	Stop condition is generated (termination of master device's transfer).
Cautions concerning set timing	
<ul style="list-style-type: none"> <li>• For master reception: Cannot be set to 1 during transfer. Can be set to 1 only in the waiting period when the ACKEn bit has been cleared to 0 and slave has been notified of final reception.</li> <li>• For master transmission: A stop condition cannot be generated normally during the acknowledge period. Therefore, set it during the wait period that follows output of the ninth clock.</li> <li>• Cannot be set to 1 at the same time as start condition trigger (STTn).</li> <li>• The SPTn bit can be set to 1 only when in master mode.</li> <li>• When the WTIMn bit has been cleared to 0, if the SPTn bit is set to 1 during the wait period that follows output of eight clocks, note that a stop condition will be generated during the high-level period of the ninth clock. The WTIMn bit should be changed from 0 to 1 during the wait period following the output of eight clocks, and the SPTn bit should be set to 1 during the wait period that follows the output of the ninth clock.</li> <li>• Once SPTn is set (1), setting it again (1) before the clear condition is met is not allowed.</li> </ul>	
Condition for clearing (SPTn = 0)	Condition for setting (SPTn = 1)
<ul style="list-style-type: none"> <li>• Cleared by loss in arbitration</li> <li>• Automatically cleared after stop condition is detected</li> <li>• Cleared by LRELn = 1 (exit from communications)</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>	<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>

**Note** When the SPTn register is read, 0 is always read.

**Caution** When bit 3 (TRCn) of the IICA status register n (IICSn) is set to 1 (transmission status), bit 5 (WRELn) of IICA control register n0 (IICCTLn0) is set to 1 during the ninth clock and wait is canceled, after which the TRCn bit is cleared (reception status) and the SDAAAn line is set to high impedance. Release the wait performed while the TRCn bit is 1 (transmission status) by writing to the IICA shift register n.

**Remark** n = 0, 1

### 18.3.3 IICA status register n (IICSn)

This register indicates the status of I<sup>2</sup>C.

The IICSn register is read by a 1-bit or 8-bit memory manipulation instruction only when STTn = 1 and during the wait period.

Reset signal generation clears this register to 00H.

**Caution** **Reading the IICSn register while the address match wakeup function is enabled (WUPn = 1) in STOP mode is prohibited. When the WUPn bit is changed from 1 to 0 (wakeup operation is stopped), regardless of the INTIICAn interrupt request, the change in status is not reflected until the next start condition or stop condition is detected. To use the wakeup function, therefore, enable (SPIEn = 1) the interrupt generated by detecting a stop condition and read the IICSn register after the interrupt has been detected.**

**Remark** STTn: bit 1 of IICA control register n0 (IICCTLn0)

WUPn: bit 7 of IICA control register n1 (IICCTLn1)

**Figure 18 - 10 Format of IICA status register n (IICSn) (1/3)**

Address: FFF51H (IICSO), FFF55H (IICSI)								After reset: 00H	R						
Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>							
IICSn	MSTS <sub>n</sub>	ALD <sub>n</sub>	EXC <sub>n</sub>	COIn	TRC <sub>n</sub>	ACKD <sub>n</sub>	STD <sub>n</sub>	SPD <sub>n</sub>							
MSTS <sub>n</sub>		Master status check flag													
0		Slave device status or communication standby status													
1		Master device communication status													
Condition for clearing (MSTS <sub>n</sub> = 0)				Condition for setting (MSTS <sub>n</sub> = 1)											
<ul style="list-style-type: none"> <li>• When a stop condition is detected</li> <li>• When ALD<sub>n</sub> = 1 (arbitration loss)</li> <li>• Cleared by LREL<sub>n</sub> = 1 (exit from communications)</li> <li>• When the IICE<sub>n</sub> bit changes from 1 to 0 (operation stop)</li> <li>• Reset</li> </ul>				<ul style="list-style-type: none"> <li>• When a start condition is generated</li> </ul>											
ALD <sub>n</sub>		Detection of arbitration loss													
0		This status means either that there was no arbitration or that the arbitration result was a “win”.													
1		This status indicates the arbitration result was a “loss”. The MSTS <sub>n</sub> bit is cleared.													
Condition for clearing (ALD <sub>n</sub> = 0)				Condition for setting (ALD <sub>n</sub> = 1)											
<ul style="list-style-type: none"> <li>• Automatically cleared after the IICSn register is read Note</li> <li>• When the IICE<sub>n</sub> bit changes from 1 to 0 (operation stop)</li> <li>• Reset</li> </ul>				<ul style="list-style-type: none"> <li>• When the arbitration result is a “loss”.</li> </ul>											

**Note** This register is also cleared when a 1-bit memory manipulation instruction is executed for bits other than the IICSn register. Therefore, when using the ALD<sub>n</sub> bit, read the data of this bit before the data of the other bits.

**Remark 1.** LREL<sub>n</sub>: Bit 6 of IICA control register n0 (IICCTLn0)

IICE<sub>n</sub>: Bit 7 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

**Figure 18 - 11 Format of IICA status register n (IICSn) (2/3)**

EXCn	Detection of extension code reception	
0	Extension code was not received.	
1	Extension code was received.	
Condition for clearing (EXCn = 0)	<ul style="list-style-type: none"> <li>When a start condition is detected</li> <li>When a stop condition is detected</li> <li>Cleared by LRELn = 1 (exit from communications)</li> <li>When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>Reset</li> </ul>	<ul style="list-style-type: none"> <li>When the higher four bits of the received address data is either "0000" or "1111" (set at the rising edge of the eighth clock).</li> </ul>
Condition for setting (EXCn = 1)		

COIn	Detection of matching addresses	
0	Addresses do not match.	
1	Addresses match.	
Condition for clearing (COIn = 0)	<ul style="list-style-type: none"> <li>When a start condition is detected</li> <li>When a stop condition is detected</li> <li>Cleared by LRELn = 1 (exit from communications)</li> <li>When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>Reset</li> </ul>	<ul style="list-style-type: none"> <li>When the received address matches the local address (slave address register n (SVAn)) (set at the rising edge of the eighth clock).</li> </ul>
Condition for setting (COIn = 1)		

TRCn	Detection of transmit/receive status	
0	Receive status (other than transmit status). The SDAAn line is set for high impedance.	
1	Transmit status. The value in the SOn latch is enabled for output to the SDAAn line (valid starting at the falling edge of the first byte's ninth clock).	
Condition for clearing (TRCn = 0)	<p>&lt;Both master and slave&gt;</p> <ul style="list-style-type: none"> <li>When a stop condition is detected</li> <li>Cleared by LRELn = 1 (exit from communications)</li> <li>When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>Cleared by WRELn = 1 <sup>Note</sup> (wait cancel)</li> <li>When the ALDn bit changes from 0 to 1 (arbitration loss)</li> <li>Reset</li> <li>When not used for communication (MSTS<sub>n</sub>, EXC<sub>n</sub>, COIn = 0)</li> </ul> <p>&lt;Master&gt;</p> <ul style="list-style-type: none"> <li>When "1" is output to the first byte's LSB (transfer direction specification bit)</li> </ul> <p>&lt;Slave&gt;</p> <ul style="list-style-type: none"> <li>When a start condition is detected</li> <li>When "0" is input to the first byte's LSB (transfer direction specification bit)</li> </ul>	<p>&lt;Master&gt;</p> <ul style="list-style-type: none"> <li>When a start condition is generated</li> <li>When 0 (master transmission) is output to the LSB (transfer direction specification bit) of the first byte (during address transfer)</li> </ul> <p>&lt;Slave&gt;</p> <ul style="list-style-type: none"> <li>When 1 (slave transmission) is input to the LSB (transfer direction specification bit) of the first byte from the master (during address transfer)</li> </ul>
Condition for setting (TRCn = 1)		

**Note** When bit 3 (TRCn) of the IICA status register n (IICSn) is set to 1 (transmission status), bit 5 (WRELn) of IICA control register n0 (IICCTLn0) is set to 1 during the ninth clock and wait is canceled, after which the TRCn bit is cleared (reception status) and the SDAAn line is set to high impedance. Release the wait performed while the TRCn bit is 1 (transmission status) by writing to the IICA shift register n.

**Remark 1.** LRELn: Bit 6 of IICA control register n0 (IICCTLn0)  
**IICEn:** Bit 7 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

**Figure 18 - 12 Format of IICA status register n (IICSn) (3/3)**

ACKDn	Detection of acknowledge (ACK)	
0	Acknowledge was not detected.	
1	Acknowledge was detected.	
Condition for clearing (ACKDn = 0)	Condition for setting (ACKDn = 1)	
<ul style="list-style-type: none"> <li>• When a stop condition is detected</li> <li>• At the rising edge of the next byte's first clock</li> <li>• Cleared by LRELn = 1 (exit from communications)</li> <li>• When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>• Reset</li> </ul>		After the SDAAn line is set to low level at the rising edge of SCLAn line's ninth clock

STDn	Detection of start condition	
0	Start condition was not detected.	
1	Start condition was detected. This indicates that the address transfer period is in effect.	
Condition for clearing (STDn = 0)	Condition for setting (STDn = 1)	
<ul style="list-style-type: none"> <li>• When a stop condition is detected</li> <li>• At the rising edge of the next byte's first clock following address transfer</li> <li>• Cleared by LRELn = 1 (exit from communications)</li> <li>• When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>• Reset</li> </ul>		• When a start condition is detected

SPDn	Detection of stop condition	
0	Stop condition was not detected.	
1	Stop condition was detected. The master device's communication is terminated and the bus is released.	
Condition for clearing (SPDn = 0)	Condition for setting (SPDn = 1)	
<ul style="list-style-type: none"> <li>• At the rising edge of the address transfer byte's first clock following setting of this bit and detection of a start condition</li> <li>• When the WUPn bit changes from 1 to 0</li> <li>• When the IICEn bit changes from 1 to 0 (operation stop)</li> <li>• Reset</li> </ul>		• When a stop condition is detected

**Remark 1.** LRELn: Bit 6 of IICA control register n0 (IICCTLn0)

IICEn: Bit 7 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

### 18.3.4 IICA flag register n (IICFn)

This register sets the operation mode of I<sup>2</sup>C and indicates the status of the I<sup>2</sup>C bus.

The IICFn register can be set by a 1-bit or 8-bit memory manipulation instruction. However, the STTn clear flag (STCFn) and I<sup>2</sup>C bus status flag (IICBSYn) bits are read-only.

The IICRSVn bit can be used to enable/disable the communication reservation function.

The STCENn bit can be used to set the initial value of the IICBSYn bit.

The IICRSVn and STCENn bits can be written only when the operation of I<sup>2</sup>C is disabled (bit 7 (IICEn) of IICA control register n0 (IICCTLn0) = 0). When operation is enabled, the IICFn register can be read.

Reset signal generation clears this register to 00H.

**Figure 18 - 13 Format of IICA flag register n (IICFn)**

Address: FFF52H (IICF0), FFF56H (IICF1)		After reset: 00H				R/W Note			
Symbol	Symbol	<7>	<6>	5	4	3	2	<1>	<0>
IICFn	STCFn	IICBSYn	0	0	0	0	0	STCENn	IICRSVn
STCFn	STTn clear flag								
0	Generate start condition								
1	Start condition generation unsuccessful: clear the STTn flag								
Condition for clearing (STCFn = 0)		Condition for setting (STCFn = 1)							
<ul style="list-style-type: none"> <li>• Cleared by STTn = 1</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>		<ul style="list-style-type: none"> <li>• Generating start condition unsuccessful and the STTn bit cleared to 0 when communication reservation is disabled (IICRSVn = 1).</li> </ul>							
IICBSYn	I <sup>2</sup> C bus status flag								
0	Bus release status (communication initial status when STCENn = 1)								
1	Bus communication status (communication initial status when STCENn = 0)								
Condition for clearing (IICBSYn = 0)	Condition for setting (IICBSYn = 1)								
<ul style="list-style-type: none"> <li>• Detection of stop condition</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>		<ul style="list-style-type: none"> <li>• Detection of start condition</li> <li>• Setting of the IICEn bit when STCENn = 0</li> </ul>							
STCENn	Initial start enable trigger								
0	After operation is enabled (IICEn = 1), enable generation of a start condition upon detection of a stop condition.								
1	After operation is enabled (IICEn = 1), enable generation of a start condition without detecting a stop condition.								
Condition for clearing (STCENn = 0)	Condition for setting (STCENn = 1)								
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Detection of start condition</li> <li>• Reset</li> </ul>		<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>							
IICRSVn	Communication reservation function disable bit								
0	Enable communication reservation								
1	Disable communication reservation								
Condition for clearing (IICRSVn = 0)	Condition for setting (IICRSVn = 1)								
<ul style="list-style-type: none"> <li>• Cleared by instruction</li> <li>• Reset</li> </ul>		<ul style="list-style-type: none"> <li>• Set by instruction</li> </ul>							

**Note** Bits 6 and 7 are read-only.

**Caution 1. Write to the STCENn bit only when the operation is stopped (IICEn = 0).**

**Caution 2. As the bus release status (IICBSYn = 0) is recognized regardless of the actual bus status when STCENn = 1, when generating the first start condition (STTn = 1), it is necessary to verify that no third party communications are in progress in order to prevent such communications from being destroyed.**

**Caution 3. Write to IICRSVn only when the operation is stopped (IICEn = 0).**

**Remark 1.** STTn: Bit 1 of IICA control register n0 (IICCTLn0)

IICEn: Bit 7 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

### 18.3.5 IICA control register n1 (IICCTLn1)

This register is used to set the operation mode of I<sup>2</sup>C and detect the statuses of the SCLAn and SDAAn pins.

The IICCTLn1 register can be set by a 1-bit or 8-bit memory manipulation instruction. However, the CLDn and DADn bits are read-only.

Set the IICCTLn1 register, except the WUPn bit, while operation of I<sup>2</sup>C is disabled (bit 7 (IICEn) of IICA control register n0 (IICCTLn0) is 0).

Reset signal generation clears this register to 00H.

**Figure 18 - 14 Format of IICA control register n1 (IICCTLn1) (1/2)**

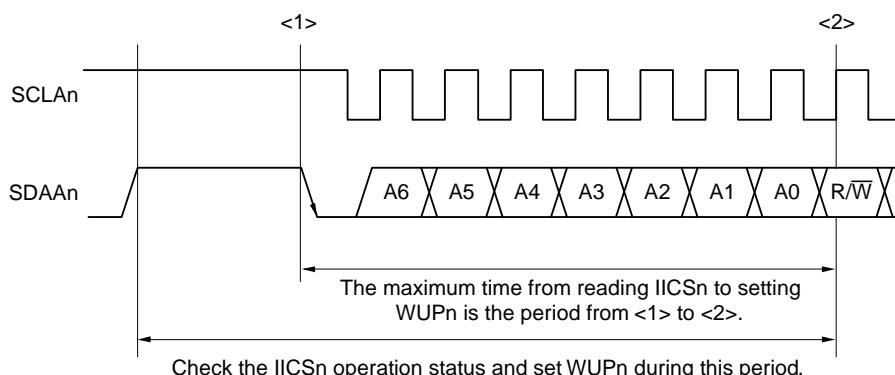
Address: F0231H (IICCTL01), F0239H (IICCTL11) After reset: 00H R/W Note 1

Symbol	<7>	6	<5>	<4>	<3>	<2>	1	<0>
IICCTLn1	WUPn	0	CLDn	DADn	SMCn	DFCn	0	PRSn

WUPn	Control of address match wakeup
0	Stops operation of address match wakeup function in STOP mode.
1	Enables operation of address match wakeup function in STOP mode.
To shift to STOP mode when WUPn = 1, execute the STOP instruction at least three fmck clocks after setting (1) the WUPn bit (see <b>Figure 18 - 29 Flow When Setting WUPn = 1</b> ).	
Clear (0) the WUPn bit after the address has matched or an extension code has been received. The subsequent communication can be entered by the clearing (0) WUPn bit. (The wait must be released and transmit data must be written after the WUPn bit has been cleared (0).)	
The interrupt timing when the address has matched or when an extension code has been received, while WUPn = 1, is identical to the interrupt timing when WUPn = 0. (A delay of the difference of sampling by the clock will occur.) Furthermore, when WUPn = 1, a stop condition interrupt is not generated even if the SPIEn bit is set to 1.	
Condition for clearing (WUPn = 0)	Condition for setting (WUPn = 1)
• Cleared by instruction (after address match or extension code reception)	• Set by instruction (when the MSTSn, EXCn, and COIn bits are “0”, and the STDn bit also “0” (communication not entered)) Note 2

**Note 1.** Bits 4 and 5 are read-only.

**Note 2.** The status of the IICA status register n (IICSn) must be checked and the WUPn bit must be set during the period shown below.



**Remark** n = 0, 1

**Figure 18 - 15 Format of IICA control register n1 (IICCTLn1) (2/2)**

CLDn	Detection of SCLAn pin level (valid only when IICEn = 1)	
0	The SCLAn pin was detected at low level.	
1	The SCLAn pin was detected at high level.	
Condition for clearing (CLDn = 0)	Condition for setting (CLDn = 1)	
<ul style="list-style-type: none"> <li>• When the SCLAn pin is at low level</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>		

DADn	Detection of SDAAAn pin level (valid only when IICEn = 1)	
0	The SDAAAn pin was detected at low level.	
1	The SDAAAn pin was detected at high level.	
Condition for clearing (DADn = 0)	Condition for setting (DADn = 1)	
<ul style="list-style-type: none"> <li>• When the SDAAAn pin is at low level</li> <li>• When IICEn = 0 (operation stop)</li> <li>• Reset</li> </ul>		

SMCn	Operation mode switching	
0	Operates in standard mode (fastest transfer rate: 100 kbps).	
1	Operates in fast mode (fastest transfer rate: 400 kbps) or fast mode plus (fastest transfer rate: 1 Mbps).	

DFCn	Digital filter operation control	
0	Digital filter off.	
1	Digital filter on.	
Use the digital filter only in fast mode and fast mode plus. The digital filter is used for noise elimination. The transfer clock does not vary, regardless of the DFCn bit being set (1) or cleared (0).		

PRSn	IICA operation clock (fmck) control	
0	Selects fCLK ( $1 \text{ MHz} \leq \text{fCLK} \leq 20 \text{ MHz}$ )	
1	Selects fCLK/2 ( $20 \text{ MHz} \leq \text{fCLK}$ )	

**Caution 1.** The maximum operating frequency of the IICA operating clock (fmck) is 20 MHz (Max.). Only when fCLK exceeds 20 MHz, set bit 0 (PRSn) of IICA control register n1 (IICCTLn1) to 1.

**Caution 2.** Note the minimum fCLK operating frequency when setting the transfer clock. The minimum fCLK operating frequency for serial interface IICA is determined according to the mode.

Fast mode:      fCLK = 3.5 MHz (MIN.)

Fast mode plus:    fCLK = 10 MHz (MIN.)

Normal mode:     fCLK = 1 MHz (MIN.)

**Caution 3.** The fast mode plus is only available in the products for “A: Consumer applications ( $T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ )” and “D: Industrial applications ( $T_A = -40^\circ\text{C}$  to  $+85^\circ\text{C}$ )”.

**Remark 1.** IICEn: Bit 7 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

### 18.3.6 IICA low-level width setting register n (IICWL<sub>n</sub>)

This register is used to set the low-level width ( $t_{LOW}$ ) of the SCLAn pin signal that is output by serial interface IICA and to control the SDAA<sub>n</sub> pin signal.

The IICWL<sub>n</sub> register can be set by an 8-bit memory manipulation instruction.

Set the IICWL<sub>n</sub> register while operation of I<sup>2</sup>C is disabled (bit 7 (IICEn) of IICA control register n0 (IICCTL<sub>n</sub>0) is 0).

Reset signal generation sets this register to FFH.

For details about setting the IICWL<sub>n</sub> register, see **18.4.2 Setting transfer clock by using IICWL<sub>n</sub> and IICWH<sub>n</sub> registers**.

The data hold time is one-quarter of the time set by the IICWL<sub>n</sub> register.

**Figure 18 - 16 Format of IICA low-level width setting register n (IICWL<sub>n</sub>)**

Address: F0232H (IICWL0), F023AH (IICWL1)								After reset: FFH	R/W
Symbol	7	6	5	4	3	2	1	0	
IICWL <sub>n</sub>									

### 18.3.7 IICA high-level width setting register n (IICWH<sub>n</sub>)

This register is used to set the high-level width of the SCLAn pin signal that is output by serial interface IICA and to control the SDAA<sub>n</sub> pin signal.

The IICWH<sub>n</sub> register can be set by an 8-bit memory manipulation instruction.

Set the IICWH<sub>n</sub> register while operation of I<sup>2</sup>C is disabled (bit 7 (IICEn) of IICA control register n0 (IICCTL<sub>n</sub>0) is 0).

Reset signal generation sets this register to FFH.

**Figure 18 - 17 Format of IICA high-level width setting register n (IICWH<sub>n</sub>)**

Address: F0233H (IICWH0), F023BH (IICWH1)								After reset: FFH	R/W
Symbol	7	6	5	4	3	2	1	0	
IICWH <sub>n</sub>									

**Remark 1.** For setting procedures of the transfer clock on master side and of the IICWL<sub>n</sub> and IICWH<sub>n</sub> registers on slave side, see **18.4.2 (1)** and **18.4.2 (2)**, respectively.

**Remark 2.**  $n = 0, 1$

### 18.3.8 Port mode register 6 (PM6)

This register sets the input/output of port 6 in 1-bit units.

When using the P60/SCLA0 pin as clock I/O and the P61/SDAA0 pin as serial data I/O, clear PM60 and PM61, and the output latches of P60 and P61 to 0.

Set the IICEn bit (bit 7 of IICA control register n0 (IICCTLn0)) to 1 before setting the output mode because the P60/SCLA0 and P61/SDAA0 pins output a low level (fixed) when the IICEn bit is 0.

The PM6 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation sets this register to FFH.

**Figure 18 - 18 Format of Port mode register 6 (PM6)**

Address: FFF26H      After reset: FFH      R/W

Symbol	7	6	5	4	3	2	1	0
PM6	1	1	1	1	1	1	PM61	PM60
PM6n	P6n pin I/O mode selection (n = 0, 1)							
0	Output mode (output buffer on)							
1	Input mode (output buffer off)							

## 18.4 I<sup>2</sup>C Bus Mode Functions

### 18.4.1 Pin configuration

The serial clock pin (SCLAn) and the serial data bus pin (SDAAn) are configured as follows.

- (1) SCLAn.....This pin is used for serial clock input and output.

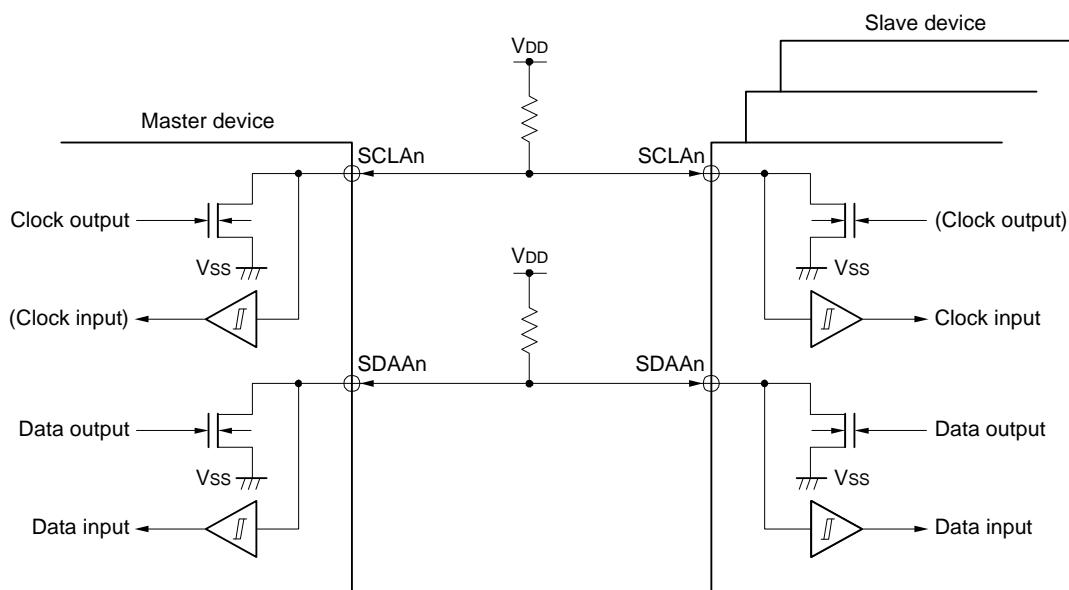
This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

- (2) SDAAn .....This pin is used for serial data input and output.

This pin is an N-ch open-drain output for both master and slave devices. Input is Schmitt input.

Since outputs from the serial clock line and the serial data bus line are N-ch open-drain outputs, an external pull-up resistor is required.

**Figure 18 - 19 Pin Configuration Diagram**



**Remark** n = 0, 1

### 18.4.2 Setting transfer clock by using IICWL<sub>n</sub> and IICWH<sub>n</sub> registers

- (1) Setting transfer clock on master side

$$\text{Transfer clock} = \frac{f_{MCK}}{IICWL + IICWH + f_{MCK} (t_R + t_F)}$$

At this time, the optimal setting values of the IICWL<sub>n</sub> and IICWH<sub>n</sub> registers are as follows.  
(The fractional parts of all setting values are rounded up.)

- When the fast mode

$$IICWL_n = \frac{0.52}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH_n = \left( \frac{0.48}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the standard mode

$$IICWL_n = \frac{0.47}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH_n = \left( \frac{0.53}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- When the fast mode plus

$$IICWL_n = \frac{0.50}{\text{Transfer clock}} \times f_{MCK}$$

$$IICWH_n = \left( \frac{0.50}{\text{Transfer clock}} - t_R - t_F \right) \times f_{MCK}$$

- (2) Setting IICWL<sub>n</sub> and IICWH<sub>n</sub> registers on slave side

(The fractional parts of all setting values are truncated.)

- When the fast mode

$$IICWL_n = 1.3 \mu s \times f_{MCK}$$

$$IICWH_n = (1.2 \mu s - t_R - t_F) \times f_{MCK}$$

- When the standard mode

$$IICWL_n = 4.7 \mu s \times f_{MCK}$$

$$IICWH_n = (5.3 \mu s - t_R - t_F) \times f_{MCK}$$

- When the fast mode plus

$$IICWL_n = 0.50 \mu s \times f_{MCK}$$

$$IICWH_n = (0.50 \mu s - t_R - t_F) \times f_{MCK}$$

**Caution 1.**The maximum operating frequency of the IICA operating clock (f<sub>MCK</sub>) is 20 MHz (Max.). Only when f<sub>CLK</sub> exceeds 20 MHz, set bit 0 (PRSN<sub>n</sub>) of IICA control register n1 (IICCTL<sub>n1</sub>) to 1.

**Caution 2.**Note the minimum f<sub>CLK</sub> operating frequency when setting the transfer clock. The minimum f<sub>CLK</sub> operating frequency for serial interface IICA is determined according to the mode.

Fast mode: f<sub>CLK</sub> = 3.5 MHz (MIN.)

Fast mode plus: f<sub>CLK</sub> = 10 MHz (MIN.)

Normal mode: f<sub>CLK</sub> = 1 MHz (MIN.)

(Remarks are listed on the next page.)

**Remark 1.** Calculate the rise time ( $t_R$ ) and fall time ( $t_F$ ) of the SDAAn and SCLAn signals separately, because they differ depending on the pull-up resistance and wire load.

**Remark 2.** IICWL $n$ : IICA low-level width setting register  $n$

IICWH $n$ : IICA high-level width setting register  $n$

$t_F$ : SDAAn and SCLAn signal falling times

$t_R$ : SDAAn and SCLAn signal rising times

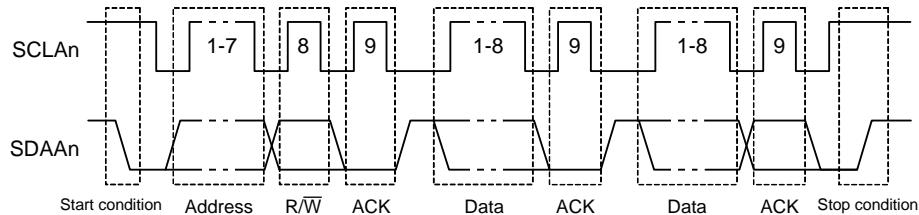
$f_{MCK}$ : IICA operating clock frequency

**Remark 3.**  $n = 0, 1$

## 18.5 I<sup>2</sup>C Bus Definitions and Control Methods

The I<sup>2</sup>C bus's serial data communication format and the signals used by the I<sup>2</sup>C bus are described below. Figure 18-20 shows the transfer timing for the "start condition", "address", "data", and "stop condition" output via the I<sup>2</sup>C bus's serial data bus.

Figure 18 - 20 I<sup>2</sup>C Bus Serial Data Transfer Timing



The master device generates the start condition, slave address, and stop condition.

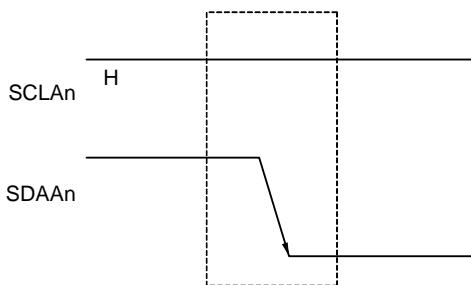
The acknowledge (ACK) can be generated by either the master or slave device (normally, it is output by the device that receives 8-bit data).

The serial clock (SCLAn) is continuously output by the master device. However, in the slave device, the SCLAn pin low level period can be extended and a wait can be inserted.

### 18.5.1 Start conditions

A start condition is met when the SCLAn pin is at high level and the SDAAn pin changes from high level to low level. The start conditions for the SCLAn pin and SDAAn pin are signals that the master device generates to the slave device when starting a serial transfer. When the device is used as a slave, start conditions can be detected.

Figure 18 - 21 Start Conditions



A start condition is output when bit 1 (STTn) of IICA control register n0 (IICCTLn0) is set (1) after a stop condition has been detected (SPDn: Bit 0 of the IICA status register n (IICSn) = 1). When a start condition is detected, bit 1 (STDn) of the IICSn register is set (1).

**Remark** n = 0, 1

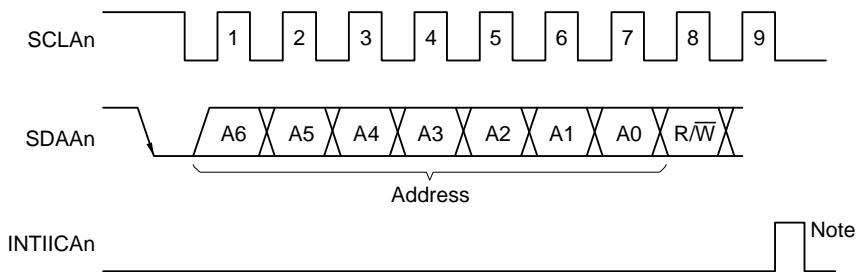
### 18.5.2 Addresses

The address is defined by the 7 bits of data that follow the start condition.

An address is a 7-bit data segment that is output in order to select one of the slave devices that are connected to the master device via the bus lines. Therefore, each slave device connected via the bus lines must have a unique address.

The slave devices include hardware that detects the start condition and checks whether or not the 7-bit address data matches the data values stored in the slave address register n (SVAn). If the address data matches the SVAn register values, the slave device is selected and communicates with the master device until the master device generates a start condition or stop condition.

**Figure 18 - 22 Address**



**Note** INTIICAn is not issued if data other than a local address or extension code is received during slave device operation.

Addresses are output when a total of 8 bits consisting of the slave address and the transfer direction described in **18.5.3 Transfer direction specification** are written to the IICA shift register n (IICAn). The received addresses are written to the IICAn register.

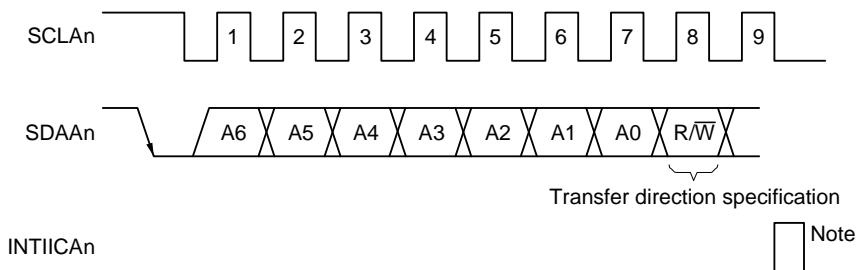
The slave address is assigned to the higher 7 bits of the IICAn register.

### 18.5.3 Transfer direction specification

In addition to the 7-bit address data, the master device sends 1 bit that specifies the transfer direction.

When this transfer direction specification bit has a value of "0", it indicates that the master device is transmitting data to a slave device. When the transfer direction specification bit has a value of "1", it indicates that the master device is receiving data from a slave device.

**Figure 18 - 23 Transfer Direction Specification**



**Note** INTIICAn is not issued if data other than a local address or extension code is received during slave device operation.

**Remark**  $n = 0, 1$

### 18.5.4 Acknowledge (ACK)

ACK is used to check the status of serial data at the transmission and reception sides.

The reception side returns ACK each time it has received 8-bit data.

The transmission side usually receives ACK after transmitting 8-bit data. When ACK is returned from the reception side, it is assumed that reception has been correctly performed and processing is continued. Whether ACK has been detected can be checked by using bit 2 (ACKDn) of the IICA status register n (IICSn).

When the master receives the last data item, it does not return ACK and instead generates a stop condition. If a slave does not return ACK after receiving data, the master outputs a stop condition or restart condition and stops transmission. If ACK is not returned, the possible causes are as follows.

- <1> Reception was not performed normally.
- <2> The final data item was received.
- <3> The reception side specified by the address does not exist.

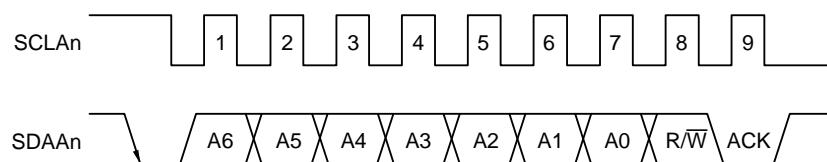
To generate ACK, the reception side makes the SDAAn line low at the ninth clock (indicating normal reception).

Automatic generation of ACK is enabled by setting bit 2 (ACKEn) of IICA control register n0 (IICCTLn0) to 1. Bit 3 (TRCn) of the IICSn register is set by the data of the eighth bit that follows 7-bit address information. Usually, set the ACKEn bit to 1 for reception (TRCn = 0).

If a slave can receive no more data during reception (TRCn = 0) or does not require the next data item, then the slave must inform the master, by clearing the ACKEn bit to 0, that it will not receive any more data.

When the master does not require the next data item during reception (TRCn = 0), it must clear the ACKEn bit to 0 so that ACK is not generated. In this way, the master informs a slave at the transmission side that it does not require any more data (transmission will be stopped).

**Figure 18 - 24 ACK**



When the local address is received, ACK is automatically generated, regardless of the value of the ACKEn bit.

When an address other than that of the local address is received, ACK is not generated (NACK).

When an extension code is received, ACK is generated if the ACKEn bit is set to 1 in advance.

How ACK is generated when data is received differs as follows depending on the setting of the wait timing.

- When 8-clock wait state is selected (bit 3 (WTIMn) of IICCTLn0 register = 0):
 

By setting the ACKEn bit to 1 before releasing the wait state, ACK is generated at the falling edge of the eighth clock of the SCLAn pin.
- When 9-clock wait state is selected (bit 3 (WTIMn) of IICCTLn0 register = 1):
 

ACK is generated by setting the ACKEn bit to 1 in advance.

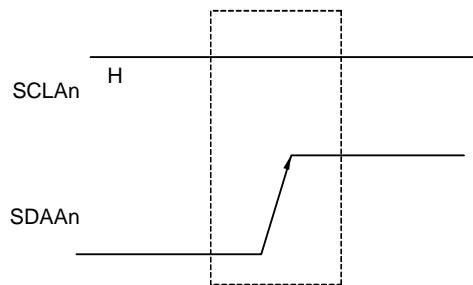
**Remark** n = 0, 1

### 18.5.5 Stop condition

When the SCLAn pin is at high level, changing the SDAAn pin from low level to high level generates a stop condition.

A stop condition is a signal that the master device generates to the slave device when serial transfer has been completed. When the device is used as a slave, stop conditions can be detected.

Figure 18 - 25 Stop Condition



A stop condition is generated when bit 0 (SPTn) of IICA control register n0 (IICCTLn0) is set to 1. When the stop condition is detected, bit 0 (SPDn) of the IICA status register n (IICSn) is set to 1 and INTIICAn is generated when bit 4 (SPIEn) of the IICCTLn0 register is set to 1.

**Remark** n = 0, 1

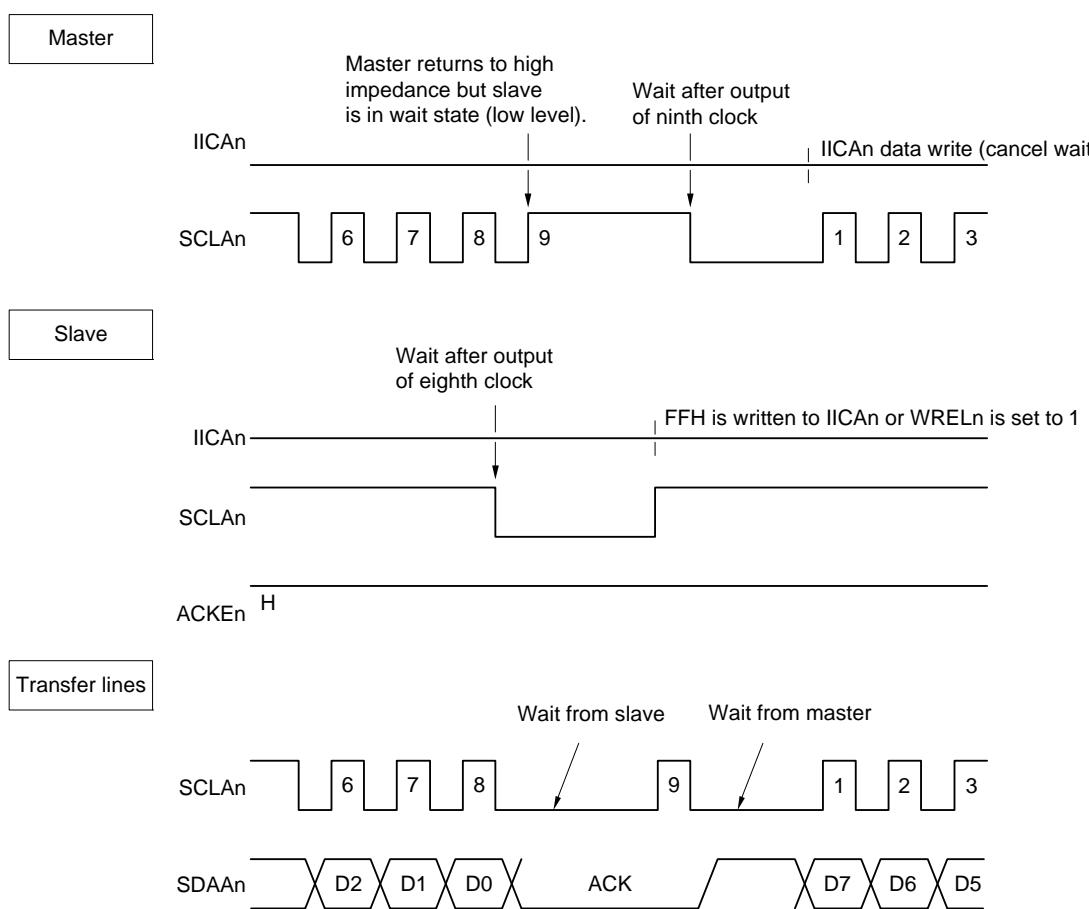
### 18.5.6 Wait

The wait is used to notify the communication partner that a device (master or slave) is preparing to transmit or receive data (i.e., is in a wait state).

Setting the SCLAn pin to low level notifies the communication partner of the wait state. When wait state has been canceled for both the master and slave devices, the next data transfer can begin.

**Figure 18 - 26 Wait (1/2)**

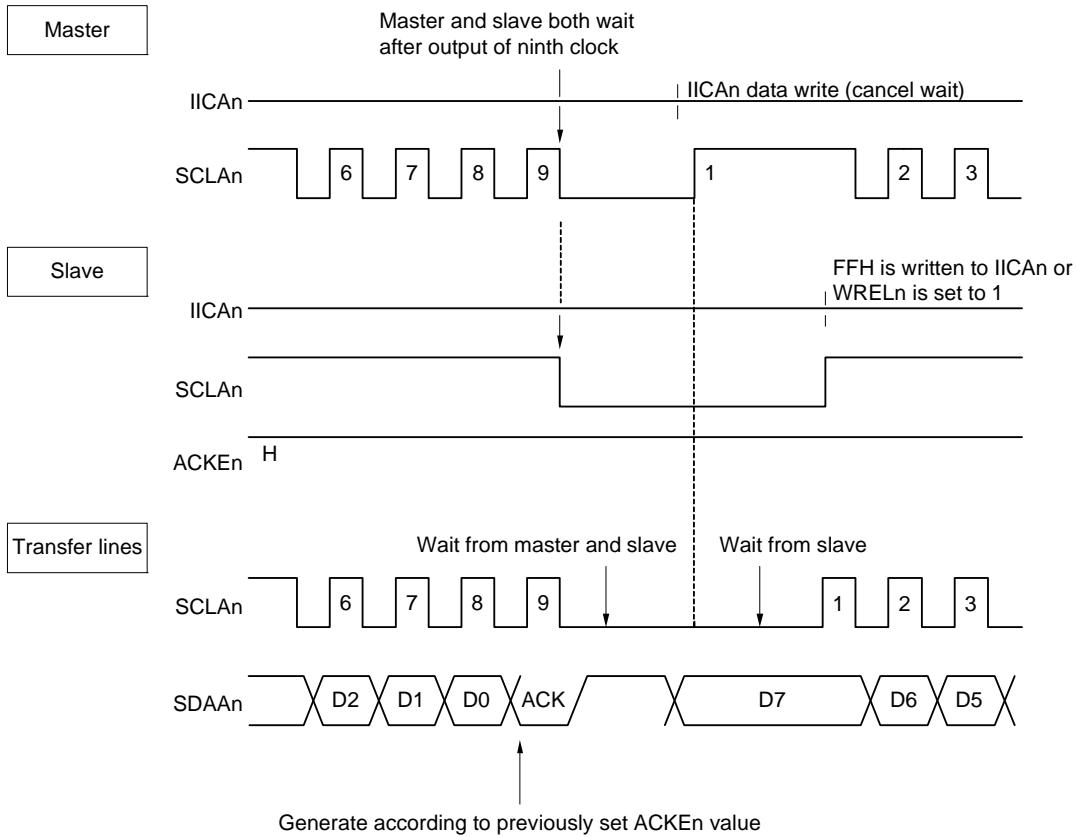
- (1) When master device has a nine-clock wait and slave device has an eight-clock wait  
(master transmits, slave receives, and ACKEn = 1)



**Remark** n = 0, 1

**Figure 18 - 27 Wait (2/2)**

- (2) When master and slave devices both have a nine-clock wait  
(master transmits, slave receives, and ACKEn = 1)



**Remark** ACKEn: Bit 2 of IICA control register n0 (IICCTLn0)

WRELn: Bit 5 of IICA control register n0 (IICCTLn0)

A wait may be automatically generated depending on the setting of bit 3 (WTIMn) of IICA control register n0 (IICCTLn0).

Normally, the receiving side cancels the wait state when bit 5 (WRELn) of the IICCTLn0 register is set to 1 or when FFH is written to the IICA shift register n (IICAn), and the transmitting side cancels the wait state when data is written to the IICAn register.

The master device can also cancel the wait state via either of the following methods.

- By setting bit 1 (STTn) of the IICCTLn0 register to 1
- By setting bit 0 (SPTn) of the IICCTLn0 register to 1

**Remark** n = 0, 1

### 18.5.7 Canceling wait

The I<sup>2</sup>C usually cancels a wait state by the following processing.

- Writing data to the IICA shift register n (IICAn)
- Setting bit 5 (WRELn) of IICA control register n0 (IICCTLn0) (canceling wait)
- Setting bit 1 (STTn) of the IICCTLn0 register (generating start condition) Note
- Setting bit 0 (SPTn) of the IICCTLn0 register (generating stop condition) Note

**Note** Master only

When the above wait canceling processing is executed, the I<sup>2</sup>C cancels the wait state and communication is resumed.

To cancel a wait state and transmit data (including addresses), write the data to the IICAn register.

To receive data after canceling a wait state, or to complete data transmission, set bit 5 (WRELn) of the IICCTLn0 register to 1.

To generate a restart condition after canceling a wait state, set bit 1 (STTn) of the IICCTLn0 register to 1.

To generate a stop condition after canceling a wait state, set bit 0 (SPTn) of the IICCTLn0 register to 1.

Execute the canceling processing only once for one wait state.

If, for example, data is written to the IICAn register after canceling a wait state by setting the WRELn bit to 1, an incorrect value may be output to SDAAn line because the timing for changing the SDAAn line conflicts with the timing for writing the IICAn register.

In addition to the above, communication is stopped if the IICEn bit is cleared to 0 when communication has been aborted, so that the wait state can be canceled.

If the I<sup>2</sup>C bus has deadlocked due to noise, processing is saved from communication by setting bit 6 (LRELn) of the IICCTLn0 register, so that the wait state can be canceled.

**Caution** If a processing to cancel a wait state is executed when WUPn = 1, the wait state will not be canceled.

**Remark** n = 0, 1

### 18.5.8 Interrupt request (INTIICAn) generation timing and wait control

The setting of bit 3 (WTIMn) of IICA control register n0 (IICCTLn0) determines the timing by which INTIICAn is generated and the corresponding wait control, as shown in Table 18 - 2.

**Table 18 - 2 INTIICAn Generation Timing and Wait Control**

WTIMn	During Slave Device Operation			During Master Device Operation		
	Address	Data Reception	Data Transmission	Address	Data Reception	Data Transmission
0	9 Notes 1, 2	8 Note 2	8 Note 2	9	8	8
1	9 Notes 1, 2	9 Note 2	9 Note 2	9	9	9

**Note 1.** The slave device's INTIICAn signal and wait period occurs at the falling edge of the ninth clock only when there is a match with the address set to the slave address register n (SVAn).

At this point, ACK is generated regardless of the value set to the IICCTLn0 register's bit 2 (ACKEn). For a slave device that has received an extension code, INTIICAn occurs at the falling edge of the eighth clock.

However, if the address does not match after restart, INTIICAn is generated at the falling edge of the 9th clock, but wait does not occur.

**Note 2.** If the received address does not match the contents of the slave address register n (SVAn) and extension code is not received, neither INTIICAn nor a wait occurs.

**Remark** The numbers in the table indicate the number of the serial clock's clock signals. Interrupt requests and wait control are both synchronized with the falling edge of these clock signals.

- (1) During address transmission/reception
  - Slave device operation: Interrupt and wait timing are determined depending on the conditions described in Notes 1 and 2 above, regardless of the WTIMn bit.
  - Master device operation: Interrupt and wait timing occur at the falling edge of the ninth clock regardless of the WTIMn bit.
- (2) During data reception
  - Master/slave device operation: Interrupt and wait timing are determined according to the WTIMn bit.
- (3) During data transmission
  - Master/slave device operation: Interrupt and wait timing are determined according to the WTIMn bit.

**Remark** n = 0, 1

(4) Wait cancellation method

The four wait cancellation methods are as follows.

- Writing data to the IICA shift register n (IICAn)
- Setting bit 5 (WREL $n$ ) of IICA control register n0 (IICCTL $n$ 0) (canceling wait)
- Setting bit 1 (STT $n$ ) of IICCTL $n$ 0 register (generating start condition) Note
- Setting bit 0 (SPT $n$ ) of IICCTL $n$ 0 register (generating stop condition) Note

**Note** Master only.

When an 8-clock wait has been selected (WTIM $n$  = 0), the presence/absence of ACK generation must be determined prior to wait cancellation.

(5) Stop condition detection

INTIICAn is generated when a stop condition is detected (only when SPIEn = 1).

### 18.5.9 Address match detection method

In I<sup>2</sup>C bus mode, the master device can select a particular slave device by transmitting the corresponding slave address.

Address match can be detected automatically by hardware. An interrupt request (INTIICAn) occurs when the address set to the slave address register n (SVAn) matches the slave address sent by the master device, or when an extension code has been received.

### 18.5.10 Error detection

In I<sup>2</sup>C bus mode, the status of the serial data bus (SDAAn) during data transmission is captured by the IICA shift register n (IICAn) of the transmitting device, so the IICA data prior to transmission can be compared with the transmitted IICA data to enable detection of transmission errors. A transmission error is judged as having occurred when the compared data values do not match.

**Remark** n = 0, 1

### 18.5.11 Extension code

- (1) When the higher 4 bits of the receive address are either "0000" or "1111", the extension code reception flag (EXCn) is set to 1 for extension code reception and an interrupt request (INTIICAn) is issued at the falling edge of the eighth clock. The local address stored in the slave address register n (SVAn) is not affected.
- (2) The settings below are specified if 11110xx0 is transferred from the master by using a 10-bit address transfer when the SVAn register is set to 11110xx0. Note that INTIICAn occurs at the falling edge of the eighth clock.
  - Higher four bits of data match: EXCn = 1
  - Seven bits of data match: COIn = 1

**Remark** EXCn: Bit 5 of IICA status register n (IICSn)  
 COIn: Bit 4 of IICA status register n (IICSn)

- (3) Since the processing after the interrupt request occurs differs according to the data that follows the extension code, such processing is performed by software.  
 If the extension code is received while a slave device is operating, then the slave device is participating in communication even if its address does not match.  
 For example, after the extension code is received, if you do not wish to operate the target device as a slave device, set bit 6 (LRELn) of IICA control register n0 (IICCTLn0) to 1 to set the standby mode for the next communication operation.

**Table 18 - 3 Bit Definitions of Major Extension Codes**

Slave Address	R/W Bit	Description
0000 000	0	General call address
1111 0xx	0	10-bit slave address specification (during address authentication)
1111 0xx	1	10-bit slave address specification (after address match, when read command is issued)

**Remark 1.** See the I<sup>2</sup>C bus specifications issued by NXP Semiconductors for details of extension codes other than those described above.

**Remark 2.** n = 0, 1

### 18.5.12 Arbitration

When several master devices simultaneously generate a start condition (when the STTn bit is set to 1 before the STDn bit is set to 1), communication among the master devices is performed as the number of clocks are adjusted until the data differs. This kind of operation is called arbitration.

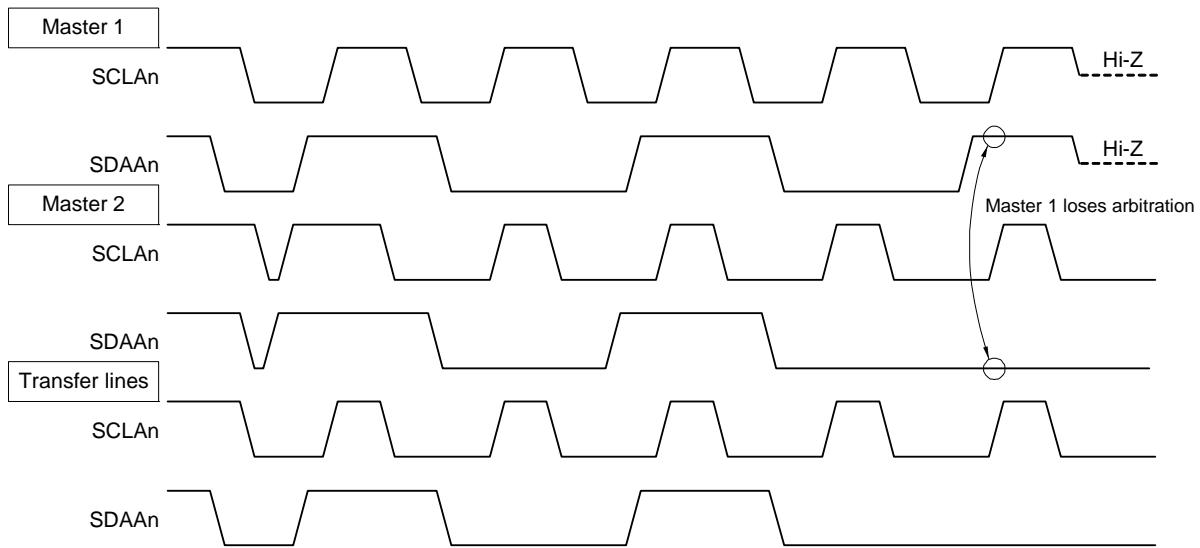
When one of the master devices loses in arbitration, an arbitration loss flag (ALDn) in the IICA status register n (IICSn) is set (1) via the timing by which the arbitration loss occurred, and the SCLAn and SDAAn lines are both set to high impedance, which releases the bus.

The arbitration loss is detected based on the timing of the next interrupt request (the eighth or ninth clock, when a stop condition is detected, etc.) and the ALDn = 1 setting that has been made by software.

For details of interrupt request timing, see **18.5.8 Interrupt request (INTIICAn) generation timing and wait control**.

**Remark** STDn: Bit 1 of IICA status register n (IICSn)  
STTn: Bit 1 of IICA control register n0 (IICCTLn0)

Figure 18 - 28 Arbitration Timing Example



**Remark** n = 0, 1

**Table 18 - 4 Status During Arbitration and Interrupt Request Generation Timing**

Status During Arbitration	Interrupt Request Generation Timing
During address transmission	At falling edge of eighth or ninth clock following byte transfer Note 1
Read/write data after address transmission	
During extension code transmission	
Read/write data after extension code transmission	
During data transmission	
During ACK transfer period after data transmission	
When restart condition is detected during data transfer	
When stop condition is detected during data transfer	When stop condition is generated (when SPIEn = 1) Note 2
When data is at low level while attempting to generate a restart condition	At falling edge of eighth or ninth clock following byte transfer Note 1
When stop condition is detected while attempting to generate a restart condition	When stop condition is generated (when SPIEn = 1) Note 2
When data is at low level while attempting to generate a stop condition	At falling edge of eighth or ninth clock following byte transfer Note 1
When SCLAn is at low level while attempting to generate a restart condition	

**Note 1.** When the WTIMn bit (bit 3 of IICA control register n0 (IICCTLn0)) = 1, an interrupt request occurs at the falling edge of the ninth clock. When WTIMn = 0 and the extension code's slave address is received, an interrupt request occurs at the falling edge of the eighth clock.

**Note 2.** When there is a chance that arbitration will occur, set SPIEn = 1 for master device operation.

**Remark 1.** SPIEn: Bit 4 of IICA control register n0 (IICCTLn0)

**Remark 2.** n = 0, 1

### 18.5.13 Wakeup function

The I<sup>2</sup>C bus slave function is a function that generates an interrupt request signal (INTIICAn) when a local address and extension code have been received.

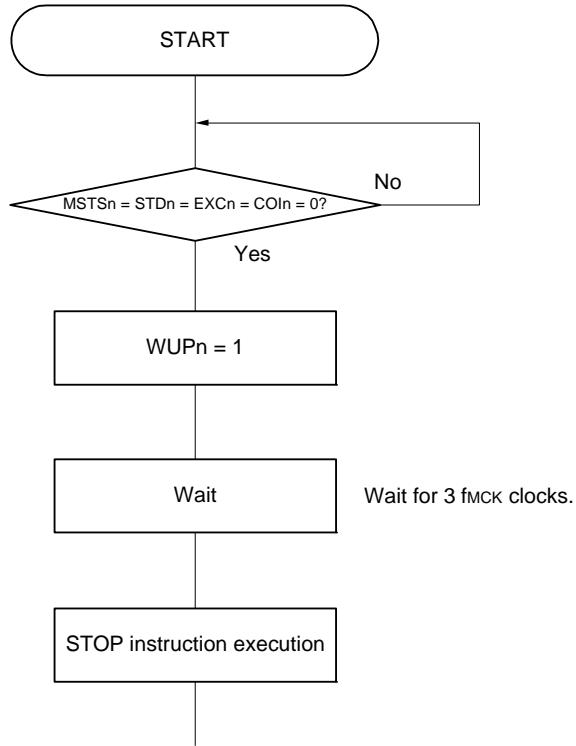
This function makes processing more efficient by preventing unnecessary INTIICAn signal from occurring when addresses do not match.

When a start condition is detected, wakeup standby mode is set. This wakeup standby mode is in effect while addresses are transmitted due to the possibility that an arbitration loss may change the master device (which has generated a start condition) to a slave device.

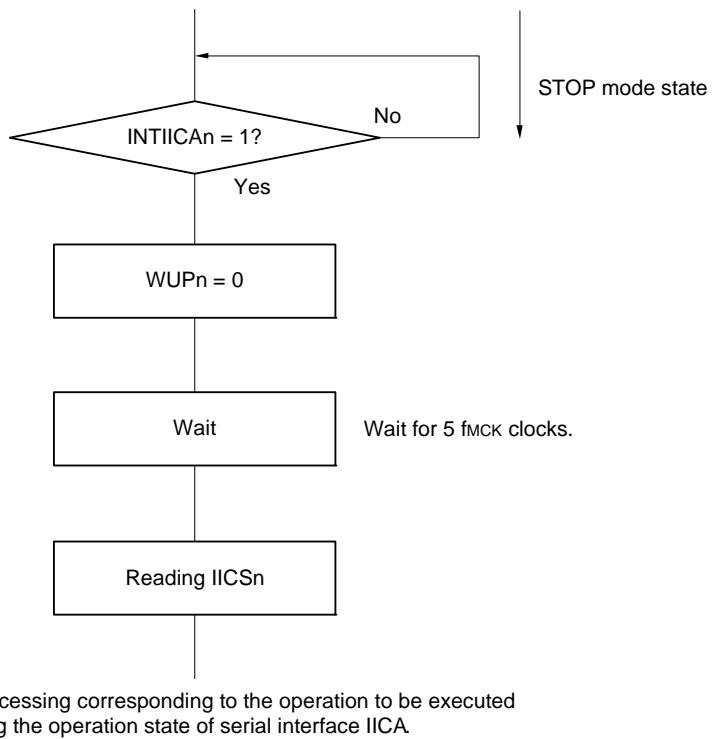
To use the wakeup function in the STOP mode, set the WUPn bit to 1. Addresses can be received regardless of the operation clock. An interrupt request signal (INTIICAn) is also generated when a local address and extension code have been received. Operation returns to normal operation by using an instruction to clear (0) the WUPn bit after this interrupt has been generated.

Figure 18 - 29 shows the flow for setting WUPn = 1 and Figure 18 - 30 shows the flow for setting WUPn = 0 upon an address match.

**Figure 18 - 29 Flow When Setting WUPn = 1**



**Remark** n = 0, 1

**Figure 18 - 30 Flow When Setting WUPn = 0 upon Address Match (Including Extension Code Reception)**

Use the following flows to perform the processing to release the STOP mode other than by an interrupt request (INTIICAn) generated from serial interface IICA.

- When operating as the master device for the next IIC communication: Flow shown in Figure 18 - 31
- When operating as a slave device for the next IIC communication:

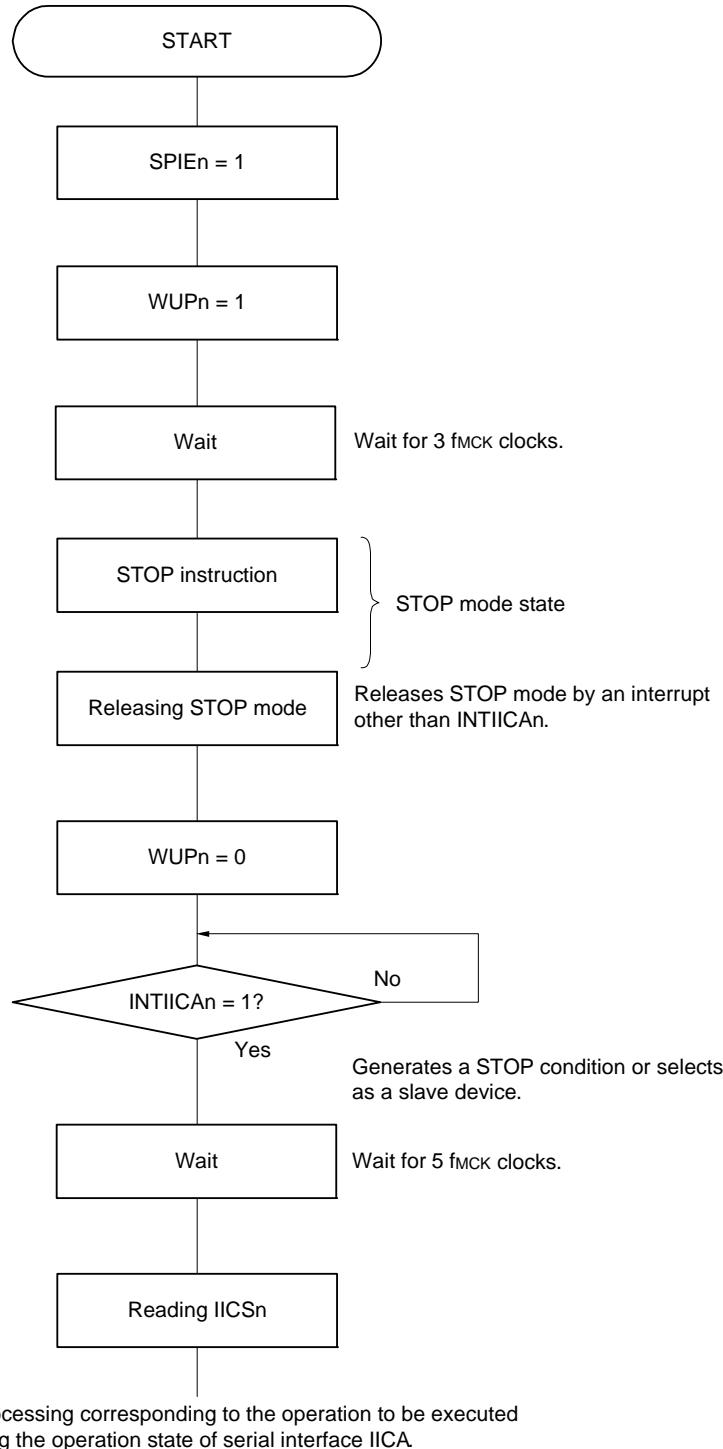
When the INTIICAn interrupt is used to return from the mode:

Same as the flow in Figure 18 - 30

When an interrupt other than the INTIICAn interrupt is used to return from the mode:

Continue operation while WUPn = 1 until an INTIICAn interrupt is generated.

**Remark** n = 0, 1

**Figure 18 - 31 When Operating as Master Device after Releasing STOP Mode other than by INTIICAn**

**Remark** n = 0, 1

### 18.5.14 Communication reservation

- (1) When communication reservation function is enabled (bit 0 (IICRSVn) of IICA flag register n (IICFn) = 0)  
 To start master device communications when not currently using a bus, a communication reservation can be made to enable transmission of a start condition when the bus is released. There are two modes under which the bus is not used.

- When arbitration results in neither master nor slave operation
- When an extension code is received and slave operation is disabled (ACK is not returned and the bus was released by setting bit 6 (LRELn) of IICA control register n0 (IICCTLn0) to 1 and saving communication).

If bit 1 (STTn) of the IICCTLn0 register is set to 1 while the bus is not used (after a stop condition is detected), a start condition is automatically generated and wait state is set.

If an address is written to the IICA shift register n (IICAn) after bit 4 (SPIEn) of the IICCTLn0 register was set to 1, and it was detected by generation of an interrupt request signal (INTIICAn) that the bus was released (detection of the stop condition), then the device automatically starts communication as the master. Data written to the IICAn register before the stop condition is detected is invalid.

When the STTn bit has been set to 1, the operation mode (as start condition or as communication reservation) is determined according to the bus status.

- If the bus has been released.....a start condition is generated
- If the bus has not been released (standby mode).....communication reservation

Check whether the communication reservation operates or not by using the MSTSn bit (bit 7 of the IICA status register n (IICSn)) after the STTn bit is set to 1 and the wait time elapses.

Use software to secure the wait time calculated by the following expression.

Wait time from setting STTn = 1 to checking the MSTSn flag:

$$(IICWL_n \text{ setting value} + IICWHL_n \text{ setting value} + 4)/f_{MCK} + t_F \times 2$$

**Remark 1.** IICWL<sub>n</sub>: IICA low-level width setting register n

IICWHL<sub>n</sub>: IICA high-level width setting register n

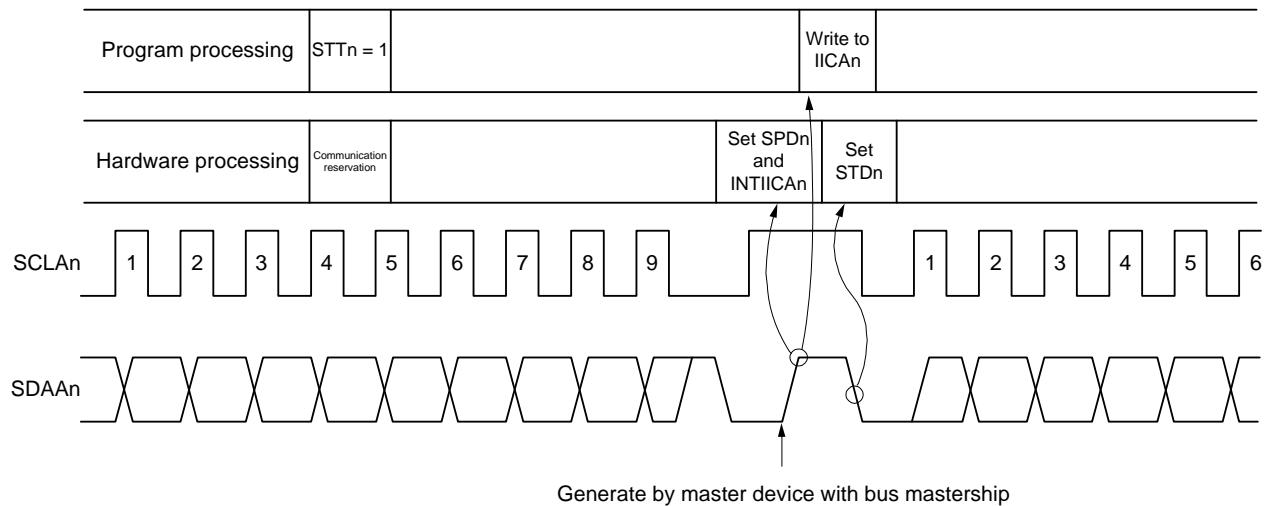
t<sub>F</sub>: SDAAn and SCLAn signal falling times

f<sub>MCK</sub>: IICA operating clock frequency

**Remark 2.** n = 0, 1

Figure 18 - 32 shows the Communication Reservation Timing.

**Figure 18 - 32 Communication Reservation Timing**



**Remark** IICAn: IICA shift register n

STTn: Bit 1 of IICA control register n0 (IICCTLn0)

STDn: Bit 1 of IICA status register n (IICSn)

SPDn: Bit 0 of IICA status register n (IICSn)

Communication reservations are accepted via the timing shown in Figure 18 - 33. After bit 1 (STDn) of the IICA status register n (IICSn) is set to 1, a communication reservation can be made by setting bit 1 (STTn) of IICA control register n0 (IICCTLn0) to 1 before a stop condition is detected.

**Figure 18 - 33 Timing for Accepting Communication Reservations**

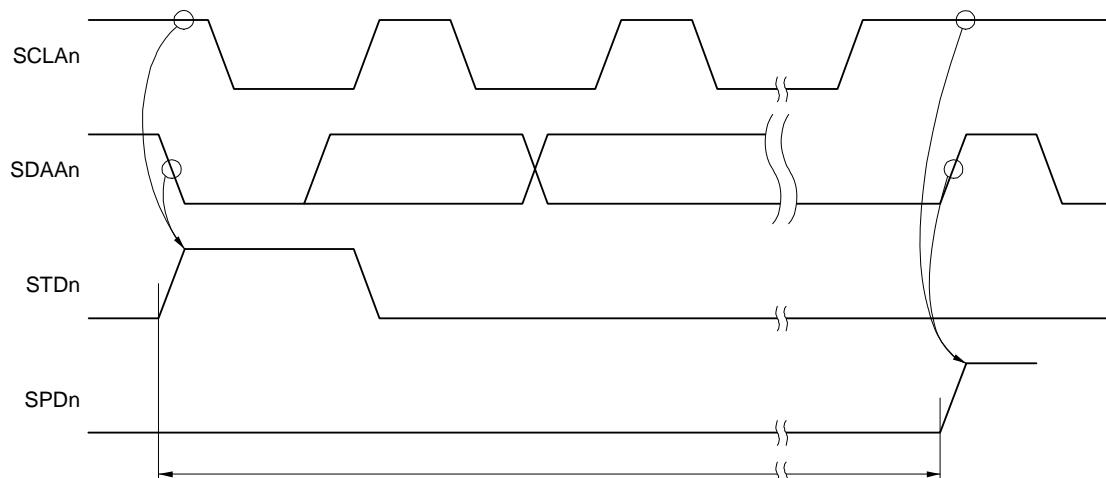
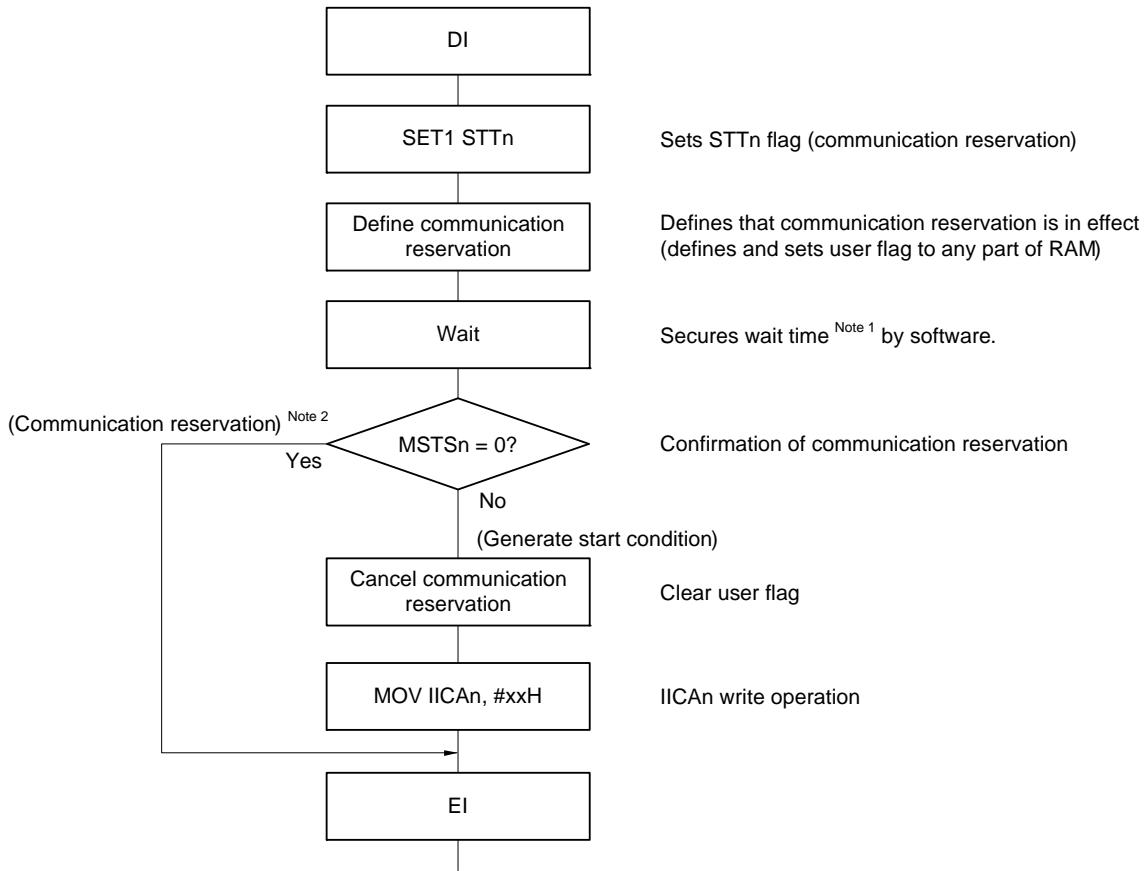


Figure 18 - 34 shows the Communication Reservation Protocol.

**Remark** n = 0, 1

**Figure 18 - 34 Communication Reservation Protocol**

**Note 1.** The wait time is calculated as follows.

$$(IICWL_n \text{ setting value} + IICWH_n \text{ setting value} + 4)/f_{MCK} + t_F \times 2$$

**Note 2.** The communication reservation operation executes a write to the IICA shift register n (IICAn) when a stop condition interrupt request occurs.

**Remark1.** STTn: Bit 1 of IICA control register n0 (IICCTLn0)

MSTS<sub>n</sub>: Bit 7 of IICA status register n (IICSn)

IICAn: IICA shift register n

IICWL<sub>n</sub>: IICA low-level width setting register n

IICWH<sub>n</sub>: IICA high-level width setting register n

t<sub>F</sub>: SDAAn and SCLAn signal falling times

f<sub>MCK</sub>: IICA operating clock frequency

**Remark2.** n = 0, 1

- (2) When communication reservation function is disabled (bit 0 (IICRSVn) of IICA flag register n (IICFn) = 1)  
When bit 1 (STTn) of IICA control register n0 (IICCTLn0) is set to 1 when the bus is not used in a communication during bus communication, this request is rejected and a start condition is not generated. The following two statuses are included in the status where bus is not used.
- When arbitration results in neither master nor slave operation
  - When an extension code is received and slave operation is disabled (ACK is not returned and the bus was released by setting bit 6 (LRELn) of the IICCTLn0 register to 1 and saving communication)

To confirm whether the start condition was generated or request was rejected, check STCFn (bit 7 of the IICFn register). It takes up to five FMCK clocks until the STCFn bit is set to 1 after setting STTn = 1. Therefore, secure this time by software.

Remark n = 0, 1

### 18.5.15 Cautions

(1) When STCENN = 0

Immediately after I<sup>2</sup>C operation is enabled (IICEn = 1), the bus communication status (IICBSYn = 1) is recognized regardless of the actual bus status. When changing from a mode in which no stop condition has been detected to a master device communication mode, first generate a stop condition to release the bus, then perform master device communication.

When using multiple masters, it is not possible to perform master device communication when the bus has not been released (when a stop condition has not been detected).

Use the following sequence for generating a stop condition.

- <1> Set IICA control register n1 (IICCTLn1).
- <2> Set bit 7 (IICEn) of IICA control register n0 (IICCTLn0) to 1.
- <3> Set bit 0 (SPTn) of the IICCTLn0 register to 1.

(2) When STCENN = 1

Immediately after I<sup>2</sup>C operation is enabled (IICEn = 1), the bus released status (IICBSYn = 0) is recognized regardless of the actual bus status. To generate the first start condition (STTn = 1), it is necessary to confirm that the bus has been released, so as to not disturb other communications.

(3) If other I<sup>2</sup>C communications are already in progress

If I<sup>2</sup>C operation is enabled and the device participates in communication already in progress when the SDAAn pin is low and the SCLAn pin is high, the macro of I<sup>2</sup>C recognizes that the SDAAn pin has gone low (detects a start condition). If the value on the bus at this time can be recognized as an extension code, ACK is returned, but this interferes with other I<sup>2</sup>C communications. To avoid this, start I<sup>2</sup>C in the following sequence.

- <1> Clear bit 4 (SPIEn) of the IICCTLn0 register to 0 to disable generation of an interrupt request signal (INTIICAn) when the stop condition is detected.
- <2> Set bit 7 (IICEn) of the IICCTLn0 register to 1 to enable the operation of I<sup>2</sup>C.
- <3> Wait for detection of the start condition.
- <4> Set bit 6 (LRELn) of the IICCTLn0 register to 1 before ACK is returned (4 to 72 fmck clocks after setting the IICEn bit to 1), to forcibly disable detection.

(4) Setting the STTn and SPTn bits (bits 1 and 0 of the IICCTLn0 register) again after they are set and before they are cleared to 0 is prohibited.

(5) When transmission is reserved, set the SPIEn bit (bit 4 of the IICCTLn0 register) to 1 so that an interrupt request is generated when the stop condition is detected. Transfer is started when communication data is written to the IICA shift register n (IICAn) after the interrupt request is generated. Unless the interrupt is generated when the stop condition is detected, the device stops in the wait state because the interrupt request is not generated when communication is started. However, it is not necessary to set the SPIEn bit to 1 when the MSTSn bit (bit 7 of the IICA status register n (IICSn)) is detected by software.

Remark n = 0, 1

### 18.5.16 Communication operations

The following shows three operation procedures with the flowchart.

(1) Master operation in single master system

The flowchart when using the RL78/G14 as the master in a single master system is shown below.

This flowchart is broadly divided into the initial settings and communication processing. Execute the initial settings at startup. If communication with the slave is required, prepare the communication and then execute communication processing.

(2) Master operation in multimaster system

In the I<sup>2</sup>C bus multimaster system, whether the bus is released or used cannot be judged by the I<sup>2</sup>C bus specifications when the bus takes part in a communication. Here, when data and clock are at a high level for a certain period (1 frame), the RL78/G14 takes part in a communication with bus released state.

This flowchart is broadly divided into the initial settings, communication waiting, and communication processing. The processing when the RL78/G14 loses in arbitration and is specified as the slave is omitted here, and only the processing as the master is shown. Execute the initial settings at startup to take part in a communication. Then, wait for the communication request as the master or wait for the specification as the slave. The actual communication is performed in the communication processing, and it supports the transmission/reception with the slave and the arbitration with other masters.

(3) Slave operation

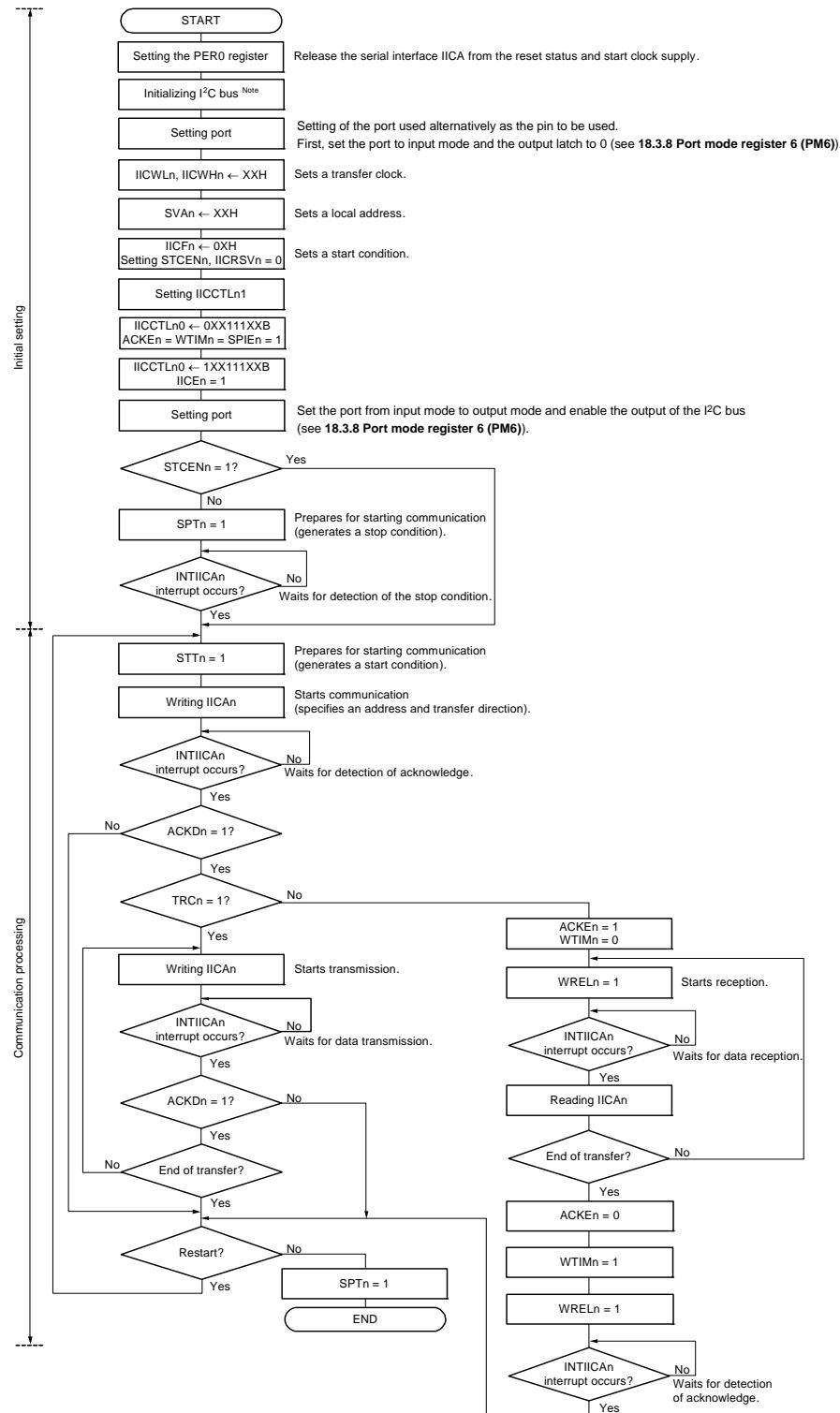
An example of when the RL78/G14 is used as the I<sup>2</sup>C bus slave is shown below.

When used as the slave, operation is started by an interrupt. Execute the initial settings at startup, then wait for the INTIICAn interrupt occurrence (communication waiting). When an INTIICAn interrupt occurs, the communication status is judged and its result is passed as a flag over to the main processing.

By checking the flags, necessary communication processing is performed.

Remark n = 0, 1

## (1) Master operation in single master system

**Figure 18 - 35 Master Operation in Single-Master System**

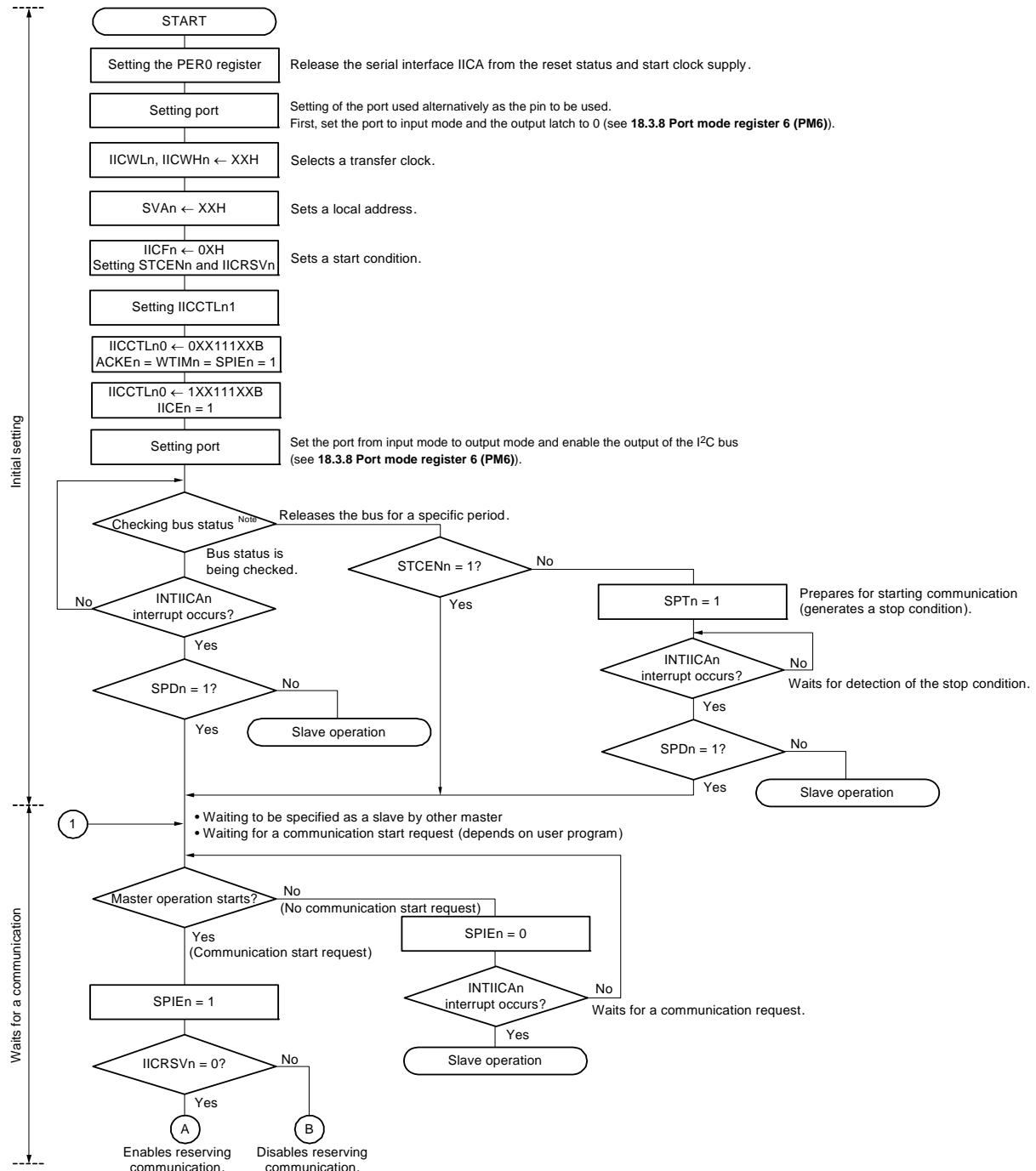
**Note** Release (SCLAn and SDAAn pins = high level) the I<sup>2</sup>C bus in conformance with the specifications of the product that is communicating. If EEPROM is outputting a low level to the SDAAn pin, for example, set the SCLAn pin in the output port mode, and output a clock pulse from the output port until the SDAAn pin is constantly at high level.

**Remark1.** Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

**Remark2.** n = 0, 1

## (2) Master operation in multimaster system

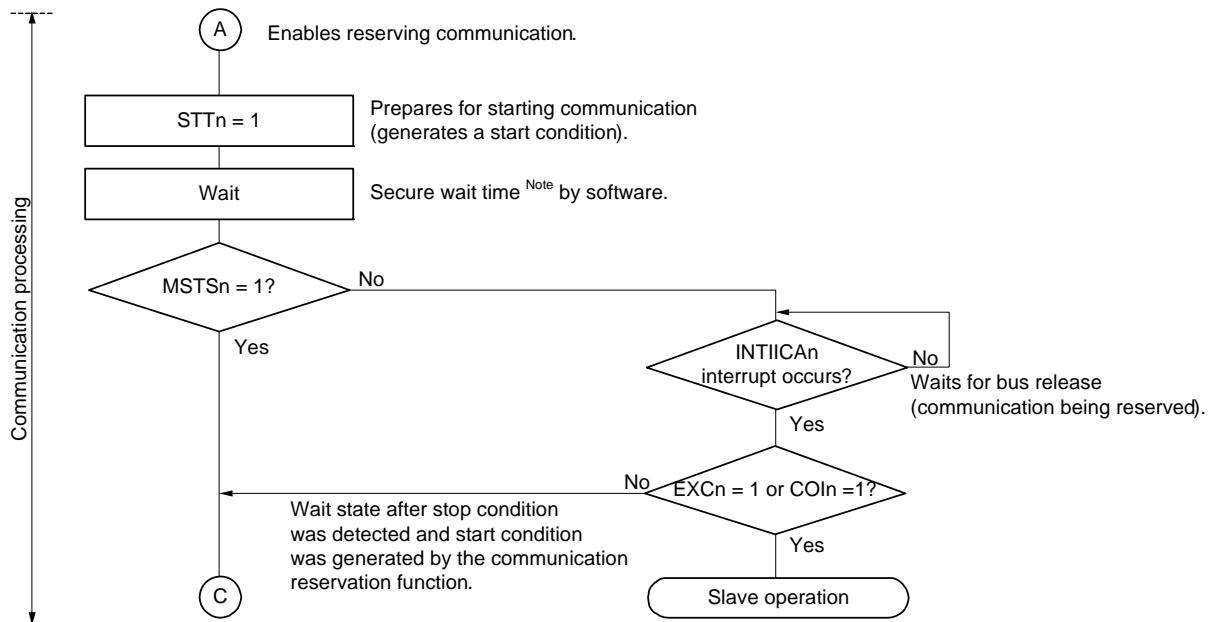
Figure 18 - 36 Master Operation in Multi-Master System (1/3)



**Note** Confirm that the bus is released (CLD<sub>n</sub> bit = 1, DAD<sub>n</sub> bit = 1) for a specific period (for example, for a period of one frame). If the SDA<sub>An</sub> pin is constantly at low level, decide whether to release the I<sup>2</sup>C bus (SCL<sub>An</sub> and SDA<sub>An</sub> pins = high level) in conformance with the specifications of the product that is communicating.

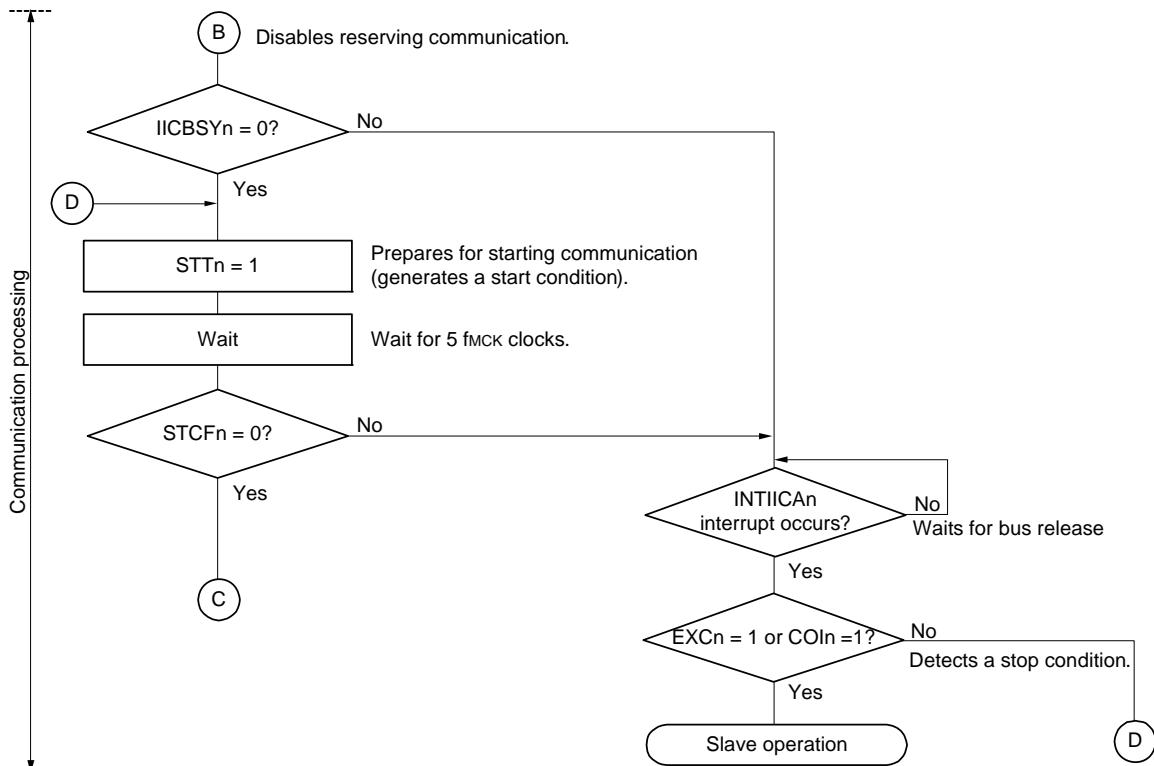
**Remark** n = 0, 1

Figure 18 - 37 Master Operation in Multi-Master System (2/3)



**Note** The wait time is calculated as follows.  

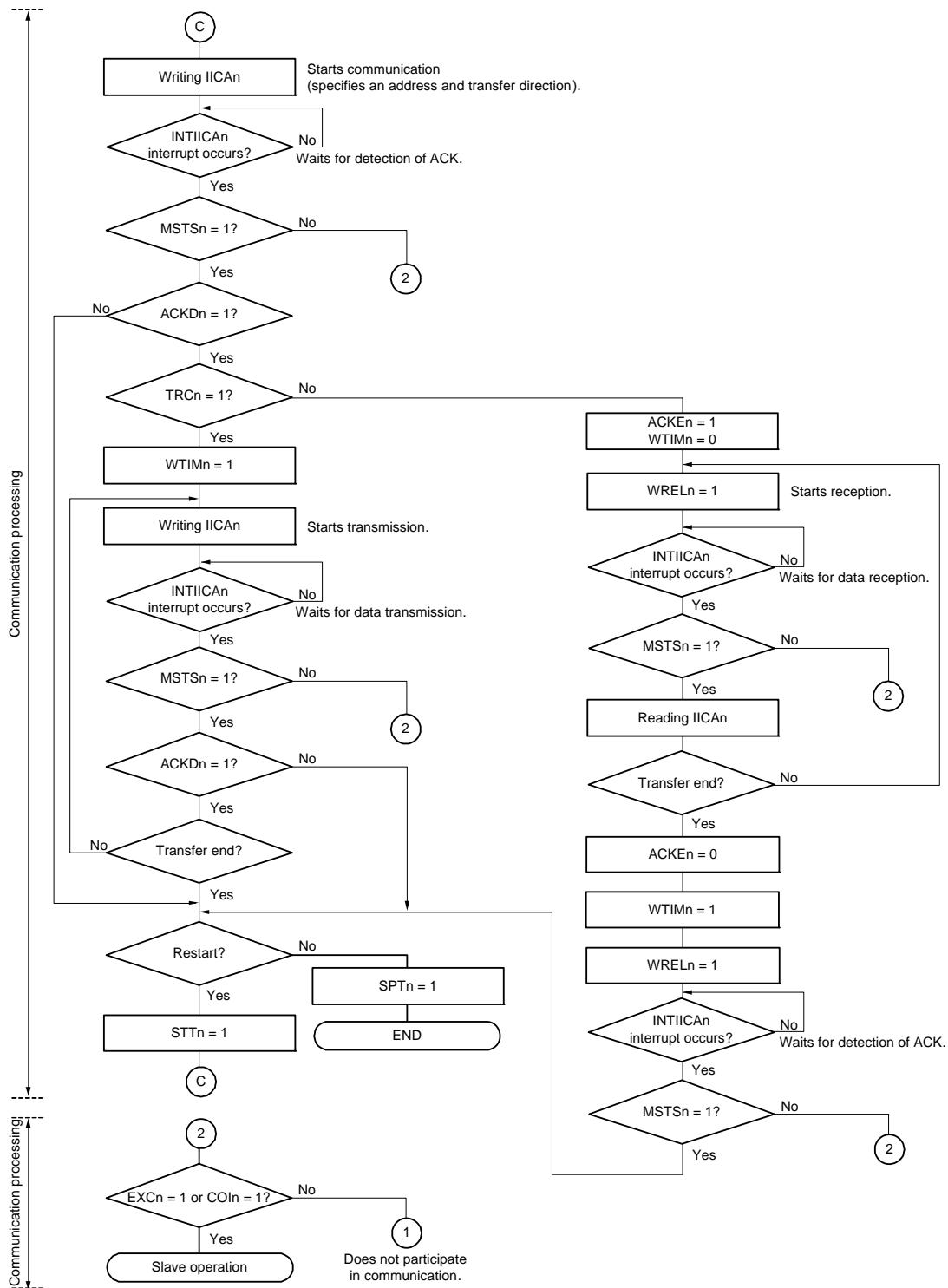
$$(\text{IICWL}_n \text{ setting value} + \text{IICWH}_n \text{ setting value} + 4) / \text{fmck} + t_F \times 2$$



**Remark1.** IICWL<sub>n</sub>: IICA low-level width setting register n  
 IICWH<sub>n</sub>: IICA high-level width setting register n  
 t<sub>F</sub>: SDAAn and SCLAn signal falling times  
 fmck: IICA operating clock frequency

**Remark2.** n = 0, 1

Figure 18 - 38 Master Operation in Multi-Master System (3/3)



**Remark 1.** Conform to the specifications of the product that is communicating, with respect to the transmission and reception formats.

**Remark 2.** To use the device as a master in a multi-master system, read the MSTSn bit each time interrupt INTIICAn has occurred to check the arbitration result.

**Remark 3.** To use the device as a slave in a multi-master system, check the status by using the IICA status register n (IICSn) and IICA flag register n (IICFn) each time interrupt INTIICAn has occurred, and determine the processing to be performed next.

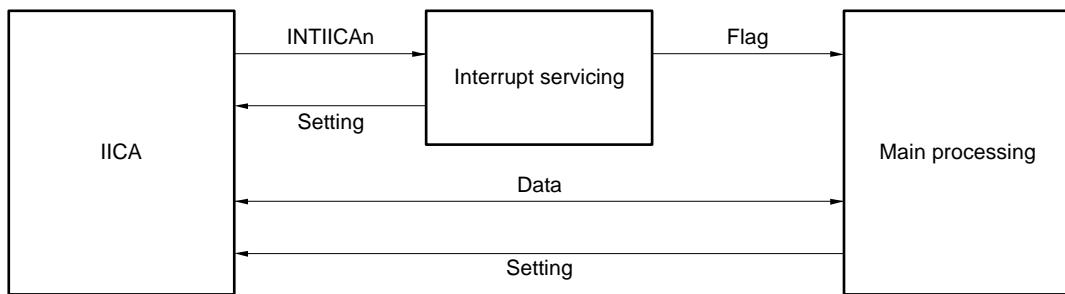
**Remark 4.**  $n = 0, 1$

(3) Slave operation

The processing procedure of the slave operation is as follows.

Basically, the slave operation is event-driven. Therefore, processing by the INTIICAn interrupt (processing that must substantially change the operation status such as detection of a stop condition during communication) is necessary.

In the following explanation, it is assumed that the extension code is not supported for data communication. It is also assumed that the INTIICAn interrupt servicing only performs status transition processing, and that actual data communication is performed by the main processing.



Therefore, data communication processing is performed by preparing the following three flags and passing them to the main processing instead of INTIICAn.

<1> Communication mode flag

This flag indicates the following two communication statuses.

- Clear mode: Status in which data communication is not performed
- Communication mode: Status in which data communication is performed (from valid address detection to stop condition detection, no detection of ACK from master, address mismatch)

<2> Ready flag

This flag indicates that data communication is enabled. Its function is the same as the INTIICAn interrupt for ordinary data communication. This flag is set by interrupt servicing and cleared by the main processing. Clear this flag by interrupt servicing when communication is started. However, the ready flag is not set by interrupt servicing when the first data is transmitted. Therefore, the first data is transmitted without the flag being cleared (an address match is interpreted as a request for the next data).

<3> Communication direction flag

This flag indicates the direction of communication. Its value is the same as the TRCn bit.

**Remark** n = 0, 1

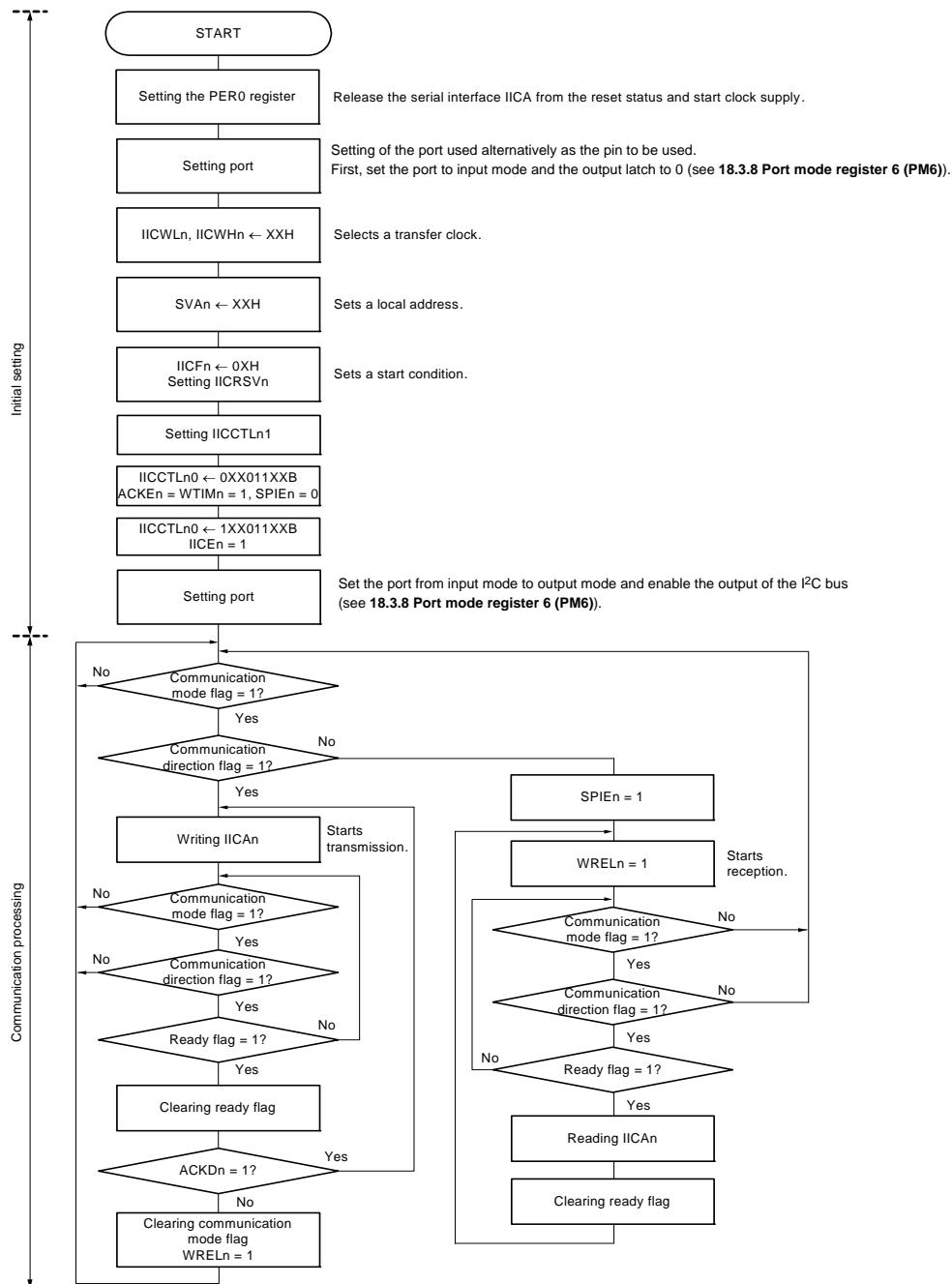
The main processing of the slave operation is explained next.

Start serial interface IICA and wait until communication is enabled. When communication is enabled, execute communication by using the communication mode flag and ready flag (processing of the stop condition and start condition is performed by an interrupt. Here, check the status by using the flags).

The transmission operation is repeated until the master no longer returns ACK. If ACK is not returned from the master, communication is completed.

For reception, the necessary amount of data is received. When communication is completed, ACK is not returned as the next data. After that, the master generates a stop condition or restart condition. Exit from the communication status occurs in this way.

**Figure 18 - 39 Slave Operation Flowchart (1)**



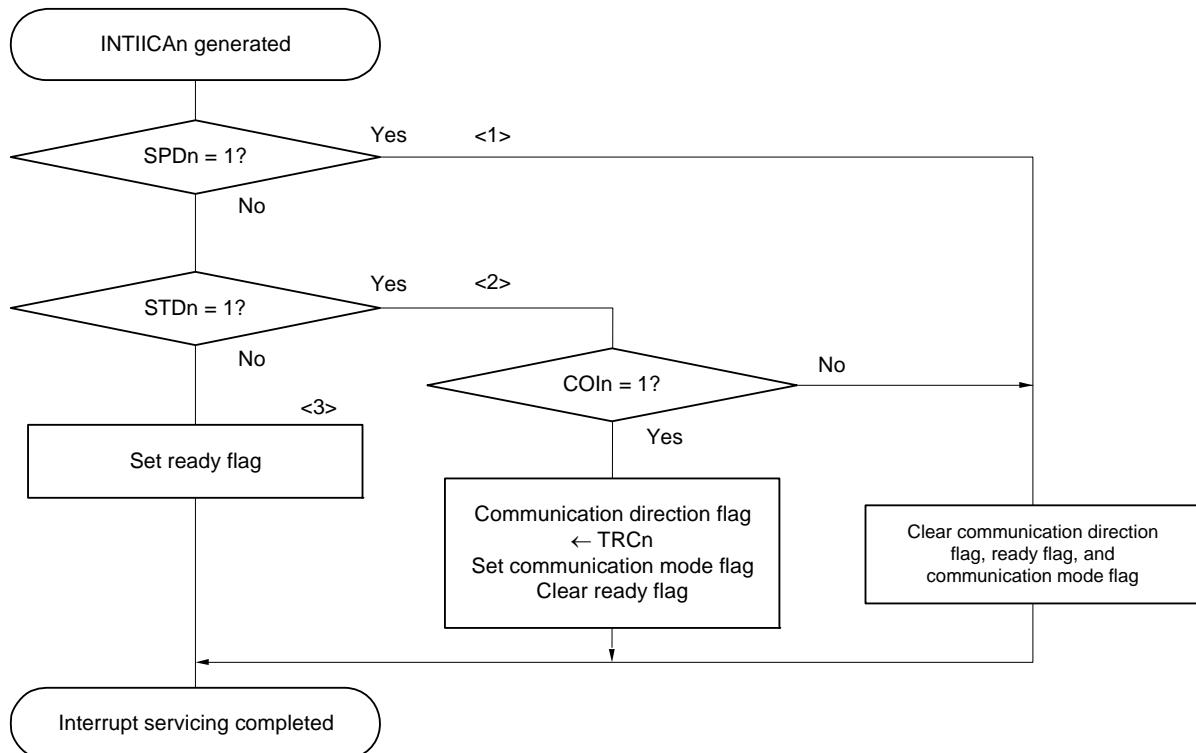
**Remark1.** Conform to the specifications of the product that is in communication, regarding the transmission and reception formats.  
**Remark2.** n = 0, 1

An example of the processing procedure of the slave with the INTIICAn interrupt is explained below (processing is performed assuming that no extension code is used). The INTIICAn interrupt checks the status, and the following operations are performed.

- <1> Communication is stopped if the stop condition is issued.
- <2> If the start condition is issued, the address is checked and communication is completed if the address does not match. If the address matches, the communication mode is set, wait is cancelled, and processing returns from the interrupt (the ready flag is cleared).
- <3> For data transmit/receive, only the ready flag is set. Processing returns from the interrupt with the I<sup>2</sup>C bus remaining in the wait state.

**Remark** <1> to <3> above correspond to <1> to <3> in Figure 18 - 40 Slave Operation Flowchart (2).

**Figure 18 - 40 Slave Operation Flowchart (2)**



**Remark** n = 0, 1

### 18.5.17 Timing of I<sup>2</sup>C interrupt request (INTIICAn) occurrence

The timing of transmitting or receiving data and generation of interrupt request signal INTIICAn, and the value of the IICA status register n (IICSn) when the INTIICAn signal is generated are shown below.

- Remark 1.** ST: Start condition  
AD6 to AD0: Address  
R/W: Transfer direction specification  
ACK: Acknowledge  
D7 to D0: Data  
SP: Stop condition

**Remark 2.** n = 0, 1

## (1) Master device operation

## (a) Start ~ Address ~ Data ~ Data ~ Stop (transmission/reception)

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1	▲2		▲3	▲4	△5

SPTn = 1  
↓

▲1: IICSn = 1000×110B  
 ▲2: IICSn = 1000×000B  
 ▲3: IICSn = 1000×000B (Sets the WTIMn bit to 1) Note  
 ▲4: IICSn = 1000××00B (Sets the SPTn bit to 1) Note  
 △5: IICSn = 00000001B

**Note** To generate a stop condition, set the WTIMn bit to 1 and change the timing for generating the INTIICAn interrupt request signal.

**Remark** ▲: Always generated  
△: Generated only when SPIEn = 1  
×: Don't care

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1	▲2		▲3	△4	

SPTn = 1  
↓

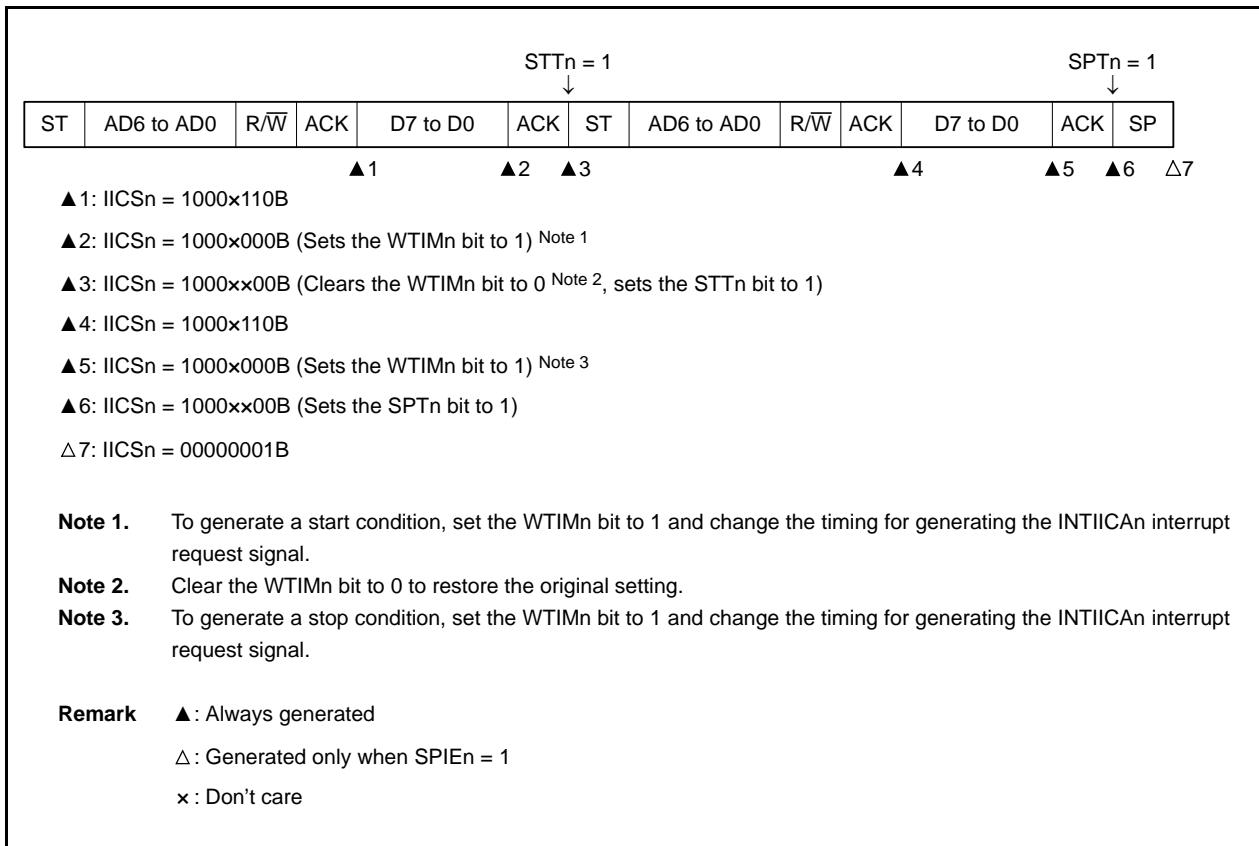
▲1: IICSn = 1000×110B  
 ▲2: IICSn = 1000×100B  
 ▲3: IICSn = 1000××00B (Sets the SPTn bit to 1)  
 △4: IICSn = 00000001B

**Remark** ▲: Always generated  
△: Generated only when SPIEn = 1  
×: Don't care

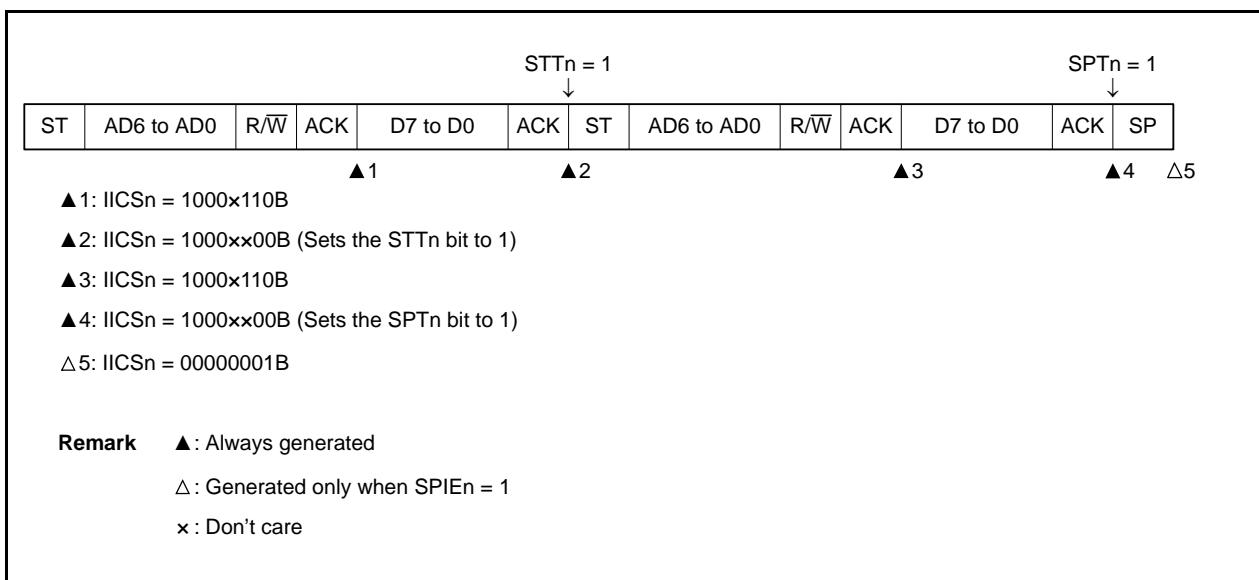
**Remark** n = 0, 1

- (b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop (restart)

- (i) When  $WTIMn = 0$



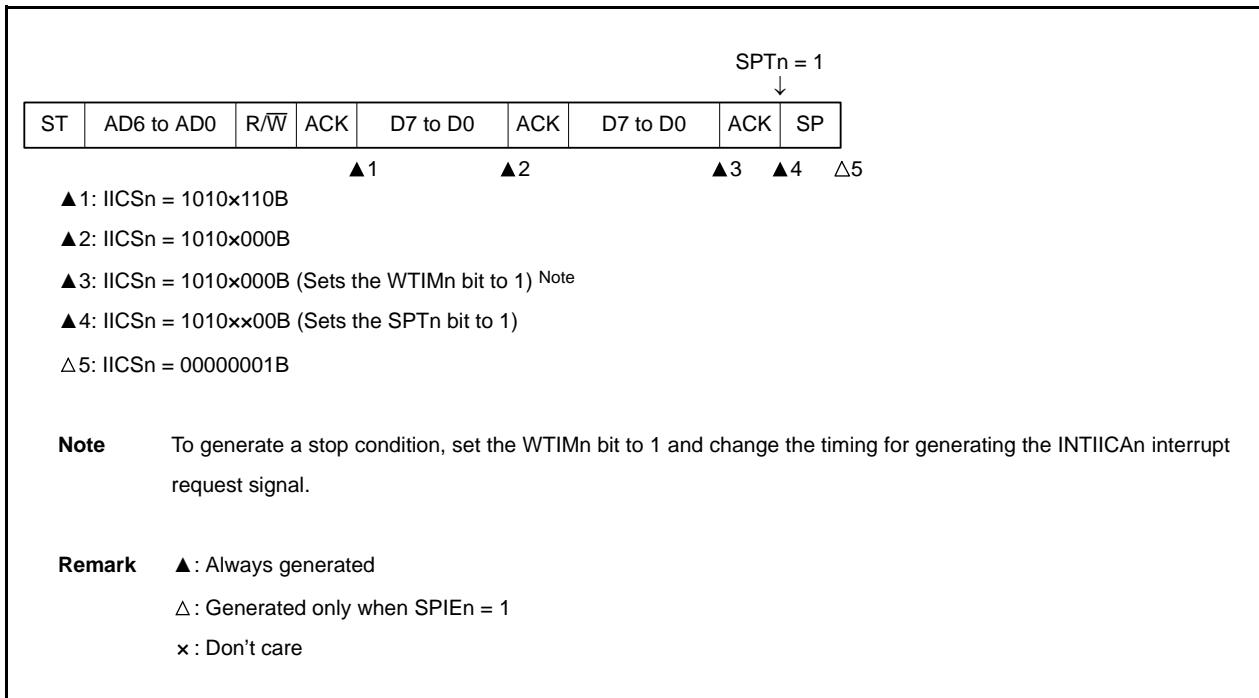
- (ii) When  $WTIMn = 1$



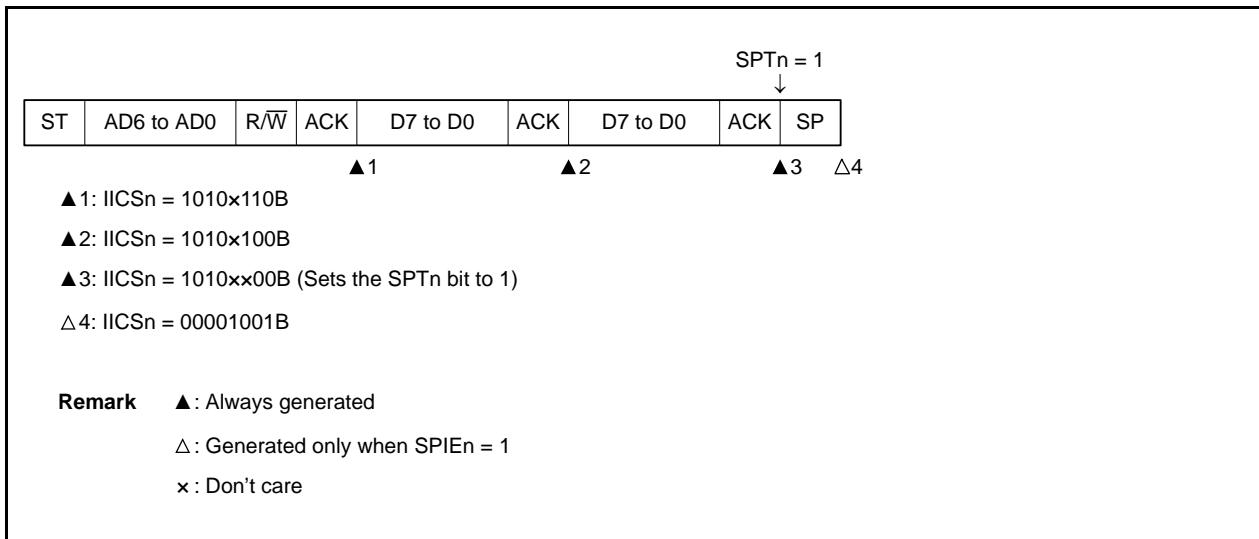
**Remark**     $n = 0, 1$

(c) Start ~ Code ~ Data ~ Data ~ Stop (extension code transmission)

(i) When WTIMn = 0



(ii) When WTIMn = 1



**Remark** n = 0, 1

## (2) Slave device operation (slave address data reception)

## (a) Start ~ Address ~ Data ~ Data ~ Stop

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

▲1                   ▲2                   ▲3                   △4

▲1: IICSn = 0001x110B

▲2: IICSn = 0001x000B

▲3: IICSn = 0001x000B

△4: IICSn = 00000001B

**Remark**   ▲: Always generated

△: Generated only when SPIEn = 1

x : Don't care

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

▲1                   ▲2                   ▲3    △4

▲1: IICSn = 0001x110B

▲2: IICSn = 0001x100B

▲3: IICSn = 0001xx00B

△4: IICSn = 00000001B

**Remark**   ▲: Always generated

△: Generated only when SPIEn = 1

x : Don't care

**Remark**   n = 0, 1

(b) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, matches with SVAn)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICSn = 0001x110B  
 ▲2: IICSn = 0001x000B  
 ▲3: IICSn = 0001x110B  
 ▲4: IICSn = 0001x000B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗ : Don't care

(ii) When WTIMn = 1 (after restart, matches with SVAn)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICSn = 0001x110B  
 ▲2: IICSn = 0001xx00B  
 ▲3: IICSn = 0001x110B  
 ▲4: IICSn = 0001xx00B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗ : Don't care

**Remark**    n = 0, 1

(c) Start ~ Address ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, does not match address (= extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3		▲4	△5

▲1: IICSn = 0001×110B  
 ▲2: IICSn = 0001×000B  
 ▲3: IICSn = 0010×010B  
 ▲4: IICSn = 0010×000B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

(ii) When WTIMn = 1 (after restart, does not match address (= extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3	▲4		▲5 △6

▲1: IICSn = 0001×110B  
 ▲2: IICSn = 0001×x00B  
 ▲3: IICSn = 0010×010B  
 ▲4: IICSn = 0010×110B  
 ▲5: IICSn = 0010×x00B  
 △6: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

**Remark**    n = 0, 1

(d) Start ~ Address ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, does not match address (= not extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3			△4

▲1: IICSn = 0001x110B  
 ▲2: IICSn = 0001x000B  
 ▲3: IICSn = 00000110B  
 △4: IICSn = 00000001B

**Remark**    ▲: Always generated  
              △: Generated only when SPIEn = 1  
              x: Don't care

(ii) When WTIMn = 1 (after restart, does not match address (= not extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
			▲1		▲2				▲3			△4

▲1: IICSn = 0001x110B  
 ▲2: IICSn = 0001x000B  
 ▲3: IICSn = 00000110B  
 △4: IICSn = 00000001B

**Remark**    ▲: Always generated  
              △: Generated only when SPIEn = 1  
              x: Don't care

**Remark**    n = 0, 1

## (3) Slave device operation (when receiving extension code)

The device is always participating in communication when it receives an extension code.

## (a) Start ~ Code ~ Data ~ Data ~ Stop

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1		▲2		▲3	△4

▲1: IICSn = 0010x010B  
 ▲2: IICSn = 0010x000B  
 ▲3: IICSn = 0010x000B  
 △4: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1	▲2		▲3	▲4	△5

▲1: IICSn = 0010x010B  
 ▲2: IICSn = 0010x110B  
 ▲3: IICSn = 0010x100B  
 ▲4: IICSn = 0010xx00B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

**Remark**    n = 0, 1

(b) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, matches SVAn)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
▲1			▲2						▲3		▲4	△5

▲1: IICSn = 0010×010B  
 ▲2: IICSn = 0010×000B  
 ▲3: IICSn = 0001×110B  
 ▲4: IICSn = 0001×000B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

(ii) When WTIMn = 1 (after restart, matches SVAn)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
▲1	▲2				▲3				▲4		▲5	△6

▲1: IICSn = 0010×010B  
 ▲2: IICSn = 0010×110B  
 ▲3: IICSn = 0010××00B  
 ▲4: IICSn = 0001×110B  
 ▲5: IICSn = 0001××00B  
 △6: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

**Remark**    n = 0, 1

(c) Start ~ Code ~ Data ~ Start ~ Code ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, extension code reception)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
		▲1		▲2				▲3		▲4		△5

▲1: IICSn = 0010x010B  
 ▲2: IICSn = 0010x000B  
 ▲3: IICSn = 0010x010B  
 ▲4: IICSn = 0010x000B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

(ii) When WTIMn = 1 (after restart, extension code reception)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
		▲1	▲2		▲3			▲4	▲5		▲6	△7

▲1: IICSn = 0010x010B  
 ▲2: IICSn = 0010x110B  
 ▲3: IICSn = 0010xx00B  
 ▲4: IICSn = 0010x010B  
 ▲5: IICSn = 0010x110B  
 ▲6: IICSn = 0010xx00B  
 △7: IICSn = 00000001B

**Remark**    ▲: Always generated

△: Generated only when SPIEn = 1

✗: Don't care

**Remark**    n = 0, 1

(d) Start ~ Code ~ Data ~ Start ~ Address ~ Data ~ Stop

(i) When WTIMn = 0 (after restart, does not match address (= not extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
		▲1		▲2					▲3			△4

▲1: IICSn = 0010 × 010B  
 ▲2: IICSn = 0010 × 000B  
 ▲3: IICSn = 00000 × 10B  
 △4: IICSn = 00000001B

**Remark**    ▲: Always generated  
 △: Generated only when SPIEn = 1  
 × : Don't care

(ii) When WTIMn = 1 (after restart, does not match address (= not extension code))

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
		▲1	▲2		▲3				▲4			△5

▲1: IICSn = 0010 × 010B  
 ▲2: IICSn = 0010 × 110B  
 ▲3: IICSn = 0010 × × 00B  
 ▲4: IICSn = 00000 × 10B  
 △5: IICSn = 00000001B

**Remark**    ▲: Always generated  
 △: Generated only when SPIEn = 1  
 × : Don't care

**Remark**    n = 0, 1

## (4) Operation without communication

## (a) Start ~ Code ~ Data ~ Data ~ Stop

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

△1

△1: IICSn = 00000001B

**Remark** △: Generated only when SPIEn = 1

## (5) Arbitration loss operation (operation as slave after arbitration loss)

When the device is used as a master in a multi-master system, read the MSTSn bit each time interrupt request signal INTIICAn has occurred to check the arbitration result.

## (a) When arbitration loss occurs during transmission of slave address data

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

▲1

▲2

▲3

△4

▲1: IICSn = 0101x110B

▲2: IICSn = 0001x000B

▲3: IICSn = 0001x000B

△4: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

x: Don't care

**Remark** n = 0, 1

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1		▲2		▲3	△4

▲1: IICSn = 0101x110B

▲2: IICSn = 0001x100B

▲3: IICSn = 0001xx00B

△4: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

x: Don't care

## (b) When arbitration loss occurs during transmission of extension code

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1		▲2		▲3	△4

▲1: IICSn = 0110x010B

▲2: IICSn = 0010x000B

▲3: IICSn = 0010x000B

△4: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

x: Don't care

**Remark** n = 0, 1

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1	▲2		▲3		▲4 △5

▲1: IICSn = 0110x010B  
 ▲2: IICSn = 0010x110B  
 ▲3: IICSn = 0010x100B  
 ▲4: IICSn = 0010xx00B  
 △5: IICSn = 00000001B

**Remark**

- ▲: Always generated
- △: Generated only when SPIEn = 1
- ×: Don't care

## (6) Operation when arbitration loss occurs (no communication after arbitration loss)

When the device is used as a master in a multi-master system, read the MSTSn bit each time interrupt request signal INTIICAn has occurred to check the arbitration result.

## (a) When arbitration loss occurs during transmission of slave address data (when WTIMn = 1)

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1				△2	

▲1: IICSn = 01000110B  
 △2: IICSn = 00000001B

**Remark**

- ▲: Always generated
- △: Generated only when SPIEn = 1

**Remark** n = 0, 1

## (b) When arbitration loss occurs during transmission of extension code

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

▲1

△2

▲1: IICSn = 0110x010B

Sets LRELn = 1 by software

△2: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

x: Don't care

## (c) When arbitration loss occurs during transmission of data

## (i) When WTIMn = 0

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	-----	----------	-----	----

▲1

▲2

△3

▲1: IICSn = 10001110B

▲2: IICSn = 01000000B

△3: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

**Remark** n = 0, 1

## (ii) When WTIMn = 1

ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	D7 to D0	ACK	SP
			▲1		▲2			△3
▲1: IICSn = 10001110B								
▲2: IICSn = 01000100B								

△3: IICSn = 00000001B

**Remark** ▲: Always generated  
△: Generated only when SPIEn = 1

## (d) When loss occurs due to restart condition during data transfer

## (i) Not extension code (Example: unmatches with SVAn)

ST	AD6 to AD0	R/W	ACK	D7 to Dm	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
				▲1				▲2			△3
▲1: IICSn = 1000x110B											
▲2: IICSn = 01000110B											
△3: IICSn = 00000001B											

▲: Always generated  
△: Generated only when SPIEn = 1  
x: Don't care  
m = 6 to 0

**Remark** n = 0, 1

## (ii) Extension code

ST	AD6 to AD0	R/W	ACK	D7 to Dm	ST	AD6 to AD0	R/W	ACK	D7 to D0	ACK	SP
----	------------	-----	-----	----------	----	------------	-----	-----	----------	-----	----

▲1

▲2

△3

▲1: IICSn = 1000x110B

▲2: IICSn = 01100010B

Sets LRELn = 1 by software

△3: IICSn = 00000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

x: Don't care

m = 6 to 0

## (e) When loss occurs due to stop condition during data transfer

ST	AD6 to AD0	R/W	ACK	D7 to Dm	SP
----	------------	-----	-----	----------	----

▲1

△2

▲1: IICSn = 10000110B

△2: IICSn = 01000001B

**Remark** ▲: Always generated

△: Generated only when SPIEn = 1

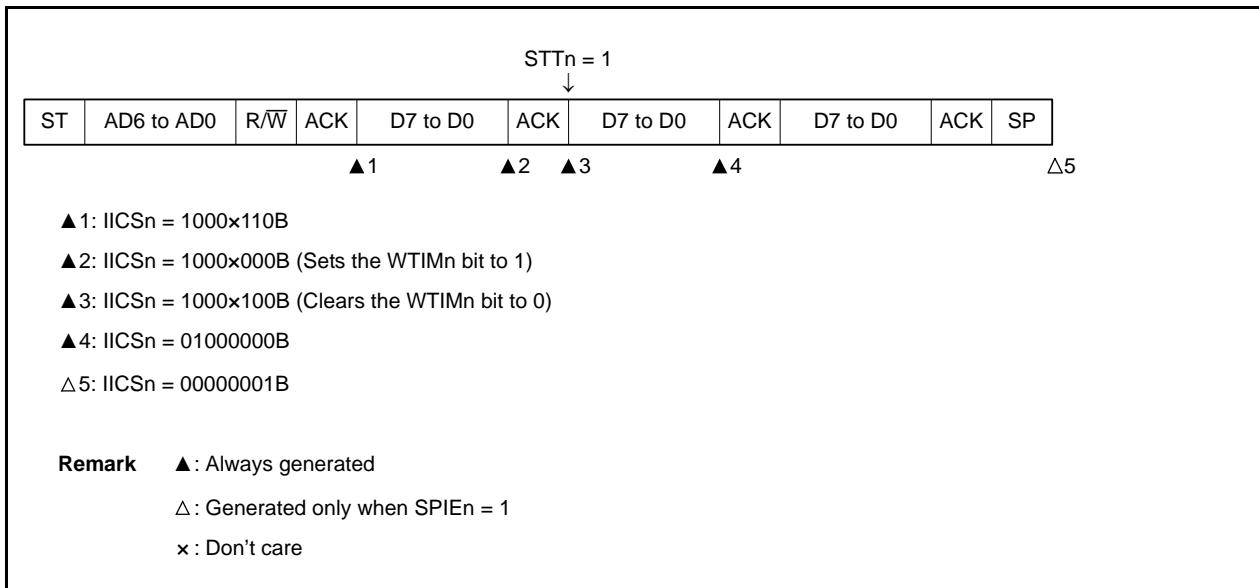
x: Don't care

m = 6 to 0

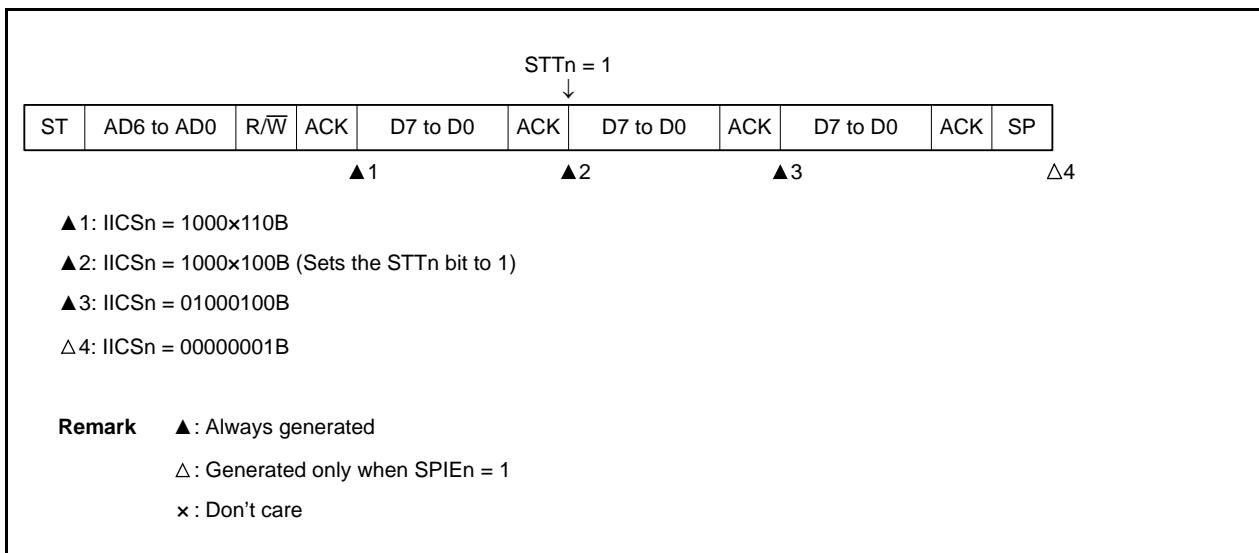
**Remark** n = 0, 1

(f) When arbitration loss occurs due to low-level data when attempting to generate a restart condition

(i) When WTIMn = 0



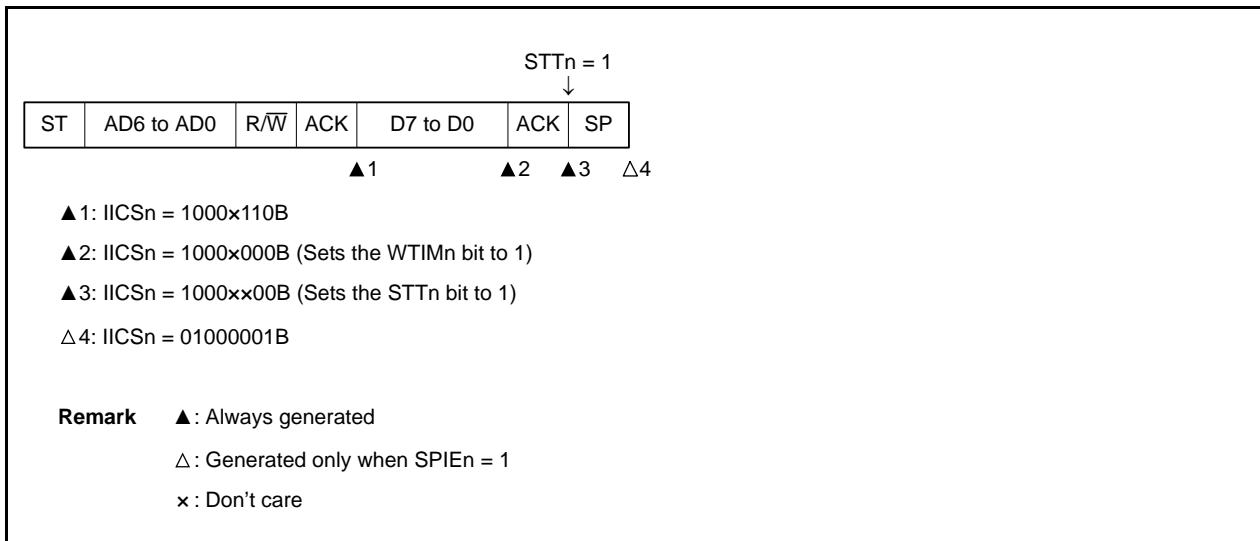
(ii) When WTIMn = 1



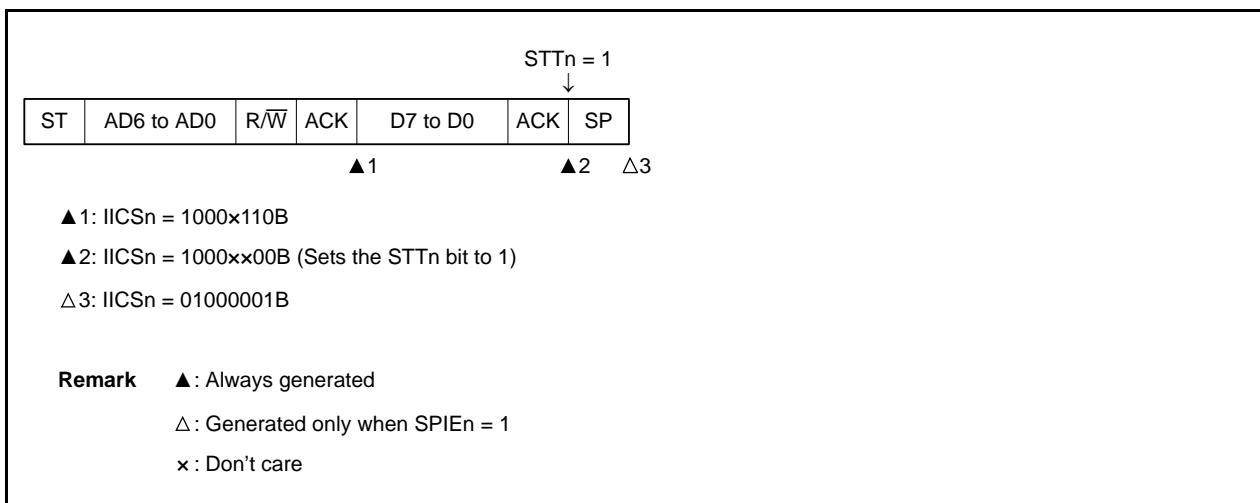
**Remark**    n = 0, 1

(g) When arbitration loss occurs due to a stop condition when attempting to generate a restart condition

(i) When  $\text{WTIMn} = 0$



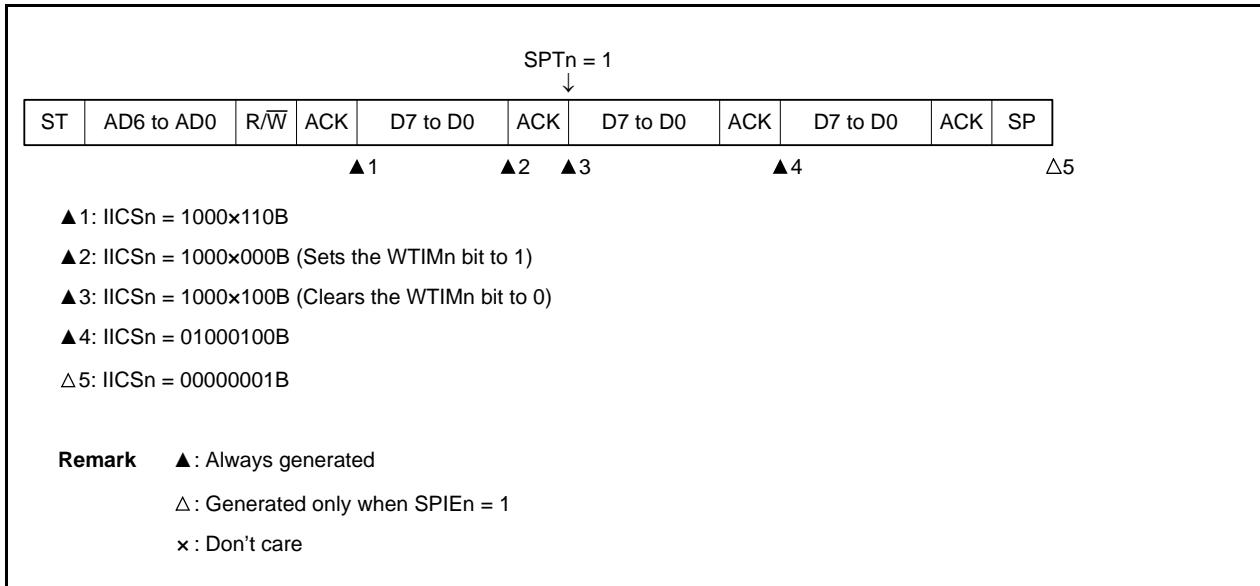
(ii) When  $\text{WTIMn} = 1$



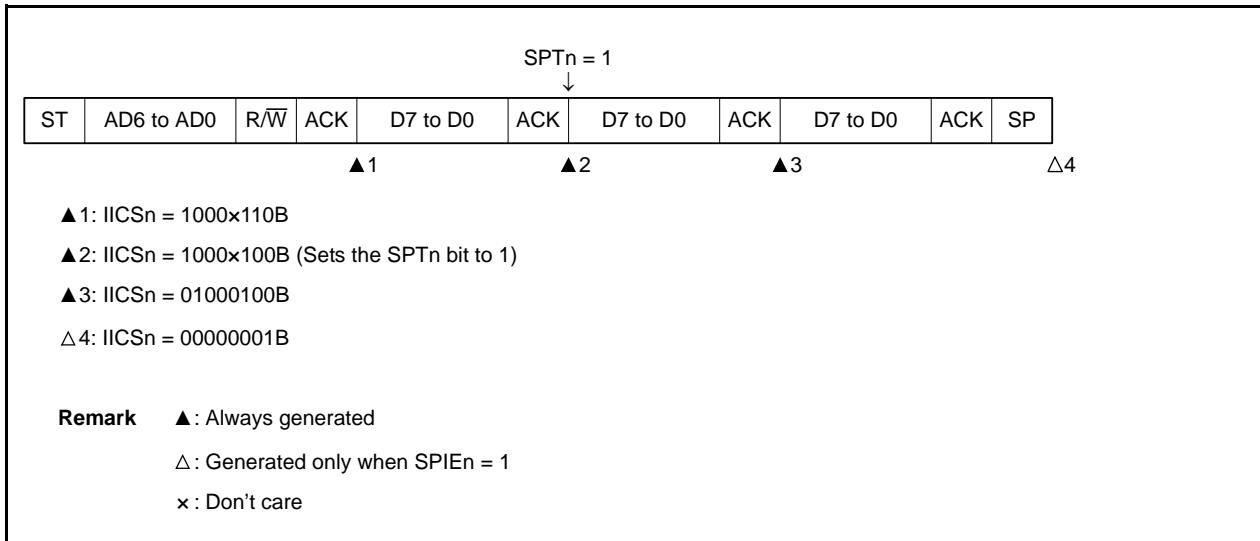
**Remark** n = 0, 1

(h) When arbitration loss occurs due to low-level data when attempting to generate a stop condition

(i) When  $\text{WTIMn} = 0$



(ii) When  $\text{WTIMn} = 1$



**Remark**    n = 0, 1

## 18.6 Timing Charts

When using the I<sup>2</sup>C bus mode, the master device outputs an address via the serial bus to select one of several slave devices as its communication partner.

After outputting the slave address, the master device transmits the TRCn bit (bit 3 of the IICA status register n (IICSn)), which specifies the data transfer direction, and then starts serial communication with the slave device.

Figures 18 - 41 to 18 - 47 show timing charts of the data communication.

The IICA shift register n (IICAn)'s shift operation is synchronized with the falling edge of the serial clock (SCLAn).

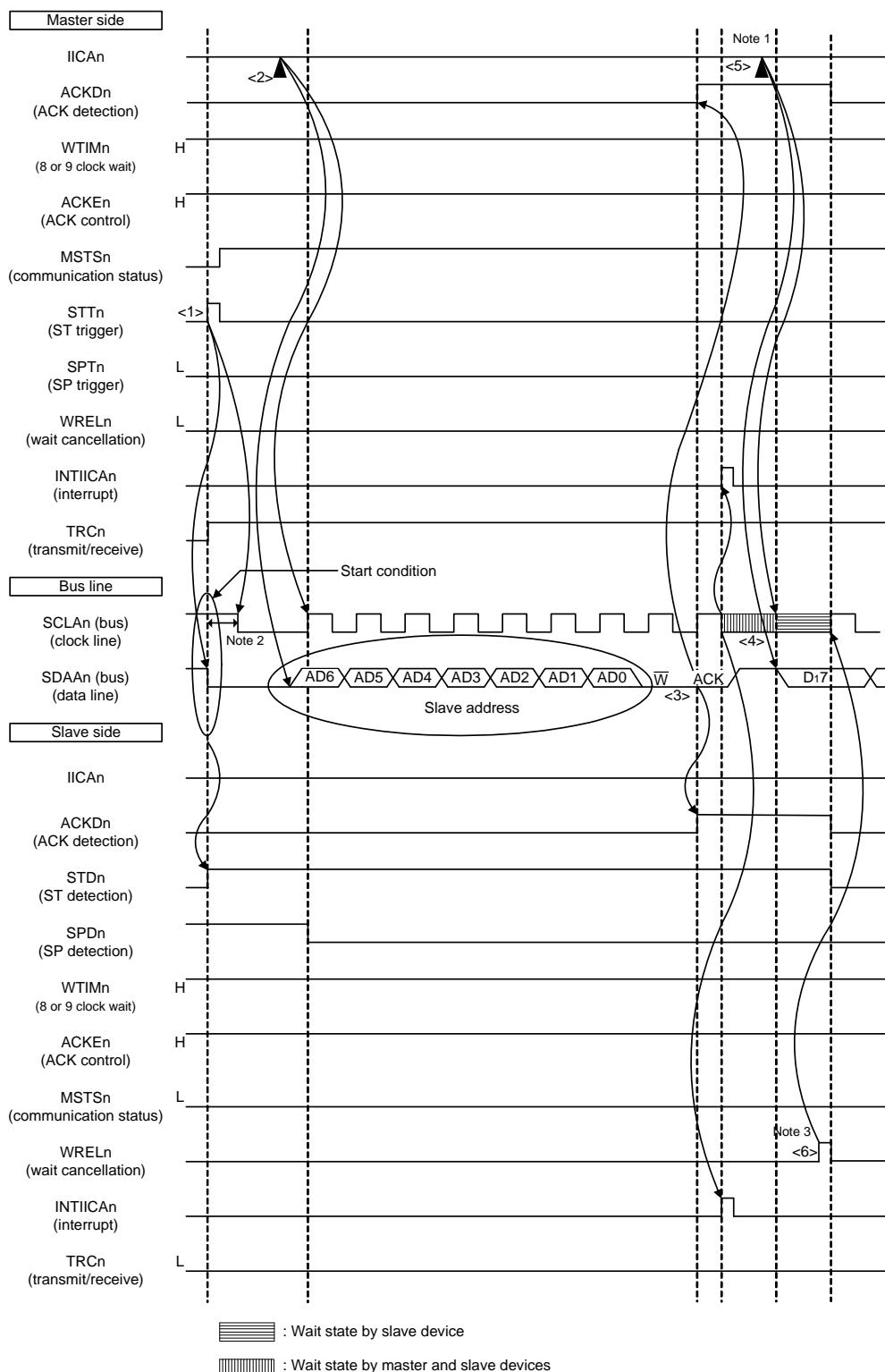
The transmit data is transferred to the SO latch and is output (MSB first) via the SDAAn pin.

Data input via the SDAAn pin is captured into IICAn at the rising edge of SCLAn.

**Remark** n = 0, 1

**Figure 18 - 41 Example of Master to Slave Communication  
(When 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (1/4)**

(1) Start condition ~ address ~ data



- Note 1.** Write data to IICAn, not setting the WRELn bit, in order to cancel a wait state during transmission by a master device.
- Note 2.** Make sure that the time between the fall of the SDAAn pin signal and the fall of the SCLAn pin signal is at least 4.0  $\mu$ s when specifying standard mode and at least 0.6  $\mu$ s when specifying fast mode.
- Note 3.** For releasing wait state during reception of a slave device, write "FFH" to IICAn or set the WRELn bit.
- Remark**  $n = 0, 1$

The meanings of <1> to <6> in (1) Start condition ~ address ~ data in Figure 18 - 41 are explained below.

- <1> The start condition trigger is set by the master device ( $STTn = 1$ ) and a start condition (i.e.  $SCLAn = 1$  changes  $SDAAn$  from 1 to 0) is generated once the bus data line goes low ( $SDAAn$ ). When the start condition is subsequently detected, the master device enters the master device communication status ( $MSTS_n = 1$ ). The master device is ready to communicate once the bus clock line goes low ( $SCLAn = 0$ ) after the hold time has elapsed.
- <2> The master device writes the address + W (transmission) to the IICA shift register n ( $IICAn$ ) and transmits the slave address.
- <3> In the slave device if the address received matches the address ( $SVAn$  value) of a slave device Note, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device ( $ACKD_n = 1$ ) at the rising edge of the 9th clock.
- <4> The master device issues an interrupt (INTIICAn: end of address transmission) at the falling edge of the 9th clock. The slave device whose address matched the transmitted slave address sets a wait status ( $SCLAn = 0$ ) and issues an interrupt (INTIICAn: address match) Note.
- <5> The master device writes the data to transmit to the  $IICAn$  register and releases the wait status that it set by the master device.
- <6> If the slave device releases the wait status ( $WREL_n = 1$ ), the master device starts transferring data to the slave device.

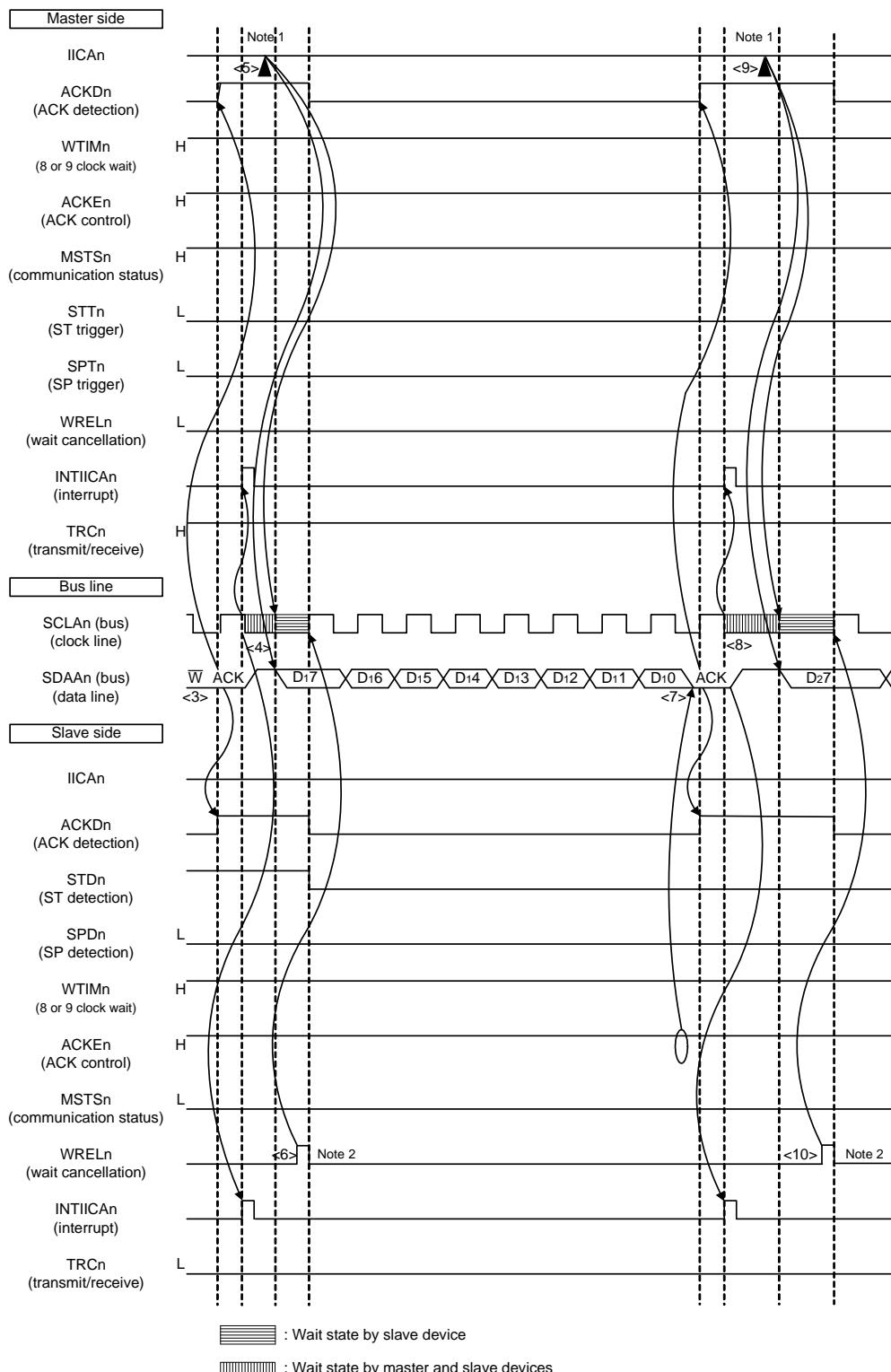
**Note** If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK:  $SDAAn = 1$ ). The slave device also does not issue the INTIICAn interrupt (address match) and does not set a wait status. The master device, however, issues the INTIICAn interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

**Remark 1.** <1> to <15> in Figures 18 - 41 to 18 - 43 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 41 (1) Start condition ~ address ~ data shows the processing from <1> to <6>, Figure 18 - 42 (3) Address ~ data ~ data shows the processing from <3> to <10>, and Figure 18 - 43 (3) Data ~ data ~ stop condition shows the processing from <7> to <15>.

**Remark 2.**  $n = 0, 1$

**Figure 18 - 42 Example of Master to Slave Communication  
(When 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (2/4)**

(3) Address ~ data ~ data



**Note 1.** Write data to IICAn, not setting the WRELn bit, in order to cancel a wait state during transmission by a master device.

**Note 2.** For releasing wait state during reception of a slave device, write "FFH" to IICAn or set the WRELn bit.

**Remark** n = 0, 1

The meanings of <3> to <10> in (3) Address ~ data ~ data in Figure 18 - 42 are explained below.

- <3> In the slave device if the address received matches the address (SVAn value) of a slave device Note, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKDn = 1) at the rising edge of the 9th clock.
- <4> The master device issues an interrupt (INTIICAn: end of address transmission) at the falling edge of the 9th clock. The slave device whose address matched the transmitted slave address sets a wait status (SCLAn = 0) and issues an interrupt (INTIICAn: address match) Note.
- <5> The master device writes the data to transmit to the IICA shift register n (IICAn) and releases the wait status that it set by the master device.
- <6> If the slave device releases the wait status (WRELn = 1), the master device starts transferring data to the slave device.
- <7> After data transfer is completed, because of ACKEn = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKDn = 1) at the rising edge of the 9th clock.
- <8> The master device and slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (INTIICAn: end of transfer).
- <9> The master device writes the data to transmit to the IICAn register and releases the wait status that it set by the master device.
- <10> The slave device reads the received data and releases the wait status (WRELn = 1). The master device then starts transferring data to the slave device.

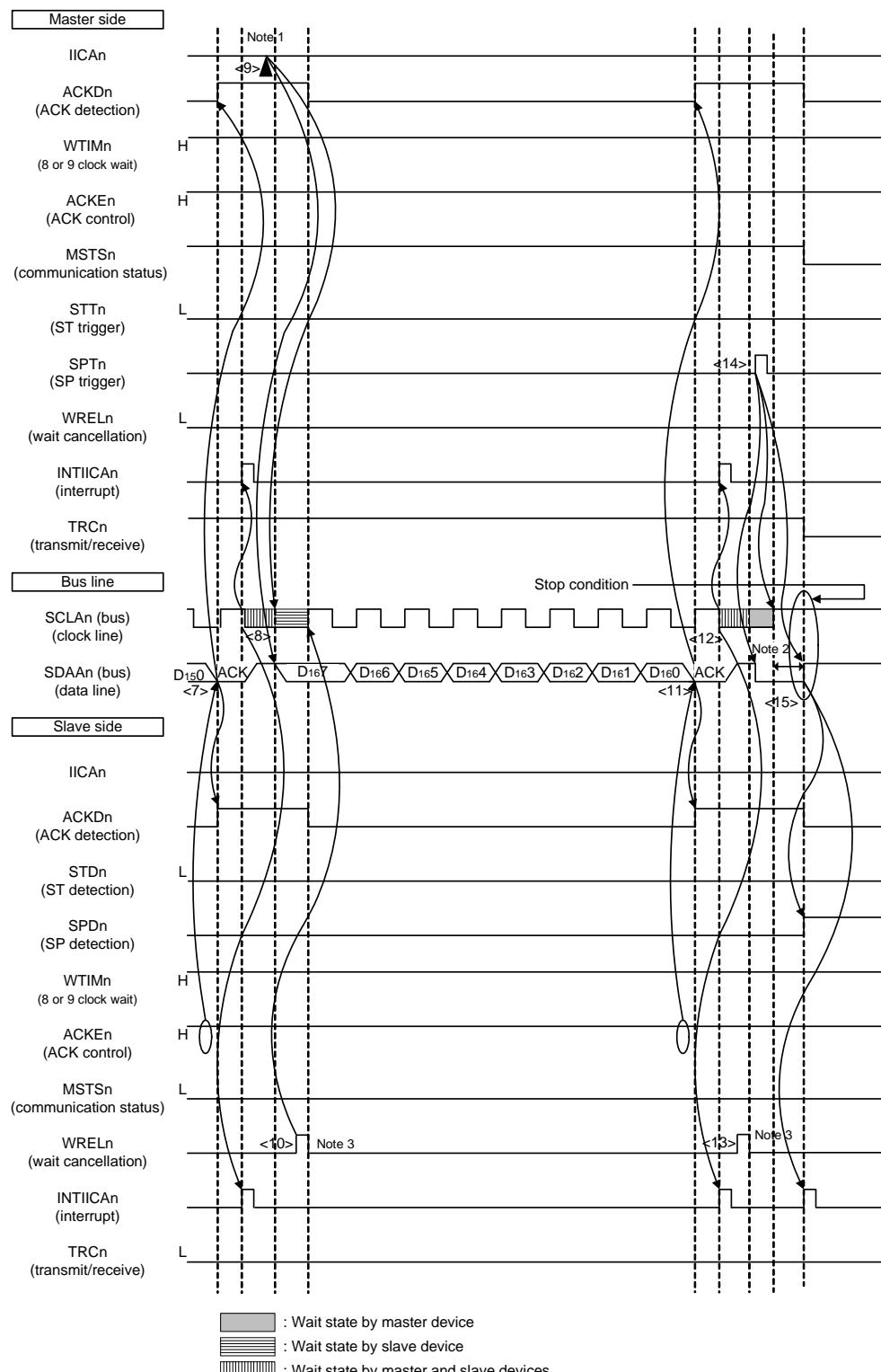
**Note** If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDAAn = 1). The slave device also does not issue the INTIICAn interrupt (address match) and does not set a wait status. The master device, however, issues the INTIICAn interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

**Remark 1.** <1> to <15> in Figures 18 - 41 to 18 - 43 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 41 (1) Start condition ~ address ~ data shows the processing from <1> to <6>, Figure 18 - 42 (3) Address ~ data ~ data shows the processing from <3> to <10>, and Figure 18 - 43 (3) Data ~ data ~ stop condition shows the processing from <7> to <15>.

**Remark 2.** n = 0, 1

**Figure 18 - 43 Example of Master to Slave Communication  
(When 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (3/4)**

(3) Data ~ data ~ stop condition



**Note 1.** Write data to IICAn, not setting the WRELn bit, in order to cancel a wait state during transmission by a master device.

**Note 2.** Make sure that the time between the rise of the SCLAn pin signal and the generation of the stop condition after a stop condition has been issued is at least 4.0  $\mu$ s when specifying standard mode and at least 0.6  $\mu$ s when specifying fast mode.

**Note 3.** For releasing wait state during reception of a slave device, write "FFH" to IICAn or set the WRELn bit.  
**Remark** n = 0, 1

The meanings of <7> to <15> in (3) Data ~ data ~ stop condition in Figure 18 - 43 are explained below.

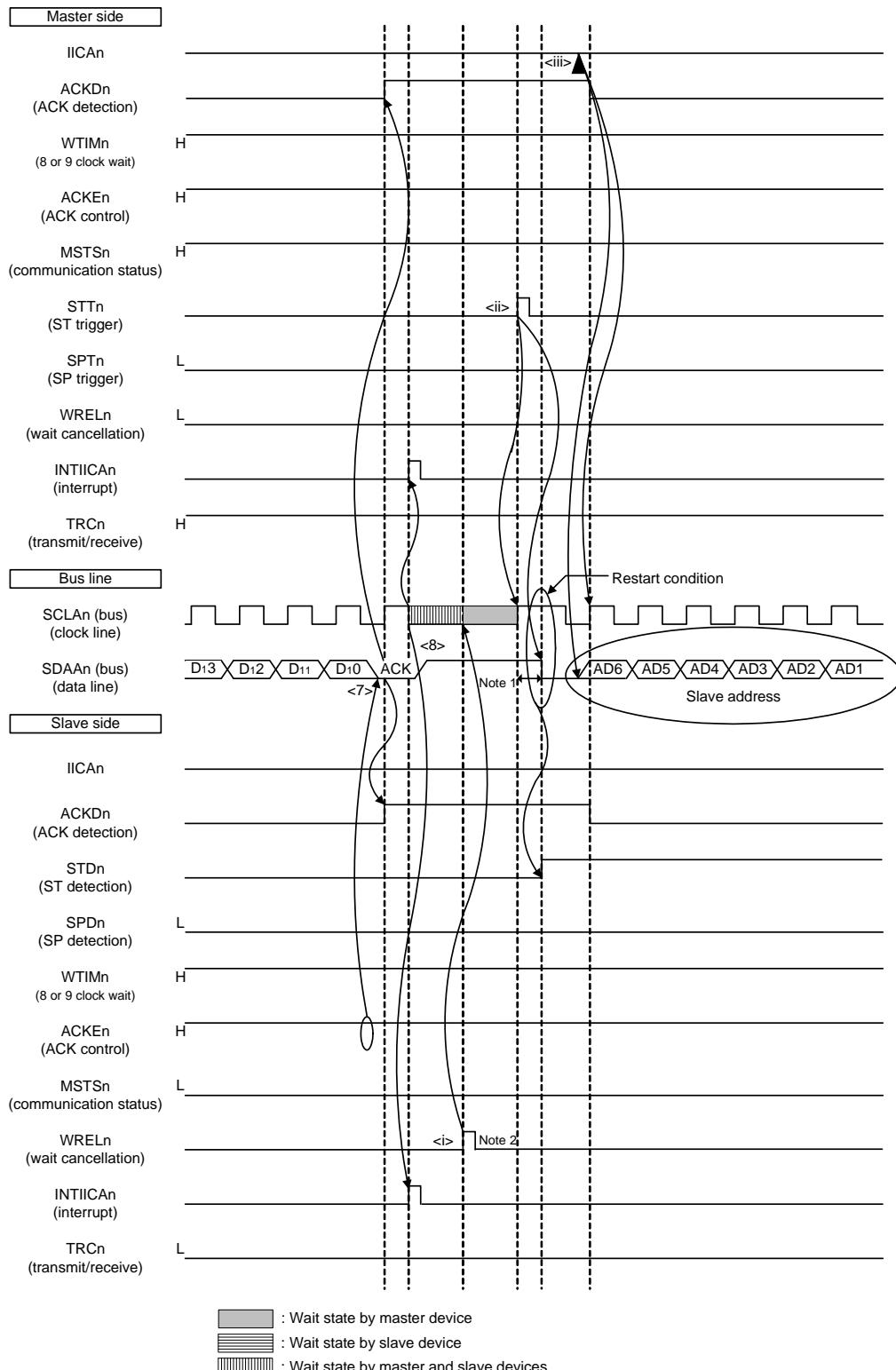
- <7> After data transfer is completed, because of ACKEn = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKDn = 1) at the rising edge of the 9th clock.
- <8> The master device and slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (INTIICAn: end of transfer).
- <9> The master device writes the data to transmit to the IICA shift register n (IICAn) and releases the wait status that it set by the master device.
- <10> The slave device reads the received data and releases the wait status (WRELn = 1). The master device then starts transferring data to the slave device.
- <11> When data transfer is complete, the slave device (ACKEn = 1) sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKDn = 1) at the rising edge of the 9th clock.
- <12> The master device and slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (INTIICAn: end of transfer).
- <13> The slave device reads the received data and releases the wait status (WRELn = 1).
- <14> By the master device setting a stop condition trigger (SPTn = 1), the bus data line is cleared (SDAAn = 0) and the bus clock line is set (SCLAn = 1). After the stop condition setup time has elapsed, by setting the bus data line (SDAAn = 1), the stop condition is then generated (i.e. SCLAn = 1 changes SDAAn from 0 to 1).
- <15> When a stop condition is generated, the slave device detects the stop condition and issues an interrupt (INTIICAn: stop condition).

**Remark 1.** <1> to <15> in Figures 18 - 41 to 18 - 43 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 41 (1) Start condition ~ address ~ data shows the processing from <1> to <6>, Figure 18 - 42 (3) Address ~ data ~ data shows the processing from <3> to <10>, and Figure 18 - 43 (3) Data ~ data ~ stop condition shows the processing from <7> to <15>.

**Remark 2.** n = 0, 1

**Figure 18 - 44 Example of Master to Slave Communication  
(When 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (4/4)**

(3) Data ~ restart condition ~ address



**Note 1.** Make sure that the time between the rise of the SCLAn pin signal and the generation of the start condition after a restart condition has been issued is at least 4.7 µs when specifying standard mode and at least 0.6 µs when specifying fast mode.

**Note 2.** For releasing wait state during reception of a slave device, write “FFH” to IICAn or set the WRELn bit.  
**Remark** n = 0, 1

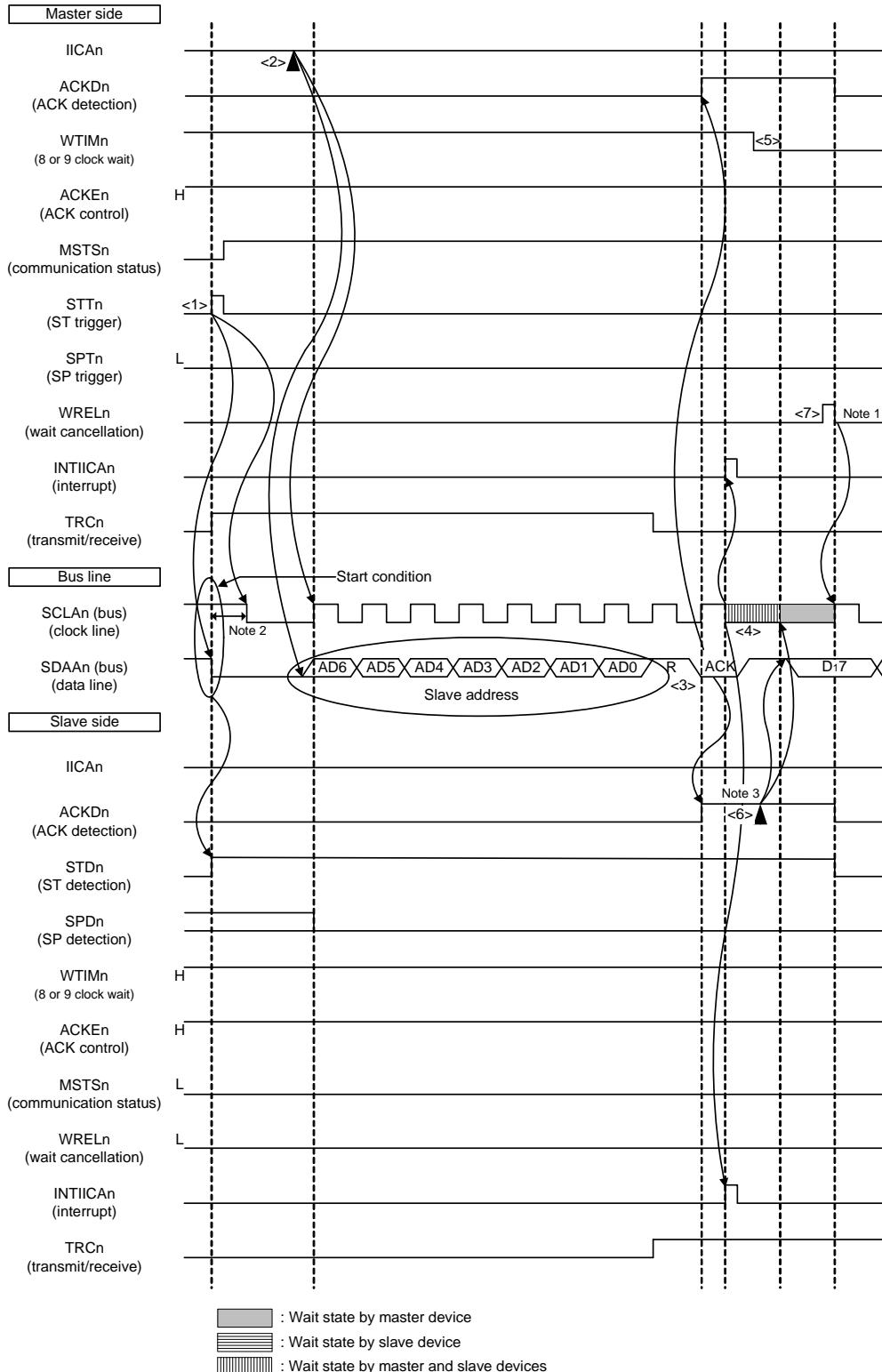
The following describes the operations in Figure 18 - 44 (3) Data ~ restart condition ~ address. After the operations in steps <7> and <8>, the operations in steps <i> to <iii> are performed. These steps return the processing to step <iii>, the data transmission step.

- <7> After data transfer is completed, because of ACKEn = 1, the slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKDn = 1) at the rising edge of the 9th clock.
- <8> The master device and slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (INTIICAn: end of transfer).
- <i> The slave device reads the received data and releases the wait status (WRELn = 1).
- <ii> The start condition trigger is set again by the master device (STTn = 1) and a start condition (i.e. SCLAn = 1 changes SDAAn from 1 to 0) is generated once the bus clock line goes high (SCLAn = 1) and the bus data line goes low (SDAAn = 0) after the restart condition setup time has elapsed. When the start condition is subsequently detected, the master device is ready to communicate once the bus clock line goes low (SCLAn = 0) after the hold time has elapsed.
- <iii> The master device writing the address + R/W (transmission) to the IICA shift register (IICAn) enables the slave address to be transmitted.

**Remark** n = 0, 1

**Figure 18 - 45 Example of Slave to Master Communication  
(When 8-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (1/3)**

(1) Start condition ~ address ~ data



**Note 1.** For releasing wait state during reception of a master device, write "FFH" to IICAn or set the WREL<sub>n</sub> bit.

**Note 2.** Make sure that the time between the fall of the SDAAn pin signal and the fall of the SCLAn pin signal is at least 4.0  $\mu$ s when specifying standard mode and at least 0.6  $\mu$ s when specifying fast mode.

**Note 3.** Write data to IICAn, not setting the WREL<sub>n</sub> bit, in order to cancel a wait state during transmission by a slave device.

**Remark** n = 0, 1

The meanings of <1> to <7> in (1) Start condition ~ address ~ data in Figure 18 - 45 are explained below.

- <1> The start condition trigger is set by the master device ( $STTn = 1$ ) and a start condition (i.e.  $SCLAn = 1$  changes  $SDAAn$  from 1 to 0) is generated once the bus data line goes low ( $SDAAn$ ). When the start condition is subsequently detected, the master device enters the master device communication status ( $MSTS_n = 1$ ). The master device is ready to communicate once the bus clock line goes low ( $SCLAn = 0$ ) after the hold time has elapsed.
- <2> The master device writes the address + R (reception) to the IICA shift register n ( $IICAn$ ) and transmits the slave address.
- <3> In the slave device if the address received matches the address ( $SVAn$  value) of a slave device Note, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device ( $ACKDn = 1$ ) at the rising edge of the 9th clock.
- <4> The master device issues an interrupt (INTIICAn: end of address transmission) at the falling edge of the 9th clock. The slave device whose address matched the transmitted slave address sets a wait status ( $SCLAn = 0$ ) and issues an interrupt (INTIICAn: address match) Note.
- <5> The timing at which the master device sets the wait status changes to the 8th clock ( $WTIMn = 0$ ).
- <6> The slave device writes the data to transmit to the  $IICAn$  register and releases the wait status that it set by the slave device.
- <7> The master device releases the wait status ( $WRELn = 1$ ) and starts transferring data from the slave device to the master device.

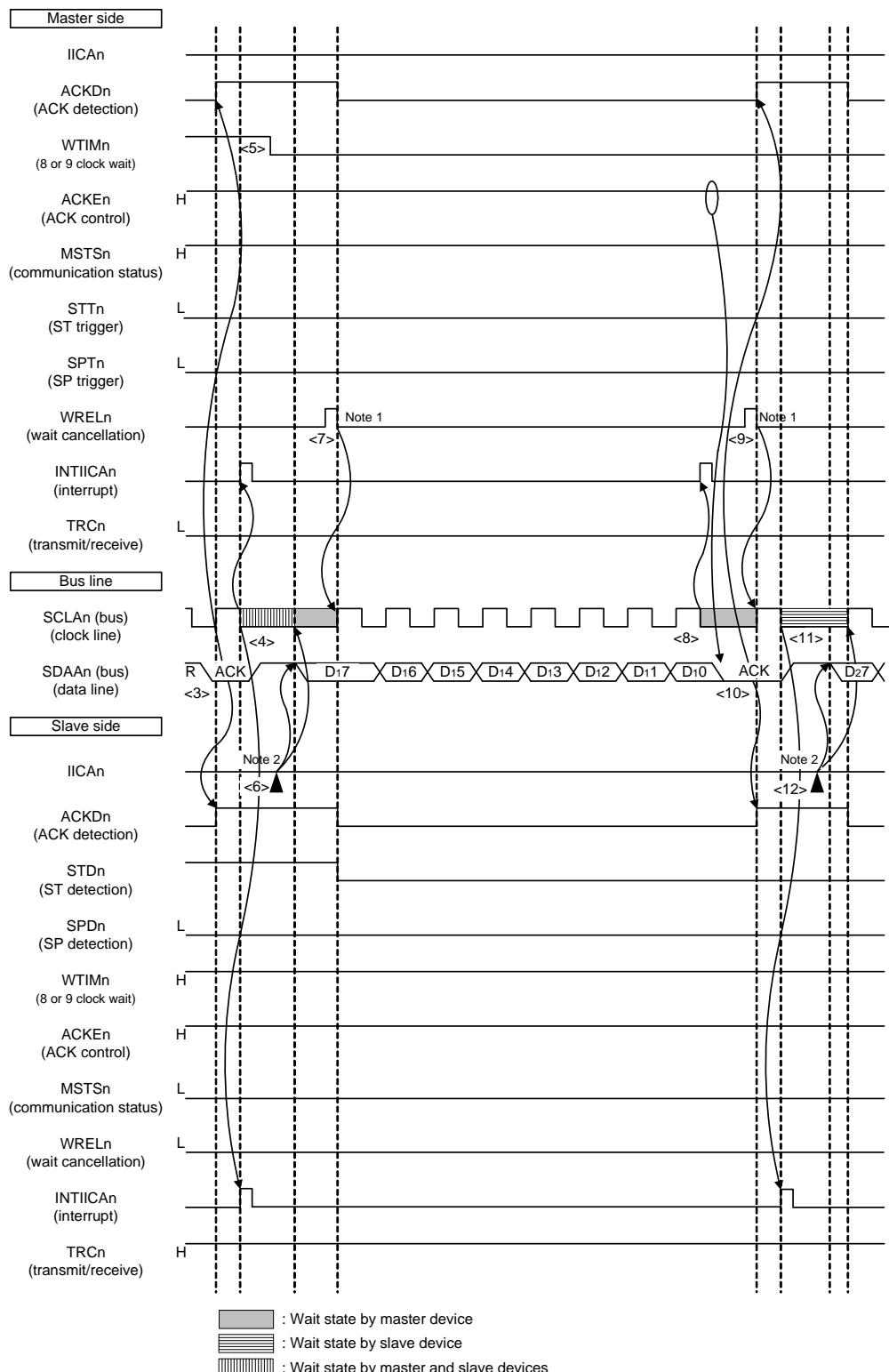
**Note** If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK:  $SDAAn = 1$ ). The slave device also does not issue the INTIICAn interrupt (address match) and does not set a wait status. The master device, however, issues the INTIICAn interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

**Remark 1.** <1> to <19> in Figures 18 - 45 to 18 - 47 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 45 (1) Start condition ~ address ~ data shows the processing from <1> to <7>, Figure 18 - 46 (3) Address ~ data ~ data shows the processing from <3> to <12>, and Figure 18 - 47 (3) Data ~ data ~ stop condition shows the processing from <8> to <19>.

**Remark 2.**  $n = 0, 1$

**Figure 18 - 46 Example of Slave to Master Communication  
(When 8-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (2/3)**

(3) Address ~ data ~ data



- Note 1.** For releasing wait state during reception of a master device, write “FFH” to IICAn or set the WRELn bit.  
**Note 2.** Write data to IICAn, not setting the WRELn bit, in order to cancel a wait state during transmission by a slave device.  
**Remark** n = 0, 1

The meanings of <3> to <12> in (3) Address ~ data ~ data in Figure 18 - 46 are explained below.

- <3> In the slave device if the address received matches the address (SVA<sub>n</sub> value) of a slave device Note, that slave device sends an ACK by hardware to the master device. The ACK is detected by the master device (ACKD<sub>n</sub> = 1) at the rising edge of the 9th clock.
- <4> The master device issues an interrupt (INTIICAn: end of address transmission) at the falling edge of the 9th clock. The slave device whose address matched the transmitted slave address sets a wait status (SCLAn = 0) and issues an interrupt (INTIICAn: address match) Note.
- <5> The master device changes the timing of the wait status to the 8th clock (WTIM<sub>n</sub> = 0).
- <6> The slave device writes the data to transmit to the IICA shift register n (IICAn) and releases the wait status that it set by the slave device.
- <7> The master device releases the wait status (WREL<sub>n</sub> = 1) and starts transferring data from the slave device to the master device.
- <8> The master device sets a wait status (SCLAn = 0) at the falling edge of the 8th clock, and issues an interrupt (INTIICAn: end of transfer). Because of ACKEn = 1 in the master device, the master device then sends an ACK by hardware to the slave device.
- <9> The master device reads the received data and releases the wait status (WREL<sub>n</sub> = 1).
- <10> The ACK is detected by the slave device (ACKD<sub>n</sub> = 1) at the rising edge of the 9th clock.
- <11> The slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and the slave device issue an interrupt (INTIICAn: end of transfer).
- <12> By the slave device writing the data to transmit to the IICAn register, the wait status set by the slave device is released. The slave device then starts transferring data to the master device.

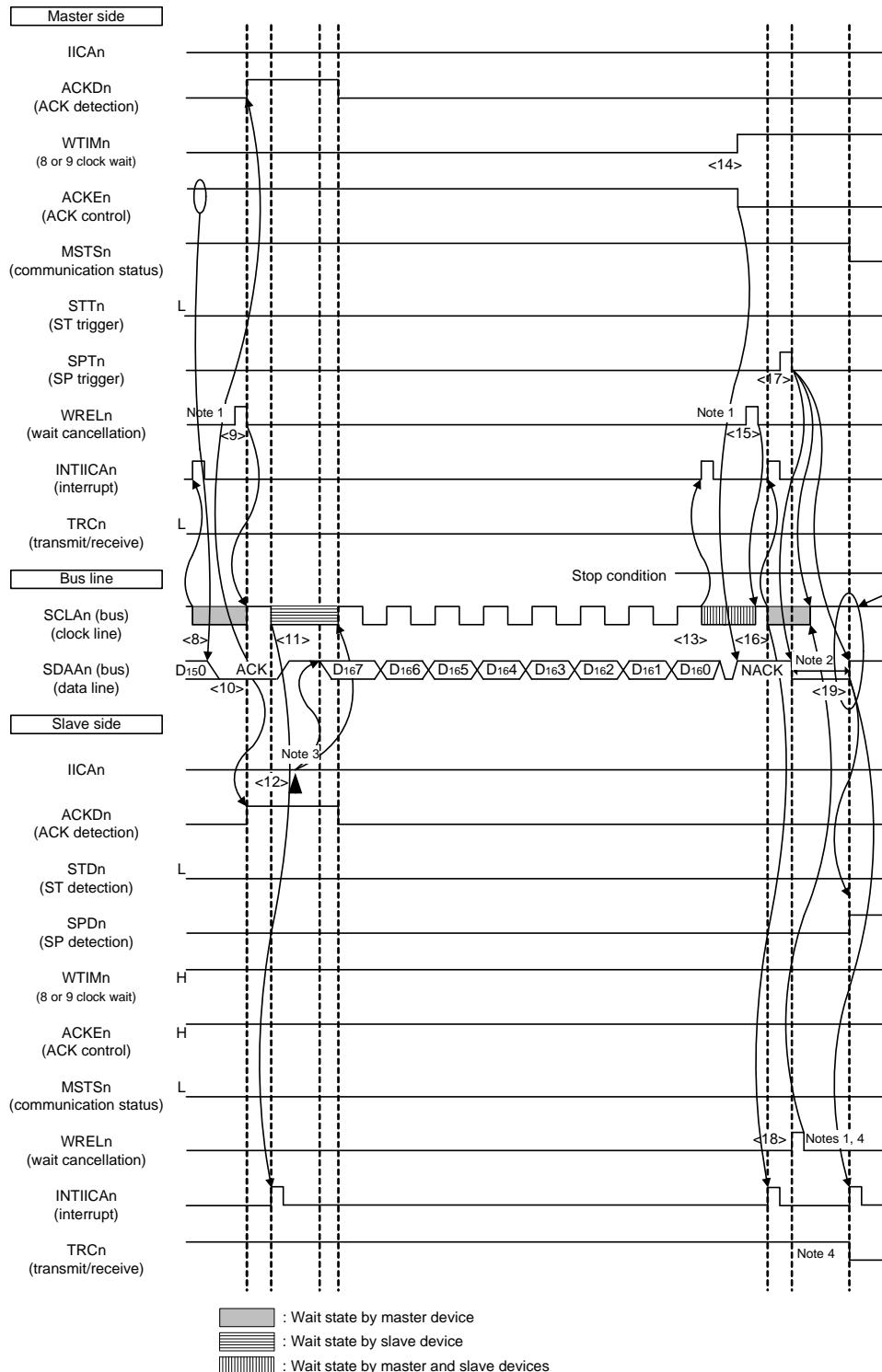
**Note** If the transmitted address does not match the address of the slave device, the slave device does not return an ACK to the master device (NACK: SDA<sub>An</sub> = 1). The slave device also does not issue the INTIICAn interrupt (address match) and does not set a wait status. The master device, however, issues the INTIICAn interrupt (end of address transmission) regardless of whether it receives an ACK or NACK.

**Remark 1.** <1> to <19> in Figures 18 - 45 to 18 - 47 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 45 (1) Start condition ~ address ~ data shows the processing from <1> to <7>, Figure 18 - 46 (3) Address ~ data ~ data shows the processing from <3> to <12>, and Figure 18 - 47 (3) Data ~ data ~ stop condition shows the processing from <8> to <19>.

**Remark 2.** n = 0, 1

**Figure 18 - 47 Example of Slave to Master Communication  
(When 8-Clock and 9-Clock Wait Is Selected for Master, 9-Clock Wait Is Selected for Slave) (3/3)**

(3) Data ~ data ~ stop condition



- Note 1.** To cancel a wait state, write “FFH” to IICAn or set the WRELn bit.  
**Note 2.** Make sure that the time between the rise of the SCLAn pin signal and the generation of the stop condition after a stop condition has been issued is at least 4.0  $\mu$ s when specifying standard mode and at least 0.6  $\mu$ s when specifying fast mode.  
**Note 3.** Write data to IICAn, not setting the WRELn bit, in order to cancel a wait state during transmission by a slave device.  
**Note 4.** If a wait state during transmission by a slave device is canceled by setting the WRELn bit, the TRCn bit will be cleared.  
**Remark** n = 0, 1

The meanings of <8> to <19> in (3) Data ~ data ~ stop condition in Figure 18 - 47 are explained below.

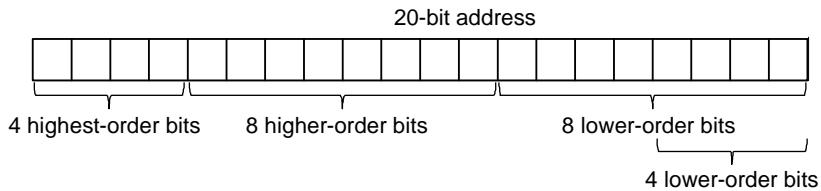
- <8> The master device sets a wait status (SCLAn = 0) at the falling edge of the 8th clock, and issues an interrupt (INTIICAn: end of transfer). Because of ACKEn = 0 in the master device, the master device then sends an ACK by hardware to the slave device.
- <9> The master device reads the received data and releases the wait status (WRELn = 1).
- <10> The ACK is detected by the slave device (ACKDn = 1) at the rising edge of the 9th clock.
- <11> The slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and the slave device issue an interrupt (INTIICAn: end of transfer).
- <12> By the slave device writing the data to transmit to the IICA register, the wait status set by the slave device is released. The slave device then starts transferring data to the master device.
- <13> The master device issues an interrupt (INTIICAn: end of transfer) at the falling edge of the 8th clock, and sets a wait status (SCLAn = 0). Because ACK control (ACKEn = 1) is performed, the bus data line is at the low level (SDAAn = 0) at this stage.
- <14> The master device sets NACK as the response (ACKEn = 0) and changes the timing at which it sets the wait status to the 9th clock (WTIMn = 1).
- <15> If the master device releases the wait status (WRELn = 1), the slave device detects the NACK (ACKDn = 0) at the rising edge of the 9th clock.
- <16> The master device and slave device set a wait status (SCLAn = 0) at the falling edge of the 9th clock, and both the master device and slave device issue an interrupt (INTIICAn: end of transfer).
- <17> When the master device issues a stop condition (SPTn = 1), the bus data line is cleared (SDAAn = 0) and the master device releases the wait status. The master device then waits until the bus clock line is set (SCLAn = 1).
- <18> The slave device acknowledges the NACK, halts transmission, and releases the wait status (WRELn = 1) to end communication. Once the slave device releases the wait status, the bus clock line is set (SCLAn = 1).
- <19> Once the master device recognizes that the bus clock line is set (SCLAn = 1) and after the stop condition setup time has elapsed, the master device sets the bus data line (SDAAn = 1) and issues a stop condition (i.e. SCLAn = 1 changes SDAAn from 0 to 1). The slave device detects the generated stop condition and slave device issue an interrupt (INTIICAn: stop condition).

**Remark 1.** <1> to <19> in Figures 18 - 45 to 18 - 47 represent the entire procedure for communicating data using the I<sup>2</sup>C bus. Figure 18 - 45 (1) Start condition ~ address ~ data shows the processing from <1> to <7>, Figure 18 - 46 (3) Address ~ data ~ data shows the processing from <3> to <12>, and Figure 18 - 47 (3) Data ~ data ~ stop condition shows the processing from <8> to <19>.

**Remark 2.** n = 0, 1

## CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)

The term “8 higher-order bits of the address” in this chapter indicates bits 15 to 8 of 20-bit address as shown below.



Unless otherwise specified, the 4 highest-order address bits all become 1 (values are of the form FxxxxH).

### 19.1 Functions of DTC

The data transfer controller (DTC) is a function that transfers data between memories without using the CPU. The DTC is activated by a peripheral function interrupt to perform data transfers. The DTC and CPU use the same bus, and the DTC takes priority over the CPU in using the bus.

Table 19 - 1 lists the DTC Specifications.

**Table 19 - 1 DTC Specifications (1/2)**

Item		Specification
Activation sources		31 sources (30-, 32-, 36-, 40-, and 44-pin products)/32 sources (48- and 52-pin products)/33 sources (64-pin products)/39 sources (80- and 100-pin products)
Allocatable control data		24 sets
Address space which can be transferred	Address space	64 Kbytes (F0000H to FFFFFH), excluding general-purpose registers
	Sources	Special function register (SFR), RAM area (excluding general-purpose registers), mirror area Note, data flash memory area Note, extended special function register (2nd SFR)
	Destinations	Special function register (SFR), RAM area (excluding general-purpose registers), extended special function register (2nd SFR)
Maximum number of transfers	Normal mode	256 times
	Repeat mode	255 times
Maximum size of block to be transferred	Normal mode (8-bit transfer)	256 bytes
	Normal mode (16-bit transfer)	512 bytes
	Repeat mode	255 bytes
Unit of transfers		8 bits/16 bits
Transfer mode	Normal mode	Transfers end on completion of the transfer causing the DTCCCTj register value to change from 1 to 0.
	Repeat mode	On completion of the transfer causing the DTCCCTj register value to change from 1 to 0, the repeat area address is initialized and the DTRLDJ register value is reloaded to the DTCCCTj register to continue transfers.
Address control	Normal mode	Fixed or incremented
	Repeat mode	Addresses of the area not selected as the repeat area are fixed or incremented.
Priority of activation sources		Refer to <b>Table 19 - 5 DTC Activation Sources and Vector Addresses</b> .
Interrupt request	Normal mode	When the data transfer causing the DTCCCTj register value to change from 1 to 0 is performed, the activation source interrupt request is generated for the CPU, and interrupt handling is performed on completion of the data transfer.
	Repeat mode	When the data transfer causing the DTCCCTj register value to change from 1 to 0 is performed while the RPTINT bit in the DTCCRJ register is 1 (interrupt generation enabled), the activation source interrupt request is generated for the CPU, and interrupt handling is performed on completion of the transfer.

(Note and Remark are listed on the next page.)

**Table 19 - 1 DTC Specifications (2/2)**

Item		Specification
Transfer start		When bits DTCENi0 to DTCENi7 in the DTCENi registers are 1 (activation enabled), data transfer is started each time the corresponding DTC activation sources are generated.
Transfer stop	Normal mode	<ul style="list-style-type: none"> <li>• When bits DTCENi0 to DTCENi7 are set to 0 (activation disabled).</li> <li>• When the data transfer causing the DTCCCTj register value to change from 1 to 0 is completed.</li> </ul>
	Repeat mode	<ul style="list-style-type: none"> <li>• When bits DTCENi0 to DTCENi7 are set to 0 (activation disabled).</li> <li>• When the data transfer causing the DTCCCTj register value to change from 1 to 0 is completed while the RPTINT bit is 1 (interrupt generation enabled).</li> </ul>

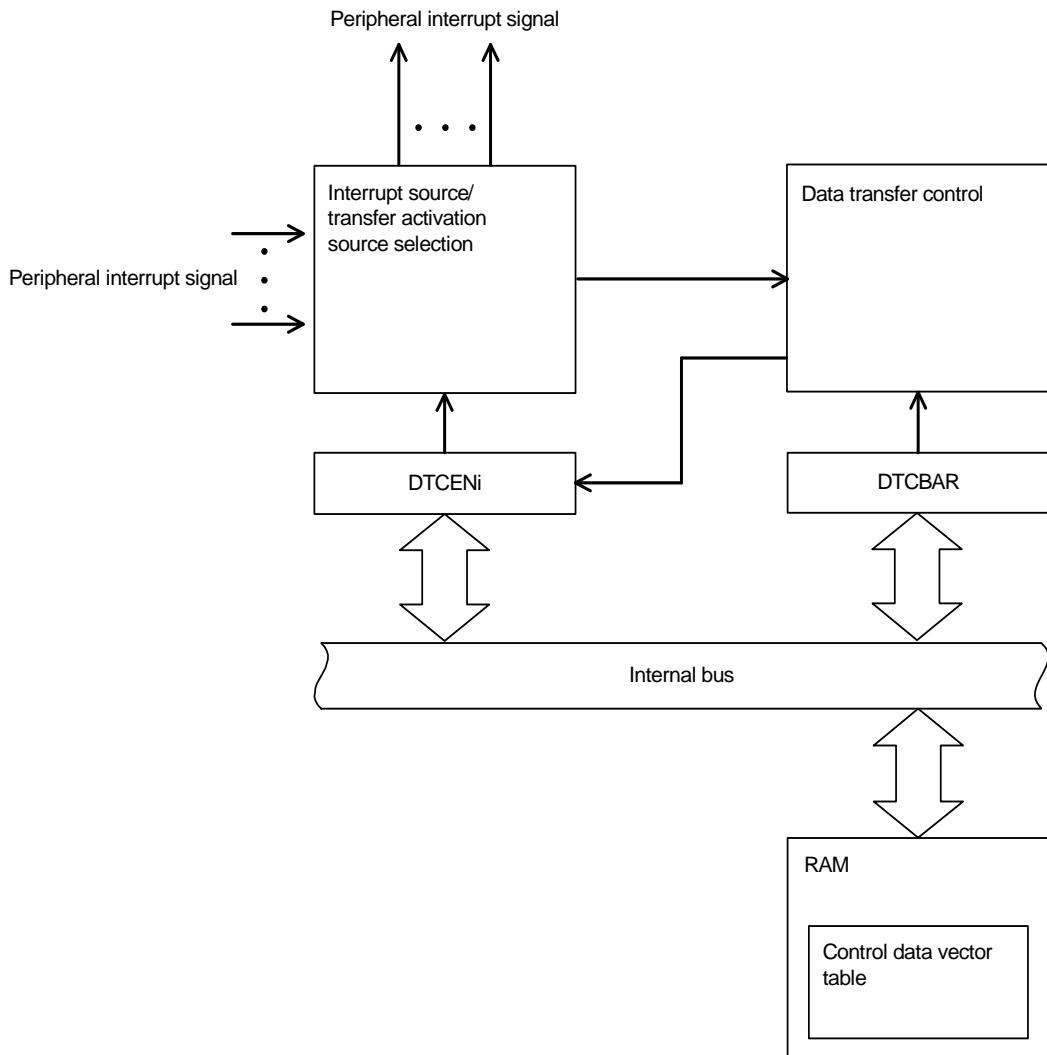
**Note** In the HALT mode or SNOOZE mode, these areas cannot be set as the sources for DTC transfer since the flash memory is stopped.

**Remark** i = 0 to 4, j = 0 to 23

## 19.2 Configuration of DTC

Figure 19 - 1 shows the DTC Block Diagram.

Figure 19 - 1 DTC Block Diagram



### 19.3 Registers Controlling DTC

Table 19 - 2 lists the Registers Controlling DTC.

**Table 19 - 2 Registers Controlling DTC**

Register Name	Symbol
Peripheral enable register 1	PER1
DTC activation enable register 0	DTCEN0
DTC activation enable register 1	DTCEN1
DTC activation enable register 2	DTCEN2
DTC activation enable register 3	DTCEN3
DTC activation enable register 4	DTCEN4
DTC base address register	DTCBAR

Table 19 - 3 lists DTC Control Data.

DTC control data is allocated in the DTC control data area in RAM.

The DTCBAR register is used to set the 256-byte area, including the DTC control data area and the DTC vector table area where the start address for control data is stored.

**Table 19 - 3 DTC Control Data**

Register Name	Symbol
DTC Control Register j	DTCCRj
DTC Block Size Register j	DTBLSj
DTC Transfer Count Register j	DTCCTj
DTC Transfer Count Reload Register j	DTRLDj
DTC Source Address Register j	DTSARj
DTC Destination Address Register j	DTDARj

**Remark**    j = 0 to 23

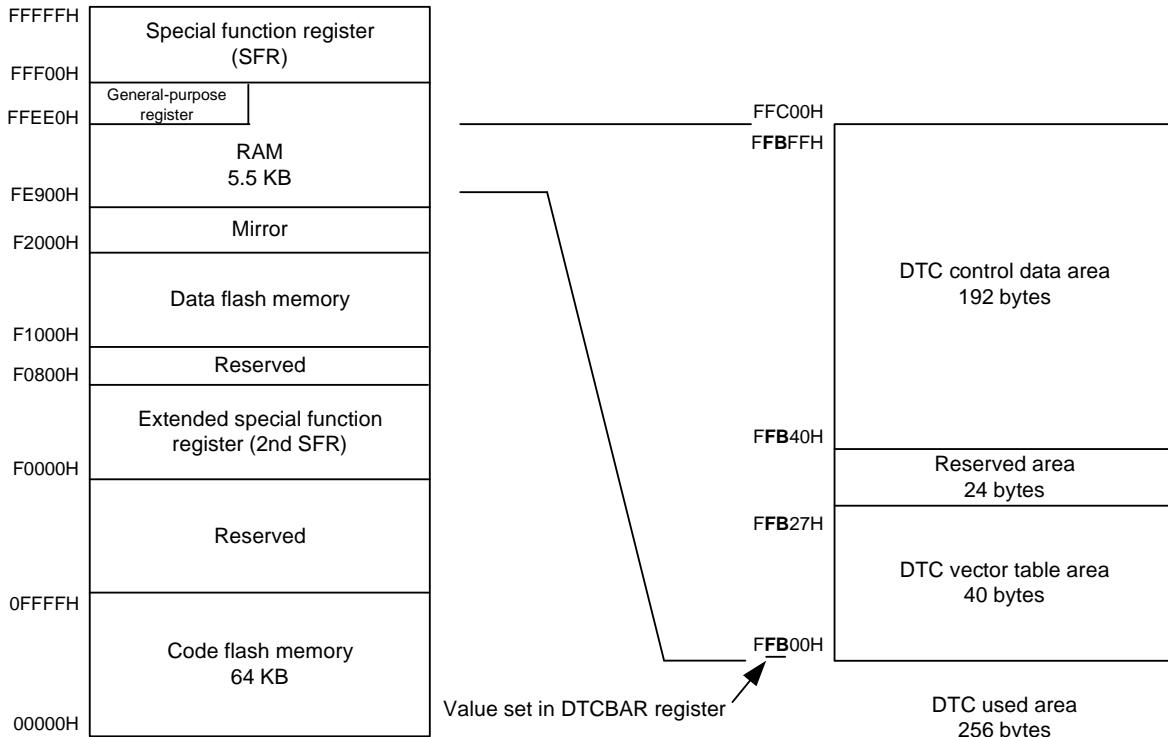
### 19.3.1 Allocation of DTC Control Data Area and DTC Vector Table Area

The DTCBAR register is used to set the 256-byte area where DTC control data and the vector table within the RAM area.

Figure 19 - 2 shows a Memory Map Example when DTCBAR Register is Set to FBH.

In the 192-byte DTC control data area, the space not used by the DTC can be used as RAM.

**Figure 19 - 2 Memory Map Example when DTCBAR Register is Set to FBH**



The areas where the DTC control data and vector table can be allocated differ depending on the product.

**Caution 1. It is prohibited to use the general-purpose register (FFEE0H to FFEFFFH) space as the DTC control data area or DTC vector table area.**

**Caution 2. Make sure the stack area, the DTC control data area, and the DTC vector table area do not overlap.**

**Caution 3. The internal RAM area in the following products cannot be used as the DTC control data area or DTC vector table area when using the self-programming and data-flash functions.**

R5F104xD (x = A to C, E to G, J, L): FE900H to FED09H

R5F104xE (x = A to C, E to G, J, L): FE900H to FED09H

R5F104xJ (x = A to C, E to G, J, L): F9F00H to FA309H

R5F104xL (x = G, L, M, P): F3F00H to F4309H

**Caution 4. The internal RAM area in the following products cannot be used as the DTC control data area or DTC vector table area when using the on-chip debugging trace function.**

R5F104xJ (x = A to C, E to G, J, L): FA300H to FA6FFH

R5F104xL (x = G, L, M, P): F4300H to F46FFH

### 19.3.2 Control Data Allocation

Control data is allocated beginning with each start address in the order: Registers DTCCRj, DTBLSj, DTCCTj, DTRLDj, DTSARj, and DTDARj ( $j = 0$  to 23).

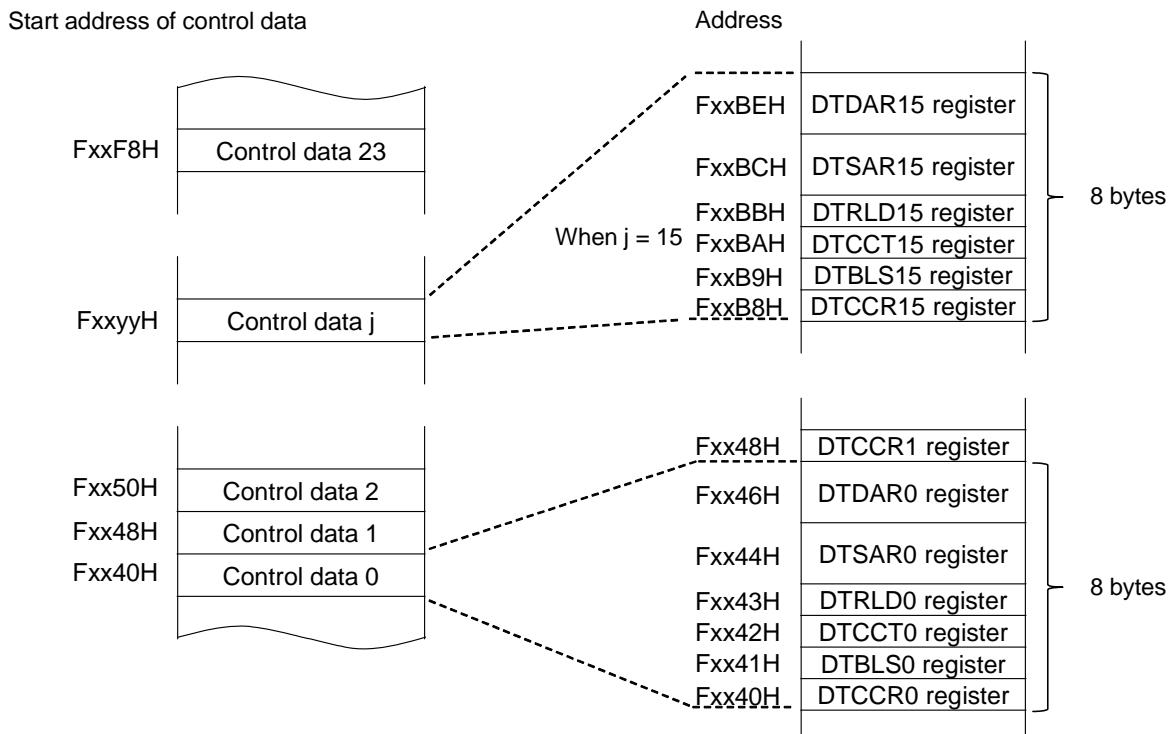
The higher 8 bits for start addresses 0 to 23 are set by the DTCBAR register, and the lower 8 bits are separately set according to the vector table assigned to each activation source.

Figure 19 - 3 shows Control Data Allocation.

**Note 1.** Change the data in registers DTCCRj, DTBLSj, DTCCTj, DTRLDj, DTSARj, and DTDARj when the corresponding bit among bits DTCENi0 to DTCENi7 ( $i = 0$  to 4) in the DTCENi register is set to 0 (activation disabled).

**Note 2.** Do not access DTCCRj, DTBLSj, DTCCTj, DTRLDj, DTSARj, or DTDARj using a DTC transfer.

Figure 19 - 3 Control Data Allocation



**Remark** xx: Value set in DTCBAR register

**Table 19 - 4 Start Address of Control Data**

j	Address
11	Fxx98H
10	Fxx90H
9	Fxx88H
8	Fxx80H
7	Fxx78H
6	Fxx70H
5	Fxx68H
4	Fxx60H
3	Fxx58H
2	Fxx50H
1	Fxx48H
0	Fxx40H

j	Address
23	FxxF8H
22	FxxF0H
21	FxxE8H
20	FxxE0H
19	FxxD8H
18	FxxD0H
17	FxxC8H
16	FxxC0H
15	FxxB8H
14	FxxB0H
13	FxxA8H
12	FxxA0H

**Remark** xx: Value set in DTCBAR register

### 19.3.3 Vector Table

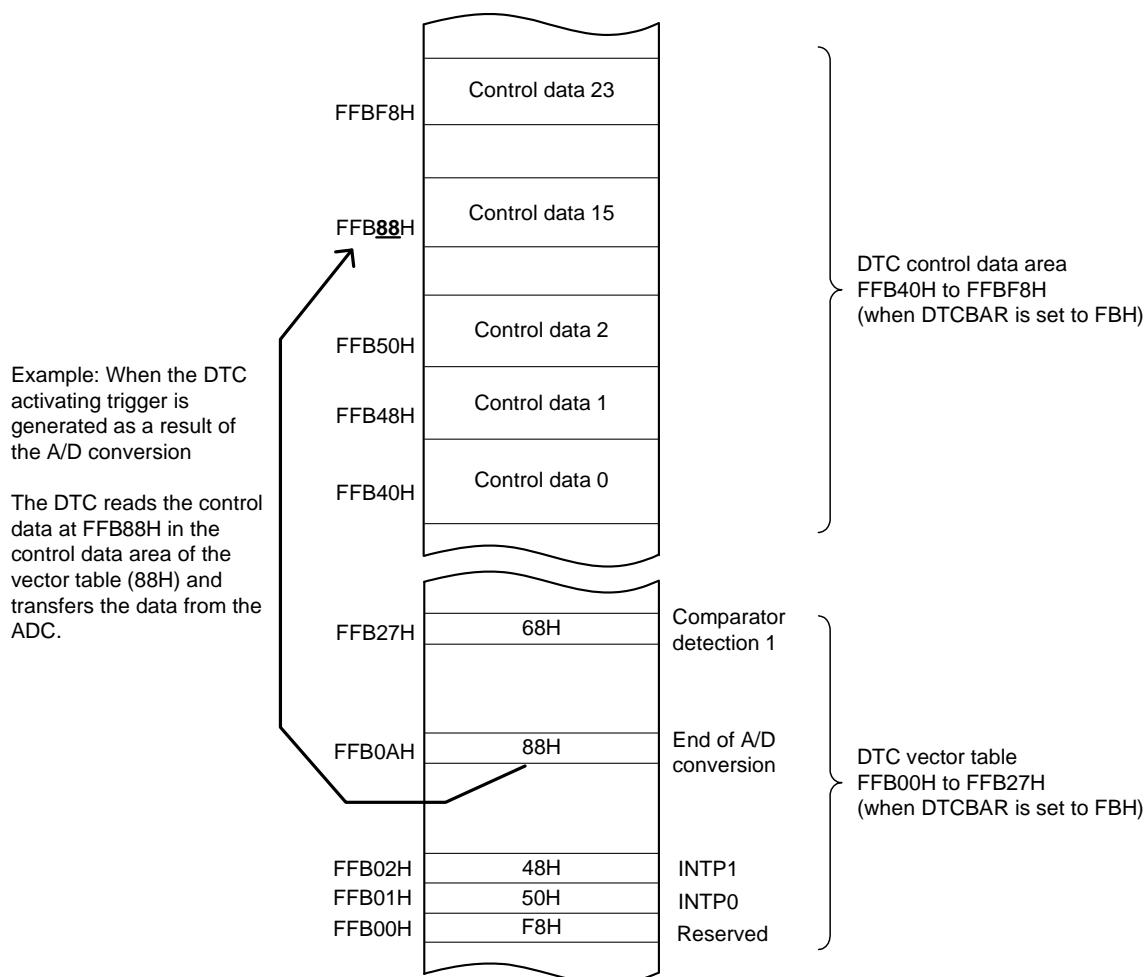
When the DTC is activated, one control data is selected according to the data read from the vector table which has been assigned to each activation source, and the selected control data is read from the DTC control data area.

Table 19 - 5 lists the DTC Activation Sources and Vector Addresses. A one byte of the vector table is assigned to each activation source, and data from 40H to F8H is stored in each area to select one of the 24 control data sets. The higher 8 bits for the vector address are set by the DTCBAR register, and 00H to 27H are allocated to the lower 8 bits corresponding to the activation source.

**Note** Change the start address of the DTC control data area to be set in the vector table when the corresponding bit among bits DTCENi0 to DTCENi7 ( $i = 0$  to  $4$ ) in the DTCENi register is set to 0 (activation disabled).

Figure 19 - 4 Start Address of Control Data and Vector Table

Example: When DTCBAR is set to FBH.



**Table 19 - 5 DTC Activation Sources and Vector Addresses**

DTC Activation Sources (Interrupt Request Source)	Source No.	Vector Address	Priority
Reserved	0	Address set in DTCBAR register +00H	Highest
INTP0	1	Address set in DTCBAR register +01H	
INTP1	2	Address set in DTCBAR register +02H	
INTP2	3	Address set in DTCBAR register +03H	
INTP3	4	Address set in DTCBAR register +04H	
INTP4	5	Address set in DTCBAR register +05H	
INTP5	6	Address set in DTCBAR register +06H	
INTP6	7	Address set in DTCBAR register +07H	
INTP7	8	Address set in DTCBAR register +08H	
Key input	9	Address set in DTCBAR register +09H	
A/D conversion end	10	Address set in DTCBAR register +0AH	
UART0 reception transfer end/CSI01 transfer end or buffer empty/IIC01 transfer end	11	Address set in DTCBAR register +0BH	
UART0 transmission transfer end/CSI00 transfer end or buffer empty/IIC00 transfer end	12	Address set in DTCBAR register +0CH	
UART1 reception transfer end/CSI11 transfer end or buffer empty/IIC11 transfer end	13	Address set in DTCBAR register +0DH	
UART1 transmission transfer end/CSI10 transfer end or buffer empty/IIC10 transfer end	14	Address set in DTCBAR register +0EH	
UART2 reception transfer end/CSI21 transfer end or buffer empty/IIC21 transfer end	15	Address set in DTCBAR register +0FH	
UART2 transmission transfer end/CSI20 transfer end or buffer empty/IIC20 transfer end	16	Address set in DTCBAR register +10H	
UART3 reception transfer end/CSI31 transfer end or buffer empty/IIC31 transfer end Note 1	17	Address set in DTCBAR register +11H	
UART3 transmission transfer end/CSI30 transfer end or buffer empty/IIC30 transfer end Note 1	18	Address set in DTCBAR register +12H	
End of channel 0 of timer array unit 0 count or capture	19	Address set in DTCBAR register +13H	
End of channel 1 of timer array unit 0 count or capture	20	Address set in DTCBAR register +14H	
End of channel 2 of timer array unit 0 count or capture	21	Address set in DTCBAR register +15H	
End of channel 3 of timer array unit 0 count or capture	22	Address set in DTCBAR register +16H	
End of channel 0 of timer array unit 1 count or capture Note 1	23	Address set in DTCBAR register +17H	
End of channel 1 of timer array unit 1 count or capture Note 1	24	Address set in DTCBAR register +18H	
End of channel 2 of timer array unit 1 count or capture Note 1	25	Address set in DTCBAR register +19H	
End of channel 3 of timer array unit 1 count or capture Note 1	26	Address set in DTCBAR register +1AH	
Timer RD compare match A0	27	Address set in DTCBAR register +1BH	
Timer RD compare match B0	28	Address set in DTCBAR register +1CH	
Timer RD compare match C0	29	Address set in DTCBAR register +1DH	
Timer RD compare match D0	30	Address set in DTCBAR register +1EH	
Timer RD compare match A1	31	Address set in DTCBAR register +1FH	
Timer RD compare match B1	32	Address set in DTCBAR register +20H	
Timer RD compare match C1	33	Address set in DTCBAR register +21H	
Timer RD compare match D1	34	Address set in DTCBAR register +22H	
Timer RG compare match A	35	Address set in DTCBAR register +23H	
Timer RG compare match B	36	Address set in DTCBAR register +24H	
Timer RJ0 underflow	37	Address set in DTCBAR register +25H	
Comparator detection 0 Note 2	38	Address set in DTCBAR register +26H	
Comparator detection 1 Note 2	39	Address set in DTCBAR register +27H	Lowest

**Note 1.** For 80- and 100-pin products only.

**Note 2.** 96 KB or more code flash memory products only.

### 19.3.4 Peripheral enable register 1 (PER1)

The PER1 register is used to enable or disable supplying the clock to the peripheral hardware. Clock supply to the hardware that is not used is also stopped so as to decrease the power consumption and noise.

When using the DTC, be sure to set bit 3 (DTCEN) to 1.

The PER1 register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 19 - 5 Format of Peripheral enable register 1 (PER1)**

Address: F007AH      After reset: 00H      R/W

Symbol	<7>	<6>	<5>	<4>	<3>	2	1	<0>
PER1	DACEN Note	TRGEN	CMPEN Note	TRD0EN	DTCEN	0	0	TRJ0EN
DTCEN	Control of DTC input clock supply							
0	Stops input clock supply. • DTC cannot run.							
1	Enables input clock supply. • DTC can run.							

**Note** Only for products with 96 KB or more code flash memory.

**Caution** Be sure to set the following bits to 0.

Products with 64 KB or less code flash memory: bits 1, 2, 5, and 7

Products with 96 KB or more code flash memory: bits 1 and 2

### 19.3.5 DTC control register j (DTCCRj) (j = 0 to 23)

The DTCCRj register is used to control the DTC operating mode.

**Figure 19 - 6 Format of DTC control register j (DTCCRj)**

Symbol	7	6	5	4	3	2	1	0	Address: Refer to 19.3.2 Control Data Allocation.	After reset: Undefined	R/W														
DTCCRj	0	SZ	RPTINT	CHNE	DAMOD	SAMOD	RPTSEL	MODE																	
Transfer Data size selection																									
SZ	Transfer Data size selection																								
0	8 bits																								
1	16 bits																								
Enabling/disabling repeat mode interrupts																									
RPTINT	Enabling/disabling repeat mode interrupts																								
0	Interrupt generation disabled																								
1	Interrupt generation enabled																								
The setting of the RPTINT bit is invalid when the MODE bit is 0 (normal mode).																									
Enabling/disabling chain transfers																									
CHNE	Enabling/disabling chain transfers																								
0	Chain transfers disabled																								
1	Chain transfers enabled																								
Set the CHNE bit in the DTCCR23 register to 0 (chain transfers disabled).																									
Transfer destination address control																									
DAMOD	Transfer destination address control																								
0	Fixed																								
1	Incremented																								
The setting of the DAMOD bit is invalid when the MODE bit is 1 (repeat mode) and the RPTSEL bit is 0 (transfer destination is the repeat area).																									
Transfer source address control																									
SAMOD	Transfer source address control																								
0	Fixed																								
1	Incremented																								
The setting of the SAMOD bit is invalid when the MODE bit is 1 (repeat mode) and the RPTSEL bit is 1 (transfer source is the repeat area).																									
Repeat area selection																									
RPTSEL	Repeat area selection																								
0	Transfer destination is the repeat area																								
1	Transfer source is the repeat area																								
The setting of the RPTSEL bit is invalid when the MODE bit is 0 (normal mode).																									
Transfer mode selection																									
MODE	Transfer mode selection																								
0	Normal mode																								
1	Repeat mode																								

**Caution** Do not access the DTCCRj register using a DTC transfer.

### 19.3.6 DTC block size register j (DTBLSj) (j = 0 to 23)

This register is used to set the block size of the data to be transferred by one activation.

**Figure 19 - 7 Format of DTC block size register j (DTBLSj)**

Symbol	7	6	5	4	3	2	1	0
DTBLSj	DTBLSj7	DTBLSj6	DTBLSj5	DTBLSj4	DTBLSj3	DTBLSj2	DTBLSj1	DTBLSj0
Transfer Block Size								
8-Bit Transfer								
00H	256 bytes				512 bytes			
01H	1 byte				2 bytes			
02H	2 bytes				4 bytes			
03H	3 bytes				6 bytes			
•	•				•			
•	•				•			
•	•				•			
FDH	253 bytes				506 bytes			
FEH	254 bytes				508 bytes			
FFH	255 bytes				510 bytes			

**Caution** Do not access the DTBLSj register using a DTC transfer.

### 19.3.7 DTC transfer count register j (DTCCTj) (j = 0 to 23)

This register is used to set the number of DTC data transfers. The value is decremented by 1 each time DTC transfer is activated once.

**Figure 19 - 8 Format of DTC transfer count register j (DTCCTj)**

Symbol	7	6	5	4	3	2	1	0
DTCCTj	DTCCTj7	DTCCTj6	DTCCTj5	DTCCTj4	DTCCTj3	DTCCTj2	DTCCTj1	DTCCTj0
Number of Transfers								
Once								
00H	256 times							
01H	Once							
02H	2 times							
03H	3 times							
•	•							
•	•							
•	•							
FDH	253 times							
FEH	254 times							
FFH	255 times							

**Caution** Do not access the DTCCTj register using a DTC transfer.

### 19.3.8 DTC transfer count reload register j (DTRLDj) (j = 0 to 23)

This register is used to set the initial value of the transfer count register in repeat mode. Since the value of this register is reloaded to the DTCTC register in repeat mode, set the same value as the initial value of the DTCTC register.

**Figure 19 - 9 Format of DTC transfer count reload register j (DTRLDj)**

Address: Refer to 19.3.2 Control Data Allocation.	After reset: Undefined	R/W
7 6 5 4 3 2 1 0		
DTRLDj	DTRLDj7   DTRLDj6   DTRLDj5   DTRLDj4   DTRLDj3   DTRLDj2   DTRLDj1   DTRLDj0	

**Caution** Do not access the DTRLDj register using a DTC transfer.

### 19.3.9 DTC source address register j (DTSARj) (j = 0 to 23)

This register is used to specify the transfer source address for data transfer.

When the SZ bit in the DTCCRj register is set to 1 (16-bit transfer), the lowest bit is ignored and the address is handled as an even address.

**Figure 19 - 10 Format of DTC source address register j (DTSARj)**

Address: Refer to 19.3.2 Control Data Allocation.	After reset: Undefined	R/W
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
DTSARj	DTS ARj15   DTS ARj14   DTS ARj13   DTS ARj12   DTS ARj11   DTS ARj10   DTS ARj9   DTS ARj8   DTS ARj7   DTS ARj6   DTS ARj5   DTS ARj4   DTS ARj3   DTS ARj2   DTS ARj1   DTS ARj0	

**Caution 1.** Do not set the general-purpose register (FFEE0H to FFEFFFH) space to the transfer source address.

**Caution 2.** Do not access the DTSARj register using a DTC transfer.

### 19.3.10 DTC destination address register j (DTDARj) (j = 0 to 23)

This register is used to specify the transfer destination address for data transfer.

When the SZ bit in the DTCCRj register is set to 1 (16-bit transfer), the lowest bit is ignored and the address is handled as an even address.

**Figure 19 - 11 Format of DTC destination address register j (DTDARj)**

Address: Refer to 19.3.2 Control Data Allocation.	After reset: Undefined	R/W
15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0		
DTDARj	DTD ARj15   DTD ARj14   DTD ARj13   DTD ARj12   DTD ARj11   DTD ARj10   DTD ARj9   DTD ARj8   DTD ARj7   DTD ARj6   DTD ARj5   DTD ARj4   DTD ARj3   DTD ARj2   DTD ARj1   DTD ARj0	

**Caution 1.** Do not set the general-purpose register (FFEE0H to FFEFFFH) space to the transfer source address.

**Caution 2.** Do not access the DTDARj register using a DTC transfer.

### 19.3.11 DTC activation enable register i (DTCENi) (i = 0 to 4)

This is an 8-bit register which enables or disables DTC activation by interrupt sources. Table 19 - 6 lists the Correspondences between Interrupt Sources and Bits DTCENi0 to DTCENi7.

The DTCENi register can be set by an 8-bit memory manipulation instruction and a 1-bit memory manipulation instruction.

**Caution 1. Modify bits DTCENi0 to DTCENi7 if an activation source corresponding to the bit has not been generated.**

**Caution 2. Do not access the DTCENi register using a DTC transfer.**

**Caution 3. The assigned functions differ depending on the product. For the bits to which no function is assigned, be sure to set their values to 0.**

**Figure 19 - 12 Format of DTC activation enable register i (DTCENi) (i = 0 to 4)**

Address: F02E8H (DTCEN0), F02E9H (DTCEN1), F02EAH (DTCEN2), After reset: 00H R/W  
F02EBH (DTCEN3), F02ECH (DTCEN4)

Symbol	7	6	5	4	3	2	1	0								
DTCENi	DTCENi7	DTCENi6	DTCENi5	DTCENi4	DTCENi3	DTCENi2	DTCENi1	DTCENi0								
DTCENi7   DTC activation enable i7																
0	Activation disabled															
1	Activation enabled															
The DTCENi7 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.																
DTCENi6   DTC activation enable i6																
0	Activation disabled															
1	Activation enabled															
The DTCENi6 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.																
DTCENi5   DTC activation enable i5																
0	Activation disabled															
1	Activation enabled															
The DTCENi5 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.																
DTCENi4   DTC activation enable i4																
0	Activation disabled															
1	Activation enabled															
The DTCENi4 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.																
DTCENi3   DTC activation enable i3																
0	Activation disabled															
1	Activation enabled															
The DTCENi3 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.																

DTCENi2		DTC activation enable i2
0	Activation disabled	
1	Activation enabled	

The DTCENi2 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.

DTCENi1		DTC activation enable i1
0	Activation disabled	
1	Activation enabled	

The DTCENi1 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.

DTCENi0		DTC activation enable i0
0	Activation disabled	
1	Activation enabled	

The DTCENi0 bit is set to 0 (activation disabled) by a condition for generating a transfer end interrupt.

**Table 19 - 6 Correspondences between Interrupt Sources and Bits DTCENi0 to DTCENi7**

Register	DTCENi7 Bit	DTCENi6 Bit	DTCENi5 Bit	DTCENi4 Bit	DTCENi3 Bit	DTCENi2 Bit	DTCENi1 Bit	DTCENi0 Bit
DTCEN0	Reserved	INTP0	INTP1	INTP2	INTP3	INTP4	INTP5	INTP6
DTCEN1	INTP7	Key input	A/D conversion end	UART0 reception transfer end/CSI01 transfer end or buffer empty/IIC01 transfer end	UART0 transmission transfer end/CSI00 transfer end or buffer empty/IIC00 transfer end	UART1 reception transfer end/CSI11 transfer end or buffer empty/IIC11 transfer end	UART1 transmission transfer end/CSI10 transfer end or buffer empty/IIC10 transfer end	UART2 reception transfer end/CSI21 transfer end or buffer empty/IIC21 transfer end
DTCEN2	UART2 transmission transfer end/CSI20 transfer end or buffer empty/IIC20 transfer end Note 1	UART3 reception transfer end/CSI31 transfer end or buffer empty/IIC31 transfer end Note 1	UART3 transmission transfer end/CSI30 transfer end or buffer empty/IIC30 transfer end Note 1	End of channel 0 of timer array unit 0 count or capture	End of channel 1 of timer array unit 0 count or capture	End of channel 2 of timer array unit 0 count or capture	End of channel 3 of timer array unit 0 count or capture	End of channel 0 of timer array unit 1 count or capture Note 1
DTCEN3	End of channel 1 of timer array unit 1 count or capture Note 1	End of channel 2 of timer array unit 1 count or capture Note 1	End of channel 3 of timer array unit 1 count or capture Note 1	Timer RD compare match A0	Timer RD compare match B0	Timer RD compare match C0	Timer RD compare match D0	Timer RD compare match A1
DTCEN4	Timer RD compare match B1	Timer RD compare match C1	Timer RD compare match D1	Timer RG compare match A	Timer RG compare match B	Timer RJ0 underflow	Comparator detection 0 Note 2	Comparator detection 1 Note 2

**Note 1.** For 80- and 100-pin products only.

**Note 2.** For 96 KB or more code flash memory products only.

**Caution** **For the bits to which no function is assigned, be sure to set their values to 0.**

**Remark** i = 0 to 4

### 19.3.12 DTC base address register (DTCBAR)

This is an 8-bit register used to set the following addresses: the vector address where the start address of the DTC control data area is stored and the address of the DTC control data area. The value of the DTCBAR register is handled as the higher 8 bits to generate a 16-bit address.

**Caution 1. Change the DTCBAR register value with all DTC activation sources set to activation disabled.**

**Caution 2. Do not rewrite the DTCBAR register more than once.**

**Caution 3. Do not access the DTCBAR register using a DTC transfer.**

**Caution 4. For the allocation of the DTC control data area and the DTC vector table area, refer to the notes on 19.3.1 Allocation of DTC Control Data Area and DTC Vector Table Area.**

**Figure 19 - 13 Format of DTC base address register (DTCBAR)**

Address: F02E0H      After reset: FDH      R/W

Symbol	7	6	5	4	3	2	1	0
DTCBAR	DTCBAR7	DTCBAR6	DTCBAR5	DTCBAR4	DTCBAR3	DTCBAR2	DTCBAR1	DTCBAR0

## 19.4 DTC Operation

When the DTC is activated, control data is read from the DTC control data area to perform data transfers and control data after data transfer is written back to the DTC control data area. Twenty-four sets of control data can be stored in the DTC control data area, which allows 24 types of data transfers to be performed.

There are two transfer modes (normal mode and repeat mode) and two transfer sizes (8-bit transfer and 16-bit transfer). When the CHNE bit in the DTCCR $j$  ( $j = 0$  to 23) register is set to 1 (chain transfers enabled), multiple control data is read and data transfers are continuously performed by one activation source (chain transfers).

A transfer source address is specified by the 16-bit register DTSAR $j$ , and a transfer destination address is specified by the 16-bit register DTDAR $j$ .

The values in registers DTSAR $j$  and DTDAR $j$  are separately incremented or fixed according to the control data after the data transfer.

### 19.4.1 Activation Sources

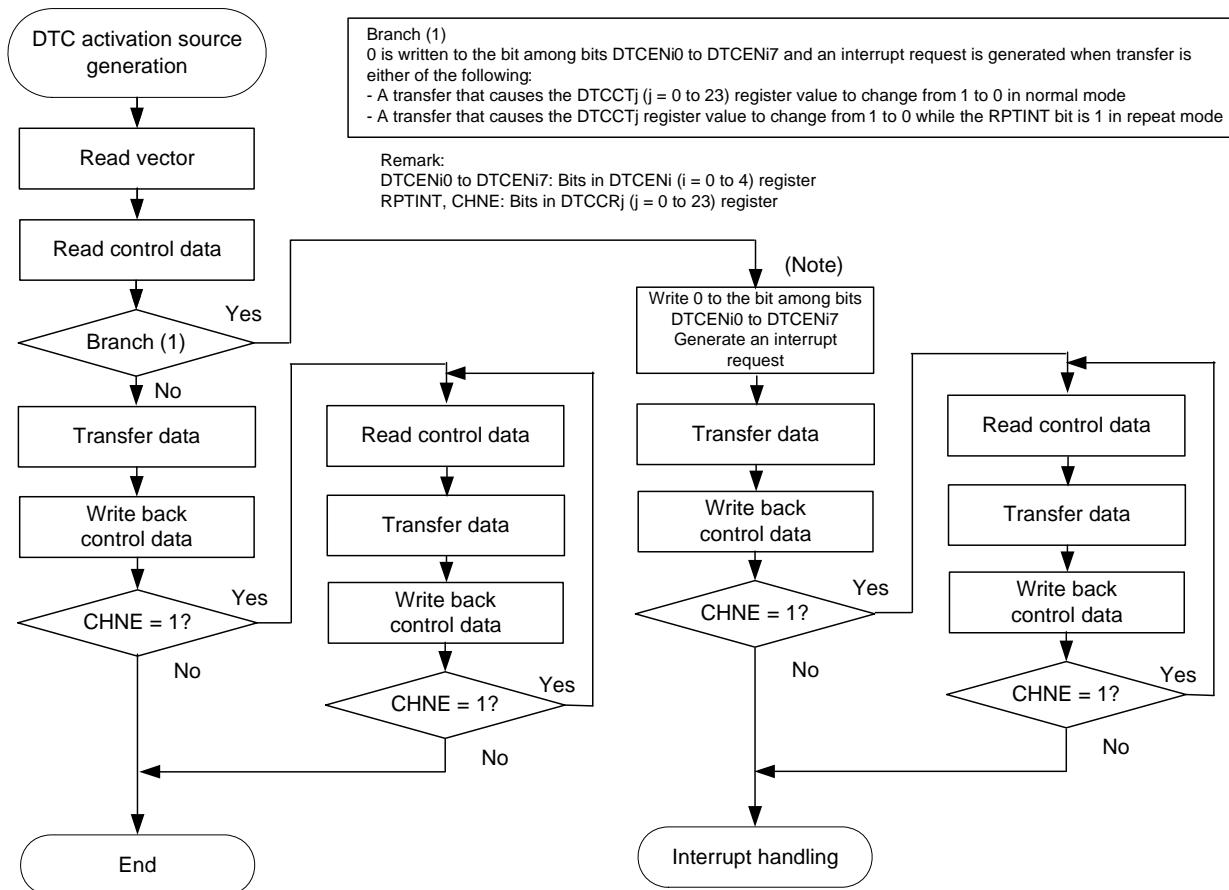
The DTC is activated by an interrupt signal from the peripheral functions. The interrupt signals to activate the DTC are selected with the DTCENi ( $i = 0$  to 4) register.

The DTC sets the corresponding bit among bits DTCENi0 to DTCENi7 in the DTCENi register to 0 (activation disabled) during operation when the setting of data transfer (the first transfer in chain transfers) is either of the following:

- A transfer that causes the DTCTCj ( $j = 0$  to 23) register value to change to 0 in normal mode
- A transfer that causes the DTCTCj register value to change to 0 while the RPTINT bit in the DTCCRj register is 1 (interrupt generation enabled) in repeat mode

Figure 19 - 14 shows the DTC Internal Operation Flowchart.

**Figure 19 - 14 DTC Internal Operation Flowchart**



**Note** 0 is not written to the bit among bits DTCENi0 to DTCENi7 for data transfers activated by the setting to enable chain transfers (the CHNE bit is 1). Also, no interrupt request is generated.

### 19.4.2 Normal Mode

One to 256 bytes of data are transferred by one activation during 8-bit transfer and 2 to 512 bytes during 16-bit transfer. The number of transfers can be 1 to 256 times. When the data transfer causing the DTCCT<sub>j</sub> ( $j = 0$  to 23) register value to change to 0 is performed, the DTC generates an interrupt request corresponding to the activation source to the interrupt controller during DTC operation, and sets the corresponding bit among bits DTCEN<sub>i0</sub> to DTCEN<sub>i7</sub> ( $i = 0$  to 4) in the DTCEN<sub>i</sub> register to 0 (activation disabled).

Table 19 - 7 shows Register Functions in Normal Mode. Figure 19 - 15 shows Data Transfers in Normal Mode.

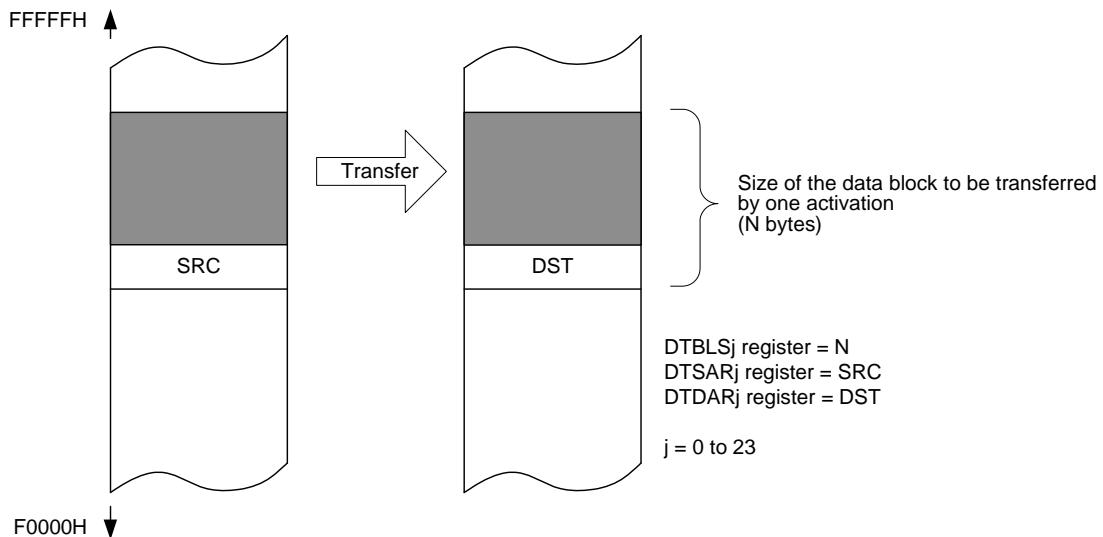
**Table 19 - 7 Register Functions in Normal Mode**

Register Name	Symbol	Function
DTC block size register j	DTBLS <sub>j</sub>	Size of the data block to be transferred by one activation
DTC transfer count register j	DTCCT <sub>j</sub>	Number of data transfers
DTC transfer count reload register j	DTRLD <sub>j</sub>	Not used Note
DTC source address register j	DTSAR <sub>j</sub>	Data transfer source address
DTC destination address register j	DTDAR <sub>j</sub>	Data transfer destination address

**Note** Initialize this register to 00H when parity error resets are enabled (RPERDIS = 0) using the RAM parity error detection function.

**Remark**  $j = 0$  to 23

**Figure 19 - 15 Data Transfers in Normal Mode**



DTCCR Register Setting				Source Address Control	Destination Address Control	Source Address after Transfer	Destination Address after Transfer
DAMOD	SAMOD	RPTSEL	MODE				
0	0	X	0	Fixed	Fixed	SRC	DST
0	1	X	0	Incremented	Fixed	SRC + N	DST
1	0	X	0	Fixed	Incremented	SRC	DST + N
1	1	X	0	Incremented	Incremented	SRC + N	DST + N

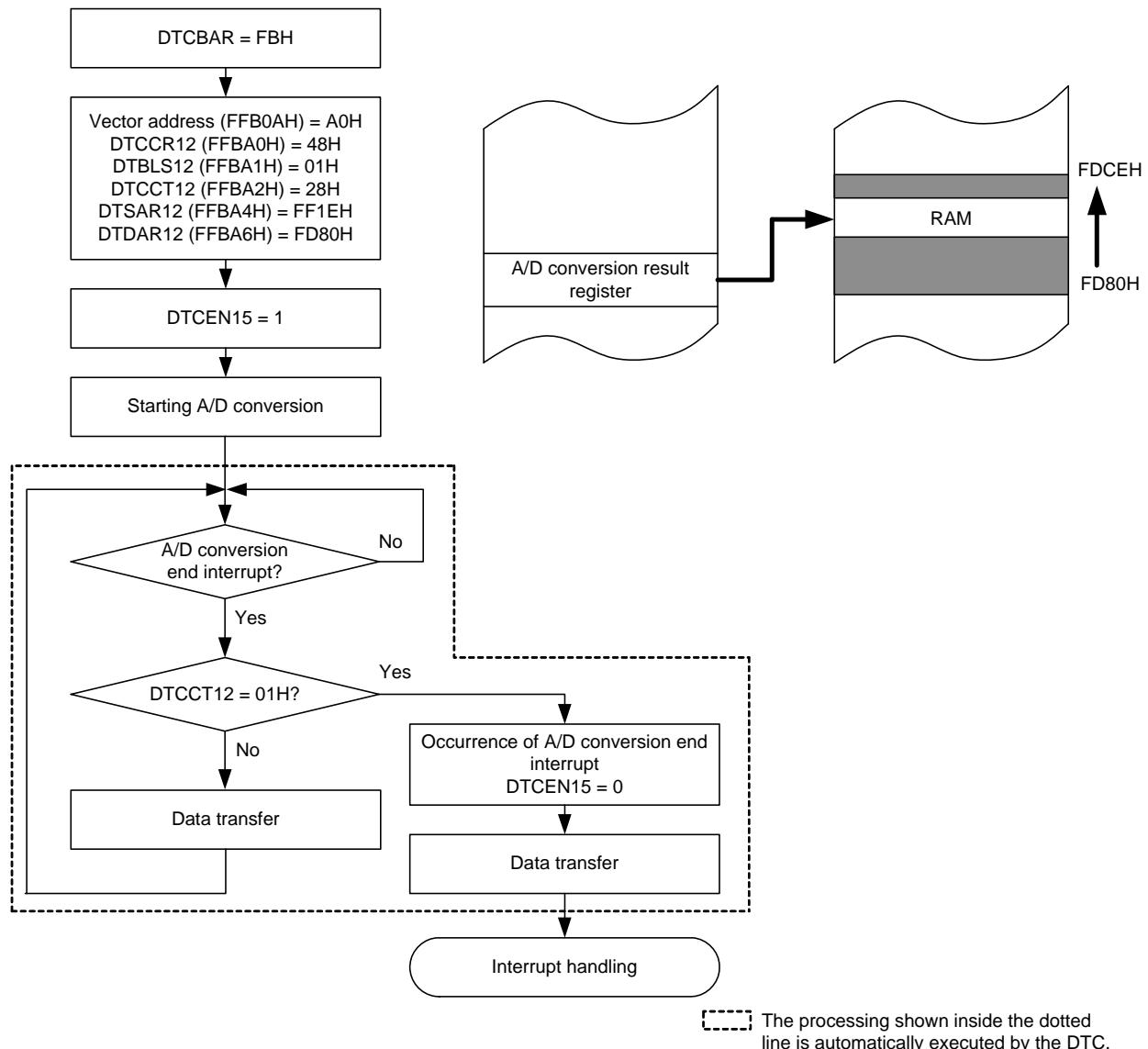
X: 0 or 1

## (1) Example 1 of using normal mode: Consecutively capturing A/D conversion results

The DTC is activated by an A/D conversion end interrupt and the value of the A/D conversion result register is transferred to RAM.

- The vector address is FFB0AH and control data is allocated at FFBA0H to FFBA7H.
- Transfers 2-byte data of the A/D conversion result register (FFF1EH, FFF1FH) to 80 bytes of FFD80H to FFDCFH of RAM 40 times.

**Figure 19 - 16 Example 1 of using normal mode: Consecutively capturing A/D conversion results**



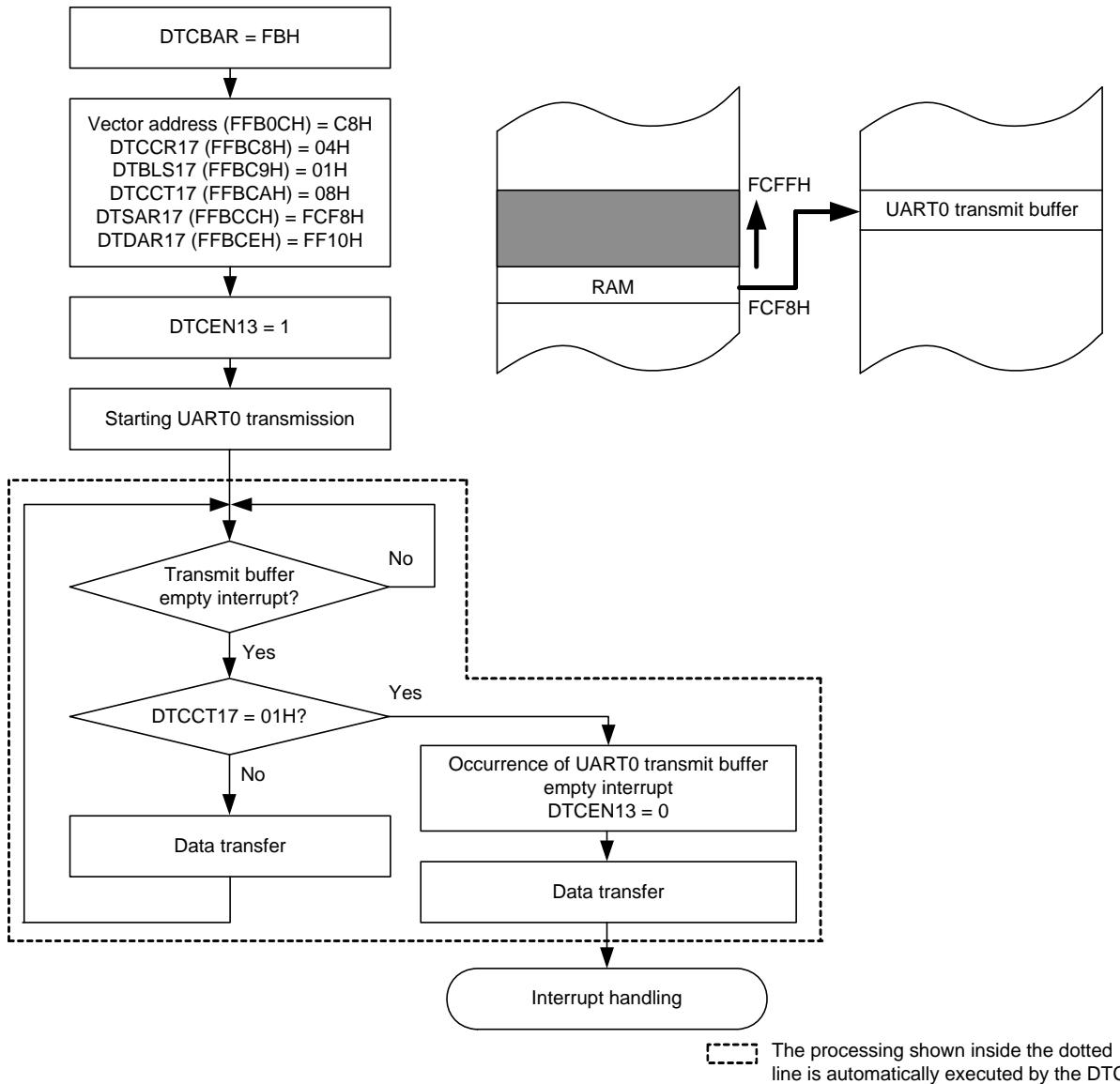
The value of the DTRLD12 register is not used because of normal mode, but initialize the register to 00H when parity error resets are enabled (RPERDIS = 0) using the RAM parity error detection function.

## (2) Example 2 of using normal mode: UART0 consecutive transmission

The DTC is activated by a UART0 transmit buffer empty interrupt and the value of RAM is transferred to the UART0 transmit buffer.

- The vector address is FFB0CH and control data is allocated at FFBC8H to FFBCFH
- Transfers 8 bytes of FFCF8H to FFCFFH of RAM to the UART0 transmit buffer (FFF10H)

**Figure 19 - 17 Example 2 of using normal mode: UART0 consecutive transmission**



The value of the DTRLD17 register is not used because of normal mode, but initialize the register to 00H when parity error resets are enabled (RPERDIS = 0) using the RAM parity error detection function.

Start the first UART0 transmission by software. The second and subsequent transmissions are automatically sent when the DTC is activated by a transmit buffer empty interrupt.

### 19.4.3 Repeat Mode

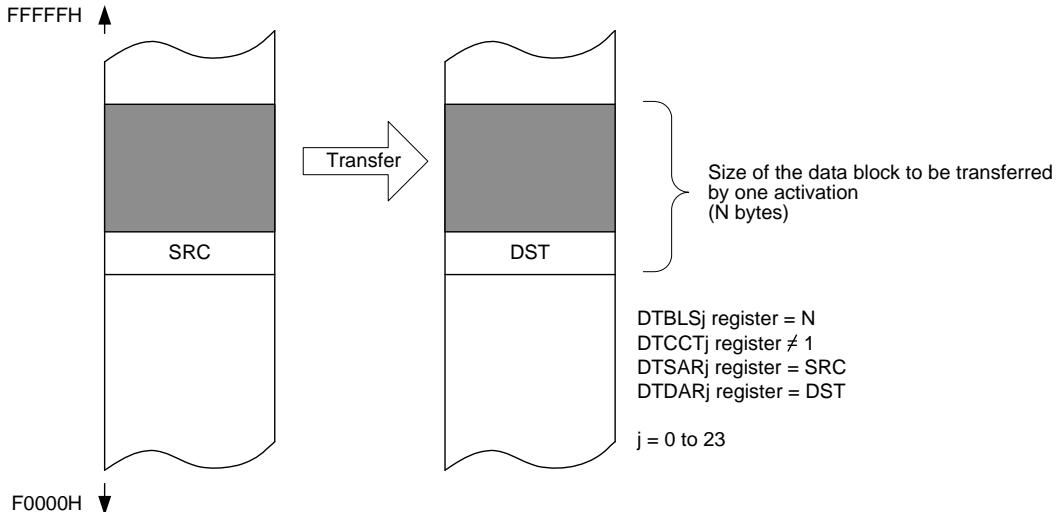
One to 255 bytes of data are transferred by one activation. Either of the transfer source or destination should be specified as the repeat area. The number of transfers can be 1 to 255 times. On completion of the specified number of transfers, the DTCCT $j$  ( $j = 0$  to 23) register and the address specified for the repeat area are initialized to continue transfers. When the data transfer causing the DTCCT $j$  register value to change to 0 is performed while the RPTINT bit in the DTCCR $j$  register is 1 (interrupt generation enabled), the DTC generates an interrupt request corresponding to the activation source to the interrupt controller during DTC operation, and sets the corresponding bit among bits DTCENi0 to DTCENi7 to 0 (activation disabled). When the RPTINT bit in the DTCCR $j$  register is 0 (interrupt generation disabled), no interrupt request is generated even if the data transfer causing the DTCCT $j$  register value to change to 0 is performed. Also, bits DTCENi0 to DTCENi7 are not set to 0. Table 19 - 8 lists Register Functions in Repeat Mode. Figure 19 - 18 shows Data Transfers in Repeat Mode.

**Table 19 - 8 Register Functions in Repeat Mode**

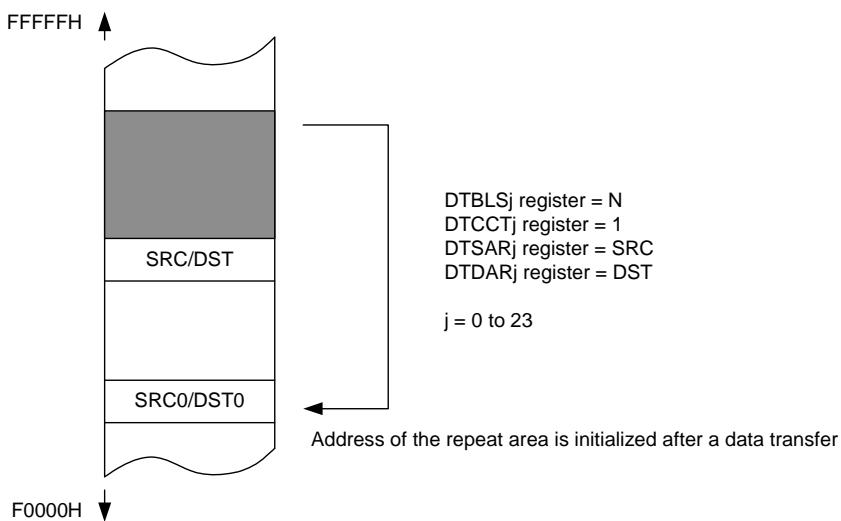
Register Name	Symbol	Function
DTC block size register $j$	DTBLS $j$	Size of the data block to be transferred by one activation
DTC transfer count register $j$	DTCCT $j$	Number of data transfers
DTC transfer count reload register $j$	DTRLD $j$	This register value is reloaded to the DTCCT register (the number of transfers is initialized).
DTC source address register $j$	DTSAR $j$	Data transfer source address
DTC destination address register $j$	DTDAR $j$	Data transfer destination address

**Remark**  $j = 0$  to 23

Figure 19 - 18 Data Transfers in Repeat Mode

DTCCT<sub>j</sub> register ≠ 1

X: 0 or 1

DTCCT<sub>j</sub> register = 1

SRC0: Initial source address value

DST0: Initial destination address value

X: 0 or 1

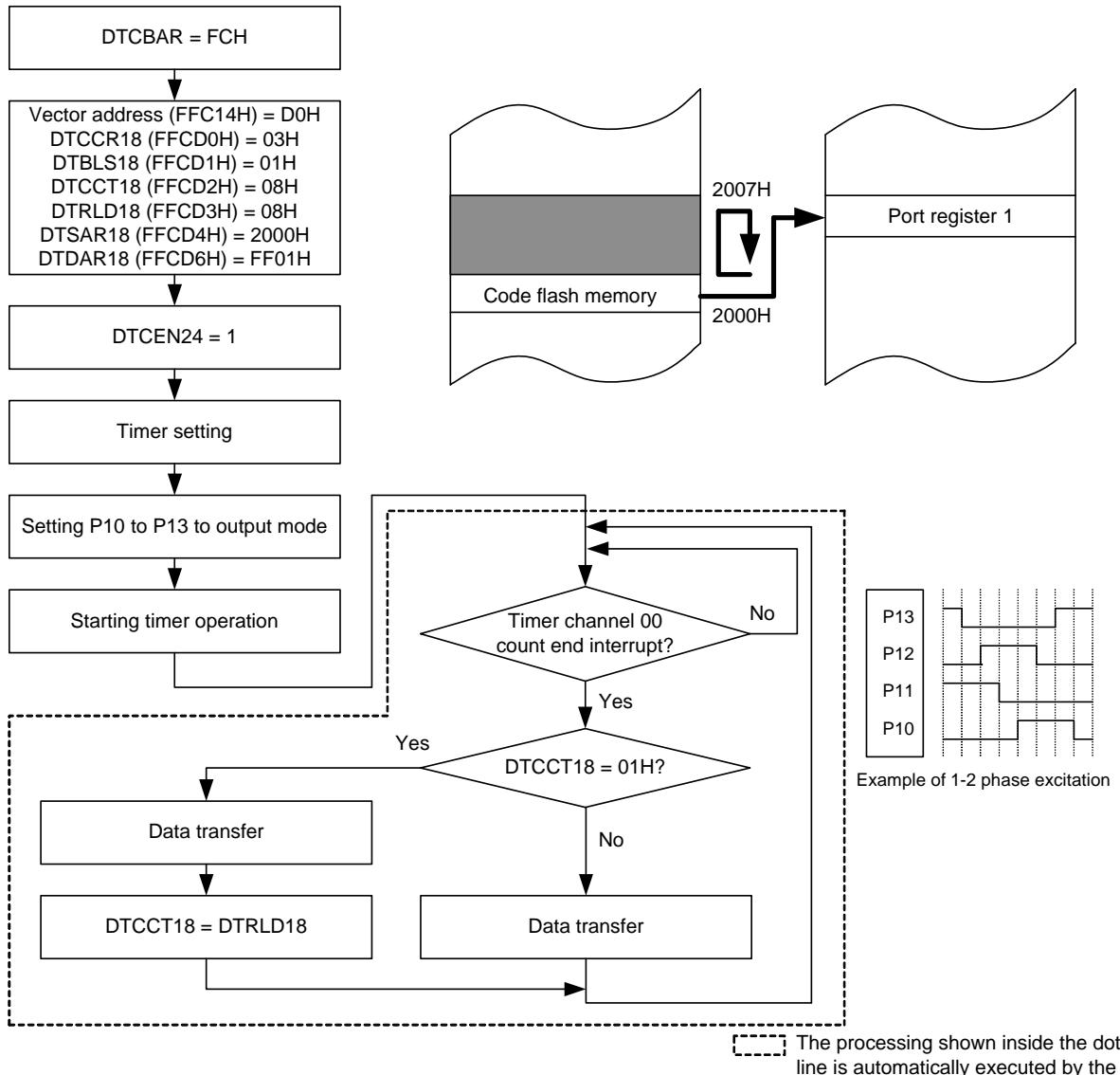
**Caution 1.** When repeat mode is used, the lower 8 bits of the initial value for the repeat area address must be 00H.**Caution 2.** When repeat mode is used, the data size of the repeat area must be set to 255 bytes or less.

## (1) Example 1 of using repeat mode: Outputting a stepping motor control pulse using ports

The DTC is activated using the interval timer function of channel 0 of timer array unit 0, and the pattern of the motor control pulse stored in the code flash memory is transferred to the general-purpose port.

- The vector address is FFC14H and control data is allocated at FFCD0H to FFCD7H
- Transfers 8-byte data of 02000H to 02007H of the code flash memory from the mirror area (F2000H to F2007H) to port register 1 (FFF01H)
- A repeat mode interrupt is disabled

**Figure 19 - 19 Example 1 of using repeat mode: Outputting a stepping motor control pulse using ports**



To stop the output, stop the timer first and then clear DTCEN24.

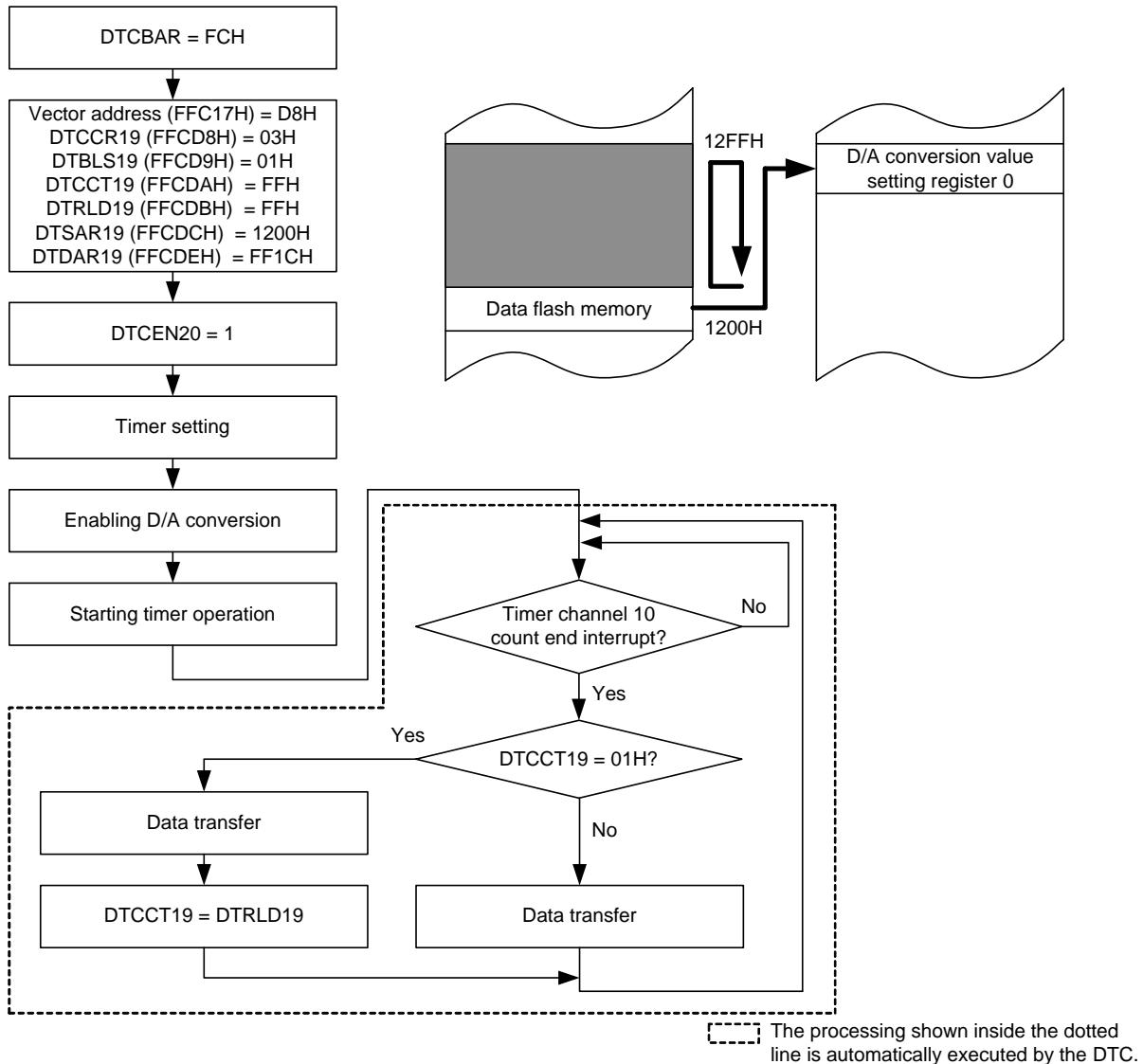
- (2) Example 2 of using repeat mode: Outputting a sine wave using the 8-bit D/A converter

The DTC is activated using the interval timer function of channel 0 of timer array unit 1, and the table of the sine wave stored in the data flash memory is transferred to the 8-bit D/A conversion value setting register 0 (FFF34H).

The timer interval time is set to the D/A output setup time.

- The vector address is FFC17H and control data is allocated at FFCD8H to FFCD9H
- Transfers 255-byte data of F1200H to F12FEH of the data flash memory to the D/A conversion value setting register (FFF34H)
- A repeat mode interrupt is disabled

**Figure 19 - 20 Example 2 of using repeat mode: Outputting a sine wave using the 8-bit D/A converter**



To stop the output, stop the timer first and then clear DTCEN20.

**Caution** A D/A converter is provided in products with 96 KB or more code flash memory.  
Timer array unit 1 is provided in 80 and 100-pin products.

#### 19.4.4 Chain Transfers

When the CHNE bit in the DTCCR $j$  ( $j = 0$  to  $22$ ) register is 1 (chain transfers enabled), multiple data transfers can be continuously performed by one activation source.

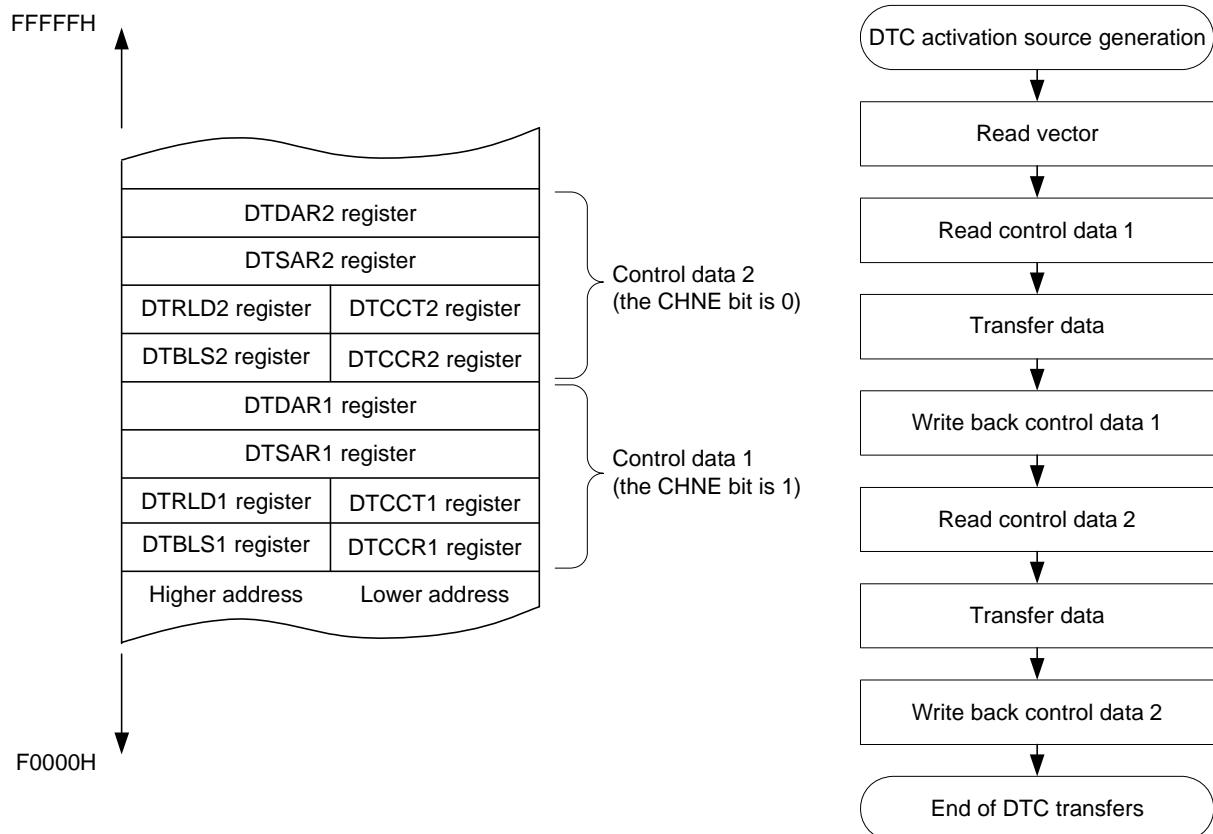
When the DTC is activated, one control data is selected according to the data read from the vector address corresponding to the activation source, and the selected control data is read from the DTC control data area.

When the CHNE bit for the control data is 1 (chain transfers enabled), the next control data immediately following the current control data is read and transferred after the current transfer is completed. This operation is repeated until the data transfer with the control data for which the CHNE bit is 0 (chain transfers disabled) is completed.

When chain transfers are performed using multiple control data, the number of transfers set for the first control data is enabled, and the number of transfers set for the second and subsequent control data to be processed will be invalid.

Figure 19 - 21 shows Data Transfers during Chain Transfers.

**Figure 19 - 21 Data Transfers during Chain Transfers**

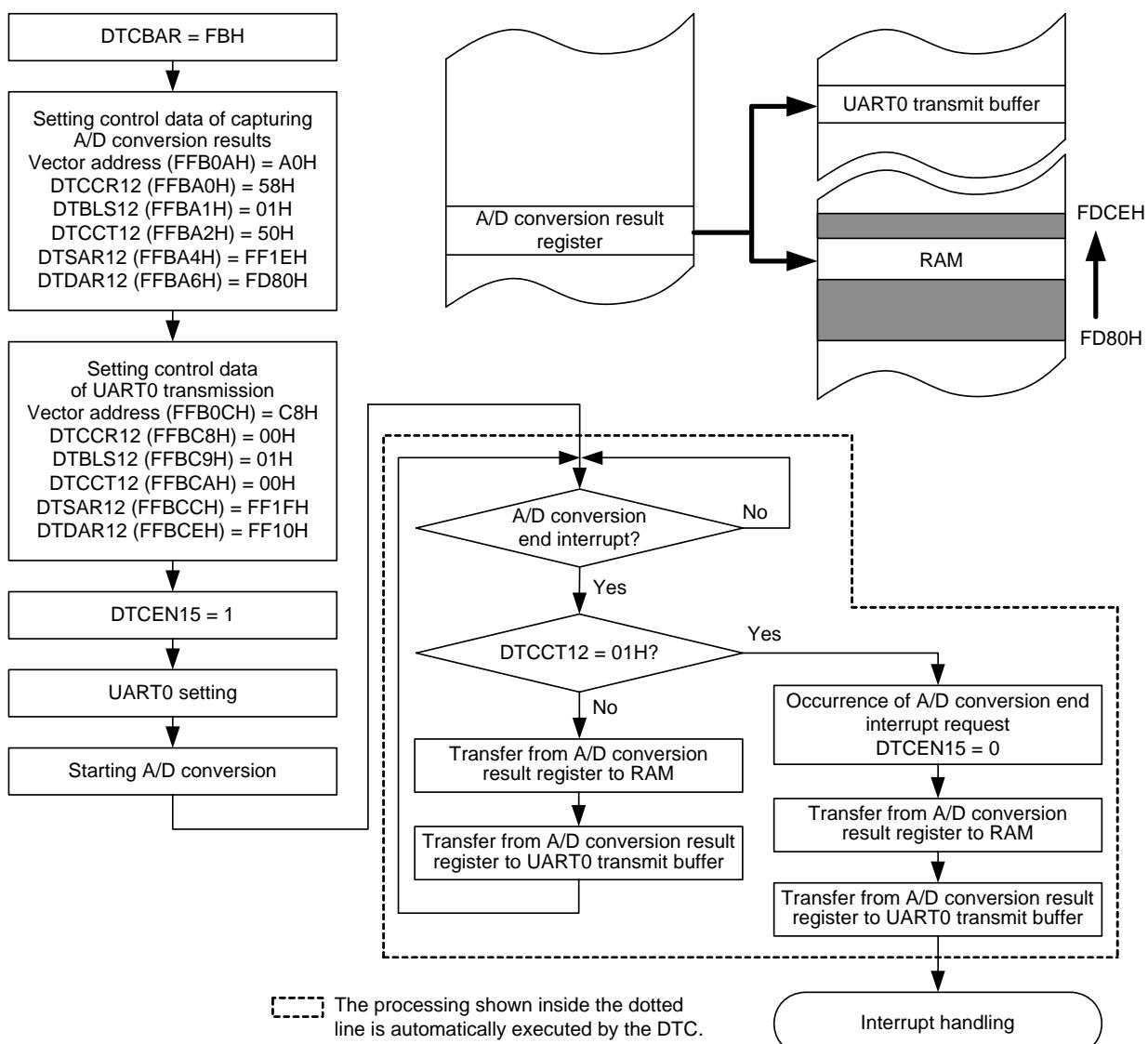


**Note 1.** Set the CHNE bit in the DTCCR23 register to 0 (chain transfers disabled).

**Note 2.** During chain transfers, bits DTCENi0 to DTCENi7 ( $i = 0$  to  $4$ ) in the DTCENi register are not set to 0 (activation disabled) for the second and subsequent transfers. Also, no interrupt request is generated.

- (1) Example of using chain transfers: Consecutively capturing A/D conversion results and UART0 transmission
- The DTC is activated by an A/D conversion end interrupt and A/D conversion results are transferred to RAM, and then transmitted using the UART0.
- The vector address is FFB0AH
  - Control data of capturing A/D conversion results is allocated at FFBA0H to FFBA7H
  - Control data of UART0 transmission is allocated at FFBA8H at FFBAFH
  - Transfers 2-byte data of the A/D conversion result register (FFF1FH, FFF1EH) to FFD80H to FFDCFH of RAM, and transfers the upper 1 byte (FFF1FH) of the A/D conversion result register to the UART transmit buffer (FFF10H)

**Figure 19 - 22 Example of using chain transfers: Consecutively capturing A/D conversion results and UART0 transmission**



## 19.5 Cautions for DTC

### 19.5.1 Setting DTC Control Data and Vector Table

- Do not access the DTC extended special function register (2nd SFR), the DTC control data area, the DTC vector table area, or the general-register (FFEE0H to FFEFFFH) space using a DTC transfer.
- Modify the DTC base address register (DTCBAR) while all DTC activation sources are set to activation disabled.
- Do not rewrite the DTC base address register (DTCBAR) twice or more.
- Modify the data of the DTCCRj, DTBLSj, DTCCTj, DTRL Dj, DTSARj, or DTDARj register when the corresponding bit among bits DTCENi0 to DTCENi7 in the DTCENi (i = 0 to 4) register is 0 (activation disabled).
- Modify the start address of the DTC control data area to be set in the vector table when the corresponding bit among bits DTCENi0 to DTCENi7 in the DTCENi (i = 0 to 4) register is 0 (activation disabled).
- Do not allocate RAM addresses which are used as a DTC transfer destination/transfer source to the area FFE20H to FFEDFH when performing self-programming and rewriting the data flash memory.

### 19.5.2 Allocation of DTC Control Data Area and DTC Vector Table Area

The areas where the DTC control data and vector table can be allocated differ, depending on the usage conditions.

- It is prohibited to use the general-purpose register (FFEE0H to FFEFFFH) space as the DTC control data area or DTC vector table area.
- Make sure the stack area, the DTC control data area, and the DTC vector table area do not overlap.
- The internal RAM area in the following products cannot be used as the DTC control data area or DTC vector table area when using the self-programming and data-flash functions.

R5F104xD (x = A to C, E to G, J, L): FE900H to FED09H

R5F104xE (x = A to C, E to G, J, L): FE900H to FED09H

R5F104xJ (x = A to C, E to G, J, L): F9F00H to FA309H

R5F104xL (x = G, L, M, P): F3F00H to F4309H

- The internal RAM area in the following products cannot be used as the DTC control data area or DTC vector table area when using the on-chip debugging trace function.

R5F104xJ (x = A to C, E to G, J, L): FA300H to FA6FFH

R5F104xL (x = G, L, M, P): F4300H to F46FFH

- Initialize the DTRLD register to 00H even in normal mode when parity error resets are enabled (RPERDIS = 0) using the RAM parity error detection function.

### 19.5.3 DTC Pending Instruction

Even if a DTC transfer request is generated, DTC transfer is held pending immediately after the following instructions. Also, the DTC is not activated between PREFIX instruction code and the instruction immediately after that code.

- Call/return instruction
- Unconditional branch instruction
- Conditional branch instruction
- Read access instruction for code flash memory
- Bit manipulation instructions for IFxx, MKxx, PRxx, and PSW, and an 8-bit manipulation instruction that has the ES register as operand
- Instruction for accessing the data flash memory
- Instruction of Multiply, Divide, Multiply & Accumulate (excluding MULU)

**Caution 1. When a DTC transfer request is acknowledged, all interrupt requests are held pending until DTC transfer is completed.**

**Caution 2. While the DTC is held pending by the DTC pending instruction, all interrupt requests are held pending.**

### 19.5.4 Operation when Accessing Data Flash Memory Space

When accessing the data flash space after an instruction execution from the start of DTC data transfer, a wait of three clock cycles will be inserted to the next instruction.

Instruction 1  
DTC data transfer  
Instruction ← The wait of three clock cycles occurs.  
MOV A, ! Data Flash space

### 19.5.5 Number of DTC Execution Clock Cycles

Table 19 - 9 lists the Operations Following DTC Activation and Required Number of Cycles for each operation.

**Table 19 - 9 Operations Following DTC Activation and Required Number of Cycles**

Vector Read	Control Data		Data Read	Data Write
	Read	Write-back		
1	4	Note 1	Note 2	Note 2

**Note 1.** For the number of clock cycles required for control data write-back, refer to **Table 19 - 10 Number of Clock Cycles Required for Control Data Write-Back Operation**.

**Note 2.** For the number of clock cycles required for data read/write, refer to **Table 19 - 11 Number of Clock Cycles Required for One Data Read/Write Operation**.

**Table 19 - 10 Number of Clock Cycles Required for Control Data Write-Back Operation**

DTCCR Register Setting				Address Setting		Control Register to be Written Back				Number of Clock Cycles
DAMOD	SAMOD	RPTSEL	MODE	Source	Destination	DTCCTj Register	DTRLDj Register	DTSARj Register	DTDARj Register	
0	0	X	0	Fixed	Fixed	Written back	Written back	Not written back	Not written back	1
0	1	X	0	Incremented	Fixed	Written back	Written back	Written back	Not written back	2
1	0	X	0	Fixed	Incremented	Written back	Written back	Not written back	Written back	2
1	1	X	0	Incremented	Incremented	Written back	Written back	Written back	Written back	3
0	X	1	1	Repeat area	Fixed	Written back	Written back	Written back	Not written back	2
1	X	1	1		Incremented	Written back	Written back	Written back	Written back	3
X	0	0	1	Fixed	Repeat area	Written back	Written back	Not written back	Written back	2
X	1	0	1	Incremented		Written back	Written back	Written back	Written back	3

**Remark** j = 0 to 23; X: 0 or 1

**Table 19 - 11 Number of Clock Cycles Required for One Data Read/Write Operation**

Operation	RAM	Code Flash Memory	Data Flash Memory	Special function register (SFR)	Extended special function register (2nd SFR)		
					No Wait State	Wait States	
Data read	1	2	4	1	1	1 + number of wait states	Note
Data write	1	—	—	1	1	1 + number of wait states	Note

**Note** The number of wait states differs depending on the specifications of the register allocated to the extended special function register (2nd SFR) to be accessed.

### 19.5.6 DTC Response Time

Table 19 - 12 lists the DTC Response Time. The DTC response time is the time from when the DTC activation source is detected until DTC transfer starts. It does not include the number of DTC execution clocks.

**Table 19 - 12 DTC Response Time**

	Minimum Time	Maximum Time
Response Time	3 clocks	19 clocks

Note that the response from the DTC may be further delayed under the following cases. The number of delayed clock cycles differs depending on the conditions.

- When executing an instruction from the internal RAM  
Maximum response time: 20 clocks
- When executing a DTC pending instruction (refer to **19.5.3 DTC Pending Instruction**)
- Maximum response time: Maximum response time for each condition + execution clock cycles for the instruction to be held pending under the condition.
- When accessing the TRJ0 register that a wait occurs  
Maximum response time: Maximum response time for each condition + 1 clock

**Remark** 1 clock: 1/fCLK (fCLK: CPU/peripheral hardware clock)

### 19.5.7 DTC Activation Sources

- After inputting a DTC activation source, do not input the same activation source again until DTC transfer is completed.
- While a DTC activation source is generated, do not manipulate the DTC activation enable bit corresponding to the source.
- If DTC activation sources conflict, their priority levels are determined in order to select the source for activation when the CPU acknowledges the DTC transfer. For details on the priority levels of activation sources, refer to **19.3.3 Vector Table**.
- When DTC activation is enabled under either of the following conditions, a DTC transfer is started and an interrupt is generated after completion of the transfer. Therefore, enable DTC activation after confirming the comparator monitor flag (CnMON) as necessary. (n = 0, 1)
  - The comparator Note is set to an interrupt request on one-edge detection (CnEDG = 0), an interrupt request at the rising edge for the comparator, and IVCMP > IVREF (or internal reference voltage: 1.45 V)
  - The comparator Note is set to an interrupt request on one-edge detection (CnEDG = 0), an interrupt request at the falling edge for the comparator, and IVCMP < IVREF (or internal reference voltage: 1.45 V)

**Note** A comparator is provided in products with 96 KB or more code flash memory.

### 19.5.8 Operation in Standby Mode Status

Status	DTC Operation
HALT mode	Operable (Operation is disabled while in the low power consumption RTC mode)
STOP mode	DTC activation sources can be accepted Note 2
SNOOZE mode	Operable Notes 1, 3, 4, 5

**Note 1.** The SNOOZE mode can only be specified when the high-speed on-chip oscillator clock is selected as fCLK.

**Note 2.** In the STOP mode, detecting a DTC activation source enables transition to SNOOZE mode and DTC transfer.

After completion of transfer, the system returns to the STOP mode. However, since the code flash memory and the data flash memory are stopped during the SNOOZE mode, the flash memory cannot be set as the transfer source.

**Note 3.** When a transfer end interrupt is set as a DTC activation source from the CSIp SNOOZE mode function, release the SNOOZE mode using the transfer end interrupt to start CPU processing after completion of DTC transfer, or use a chain transfer to set CSIp reception again (writing 1 to the STm0 bit, writing 0 to the SWCm bit, setting of the SSCm register, and writing 1 to the SSm0 bit).

**Note 4.** When a transfer end interrupt is set as a DTC activation source from the UARTq SNOOZE mode function, release the SNOOZE mode using the transfer end interrupt to start CPU processing after completion of DTC transfer, or use a chain transfer to set UARTq reception again (writing 1 to the STm1 bit, writing 0 to the SWCm bit, setting of the SSCm register, and writing 1 to the SSm1 bit).

**Note 5.** When an A/D conversion end interrupt is set as a DTC activation source from the A/D converter SNOOZE mode function, release the SNOOZE mode using the A/D conversion end interrupt to start CPU processing after completion of DTC transfer, or use a chain transfer to set the A/D converter SNOOZE mode function again (writing 0 to the AWC bit and then writing 1 to the AWC bit).

**Remark** 30 to 64-pin products: p = 00; q = 0; m = 0  
80, 100-pin products: p = 00, 20; q = 0, 2; m = 0, 1

## CHAPTER 20 EVENT LINK CONTROLLER (ELC)

### 20.1 Functions of ELC

The event link controller (ELC) mutually connects (links) events output from each peripheral function. By linking events, it becomes possible to coordinate operation between peripheral functions directly without going through the CPU.

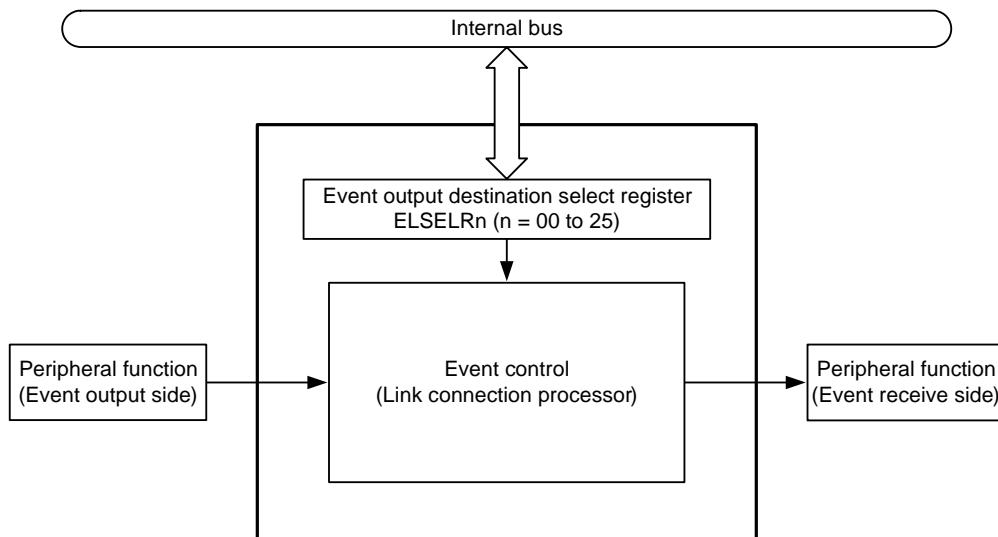
The ELC has the following functions.

- Capable of directly linking event signals from 19 to 26 types of peripheral functions to specified peripheral functions (the number of types of peripheral functions varies depending on the product).
- Event signals can be used as activation sources for operating any one of 7 to 9 types of peripheral functions (the number of types of peripheral functions varies depending on the product).

### 20.2 Configuration of ELC

Figure 20 - 1 shows the ELC Block Diagram.

**Figure 20 - 1 ELC Block Diagram**



## 20.3 Registers Controlling ELC

Table 20 - 1 lists the Registers Controlling ELC.

**Table 20 - 1 Registers Controlling ELC**

Register name	Symbol
Event output destination select register 00	ELSELR00
Event output destination select register 01	ELSELR01
Event output destination select register 02	ELSELR02
Event output destination select register 03	ELSELR03
Event output destination select register 04	ELSELR04
Event output destination select register 05	ELSELR05
Event output destination select register 06 Note 3	ELSELR06
Event output destination select register 07	ELSELR07
Event output destination select register 08	ELSELR08
Event output destination select register 09	ELSELR09
Event output destination select register 10	ELSELR10
Event output destination select register 11	ELSELR11
Event output destination select register 12	ELSELR12
Event output destination select register 13	ELSELR13
Event output destination select register 14	ELSELR14
Event output destination select register 15	ELSELR15
Event output destination select register 16	ELSELR16
Event output destination select register 17	ELSELR17
Event output destination select register 18	ELSELR18
Event output destination select register 19	ELSELR19
Event output destination select register 20 Note 1	ELSELR20
Event output destination select register 21 Note 1	ELSELR21
Event output destination select register 22 Note 1	ELSELR22
Event output destination select register 23 Note 1	ELSELR23
Event output destination select register 24 Note 2	ELSELR24
Event output destination select register 25 Note 2	ELSELR25

**Note 1.** 80- and 100-pin products only.

**Note 2.** 96 KB or more code flash memory products only.

**Note 3.** Products with at least 40 pins only.

### 20.3.1 Event output destination select register n (ELSELRn) (n = 00 to 25)

An ELSELRn register links each event signal to an operation of an event-receiving peripheral function (link destination peripheral function) after reception.

Do not set multiple event inputs to the same event output destination (event receive side). The operation of the event-receiving peripheral function will become undefined, and event signals may not be received correctly. In addition, do not set the event link generation source and the event link output destination to the same function.

Set an ELSELRn register during a period when no event output peripheral functions are generating event signals.

Table 20 - 2 lists the Correspondence Between ELSELRn (n = 00 to 25) Registers and Peripheral Functions, and Table 20 - 3 lists the Correspondence Between Values Set to ELSELRn (n = 00 to 25) Registers and Operation of Link Destination Peripheral Functions at Reception.

**Figure 20 - 2 Format of Event output destination select register n (ELSELRn)**

Address: F0300H (ELSELR00) to F0319H (ELSELR25) After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

ELSELRn	0	0	0	0	ELSELn3 Note 1	ELSELn2	ELSELn1	ELSELn0
ELSELn3 Note 1	ELSELn2	ELSELn1	ELSELn0		Event Link Selection			
0	0	0	0		Event link disabled			
0	0	0	1		Select operation of peripheral function 1 to link Note 2			
0	0	1	0		Select operation of peripheral function 2 to link Note 2			
0	0	1	1		Select operation of peripheral function 3 to link Note 2			
0	1	0	0		Select operation of peripheral function 4 to link Note 2			
0	1	0	1		Select operation of peripheral function 5 to link Note 2			
0	1	1	0		Select operation of peripheral function 6 to link Note 2			
0	1	1	1		Select operation of peripheral function 7 to link Note 2			
1	0	0	0		Select operation of peripheral function 8 to link Note 2			
1	0	0	1		Select operation of peripheral function 9 to link Note 2			
Other than above					Setting prohibited			

**Note 1.** Only for products with 96 KB or more code flash memory. Set this bit to 0 in products with 64 KB or less code flash memory.

**Note 2.** See **Table 20 - 3 Correspondence Between Values Set to ELSELRn (n = 00 to 25) Registers and Operation of Link Destination Peripheral Functions at Reception.**

**Table 20 - 2 Correspondence Between ELSELR<sub>n</sub> (n = 00 to 25) Registers and Peripheral Functions**

Register Name	Event Generator (Output Origin of Event Input n)	Event Description
ELSELR00	External interrupt edge detection 0	INTP0
ELSELR01	External interrupt edge detection 1	INTP1
ELSELR02	External interrupt edge detection 2	INTP2
ELSELR03	External interrupt edge detection 3	INTP3
ELSELR04	External interrupt edge detection 4	INTP4
ELSELR05	External interrupt edge detection 5	INTP5
ELSELR06 Note 3	Key return signal detection	INTKR
ELSELR07	RTC fixed-cycle signal/Alarm match detection	INTRTC
ELSELR08	Timer RD0 input capture A/compare match A	INTTRD0
ELSELR09	Timer RD0 input capture B/compare match B	INTTRD0
ELSELR10	Timer RD1 input capture A/compare match A	INTTRD1
ELSELR11	Timer RD1 input capture B/compare match B	INTTRD1
ELSELR12	Timer RD1 underflow	TRD1 underflow signal
ELSELR13	Timer RJ0 underflow/end of pulse width measurement period/end of pulse period measurement period	INTTRJ0
ELSELR14	Timer RG input capture A/compare match A	INTTRG
ELSELR15	Timer RG input capture B/compare match B	INTTRG
ELSELR16	TAU channel 00 count end/capture end	INTTM00
ELSELR17	TAU channel 01 count end/capture end	INTTM01
ELSELR18	TAU channel 02 count end/capture end	INTTM02
ELSELR19	TAU channel 03 count end/capture end	INTTM03
ELSELR20 Note 1	TAU channel 10 count end/capture end	INTTM10
ELSELR21 Note 1	TAU channel 11 count end/capture end	INTTM11
ELSELR22 Note 1	TAU channel 12 count end/capture end	INTTM12
ELSELR23 Note 1	TAU channel 13 count end/capture end	INTTM13
ELSELR24 Note 2	Comparator detection 0	INTCMP0
ELSELR25 Note 2	Comparator detection 1	INTCMP1

**Note 1.** 80- and 100-pin products only.

**Note 2.** 96 KB or more code flash memory products only.

**Note 3.** Products with at least 40 pins only.

**Table 20 - 3 Correspondence Between Values Set to ELSELn (n = 00 to 25) Registers and Operation of Link Destination Peripheral Functions at Reception**

Bits ELSELn3 to ELSELn0 in ELSELn Register	Link Destination Number	Link Destination Peripheral Function	Operation When Receiving Event
0001B	1	A/D converter	A/D conversion starts
0010B	2	Timer input of timer array unit 0 channel 0 Note 1	Delay counter, input pulse interval measurement, external event counter
0011B	3	Timer input of timer array unit 0 channel 1 Note 2	Delay counter, input pulse interval measurement, external event counter
0100B	4	Timer RJ0	Count source
0101B	5	Timer RG	TRGIOB input capture
0110B	6	Timer RD0	TRDIOD0 input capture, pulse output forced cutoff
0111B	7	Timer RD1	TRDIOD1 input capture, pulse output forced cutoff
1000B Note 4	8	DA0 Note 3	Real-time output
1001B Note 5	9	DA1 Note 3	Real-time output

- Note 1.** To select the timer input of timer array unit 0 channel 0 as the link destination peripheral function, set the operating clock for channel 0 to fCLK using timer clock select register 0 (TPS0), set the noise filter of the TI00 pin to OFF (TNFEN00 = 0) using noise filter enable register 1 (NFEN1), and then set the timer output used for channel 0 to an event input signal from the ELC using timer input select register 0 (TIS0).
- Note 2.** To select the timer input of timer array unit 0 channel 1 as the link destination peripheral function, set the operating clock for channel 1 to fCLK using timer clock select register 0 (TPS0), set the noise filter of the TI01 pin to OFF (TNFEN01 = 0) using noise filter enable register 1 (NFEN1), and then set the timer output used for channel 1 to an event input signal from the ELC using timer input select register 0 (TIS0).
- Note 3.** When entering the STOP mode while the real-time output event mode for D/A conversion is enabled, disable linking of ELC events before entering STOP mode.
- Note 4.** Products with 96 KB or more code flash memory only.
- Note 5.** Products with 96 KB or more code flash memory and at least 32 pins only.

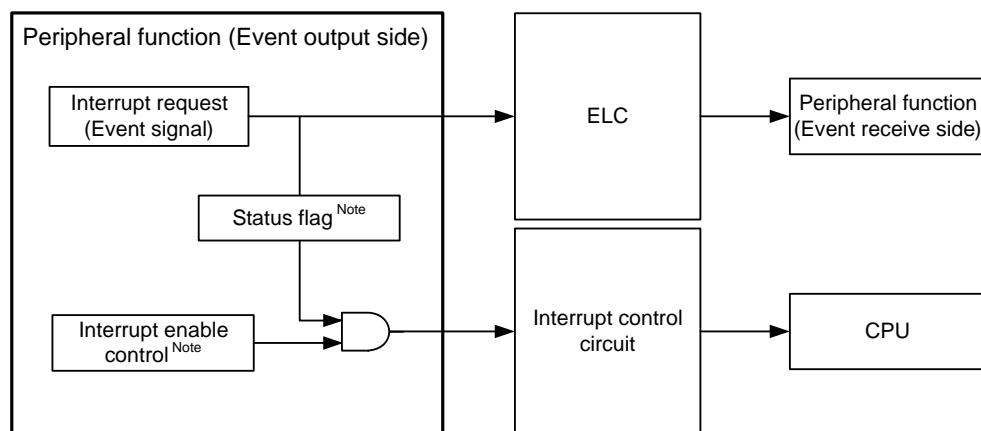
## 20.4 ELC Operation

The path for using an event signal generated by a peripheral function as an interrupt request to the interrupt control circuit is independent from the path for using it as an ELC event. Therefore, each event signal can be used as an event signal for operation of an event-receiving peripheral function, regardless of interrupt control.

Figure 20 - 3 shows the Relationship Between Interrupt Handling and ELC. The figure show an example of an interrupt request status flag and a peripheral function possessing the enable bits that control enabling/disabling of such interrupts.

A peripheral function which receives an event from the ELC will perform the operation corresponding to the event-receiving peripheral function after reception of an event (See **Table 20 - 3 Correspondence Between Values Set to ELSELR<sub>n</sub> (n = 00 to 25) Registers and Operation of Link Destination Peripheral Functions at Reception**).

**Figure 20 - 3 Relationship Between Interrupt Handling and ELC**



**Note** Not available depending on the peripheral function.

Table 20 - 4 lists the Response of Peripheral Functions That Receive Events.

**Table 20 - 4 Response of Peripheral Functions That Receive Events**

Event Receiver No.	Event Link Destination Function	Operation after Event Reception	Response
1	A/D converter	A/D conversion	An event from the ELC is directly used as a hardware trigger of A/D conversion.
2	Timer array unit 0 Timer input of channel 0	Delay counter Input pulse width measurement External event counter	The edge is detected 3 or 4 cycles of fCLK after an ELC event is generated.
3	Timer array unit 0 Timer input of channel 1	Delay counter Input pulse width measurement External event counter	The edge is detected 3 or 4 cycles of fCLK after an ELC event is generated.
4	Timer RJ	Count source	An event from the ELC is directly used as the count source of timer RJ.
5	Timer RG	TRGIOB input capture	A count start trigger is generated 2 or 3 cycles of fCLK after an ELC event is generated.
6	Timer RD0	TRDIOD0 input capture	A count start trigger is generated 2 or 3 cycles of the timer RD operating clock after an ELC event is generated.
		Pulse output forced cutoff	The pulse is forcibly cut off 2 or 3 cycles of the timer RD operating clock after an ELC event is generated.
7	Timer RD1	TRDIOD1 input capture	A count start trigger is generated 2 or 3 cycles of the timer RD operating clock after an ELC event is generated.
		Pulse output forced cutoff	The pulse is forcibly cut off 2 or 3 cycles of the timer RD operating clock after an ELC event is generated.
8 Note 1	Channel 0 of D/A converter	Real-time output (channel 0)	D/A conversion of channel 0 is started after 2 or 3 cycles of fCLK after an ELC event is generated.
9 Note 2	Channel 1 of D/A converter	Real-time output (channel 1)	D/A conversion of channel 1 is started after 2 or 3 cycles of fCLK after an ELC event is generated.

**Note 1.** Products with 96 KB or more code flash memory only.

**Note 2.** Products with 96 KB or more code flash memory and at least 32 pins only.

## CHAPTER 21 INTERRUPT FUNCTIONS

The interrupt function switches the program execution to other processing. When the branch processing is finished, the program returns to the interrupted processing.

The number of interrupt sources differs, depending on the product.

		30, 32, 36-pin	40, 44-pin	48-pin	52-pin	64-pin	80, 100-pin
Maskable interrupts	External	6	7	10	12	13	13
	Internal	24	24	24	24	24	32

### 21.1 Interrupt Function Types

The following two types of interrupt functions are used.

(1) Maskable interrupts

These interrupts undergo mask control. Maskable interrupts can be divided into four priority groups by setting the priority specification flag registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H).

Multiple interrupt servicing can be applied to low-priority interrupts when high-priority interrupts are generated. If two or more interrupt requests, each having the same priority, are simultaneously generated, then they are processed according to the default priority of vectored interrupt servicing. Default priority, see **Tables 21 - 1 to 21 - 4**.

A standby release signal is generated and STOP, HALT, and SNOOZE modes are released.

External interrupt requests and internal interrupt requests are provided as maskable interrupts.

(2) Software interrupt

This is a vectored interrupt generated by executing the BRK instruction. It is acknowledged even when interrupts are disabled. The software interrupt does not undergo interrupt priority control.

### 21.2 Interrupt Sources and Configuration

Interrupt sources include maskable interrupts and software interrupts. In addition, they also have up to seven reset sources (see **Tables 21 - 1 to 21 - 4**). The vector codes that store the program start address when branching due to the generation of a reset or various interrupt requests are two bytes each, so interrupts jump to a 64 K address of 00000H to 0FFFFH.

Table 21 - 1 Interrupt Source List (1/4)

Interrupt Type	Default Priority Note 1	Interrupt Source		Internal/External	Vector Table Address	Basic Configuration Type Note 2										
		Name	Trigger				100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
Maskable	0	INTWDTI	Watchdog timer interval Note 3 (75% of overflow time + 1/2 f <sub>IL</sub> )	Internal	00004H	(A)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	1	INTLVI	Voltage detection Note 4		00006H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	2	INTP0	Pin input edge detection	External	00008H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	3	INTP1			0000AH		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	4	INTP2			0000CH		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	5	INTP3			0000EH		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	6	INTP4			00010H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	7	INTP5			00012H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	8	INTST2/ INTCSI20/ INTIIC20	UART2 transmission transfer end or buffer empty interrupt/CSI20 transfer end or buffer empty interrupt/IIC20 transfer end	Internal	00014H	(A)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	9	INTSR2/ INTCSI21/ INTIIC21	UART2 reception transfer end/CSI21 transfer end or buffer empty interrupt/IIC21 transfer end		00016H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	10	INTSRE2	UART2 reception communication error occurrence		00018H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		INTTM11H	End of timer channel 11 count or capture (at higher 8-bit timer operation)		0001EH		✓	✓	—	—	—	—	—	—	—	—
	11	INTST0/ INTCSI00/ INTIIC00	UART0 transmission transfer end or buffer empty interrupt/CSI00 transfer end or buffer empty interrupt/IIC00 transfer end		00020H		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	12	INTSR0/ INTCSI01/ INTIIC01	UART0 reception transfer end/CSI01 transfer end or buffer empty interrupt/IIC01 transfer end		00022H		✓	✓	✓	✓	✓	✓	Note 6	Note 6	Note 6	Note 6
	13	INTSRE0	UART0 reception communication error occurrence				✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		INTTM01H	End of timer channel 01 count or capture (at higher 8-bit timer operation)				✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

**Note 1.** The default priority determines the sequence of interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 44 indicates the lowest priority.

**Note 2.** Basic configuration types (A) to (D) correspond to (A) to (D) in Figures 21 - 1 and 21 - 2.

**Note 3.** When bit 7 (WDTINT) of the option byte (000C0H) is set to 1.

**Note 4.** When bit 7 (LVIMD) of the voltage detection level register (LVIS) is cleared to 0.

**Note 5.** INTSR2 only.

**Note 6.** INTSR0 only.

Table 21 - 2 Interrupt Source List (2/4)

Maskable	Interrupt Type	Interrupt Source		Internal/External	Vector Table Address	Basic Configuration Type Note 2									
		Name	Trigger			100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin	32-pin	30-pin
	14	INTST1/ INTCSI10/ INTIIC10	UART1 transmission transfer end or buffer empty interrupt/CSI10 transfer end or buffer empty interrupt/IIC10 transfer end	Internal	00024H	(A)	✓	✓	✓	Note 3					
	15	INTSR1/ INTCSI11/ INTIIC11	UART1 reception transfer end/CSI11 transfer end or buffer empty interrupt/IIC11 transfer end		00026H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	16	INTSRE1	UART1 reception communication error occurrence		00028H		✓	✓	✓	✓	✓	✓	✓	✓	✓
		INTTM03H	End of timer channel 03 count or capture (at higher 8-bit timer operation)				✓	✓	✓	✓	✓	✓	✓	✓	✓
	17	INTIICA0	End of IICA0 communication		0002AH		✓	✓	✓	✓	✓	✓	✓	✓	✓
	18	INTTM00	End of timer channel 00 count or capture		0002CH		✓	✓	✓	✓	✓	✓	✓	✓	✓
	19	INTTM01	End of timer channel 01 count or capture		0002EH		✓	✓	✓	✓	✓	✓	✓	✓	✓
	20	INTTM02	End of timer channel 02 count or capture		00030H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	21	INTTM03	End of timer channel 03 count or capture		00032H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	22	INTAD	End of A/D conversion		00034H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	23	INTRTC	Fixed-cycle signal of real-time clock/alarm match detection		00036H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	24	INTIT	Interval signal detection		00038H		✓	✓	✓	✓	✓	✓	✓	✓	✓
	25	INTKR	Key return signal detection		0003AH	(C)	✓	✓	✓	✓	✓	✓	✓	✓	✓
	26	INTST3/ INTCSI30/ INTIIC30	UART3 transmission transfer end or buffer empty interrupt/CSI30 transfer end or buffer empty interrupt/IIC30 transfer end	Internal	0003CH	(B)	✓	✓	—	—	—	—	—	—	—
	27	INTSR3/ INTCSI31/ INTIIC31	UART3 reception transfer end/CSI31 transfer end or buffer empty interrupt/IIC31 transfer end		0003EH		✓	✓	—	—	—	—	—	—	—
	28	INTTRJ0	Timer RJ interrupt		00040H		✓	✓	✓	✓	✓	✓	✓	✓	✓

**Note 1.** The default priority determines the sequence of interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 44 indicates the lowest priority.

**Note 2.** Basic configuration types (A) to (D) correspond to (A) to (D) in Figures 21 - 1 and 21 - 2.

**Note 3.** INTST1 only.

Table 21 - 3 Interrupt Source List (3/4)

Interrupt Type	Default Priority Note 1	Interrupt Source		Internal/External	Vector Table Address	Basic Configuration Type Note 2							
		Name	Trigger				100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin
Maskable	29	INTTM10	End of timer channel 10 count or capture	Internal	00042H	(A)	✓	✓	—	—	—	—	—
	30	INTTM11	End of timer channel 11 count or capture		00044H		✓	✓	—	—	—	—	—
	31	INTTM12	End of timer channel 12 count or capture		00046H		✓	✓	—	—	—	—	—
	32	INTTM13	End of timer channel 13 count or capture		00048H		✓	✓	—	—	—	—	—
	33	INTP6	Pin input edge detection	External	0004AH	(B)	✓	✓	✓	✓	✓	—	—
	34	INTP7			0004CH		✓	✓	✓	—	—	—	—
	35	INTP8			0004EH		✓	✓	✓	✓	✓	—	—
	36	INTP9			00050H		✓	✓	✓	✓	✓	—	—
	37	INTP10	Pin input edge detection	Internal	00052H	(B)	✓	✓	✓	✓	—	—	—
		INTCMP0	Comparator detection 0				✓	✓	✓	✓	✓	—	—
	38	INTP11	Pin input edge detection	Internal	00054H	(A)	✓	✓	✓	✓	—	—	—
		INTCMP1	Comparator detection 1				✓	✓	✓	✓	✓	—	—
Non-Maskable	39	INTTRD0	Timer RD0 input capture, compare match, overflow, underflow interrupt	Internal	00056H	(A)	✓	✓	✓	✓	✓	✓	✓
	40	INTTRD1	Timer RD1 input capture, compare match, overflow, underflow interrupt				✓	✓	✓	✓	✓	✓	✓
	41	INTTRG	Timer RG input capture, compare match, overflow, underflow interrupt				✓	✓	✓	✓	✓	✓	✓
	42	INTSRE3	UART3 reception communication error occurrence				✓	✓	—	—	—	—	—
		INTTM13H	End of timer channel 13 count or capture (at 8-bit timer operation)	Internal	0005CH	(A)	✓	✓	—	—	—	—	—
	43	INTIICA1	End of IICA1 communication				✓	✓	—	—	—	—	—
	44	INTFL	Reserved Note 4	Internal	00060H	(A)	✓	✓	✓	✓	✓	✓	✓
							✓	✓	✓	✓	✓	✓	✓

**Note 1.** The default priority determines the sequence of interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 44 indicates the lowest priority.

**Note 2.** Basic configuration types (A) to (D) correspond to (A) to (D) in Figures 21 - 1 and 21 - 2.

**Note 3.** Mounted on the 96 KB or more code flash memory products.

**Note 4.** Be used at the flash self-programming library or the data flash library.

Table 21 - 4 Interrupt Source List (4/4)

Reset	Software	Interrupt Source		Internal/External	Vector Table Address	Basic Configuration Type Note 2								
		Name	Trigger				100-pin	80-pin	64-pin	52-pin	48-pin	44-pin	40-pin	36-pin
	—	BRK	Execution of BRK instruction	—	0007EH	(D)	✓	✓	✓	✓	✓	✓	✓	✓
	—	RESET	RESET pin input	—	00000H	—	✓	✓	✓	✓	✓	✓	✓	✓
		POR	Power-on-reset				✓	✓	✓	✓	✓	✓	✓	✓
		LVD	Voltage detection Note 3				✓	✓	✓	✓	✓	✓	✓	✓
		WDT	Overflow of watchdog timer				✓	✓	✓	✓	✓	✓	✓	✓
		TRAP	Execution of illegal instruction Note 4				✓	✓	✓	✓	✓	✓	✓	✓
		IAW	Illegal-memory access				✓	✓	✓	✓	✓	✓	✓	✓
		RPE	RAM parity error				✓	✓	✓	✓	✓	✓	✓	✓

**Note 1.** The default priority determines the sequence of interrupts if two or more maskable interrupts occur simultaneously. Zero indicates the highest priority and 44 indicates the lowest priority.

**Note 2.** Basic configuration types (A) to (D) correspond to (A) to (D) in Figures 21 - 1 and 21 - 2.

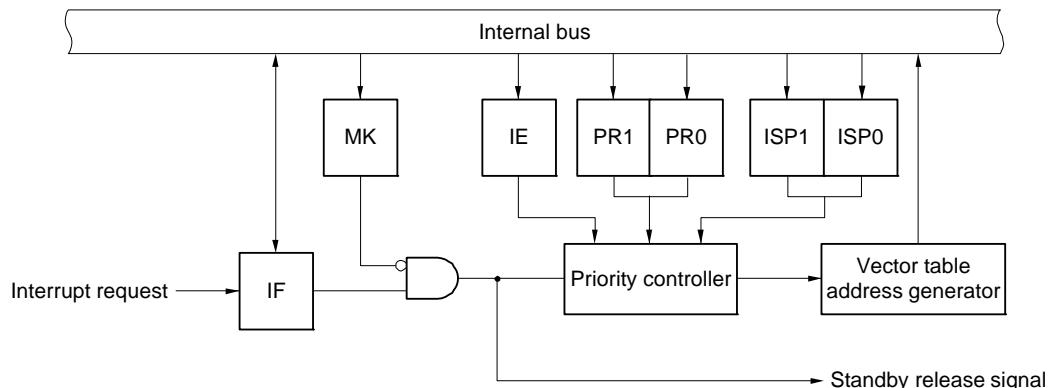
**Note 3.** When bit 7 (LVIMD) of the voltage detection level register (LVIS) is set to 1.

**Note 4.** When the instruction code in FFH is executed.

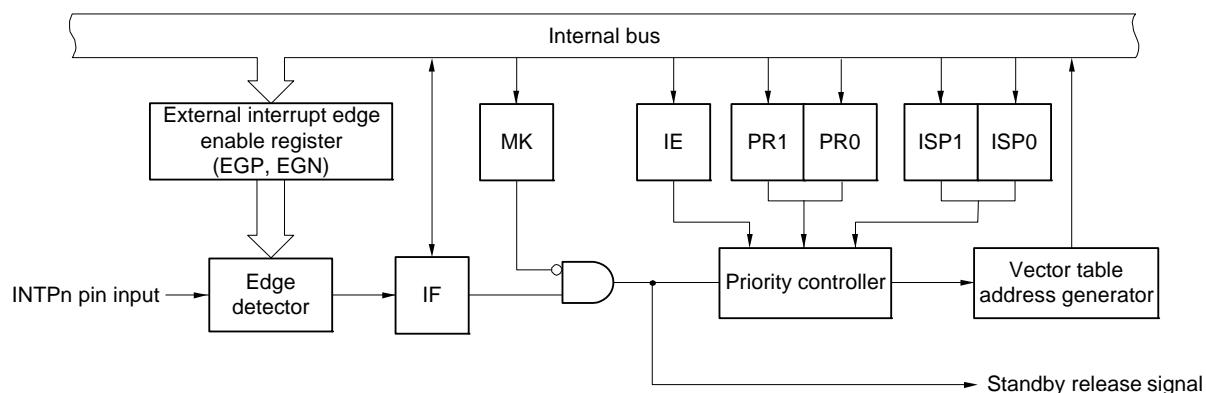
Reset by the illegal instruction execution not issued by emulation with the in-circuit emulator or on-chip debug emulator.

Figure 21 - 1 Basic Configuration of Interrupt Function (1/2)

(A) Internal maskable interrupt



(B) External maskable interrupt (INTPn)

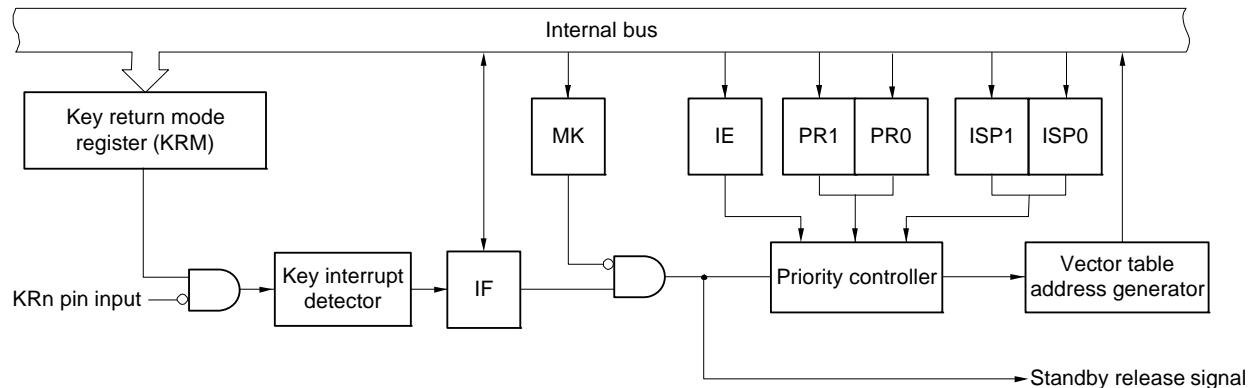


IF:	Interrupt request flag
IE:	Interrupt enable flag
ISP0:	In-service priority flag 0
ISP1:	In-service priority flag 1
MK:	Interrupt mask flag
PR0:	Priority specification flag 0
PR1:	Priority specification flag 1

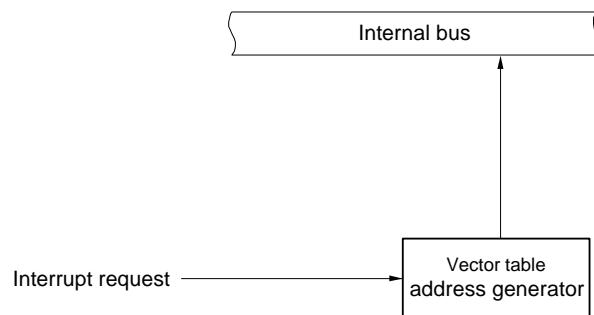
<b>Remark</b>	30, 32, 36, 40, 44-pin: n = 0 to 5
	48-pin: n = 0 to 6, 8, 9
	52-pin: n = 0 to 6, 8 to 11
	64, 80, 100-pin: n = 0 to 11

Figure 21 - 2 Basic Configuration of Interrupt Function (2/2)

(C) External maskable interrupt (INTKR)



(D) Software interrupt



IF:	Interrupt request flag
IE:	Interrupt enable flag
ISPO:	In-service priority flag 0
ISP1:	In-service priority flag 1
MK:	Interrupt mask flag
PR0:	Priority specification flag 0
PR1:	Priority specification flag 1

<b>Remark</b>	40, 44-pin:	$n = 0 \text{ to } 3$
	48-pin:	$n = 0 \text{ to } 5$
	52, 64, 80, 100-pin:	$n = 0 \text{ to } 7$

## 21.3 Registers Controlling Interrupt Functions

The following 6 types of registers are used to control the interrupt functions.

- Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H)
- Interrupt mask flag registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H)
- Priority specification flag registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H)
- External interrupt rising edge enable registers (EGP0, EGP1)
- External interrupt falling edge enable registers (EGN0, EGN1)
- Program status word (PSW)

Tables 21 - 5 to 21 - 8 show a list of interrupt request flags, interrupt mask flags, and priority specification flags corresponding to interrupt request sources.

**Table 21 - 5 Flags Corresponding to Interrupt Request Sources (1/4)**

Interrupt Source	Interrupt Request Flag	Interrupt Mask Flag		Priority Specification Flag	Register	100-pin	80-pin	64-pin	52-pin	48-pin	40, 44-pin	36-pin	30, 32-pin
		Register	Register										
INTWDTI	WDTIIF	IF0L	WDTIMK	MK0L	WDTIPR0, WDTIPR1	PR00L, PR10L	✓	✓	✓	✓	✓	✓	✓
INTLVI	LVIIF		LVIMK		LVIPR0, LVIPR1		✓	✓	✓	✓	✓	✓	✓
INTP0	PIF0		PMK0		PPR00, PPR10		✓	✓	✓	✓	✓	✓	✓
INTP1	PIF1		PMK1		PPR01, PPR11		✓	✓	✓	✓	✓	✓	✓
INTP2	PIF2		PMK2		PPR02, PPR12		✓	✓	✓	✓	✓	✓	✓
INTP3	PIF3		PMK3		PPR03, PPR13		✓	✓	✓	✓	✓	✓	✓
INTP4	PIF4		PMK4		PPR04, PPR14		✓	✓	✓	✓	✓	✓	✓
INTP5	PIF5		PMK5		PPR05, PPR15		✓	✓	✓	✓	✓	✓	✓

**Table 21 - 6 Flags Corresponding to Interrupt Request Sources (2/4)**

Interrupt Source	Interrupt Request Flag	Interrupt Mask Flag		Priority Specification Flag	Register	'100-pin	80-pin	64-pin	52-pin	48-pin	40, 44-pin	36-pin	30, 32-pin
		Register	Register										
INTST2 Note 1	STIF2 Note 1	IF0H	STMK2 Note 1	MK0H	STPR02, STPR12 Note 1	PR00H, PR10H	✓	✓	✓	✓	✓	✓	✓
INTCSI20 Note 1	CSIIFF20 Note 1		CSIMK20 Note 1		CSIPR020, CSIPR120 Note 1		✓	✓	✓	✓	✓	✓	✓
INTIIC20 Note 1	IICIF20 Note 1		IICMK20 Note 1		IICPR020, IICPR120 Note 1		✓	✓	✓	✓	✓	✓	✓
INTSR2 Note 2	SRIF2 Note 2		SRMK2 Note 2		SRPR02, SRPR12 Note 2		✓	✓	✓	✓	✓	✓	✓
INTCSI21 Note 2	CSIIFF21 Note 2		CSIMK21 Note 2		CSIPR021, CSIPR121 Note 2		✓	✓	✓	✓	✓	✓	—
INTIIC21 Note 2	IICIF21 Note 2		IICMK21 Note 2		IICPR021, IICPR121 Note 2		✓	✓	✓	✓	✓	✓	—
INTSRE2 Note 3	SREIF2 Note 3		SREMK2 Note 3		SREPR02, SREPR12 Note 3		✓	✓	✓	✓	✓	✓	✓
INTTM11H Note 3	TMIF11H Note 3		TMMK11H Note 3		TMPPR011H, TMPPR11H Note 3		✓	✓	—	—	—	—	—
INTST0 Note 4	STIF0 Note 4		STMK0 Note 4		STPR00, STPR10 Note 4		✓	✓	✓	✓	✓	✓	✓
INTCSI0 Note 4	CSIIFF0 Note 4		CSIMK00 Note 4		CSIPR000, CSIPR100 Note 4		✓	✓	✓	✓	✓	✓	✓
INTIIC00 Note 4	IICIF00 Note 4		IICMK00 Note 4		IICPR000, IICPR100 Note 4		✓	✓	✓	✓	✓	✓	✓
INTSR0 Note 5	SRIF0 Note 5		SRMK0 Note 5		SRPR00, SRPR10 Note 5		✓	✓	✓	✓	✓	✓	✓
INTCSI01 Note 5	CSIIFF01 Note 5		CSIMK01 Note 5		CSIPR001, CSIPR101 Note 5		✓	✓	✓	✓	✓	—	—
INTIIC01 Note 5	IICIF01 Note 5		IICMK01 Note 5		IICPR001, IICPR101 Note 5		✓	✓	✓	✓	✓	—	—
INTSRE0 Note 6	SREIF0 Note 6		SREMK0 Note 6		SREPR00, SREPR10 Note 6		✓	✓	✓	✓	✓	✓	✓
INTTM01H Note 6	TMIF01H Note 6		TMMK01H Note 6		TMPPR001H, TMPPR101H Note 6		✓	✓	✓	✓	✓	✓	✓

**Note 1.** If one of the interrupt sources INTST2, INTCSI20, and INTIIC20 is generated, bit 0 of the IF0H register is set to 1. Bit 0 of the MK0H, PR00H, and PR10H registers supports these three interrupt sources.

**Note 2.** If one of the interrupt sources INTSR2, INTCSI21, and INTIIC21 is generated, bit 1 of the IF0H register is set to 1. Bit 1 of the MK0H, PR00H, and PR10H registers supports these three interrupt sources.

**Note 3.** Do not use a UART2 reception error interrupt and an interrupt of channel 1 of TAU1 (at higher 8-bit timer operation) at the same time because they share flags for the interrupt request sources. When the UART2 reception error interrupt is not used (EOC01 = 0), UART2 and channel 1 of TAU1 (at higher 8-bit timer operation) can be used at the same time. If either of the interrupt sources INTSRE2 or INTTM11H is generated, bit 2 of the IF0H register is set to 1. Bit 2 of the MK0H, PR00H, and PR10H registers support these two interrupt sources.

**Note 4.** If one of the interrupt sources INTST0, INTCSI00, and INTIIC00 is generated, bit 5 of the IF0H register is set to 1. Bit 5 of the MK0H, PR00H, and PR10H registers supports these three interrupt sources.

**Note 5.** If one of the interrupt sources INTSR0, INTCSI01, and INTIIC01 is generated, bit 6 of the IF0H register is set to 1. Bit 6 of the MK0H, PR00H, and PR10H registers supports these three interrupt sources.

**Note 6.** Do not use a UART0 reception error interrupt and an interrupt of channel 1 of TAU0 (at higher 8-bit timer operation) at the same time because they share flags for the interrupt request sources. When the UART0 reception error interrupt is not used (EOC01 = 0), UART0 and channel 1 of TAU0 (at higher 8-bit timer operation) can be used at the same time. If either of the interrupt sources INTSRE0 or INTTM01H is generated, bit 7 of the IF0H register is set to 1. Bit 7 of the MK0H, PR00H, and PR10H registers support these two interrupt sources.

Table 21 - 7 Flags Corresponding to Interrupt Request Sources (3/4)

Interrupt Source	Interrupt Request Flag	Interrupt Mask Flag		Priority Specification Flag	Register	Pin							
		Register	Register			100-pin	80-pin	64-pin	52-pin	48-pin	40, 44-pin	36-pin	30, 32-pin
INTST1 Note 1	STIF1 Note 1	IF1L	STMK1 Note 1	MK1L	STPR01, STPR11 Note 1	PR01L, PR11L	✓	✓	✓	✓	✓	✓	✓
INTCSI10 Note 1	CSIIF10 Note 1		CSIMK10 Note 1		CSIPR010, CSIPR110 Note 1		✓	✓	✓	—	—	—	—
INTIIC10 Note 1	IICIF10 Note 1		IICMK10 Note 1		IICPR010, IICPR110 Note 1		✓	✓	✓	—	—	—	—
INTSR1 Note 2	SRIF1 Note 2		SRMK1 Note 2		SRPR01, SRPR11 Note 2		✓	✓	✓	✓	✓	✓	✓
INTCSI11 Note 2	CSIIF11 Note 2		CSIMK11 Note 2		CSIPR011, CSIPR111 Note 2		✓	✓	✓	✓	✓	✓	✓
INTIIC11 Note 2	IICIF11 Note 2		IICMK11 Note 2		IICPR011, IICPR111 Note 2		✓	✓	✓	✓	✓	✓	✓
INTSRE1 Note 3	SREIF1 Note 3		SREMK1 Note 3		SREPR01, SREPR11 Note 3		✓	✓	✓	✓	✓	✓	✓
INTTM03H Note 3	TMIF03H Note 3		TMMK03H Note 3		TMPPR03H, TMPPR103H Note 3		✓	✓	✓	✓	✓	✓	✓
INTIICA0	IICAIF0		IICAMK0		IICAPR00, IICAPR10		✓	✓	✓	✓	✓	✓	✓
INTTM00	TMIF00		TMMK00		TMPPR000, TMPPR100		✓	✓	✓	✓	✓	✓	✓
INTTM01	TMIF01		TMMK01		TMPPR001, TMPPR101		✓	✓	✓	✓	✓	✓	✓
INTTM02	TMIF02		TMMK02		TMPPR002, TMPPR102		✓	✓	✓	✓	✓	✓	✓
INTTM03	TMIF03		TMMK03		TMPPR003, TMPPR103		✓	✓	✓	✓	✓	✓	✓
INTAD	ADIF	IF1H	ADMK	MK1H	ADPRO, ADPR1	PR01H, PR11H	✓	✓	✓	✓	✓	✓	✓
INTRTC	RTCIF		RTCMK		RTCP0, RTCP1		✓	✓	✓	✓	✓	✓	✓
INTIT	ITIF		ITMK		ITPR0, ITPR1		✓	✓	✓	✓	✓	✓	✓
INTKR	KRIF		KRMK		KRPR0, KRPR1		✓	✓	✓	✓	✓	✓	—
INTST3 Note 4	STIF3 Note 4		STMK3 Note 4		STPR03, STPR13 Note 4		✓	✓	—	—	—	—	—
INTCSI30 Note 4	CSIIF30 Note 4		CSIMK30 Note 4		CSIPR030, CSIPR130 Note 4		✓	✓	—	—	—	—	—
INTIIC30 Note 4	IICIF30 Note 4		IICMK30 Note 4		IICPR030, IICPR130 Note 4		✓	✓	—	—	—	—	—
INTSR3 Note 5	SRIF3 Note 5		SRMK3 Note 5		SRPR03, SRPR13 Note 5		✓	✓	—	—	—	—	—
INTCSI31 Note 5	CSIIF31 Note 5		CSIMK31 Note 5		CSIPR031, CSIPR131 Note 5		✓	✓	—	—	—	—	—
INTIIC31 Note 5	IICIF31 Note 5		IICMK31 Note 5		IICPR031, IICPR131 Note 5		✓	✓	—	—	—	—	—
INTTRJ0	TRJIF0		TRJMK0		TRJPR00, TRJPR10		✓	✓	✓	✓	✓	✓	✓
INTTM10	TMIF10		TMMK10		TMPPR010, TMPPR110		✓	✓	—	—	—	—	—

- Note 1.** If one of the interrupt sources INTST1, INTCSI10, and INTIIC10 is generated, bit 0 of the IF1L register is set to 1. Bit 0 of the MK1L, PR01L, and PR11L registers supports these three interrupt sources.
- Note 2.** If one of the interrupt sources INTSR1, INTCSI11, and INTIIC11 is generated, bit 1 of the IF1L register is set to 1. Bit 1 of the MK1L, PR01L, and PR11L registers supports these three interrupt sources.
- Note 3.** Do not use a UART1 reception error interrupt and an interrupt of channel 3 of TAU0 (at higher 8-bit timer operation) at the same time because they share flags for the interrupt request sources. When the UART1 reception error interrupt is not used (EOC03 = 0), UART1 and channel 3 of TAU0 (at higher 8-bit timer operation) can be used at the same time. If either of the interrupt sources INTSRE1 or INTTM03H is generated, bit 2 of the IF1H register is set to 1. Bit 2 of the MK1L, PR01L, and PR11L registers support these two interrupt sources.
- Note 4.** If one of the interrupt sources INTST3, INTCSI30, and INTIIC30 is generated, bit 4 of the IF1H register is set to 1. Bit 4 of the MK1H, PR01H, and PR11H registers supports these three interrupt sources.
- Note 5.** If one of the interrupt sources INTSR3, INTCSI31, and INTIIC31 is generated, bit 5 of the IF1H register is set to 1. Bit 5 of the MK1H, PR01H, and PR11H registers supports these three interrupt sources.

**Table 21 - 8 Flags Corresponding to Interrupt Request Sources (4/4)**

Interrupt Source	Interrupt Request Flag	Interrupt Mask Flag		Priority Specification Flag	Register										
		Register	Register												
							100-pin	80-pin	64-pin	52-pin	48-pin	40, 44-pin	36-pin	30, 32-pin	
INTTM11	TMIF11	IF2L	TMMK11	MK2L	TMPR011, TMPR111 TMPR012, TMPR112 TMPR013, TMPR113 PPR06, PPR16 PPR07, PPR17 PPR08, PPR18 PPR09, PPR19 PPR010, PPR110 Note 1 CMPPR00, CMPPR10 Note 1	PR02L, PR12L	✓	✓	—	—	—	—	—	—	
INTTM12	TMIF12		TMMK12				✓	✓	—	—	—	—	—	—	
INTTM13	TMIF13		TMMK13				✓	✓	—	—	—	—	—	—	
INTP6	PIF6		PMK6				✓	✓	✓	✓	✓	—	—	—	
INTP7	PIF7		PMK7				✓	✓	✓	—	—	—	—	—	
INTP8	PIF8		PMK8				✓	✓	✓	✓	✓	—	—	—	
INTP9	PIF9		PMK9				✓	✓	✓	✓	✓	—	—	—	
INTP10 Note 1	PIF10 Note 1		PMK10 Note 1				✓	✓	✓	✓	—	—	—	—	
INTCMP0 Note 1	CMPIF0 Note 1		CMPMK0 Note 1				✓	✓	—	—	—	—	—	—	
INTP11 Note 2	PIF11 Note 2	IF2H	PMK11 Note 2	MK2H	PPR011, PPR111 Note 2 CMPPR01, CMPPR11 Note 2 TRDPR00, TRDPR10 TRDPR01, TRDPR11 TRGPR0, TRGPR1 SREPR03, SREPR13 Note 3 TMPR013H, TMPR113H Note 3 IICAPR01, IICAPR11 FLPR0, FLPR1	PR02H, PR12H	✓	✓	✓	✓	—	—	—	—	
INTCMP1 Note 2	CMPIF1 Note 2		CMPMK1 Note 2				✓	✓	Note 4	Note 4	Note 4	Note 4	Note 4	Note 4	
INTTRD0	TRDIF0		TRDMK0				✓	✓	✓	✓	✓	✓	✓	✓	
INTTRD1	TRDIF1		TRDMK1				✓	✓	✓	✓	✓	✓	✓	✓	
INTTRG	TRGIF		TRGMK				✓	✓	✓	✓	✓	✓	✓	✓	
INTSRE3 Note 3	SREIF3 Note 3		SREMK3 Note 3				✓	✓	—	—	—	—	—	—	
INTTM13H Note 3	TMIF13H Note 3		TMMK13H Note 3				✓	✓	—	—	—	—	—	—	
INTIICA1	IICAIF1		IICAMK1				✓	✓	—	—	—	—	—	—	
INTFL	FLIF		FLMK				✓	✓	✓	✓	✓	✓	✓	✓	

**Note 1.** Do not use INTP10 and Comparator 0 at the same time because they share flags for the interrupt request sources. If one of the interrupt sources INTP10 and INT CMP0 is generated, bit 7 of the IF2L register is set to 1. Bit 7 of the MK2L, PR02L, and PR12L registers supports these two interrupt sources.

**Note 2.** Do not use INTP11 and Comparator 1 at the same time because they share flags for the interrupt request sources. If one of the interrupt sources INTP11 and INT CMP1 is generated, bit 0 of the IF2H register is set to 1. Bit 0 of the MK2H, PR02H, and PR12H registers supports these two interrupt sources.

**Note 3.** Do not use a UART3 reception error interrupt and an interrupt of channel 3 of TAU1 (at higher 8-bit timer operation) at the same time because they share flags for the interrupt request sources. When the UART3 reception error interrupt is not used (EOC03 = 0), UART3 and channel 3 of TAU1 (at higher 8-bit timer operation) can be used at the same time. If either of the interrupt sources INTSRE3 or INTTM13H is generated, bit 4 of the IF2H register is set to 1. Bit 4 of the MK2H, PR02H, and PR12H registers support these two interrupt sources.

**Note 4.** Mounted on the 96 KB or more code flash memory products.

### 21.3.1 Interrupt request flag registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H)

The interrupt request flags are set to 1 when the corresponding interrupt request is generated or an instruction is executed. They are cleared to 0 when an instruction is executed upon acknowledgment of an interrupt request or upon reset signal generation.

When an interrupt is acknowledged, the interrupt request flag is automatically cleared and then the interrupt routine is entered.

The IF0L, IF0H, IF1L, IF1H, IF2L, and IF2H registers can be set by a 1-bit or 8-bit memory manipulation instruction. When the IF0L and IF0H registers, the IF1L and IF1H registers, and the IF2L and IF2H registers are combined to form 16-bit registers IF0, IF1, and IF2, they can be set by a 16-bit memory manipulation instruction. Reset signal generation clears these registers to 00H.

**Remark** If an instruction that writes data to this register is executed, the number of instruction execution clocks increases by 2 clocks.

**Figure 21 - 3 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H) (1/2)**

Address: FFFE0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF0L	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0	LVIIF	WDTIIF

Address: FFFE1H After reset: 00H R/W

Symbol	<7>	<6>	<5>	4	3	<2>	<1>	<0>
IF0H	SREIF0 TMIF01H	SRIF0 CSIIF01 IICIF01	STIF0 CSIIF00 IICIF00	0	0	SREIF2 TMIF11H	SRIF2 CSIIF21 IICIF21	STIF2 CSIIF20 IICIF20

Address: FFFE2H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1L	TMIF03	TMIF02	TMIF01	TMIF00	IICAIF0	SREIF1 TMIF03H	SRIF1 CSIIF11 IICIF11	STIF1 CSIIF10 IICIF10

Address: FFFE3H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF1H	TMIF10	TRJIF0	SRIF3 CSIIF31 IICIF31	STIF3 CSIIF30 IICIF30	KRIF	ITIF	RTCIF	ADIF

Address: FFFD0H After reset: 00H R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
IF2L	PIF10 CMPIF0	PIF9	PIF8	PIF7	PIF6	TMIF13	TMIF12	TMIF11

**Figure 21 - 4 Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H) (2/2)**

Symbol	<7>	<6>	5	<4>	<3>	<2>	<1>	<0>
IF2H	FLIF	IICAI1	0	SREIF3 TMIF13H	TRGIF	TRDIF1	TRDIF0	PIF11 CMPIF1
XXIFX	Interrupt request flag							
0	No interrupt request signal is generated							
1	Interrupt request is generated, interrupt request status							

**Caution 1.** The available registers and bits differ depending on the product. For details about the registers and bits available for each product, see Tables 21 - 5 to 21 - 8. Be sure to set bits that are not available to the initial value.

**Caution 2.** When manipulating a flag of the interrupt request flag register, use a 1-bit memory manipulation instruction (CLR1). When describing in C language, use a bit manipulation instruction such as “IF0L.0 = 0;” or “\_asm (“clr1 IF0L, 0”);” because the compiled assembler must be a 1-bit memory manipulation instruction (CLR1).

If a program is described in C language using an 8-bit memory manipulation instruction such as “IF0L &= 0xfe;” and compiled, it becomes the assembler of three instructions.

```
mov a, IF0L
and a, #0FEH
mov IF0L, a
```

In this case, even if the request flag of the another bit of the same interrupt request flag register (IF0L) is set to 1 at the timing between “mov a, IF0L” and “mov IF0L, a”, the flag is cleared to 0 at “mov IF0L, a”. Therefore, care must be exercised when using an 8-bit memory manipulation instruction in C language.

### 21.3.2 Interrupt mask flag registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H)

The interrupt mask flags are used to enable/disable the corresponding maskable interrupt.

The MK0L, MK0H, MK1L, MK1H, MK2L, and MK2H registers can be set by a 1-bit or 8-bit memory manipulation instruction. When the MK0L and MK0H registers, the MK1L and MK1H registers, and the MK2L and MK2H registers are combined to form 16-bit registers MK0, MK1, and MK2, they can be set by a 16-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Remark** If an instruction that writes data to this register is executed, the number of instruction execution clocks increases by 2 clocks.

**Figure 21 - 5 Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H) (1/2)**

Address: FFFE4H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK0L	PMK5	PMK4	PMK3	PMK2	PMK1	PMK0	LVIMK	WDTIMK

Address: FFFE5H After reset: FFH R/W

Symbol	<7>	<6>	<5>	4	3	<2>	<1>	<0>
MK0H	SREM <b>K</b> 0 TMMK01H	SRM <b>K</b> 0 CSIMK01 IICMK01	STM <b>K</b> 0 CSIMK00 IICMK00	1	1	SREM <b>K</b> 2 TMMK11H	SRM <b>K</b> 2 CSIMK21 IICMK21	STM <b>K</b> 2 CSIMK20 IICMK20

Address: FFFE6H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK1L	TMMK03	TMMK02	TMMK01	TMMK00	IICAMK0	SREM <b>K</b> 1 TMMK03H	SRM <b>K</b> 1 CSIMK11 IICMK11	STM <b>K</b> 1 CSIMK10 IICMK10

Address: FFFE7H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK1H	TMMK10	TRJMK0	SRM <b>K</b> 3 CSIMK31 IICMK31	STM <b>K</b> 3 CSIMK30 IICMK30	KRMK	ITMK	RTCMK	ADMK

Address: FFFD4H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
MK2L	PMK10 CMPMK0	PMK9	PMK8	PMK7	PMK6	TMMK13	TMMK12	TMMK11

**Figure 21 - 6 Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H) (2/2)**

Address: FFFD5H      After reset: FFH      R/W

Symbol      &lt;7&gt;      &lt;6&gt;      5      &lt;4&gt;      &lt;3&gt;      &lt;2&gt;      &lt;1&gt;      &lt;0&gt;

MK2H	FLMK	IICAMK1	1	SREMK3 TMMK13H	TRGMK	TRDMK1	TRDMK0	PMK11 CMPMK1
XXMKX	Interrupt servicing control							
0	Interrupt servicing enabled							
1	Interrupt servicing disabled							

**Caution** The available registers and bits differ depending on the product. For details about the registers and bits available for each product, see Tables 21 - 5 to 21 - 8. Be sure to set bits that are not available to the initial value.

### 21.3.3 Priority specification flag registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H)

The priority specification flag registers are used to set the corresponding maskable interrupt priority level.

A priority level is set by using the PR0xy and PR1xy registers in combination (xy = 0L, 0H, 1L, 1H, 2L, or 2H).

The PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, and the PR12H registers can be set by a 1-bit or 8-bit memory manipulation instruction. If the PR00L and PR00H registers, the PR01L and PR01H registers, the PR02L and PR02H registers, the PR10L and PR10H registers, the PR11L and PR11H registers, and the PR12L and PR12H registers are combined to form 16-bit registers PR00, PR01, PR02, PR10, PR11, and PR12, they can be set by a 16-bit memory manipulation instruction.

Reset signal generation sets these registers to FFH.

**Remark** If an instruction that writes data to this register is executed, the number of instruction execution clocks increases by 2 clocks.

**Figure 21 - 7 Format of Priority Specification Flag Registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H) (1/3)**

Address: FFFE8H After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR00L	PPR05	PPR04	PPR03	PPR02	PPR01	PPR00	LVIPR0	WDTIPR0

Address: FFFECH After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR10L	PPR15	PPR14	PPR13	PPR12	PPR11	PPR10	LVIPR1	WDTIPR1

Address: FFFE9H After reset: FFH R/W

Symbol	<7>	<6>	<5>	4	3	<2>	<1>	<0>
PR00H	SREPR00 TMPCR001H	SRPR00 CSIPR001 IICPR001	STPR00 CSIPR000 IICPR000	1	1	SREPR02 TMPCR011H	SRPR02 CSIPR021 IICPR021	STPR02 CSIPR020 IICPR020

Address: FFFEDH After reset: FFH R/W

Symbol	<7>	<6>	<5>	4	3	<2>	<1>	<0>
PR10H	SREPR10 TMPCR101H	SRPR10 CSIPR101 IICPR101	STPR10 CSIPR100 IICPR100	1	1	SREPR12 TMPCR111H	SRPR12 CSIPR121 IICPR121	STPR12 CSIPR120 IICPR120

Address: FFFEAH After reset: FFH R/W

Symbol	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
PR01L	TMPCR003	TMPCR002	TMPCR001	TMPCR000	IICPR000	SREPR01 TMPCR003H	SRPR01 CSIPR011 IICPR011	STPR01 CSIPR010 IICPR010

**Figure 21 - 8 Format of Priority Specification Flag Registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H) (2/3)**

Address: FFFEEH After reset: FFH R/W

Symbol <7> <6> <5> <4> <3> <2> <1> <0>

PR11L	TMPR103	TMPR102	TMPR101	TMPR100	IICAPR10	SREPR11 TMPR103H	SRPR11 CSIPR111 IICPR111	STPR11 CSIPR110 IICPR110
-------	---------	---------	---------	---------	----------	---------------------	--------------------------------	--------------------------------

Address: FFFEBH After reset: FFH R/W

Symbol <7> <6> <5> <4> <3> <2> <1> <0>

PR01H	TMPR010	TRJPR00	SRPR03 CSIPR031 IICPR031	STPR03 CSIPR030 IICPR030	KRPR0	ITPR0	RTCP0	ADPR0
-------	---------	---------	--------------------------------	--------------------------------	-------	-------	-------	-------

Address: FFFEFH After reset: FFH R/W

Symbol <7> <6> <5> <4> <3> <2> <1> <0>

PR11H	TMPR110	TRJPR10	SRPR13 CSIPR131 IICPR131	STPR13 CSIPR130 IICPR130	KRPR1	ITPR1	RTCP1	ADPR1
-------	---------	---------	--------------------------------	--------------------------------	-------	-------	-------	-------

Address: FFFD8H After reset: FFH R/W

Symbol <7> <6> <5> <4> <3> <2> <1> <0>

PR02L	PPR010 CMPPR00	PPR09	PPR08	PPR07	PPR06	TMPR013	TMPR012	TMPR011
-------	-------------------	-------	-------	-------	-------	---------	---------	---------

Address: FFFDCH After reset: FFH R/W

Symbol <7> <6> <5> <4> <3> <2> <1> <0>

PR12L	PPR110 CMPPR10	PPR19	PPR18	PPR17	PPR16	TMPR113	TMPR112	TMPR111
-------	-------------------	-------	-------	-------	-------	---------	---------	---------

Address: FFFD9H After reset: FFH R/W

Symbol <7> <6> 5 <4> <3> <2> <1> <0>

PR02H	FLPR0	IICAPR01	1	SREPR03 TMPR013H	TRGPR0	TRDPR01	TRDPR00	PPR011 CMPPR01
-------	-------	----------	---	---------------------	--------	---------	---------	-------------------

Address: FFFDDH After reset: FFH R/W

Symbol <7> <6> 5 <4> <3> <2> <1> <0>

PR12H	FLPR1	IICAPR11	1	SREPR13 TMPR113H	TRGPR1	TRDPR11	TRDPR10	PPR111 CMPPR11
-------	-------	----------	---	---------------------	--------	---------	---------	-------------------

**Figure 21 - 9 Format of Priority Specification Flag Registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H) (3/3)**

XXPR1X	XXPR0X	Priority level selection
0	0	Specify level 0 (high priority level)
0	1	Specify level 1
1	0	Specify level 2
1	1	Specify level 3 (low priority level)

**Caution** The available registers and bits differ depending on the product. For details about the registers and bits available for each product, see Tables 21 - 5 to 21 - 8. Be sure to set bits that are not available to the initial value.

### 21.3.4 External interrupt rising edge enable registers (EGP0, EGP1), external interrupt falling edge enable registers (EGN0, EGN1)

These registers specify the valid edge for INTP0 to INTP11.

The EGP0, EGP1, EGN0, and EGN1 registers can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 21 - 10 Format of External Interrupt Rising Edge Enable Registers (EGP0, EGP1) and External Interrupt Falling Edge Enable Registers (EGN0, EGN1)**

Address: FFF38H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGP0	EGP7	EGP6	EGP5	EGP4	EGP3	EGP2	EGP1	EGP0

Address: FFF39H After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGN0	EGN7	EGN6	EGN5	EGN4	EGN3	EGN2	EGN1	EGN0

Address: FFF3AH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGP1	0	0	0	0	EGP11	EGP10	EGP9	EGP8

Address: FFF3BH After reset: 00H R/W

Symbol	7	6	5	4	3	2	1	0
EGN1	0	0	0	0	EGN11	EGN10	EGN9	EGN8

EGPn	EGNn	INTPn pin valid edge selection (n = 0 to 11)
0	0	Edge detection disabled
0	1	Falling edge
1	0	Rising edge
1	1	Both rising and falling edges

Table 21 - 9 shows the Ports Corresponding to EGPN and EGNn bits.

**Table 21 - 9 Ports Corresponding to EGPn and EGNn bits**

Detection Enable Bit		Interrupt Request Signal	64, 80, 100-pin	52-pin	48-pin	30, 32, 36, 40, 44-pin
EGP0	EGN0	INTP0	√	√	√	√
EGP1	EGN1	INTP1	√	√	√	√
EGP2	EGN2	INTP2	√	√	√	√
EGP3	EGN3	INTP3	√	√	√	√
EGP4	EGN4	INTP4	√	√	√	√
EGP5	EGN5	INTP5	√	√	√	√
EGP6	EGN6	INTP6	√	√	√	—
EGP7	EGN7	INTP7	√	—	—	—
EGP8	EGN8	INTP8	√	√	√	—
EGP9	EGN9	INTP9	√	√	√	—
EGP10	EGN10	INTP10	√	√	—	—
EGP11	EGN11	INTP11	√	√	—	—

**Caution** When the input port pins used for the external interrupt functions are switched to the output mode, the INTPn interrupt might be generated upon detection of a valid edge.

When switching the input port pins to the output mode, set the port mode register (PMxx) to 0 after disabling the edge detection (by setting EGPn and EGNn to 0).

**Remark** n = 0 to 11

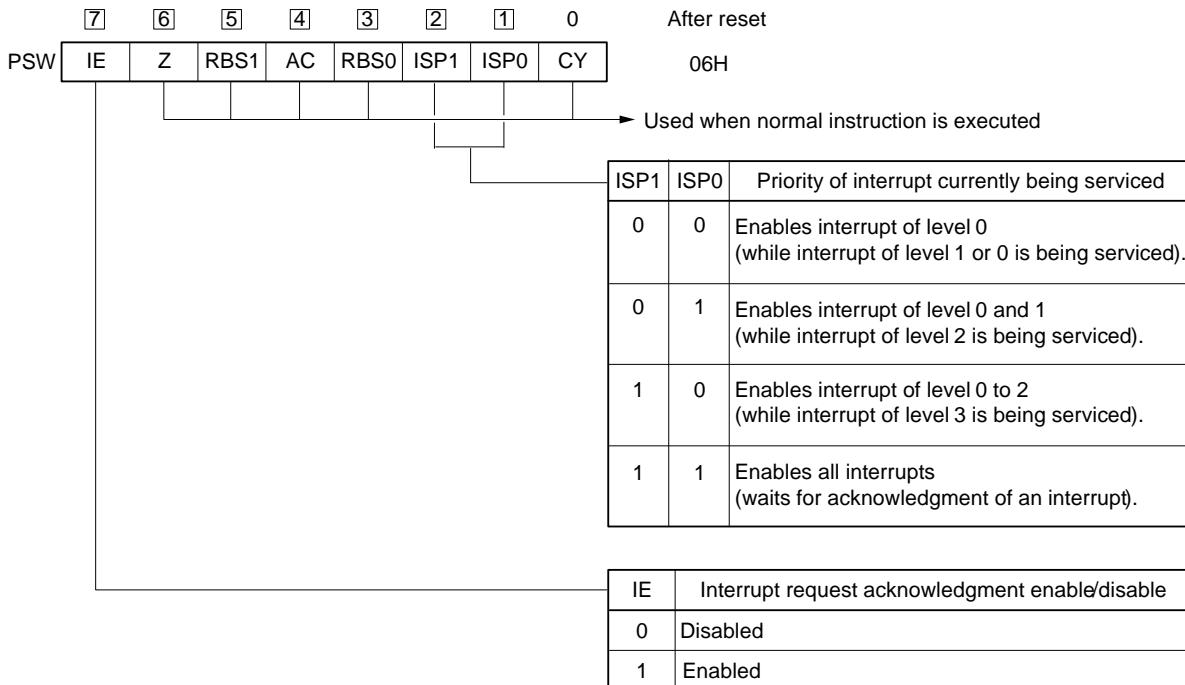
### 21.3.5 Program status word (PSW)

The program status word is a register used to hold the instruction execution result and the current status for an interrupt request. The IE flag that sets maskable interrupt enable/disable and the ISP0 and ISP1 flags that controls multiple interrupt servicing are mapped to the PSW.

Besides 8-bit read/write, this register can carry out operations using bit manipulation instructions and dedicated instructions (EI and DI). When a vectored interrupt request is acknowledged, if the BRK instruction is executed, the contents of the PSW are automatically saved into a stack and the IE flag is reset to 0. Upon acknowledgment of a maskable interrupt request, if the value of the priority specification flag register of the acknowledged interrupt is not 00, its value minus 1 is transferred to the ISP0 and ISP1 flags. The PSW contents are also saved into the stack with the PUSH PSW instruction. They are restored from the stack with the RETI, RETB, and POP PSW instructions.

Reset signal generation sets PSW to 06H.

**Figure 21 - 11 Configuration of Program Status Word**



## 21.4 Interrupt Servicing Operations

### 21.4.1 Maskable interrupt request acknowledgment

A maskable interrupt request becomes acknowledgeable when the interrupt request flag is set to 1 and the mask (MK) flag corresponding to that interrupt request is cleared to 0. A vectored interrupt request is acknowledged if interrupts are in the interrupt enabled state (when the IE flag is set to 1). However, a low-priority interrupt request is not acknowledged during servicing of a higher priority interrupt request.

The times from generation of a maskable interrupt request until vectored interrupt servicing is performed are listed in Table 21 - 10 below.

For the interrupt request acknowledgment timing, see **Figures 21 - 13** and **21 - 14**.

**Table 21 - 10 Time from Generation of Maskable Interrupt Until Servicing**

	Minimum Time	Maximum Time Note
Servicing time	9 clocks	16 clocks

**Note** Maximum time does not apply when an instruction from the internal RAM area is executed.

**Remark** 1 clock: 1/fCLK (fCLK: CPU clock)

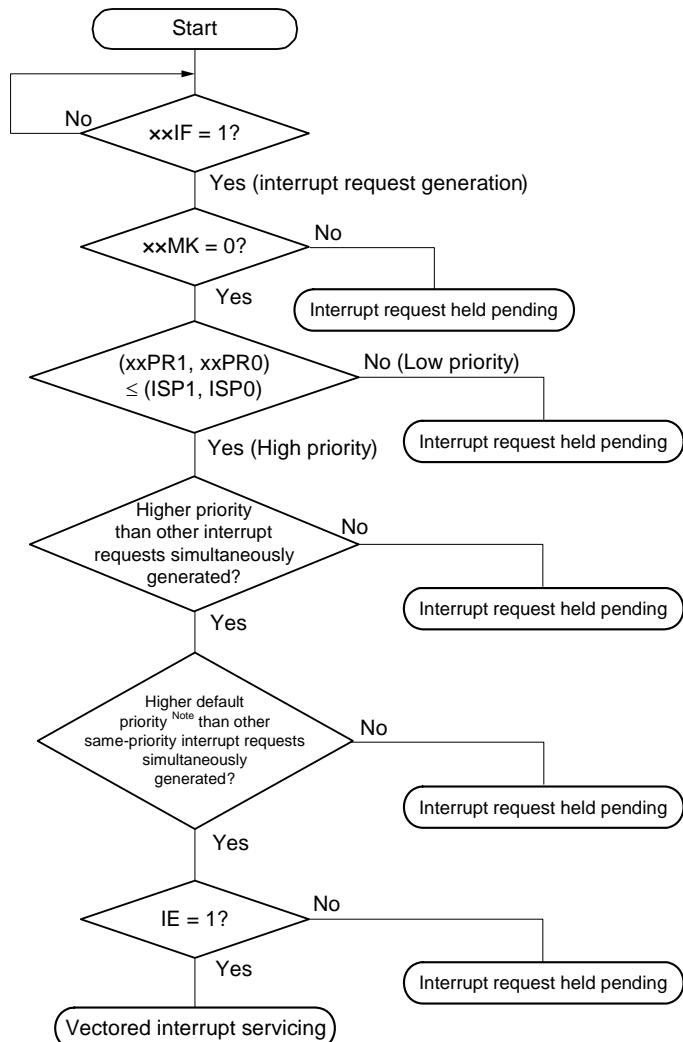
If two or more maskable interrupt requests are generated simultaneously, the request with a higher priority level specified in the priority specification flag is acknowledged first. If two or more interrupt requests have the same priority level, the request with the highest default priority is acknowledged first.

An interrupt request that is held pending is acknowledged when it becomes acknowledgeable.

Figure 21 - 12 shows the Interrupt Request Acknowledgment Processing Algorithm.

If a maskable interrupt request is acknowledged, the contents are saved into the stacks in the order of PSW, then PC, the IE flag is reset (0), and the contents of the priority specification flag corresponding to the acknowledged interrupt are transferred to the ISP1 and ISP0 flags. The vector table data determined for each interrupt request is loaded into the PC and branched.

Restoring from an interrupt is possible by using the RETI instruction.

**Figure 21 - 12 Interrupt Request Acknowledgment Processing Algorithm**

xxIF: Interrupt request flag

xxMK: Interrupt mask flag

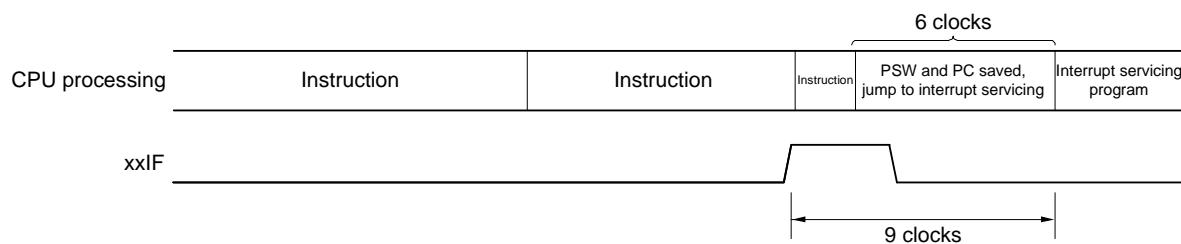
xxPR0: Priority specification flag 0

xxPR1: Priority specification flag 1

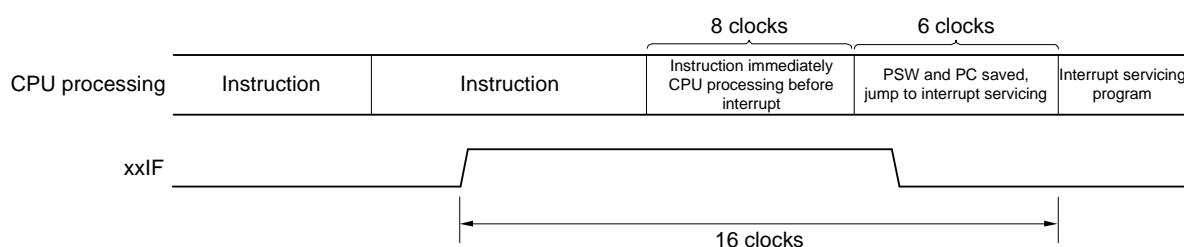
IE: Flag that controls acknowledgment of maskable interrupt request (1 = Enable, 0 = Disable)

ISP0, ISP1: Flag that indicates the priority level of the interrupt currently being serviced (see **Figure 21 - 11**)

**Note** For the default priority, refer to **Tables 21 - 1 to 21 - 4 Interrupt Source List**.

**Figure 21 - 13 Interrupt Request Acknowledgment Timing (Minimum Time)**

**Remark** 1 clock: 1/fCLK (fCLK: CPU clock)

**Figure 21 - 14 Interrupt Request Acknowledgment Timing (Maximum Time)**

**Remark** 1 clock: 1/fCLK (fCLK: CPU clock)

### 21.4.2 Software interrupt request acknowledgment

A software interrupt request is acknowledged by BRK instruction execution. Software interrupts cannot be disabled.

If a software interrupt request is acknowledged, the contents are saved into the stacks in the order of the program status word (PSW), then program counter (PC), the IE flag is reset (0), and the contents of the vector table (0007EH, 0007FH) are loaded into the PC and branched.

Restoring from a software interrupt is possible by using the RETB instruction.

**Caution** Can not use the RETI instruction for restoring from the software interrupt.

### 21.4.3 Multiple interrupt servicing

Multiple interrupt servicing occurs when another interrupt request is acknowledged during execution of an interrupt.

Multiple interrupt servicing does not occur unless the interrupt request acknowledgment enabled state is selected (IE = 1). When an interrupt request is acknowledged, interrupt request acknowledgment becomes disabled (IE = 0). Therefore, to enable multiple interrupt servicing, it is necessary to set (1) the IE flag with the EI instruction during interrupt servicing to enable interrupt acknowledgment.

Moreover, even if interrupts are enabled, multiple interrupt servicing may not be enabled, this being subject to interrupt priority control. Two types of priority control are available: default priority control and programmable priority control. Programmable priority control is used for multiple interrupt servicing.

In the interrupt enabled state, if an interrupt request with a priority equal to or higher than that of the interrupt currently being serviced is generated, it is acknowledged for multiple interrupt servicing. If an interrupt with a priority equal to or lower than that of the interrupt currently being serviced is generated during interrupt servicing, it is not acknowledged for multiple interrupt servicing. However, when setting the IE flag to 1 during the interruption at level 0, other level 0 interruptions can be allowed.

Interrupt requests that are not enabled because interrupts are in the interrupt disabled state or because they have a lower priority are held pending. When servicing of the current interrupt ends, the pending interrupt request is acknowledged following execution of at least one main processing instruction execution.

Table 21 - 11 shows Relationship Between Interrupt Requests Enabled for Multiple Interrupt Servicing During Interrupt Servicing and Figures 21 - 15 and 21 - 16 show multiple interrupt servicing examples.

**Table 21 - 11 Relationship Between Interrupt Requests Enabled for Multiple Interrupt Servicing During Interrupt Servicing**

Multiple Interrupt Request		Maskable Interrupt Request								Software Interrupt Request	
		Priority Level 0 (PR = 00)		Priority Level 1 (PR = 01)		Priority Level 2 (PR = 10)		Priority Level 3 (PR = 11)			
		IE = 1	IE = 0								
Maskable interrupt	ISP1 = 0 ISP0 = 0	✓	✗	✗	✗	✗	✗	✗	✗	✓	
	ISP1 = 0 ISP0 = 1	✓	✗	✓	✗	✗	✗	✗	✗	✓	
	ISP1 = 1 ISP0 = 0	✓	✗	✓	✗	✓	✗	✗	✗	✓	
	ISP1 = 1 ISP0 = 1	✓	✗	✓	✗	✓	✗	✓	✗	✓	
Software interrupt		✓	✗	✓	✗	✓	✗	✓	✗	✓	

**Remark 1.** ✓: Multiple interrupt servicing enabled

**Remark 2.** ✗: Multiple interrupt servicing disabled

**Remark 3.** ISP0, ISP1, and IE are flags contained in the PSW.

ISP1 = 0, ISP0 = 0: An interrupt of level 1 or level 0 is being serviced.

ISP1 = 0, ISP0 = 1: An interrupt of level 2 is being serviced.

ISP1 = 1, ISP0 = 0: An interrupt of level 3 is being serviced.

ISP1 = 1, ISP0 = 1: Wait for An interrupt acknowledgment (all interrupts enabled).

IE = 0: Interrupt request acknowledgment is disabled.

IE = 1: Interrupt request acknowledgment is enabled.

**Remark 4.** PR is a flag contained in the PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, and PR12H registers.

PR = 00: Specify level 0 with xxPR1x = 0, xxPR0x = 0 (higher priority level)

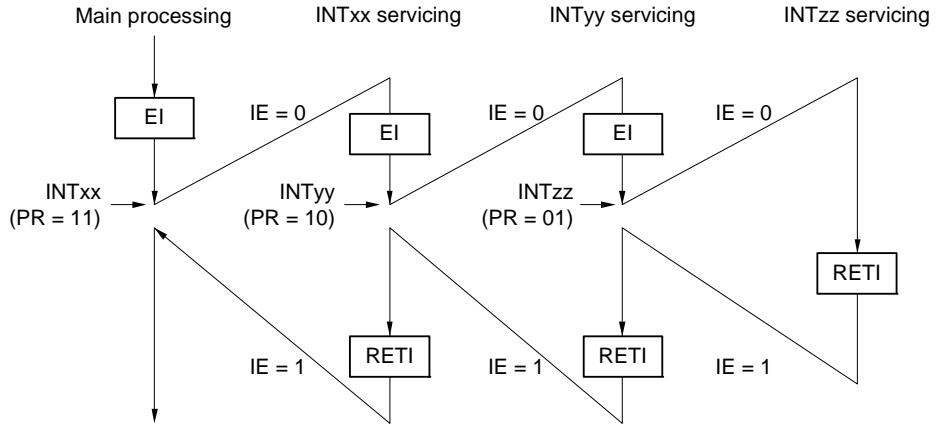
PR = 01: Specify level 1 with xxPR1x = 0, xxPR0x = 1

PR = 10: Specify level 2 with xxPR1x = 1, xxPR0x = 0

PR = 11: Specify level 3 with xxPR1x = 1, xxPR0x = 1 (lower priority level)

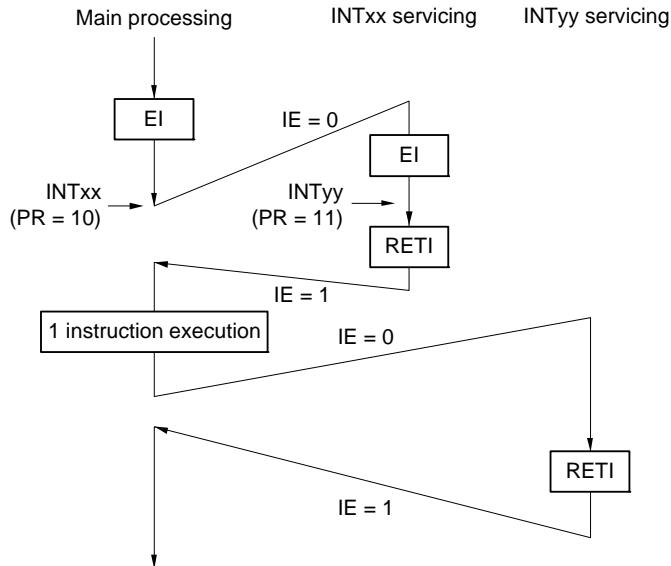
**Figure 21 - 15 Examples of Multiple Interrupt Servicing (1/2)**

Example 1. Multiple interrupt servicing occurs twice



During servicing of interrupt INTxx, two interrupt requests, INTyy and INTzz, are acknowledged, and multiple interrupt servicing takes place. Before each interrupt request is acknowledged, the EI instruction must always be issued to enable interrupt request acknowledgment.

Example 2. Multiple interrupt servicing does not occur due to priority control

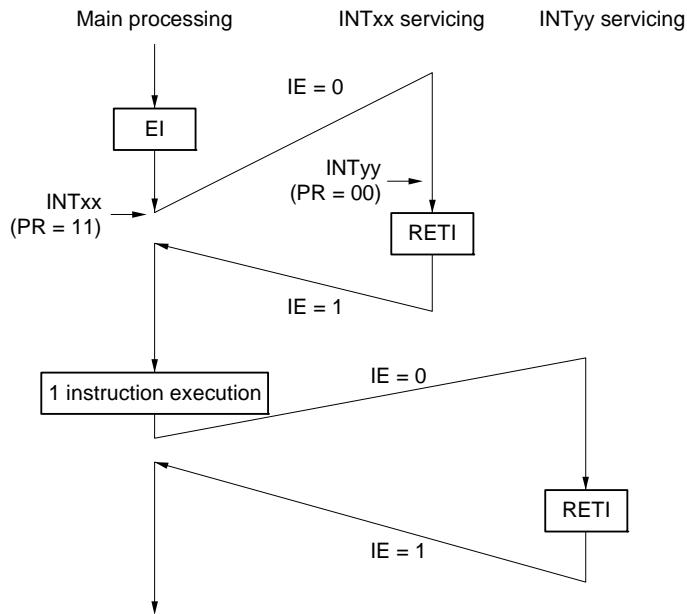


Interrupt request INTyy issued during servicing of interrupt INTxx is not acknowledged because its priority is lower than that of INTxx, and multiple interrupt servicing does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

- PR = 00: Specify level 0 with xxPR1x = 0, xxPROx = 0 (higher priority level)
- PR = 01: Specify level 1 with xxPR1x = 0, xxPROx = 1
- PR = 10: Specify level 2 with xxPR1x = 1, xxPROx = 0
- PR = 11: Specify level 3 with xxPR1x = 1, xxPROx = 1 (lower priority level)
- IE = 0: Interrupt request acknowledgment is disabled
- IE = 1: Interrupt request acknowledgment is enabled.

**Figure 21 - 16 Examples of Multiple Interrupt Servicing (2/2)**

Example 3. Multiple interrupt servicing does not occur because interrupts are not enabled



Interrupts are not enabled during servicing of interrupt INTxx (EI instruction is not issued), therefore, interrupt request INTyy is not acknowledged and multiple interrupt servicing does not take place. The INTyy interrupt request is held pending, and is acknowledged following execution of one main processing instruction.

- PR = 00: Specify level 0 with xxPR1x = 0, xxPR0x = 0 (higher priority level)
- PR = 01: Specify level 1 with xxPR1x = 0, xxPR0x = 1
- PR = 10: Specify level 2 with xxPR1x = 1, xxPR0x = 0
- PR = 11: Specify level 3 with xxPR1x = 1, xxPR0x = 1 (lower priority level)
- IE = 0: Interrupt request acknowledgment is disabled
- IE = 1: Interrupt request acknowledgment is enabled.

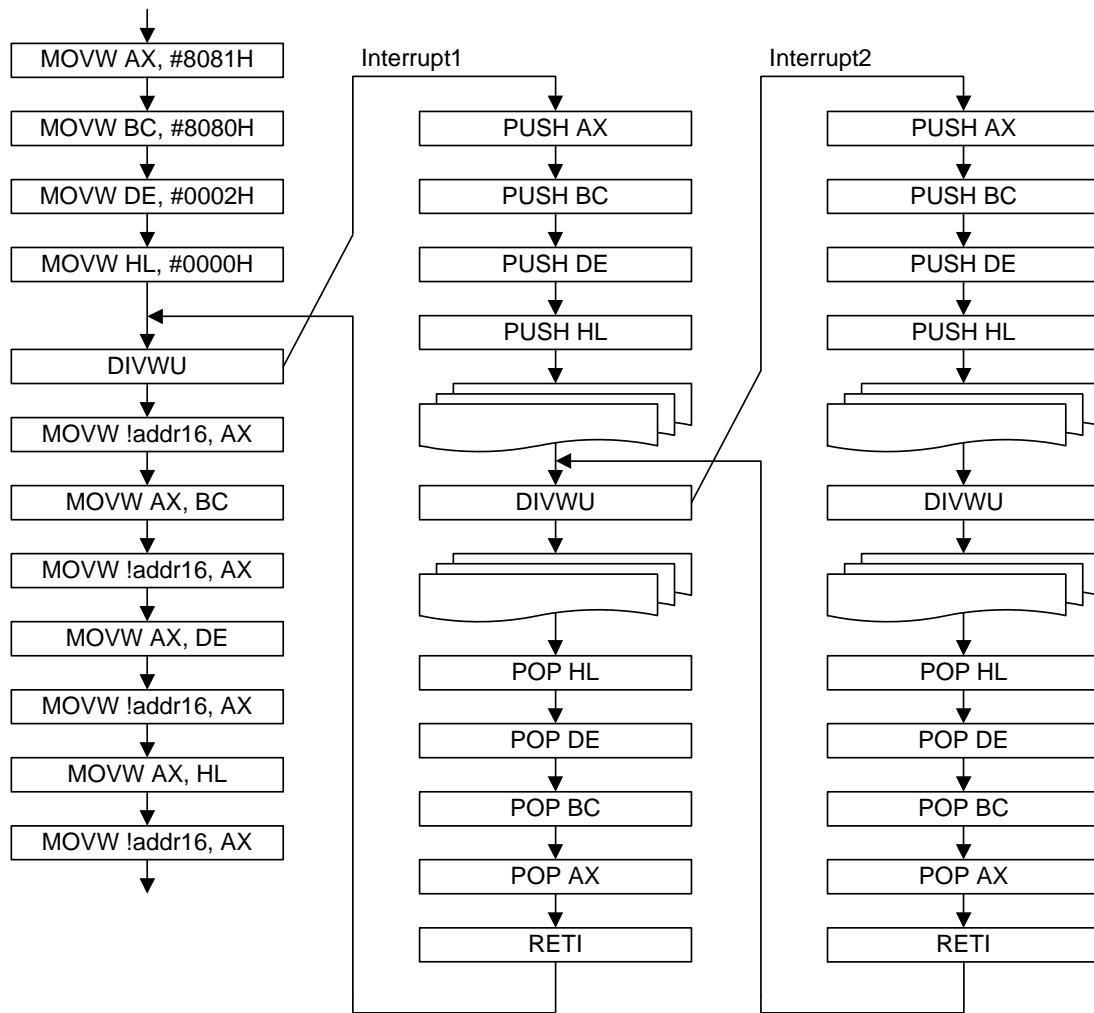
#### 21.4.4 Interrupt servicing during division instruction

The RL78/G14 handles interrupts during the DIVHU/DIVWU instruction in order to enhance the interrupt response when a division instruction is executed.

- When an interrupt is generated while the DIVHU/DIVWU instruction is executed, the instruction is suspended
- After the instruction is suspended, the PC indicates the next instruction after DIVHU/DIVWU
- An interrupt is generated by the next instruction
- PC-3 is stacked to execute the DIVHU/DIVWU instruction again

Normal interrupt	Interrupts while Executing DIVHU/DIVWU Instruction
(SP-1) ← PSW	(SP-1) ← PSW
(SP-2) ← (PC)s	(SP-2) ← (PC-3)s
(SP-3) ← (PC)H	(SP-3) ← (PC-3)H
(SP-4) ← (PC)L	(SP-4) ← (PC-3)L
PCs ← 0000	PCs ← 0000
PCh ← (Vector)	PCh ← (Vector)
PCL ← (Vector)	PCL ← (Vector)
SP ← SP-4	SP ← SP-4
IE ← 0	IE ← 0

The AX, BC, DE, and HL registers are used for DIVHU/DIVWU. Use these registers by stacking them for interrupt servicing.



**Caution** Disable interrupts when executing the DIVHU or DIVWU instruction in an interrupt servicing routine.

Alternatively, unless they are executed in the RAM area, note that execution of a DIVHU or DIVWU instruction is possible even with interrupts enabled as long as a NOP instruction is added immediately after the DIVHU or DIVWU instruction in the assembly language source code. The following compilers automatically add a NOP instruction immediately after any DIVHU or DIVWU instruction output during the build process.

- V. 1.71 and later versions of the CA78K0R (Renesas Electronics compiler), for both C and assembly language source code
- Service pack 1.40.6 and later versions of the EWRL78 (IAR compiler), for C language source code
- GNURL78 (KPIT compiler), for C language source code

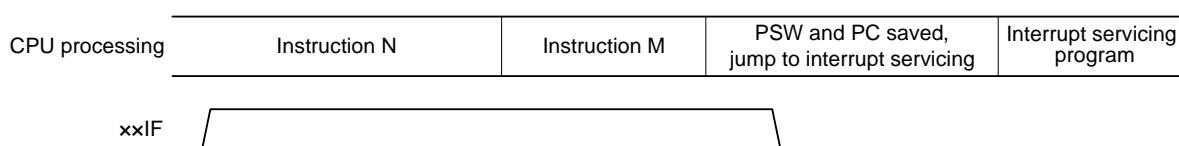
### 21.4.5 Interrupt request hold

There are instructions where, even if an interrupt request is issued while the instructions are being executed, interrupt request acknowledgment is held pending until the end of execution of the next instruction. These instructions (interrupt request hold instructions) are listed below.

- MOV PSW, #byte
- MOV PSW, A
- MOV1 PSW. bit, CY
- SET1 PSW. bit
- CLR1 PSW. bit
- RETB
- RETI
- POP PSW
- BTCLR PSW. bit, \$addr20
- EI
- DI
- SKC
- SKNC
- SKZ
- SKNZ
- SKH
- SKNH
- MULHU
- MULH
- MACHU
- MACH
- Write instructions for the IF0L, IF0H, IF1L, IF1H, IF2L, IF2H, MK0L, MK0H, MK1L, MK1H, MK2L, MK2H, PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, and PR12H registers

Figure 21 - 17 shows the timing at which interrupt requests are held pending.

**Figure 21 - 17 Interrupt Request Hold**



**Remark 1.** Instruction N: Interrupt request hold instruction

**Remark 2.** Instruction M: Instruction other than interrupt request hold instruction

## CHAPTER 22 KEY INTERRUPT FUNCTION

The number of key interrupt input channels differs, depending on the product.

	30, 32, 36-pin	40, 44-pin	48-pin	52, 64, 80, 100-pin
Key interrupt input channels	—	4 ch	6 ch	8 ch

### 22.1 Functions of Key Interrupt

A key interrupt (INTKR) can be generated by inputting a falling edge to the key interrupt input pins (KR0 to KR7).

**Table 22 - 1 Assignment of Key Interrupt Detection Pins**

Key interrupt pins	Key return mode register (KRM)
KR0	KRM0
KR1	KRM1
KR2	KRM2
KR3	KRM3
KR4	KRM4
KR5	KRM5
KR6	KRM6
KR7	KRM7

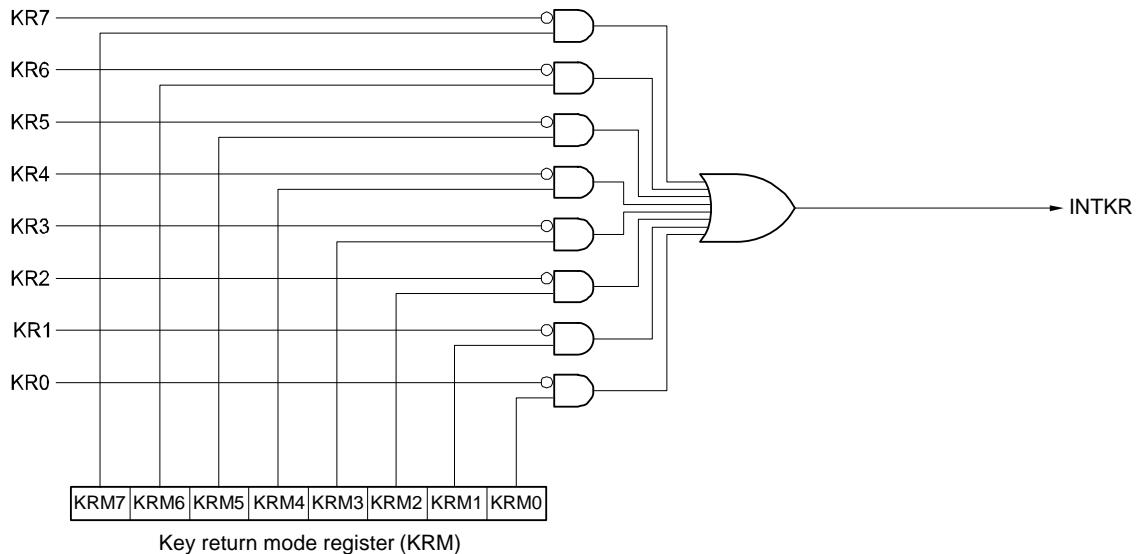
**Remark** KR0 to KR3: 40-pin, 44-pin  
KR0 to KR5: 48-pin  
KR0 to KR7: 52-pin, 64-pin, 80-pin, 100-pin

### 22.2 Configuration of Key Interrupt

The key interrupt includes the following hardware.

**Table 22 - 2 Configuration of Key Interrupt**

Item	Configuration
Control register	Key return mode register (KRM) Port mode register 7 (PM7)

**Figure 22 - 1 Block Diagram of Key Interrupt**

**Remark** KR0 to KR3: 40-pin, 44-pin

KR0 to KR5: 48-pin

KR0 to KR7: 52-pin, 64-pin, 80-pin, 100-pin

## 22.3 Register Controlling Key Interrupt

The key interrupt function is controlled by the following registers.

- Key return mode register (KRM)
- Port mode register 7 (PM7)

### 22.3.1 Key return mode register (KRM)

The KRM0 to KRM7 bits are registers for controlling signals KR0 to KR7.

The KRM register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 22 - 2 Format of Key return mode register (KRM)**

Address: FFF37H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
KRM	KRM7	KRM6	KRM5	KRM4	KRM3	KRM2	KRM1	KRM0
KRMn	Key interrupt mode control							
0	Does not detect key interrupt signal							
1	Detects key interrupt signal							

**Caution 1.** The on-chip pull-up resistors can be applied by setting the corresponding key interrupt input pins (bits) in pull-up resistor register 7 (PU7) to 1.

**Caution 2.** An interrupt will be generated if the target bit of the KRM register is set while a low level is being input to the key interrupt input pin. To ignore this interrupt, set the KRM register after disabling interrupt servicing by using the interrupt mask flag. Afterward, clear the interrupt request flag and enable interrupt servicing after waiting for the key interrupt input low-level width (tKR) (see 34.4 or 35.4 AC characteristics).

**Caution 3.** The pins not used in the key interrupt mode can be used as normal ports.

**Remark 1.** n = 0 to 7

**Remark 2.** KR0 to KR3: 40-pin, 44-pin

KR0 to KR5: 48-pin

KR0 to KR7: 52-pin, 64-pin, 80-pin, 100-pin

### 22.3.2 Port mode register 7 (PM7)

When port 7 is used as the key interrupt input pins (KR0 to KR7), set the PM7n bit to 1. The output latches of P7n at this time may be 0 or 1. The PM7 register can be set by a 1-bit or 8-bit memory manipulation instruction. Reset signal generation sets this register to FFH.

Use of an on-chip pull-up resistor can be specified in 1-bit units by the pull-up resistor option register 7 (PU7).

**Figure 22 - 3 Format of Port mode register 7 (PM7)**

Address: FFF27H      After reset: FFH      R/W

Symbol	7	6	5	4	3	2	1	0
PM7	PM77	PM76	PM75	PM74	PM73	PM72	PM71	PM70
PM7n	P7n pin I/O mode selection (n = 0 to 7)							
0	Output mode (output buffer on)							
1	Input mode (output buffer off)							

## CHAPTER 23 STANDBY FUNCTION

### 23.1 Standby Function

The standby function reduces the operating current of the system, and the following three modes are available.

(1) HALT mode

HALT instruction execution sets the HALT mode. In the HALT mode, the CPU operation clock is stopped. If the high-speed system clock oscillator, high-speed on-chip oscillator, or subsystem clock oscillator is operating before the HALT mode is set, oscillation of each clock continues. In this mode, the operating current is not decreased as much as in the STOP mode, but the HALT mode is effective for restarting operation immediately upon interrupt request generation and carrying out intermittent operations frequently.

(2) STOP mode

STOP instruction execution sets the STOP mode. In the STOP mode, the high-speed system clock oscillator and high-speed on-chip oscillator stop, stopping the whole system, thereby considerably reducing the CPU operating current.

Because this mode can be cleared by an interrupt request, it enables intermittent operations to be carried out. However, because a wait time is required to secure the oscillation stabilization time after the STOP mode is released when the X1 clock is selected, select the HALT mode if it is necessary to start processing immediately upon interrupt request generation.

(3) SNOOZE mode

In the case of CSIp or UARTq data reception, an A/D conversion request by the timer trigger signal (the interrupt request signal (INTRTC/INTIT) or ELC event input), and DTC start source, the STOP mode is exited, the CSIp or UARTq data is received without operating the CPU, A/D conversion is performed, and DTC start source. This can only be specified when the high-speed on-chip oscillator is selected for the CPU/peripheral hardware clock (fCLK).

In either of these two modes, all the contents of registers, flags and data memory just before the standby mode is set are held. The I/O port output latches and output buffer statuses are also held.

**Caution 1.** The STOP mode can be used only when the CPU is operating on the main system clock. Do not set to the STOP mode while the CPU operates with the subsystem clock. The HALT mode can be used when the CPU is operating on either the main system clock or the subsystem clock.

**Caution 2.** When shifting to the STOP mode, be sure to stop the peripheral hardware operation operating with main system clock before executing STOP instruction (except SNOOZE mode setting unit).

**Caution 3.** When using CSIP, UARTq, or the A/D converter in the SNOOZE mode, set up serial standby control register m (SSCm) and A/D converter mode register 2 (ADM2) before switching to the STOP mode. For details, see 17.3 Registers Controlling Serial Array Unit and 14.3 Registers Controlling A/D Converter.

**Caution 4.** The following sequence is recommended for operating current reduction of the A/D converter when the standby function is used: First clear bit 7 (ADCS) and bit 0 (ADCE) of A/D converter mode register 0 (ADM0) to 0 to stop the A/D conversion operation, and then execute the STOP instruction.

**Caution 5.** It can be selected by the option byte whether the low-speed on-chip oscillator continues oscillating or stops in the HALT or STOP mode. For details, see CHAPTER 29 OPTION BYTE.

**Remark** 30 to 64-pin products: p = 00; q = 0; m = 0  
80, 100-pin products: p = 00, 20; q = 0, 2; m = 0, 1

## 23.2 Registers controlling standby function

The registers which control the standby function are described below.

- Subsystem clock supply mode control register (OSMC)
- Oscillation stabilization time counter status register (OSTC)
- Oscillation stabilization time select register (OSTS)

**Remark** For details of registers described above, see CHAPTER 5 CLOCK GENERATOR. For registers which control the SNOOZE mode, CHAPTER 14 A/D CONVERTER and CHAPTER 17 SERIAL ARRAY UNIT.

## 23.3 Standby Function Operation

### 23.3.1 HALT mode

#### (1) HALT mode

The HALT mode is set by executing the HALT instruction. HALT mode can be set regardless of whether the CPU clock before the setting was the high-speed system clock, high-speed on-chip oscillator clock, or subsystem clock.

The operating statuses in the HALT mode are shown below.

**Caution** Because the interrupt request signal is used to clear the HALT mode, if the interrupt mask flag is 0 (the interrupt processing is enabled) and the interrupt request flag is 1 (the interrupt request signal is generated), the HALT mode is not entered even if the HALT instruction is executed in such a situation.

**Table 23 - 1 Operating Statuses in HALT Mode (1/2)**

Item	HALT Mode Setting	When HALT Instruction is Executed While CPU is Operating on Main System Clock			
		When CPU is Operating on High-speed On-chip Oscillator Clock (fIH)	When CPU is Operating on X1 Clock (fx)	When CPU is Operating on External Main System Clock (fEX)	
System clock		Clock supply to the CPU is stopped			
Main system clock	fIH	Operation continues (cannot be stopped)		Operation disabled	
	fx	Operation disabled		Operation continues (cannot be stopped)	Cannot operate
	fEX			Cannot operate	Operation continues (cannot be stopped)
Subsystem clock	fXT	Status before HALT mode was set is retained			
	fEXS				
Low-speed on-chip oscillator clock	fIL	Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of subsystem clock supply mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops			
CPU		Operation stopped			
Code flash memory					
Data flash memory					
RAM		Operation stopped (Operable while in the DTC is executed)			
Port (latch)		Status before HALT mode was set is retained			
Timer array unit		Operable			
Real-time clock (RTC)					
12-bit Interval timer					
Watchdog timer		See <b>CHAPTER 13 WATCHDOG TIMER</b> .			
Timer RJ		Operable			
Timer RD					
Timer RG					
Clock output/buzzer output					
A/D converter					
D/A converter Note					
Comparator Note					
Serial array unit (SAU)					
Serial interface (IICA)					
Data transfer controller (DTC)					
Event link controller (ELC)		Operable function blocks can be linked			
Power-on-reset function		Operable			
Voltage detection function					
External interrupt					
Key interrupt function					
CRC operation function	High-speed CRC				
	General-purpose CRC	In the calculation of the RAM area, operable when DTC is executed only			
Illegal-memory access detection function		Operable when DTC is executed only			
RAM parity error detection function					
RAM guard function					
SFR guard function					

**Note** Only for products with 96 KB or more code flash memory.

**Remark** Operation stopped: Operation is automatically stopped before switching to the HALT mode.

Operation disabled: Operation is stopped before switching to the HALT mode.

fIH: High-speed on-chip oscillator clock

fIL: Low-speed on-chip oscillator clock

fx: X1 clock

fEX: External main system clock

fXT: XT1 clock

fEXS: External subsystem clock

**Table 23 - 2 Operating Statuses in HALT Mode (2/2)**

Item	HALT Mode Setting	When HALT Instruction is Executed While CPU is Operating on Subsystem Clock	
		When CPU is Operating on XT1 Clock (fxT)	When CPU is Operating on External Subsystem Clock (fxExS)
System clock		Clock supply to the CPU is stopped	
Main system clock	fIH	Operation disabled	
	fx		
	fEX		
Subsystem clock	fxT	Operation continues (cannot be stopped)	Cannot operate
	fxExS	Cannot operate	Operation continues (cannot be stopped)
Low-speed on-chip oscillator clock	fIL	Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of subsystem clock supply mode control register (OSMC) • WUTMMCK0 = 1: Oscillates • WUTMMCK0 = 0 and WDTON = 0: Stops • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates • WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops	
CPU		Operation stopped	
Code flash memory			
Data flash memory			
RAM		Operation stopped (Operable while in the DTC is executed)	
Port (latch)		Status before HALT mode was set is retained	
Timer array unit		Operates when the RTCLPC bit is 0 (operation is disabled when the RTCLPC bit is not 0).	
Real-time clock (RTC)		Operable	
12-bit Interval timer			
Watchdog timer		See <b>CHAPTER 13 WATCHDOG TIMER</b> .	
Timer RJ		Operates when the RTCLPC bit is 0 (operation is disabled when the RTCLPC bit is not 0).	
Timer RD			
Timer RG			
Clock output/buzzer output			
A/D converter		Operation disabled	
D/A converter Note		Operates when the RTCLPC bit is 0 (operation is disabled when the RTCLPC bit is not 0).	
Comparator Note		Operable when the external input (IVREFn) is selected as the comparator reference voltage.	
Serial array unit (SAU)		Operates when the RTCLPC bit is 0 (operation is disabled when the RTCLPC bit is not 0).	
Serial interface (IICA)		Operation disabled	
Data transfer controller (DTC)		Operates when the RTCLPC bit is 0 (operation is disabled when the RTCLPC bit is not 0).	
Event link controller (ELC)		Operable function blocks can be linked	
Power-on-reset function		Operable	
Voltage detection function			
External interrupt			
Key interrupt function			
CRC operation function	High-speed CRC	Operation disabled	
	General-purpose CRC	In the calculation of the RAM area, operable when DTC is executed only	
Illegal-memory access detection function		Operable when DTC is executed only	
RAM parity error detection function			
RAM guard function			
SFR guard function			

**Note** Only for products with 96 KB or more code flash memory.

**Remark** Operation stopped: Operation is automatically stopped before switching to the HALT mode.

Operation disabled: Operation is stopped before switching to the HALT mode.

fIH: High-speed on-chip oscillator clock

fIL: Low-speed on-chip oscillator clock

fx: X1 clock

fEX: External main system clock

fxT: XT1 clock

fExS: External subsystem clock

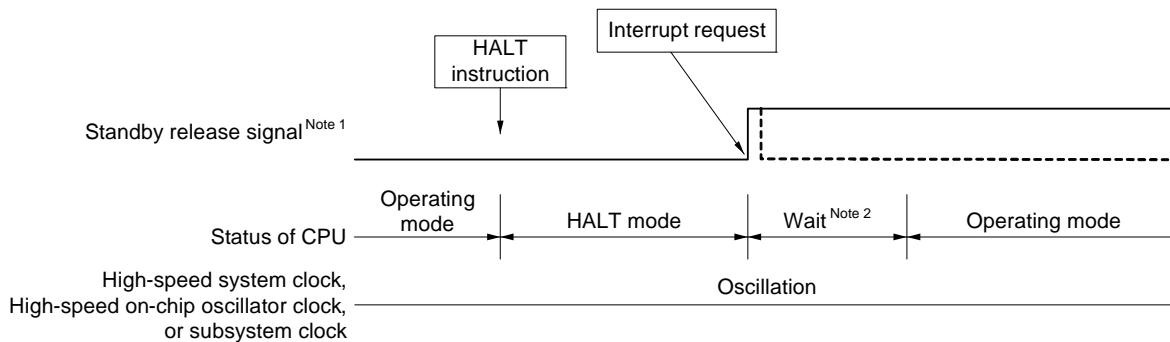
## (2) HALT mode release

The HALT mode can be released by the following two sources.

## (a) Release by unmasked interrupt request

When an unmasked interrupt request is generated, the HALT mode is released. If interrupt acknowledgment is enabled, vectored interrupt servicing is carried out. If interrupt acknowledgment is disabled, the next address instruction is executed.

**Figure 23 - 1 HALT Mode Release by Interrupt Request Generation**



**Note 1.** For details of the standby release signal, see **Figure 21 - 1 Basic Configuration of Interrupt Function**.

**Note 2.** Wait time for HALT mode release

- When vectored interrupt servicing is carried out
 

Main system clock:	15 to 16 clocks
Subsystem clock (RTCLPC = 0):	10 to 11 clocks
Subsystem clock (RTCLPC = 1):	11 to 12 clocks
- When vectored interrupt servicing is not carried out
 

Main system clock:	9 to 10 clocks
Subsystem clock (RTCLPC = 0):	4 to 5 clocks
Subsystem clock (RTCLPC = 1):	5 to 6 clocks

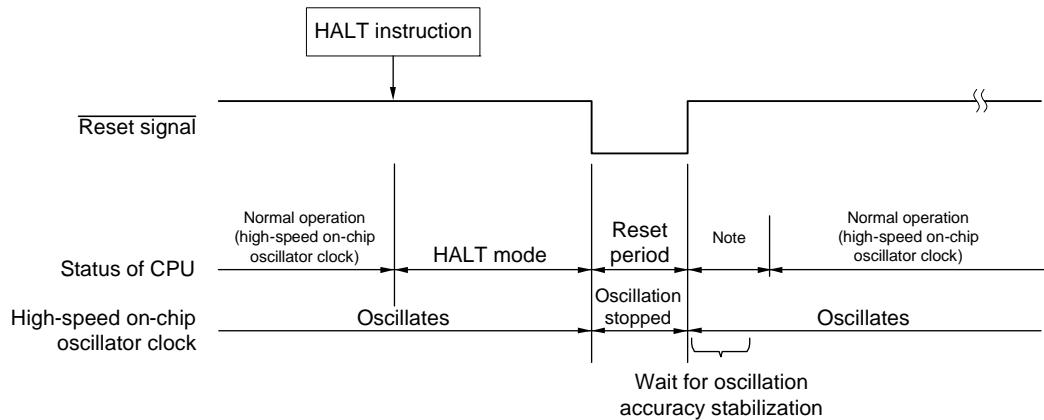
**Remark** The broken lines indicate the case when the interrupt request which has released the standby mode is acknowledged.

## (b) Release by reset signal generation

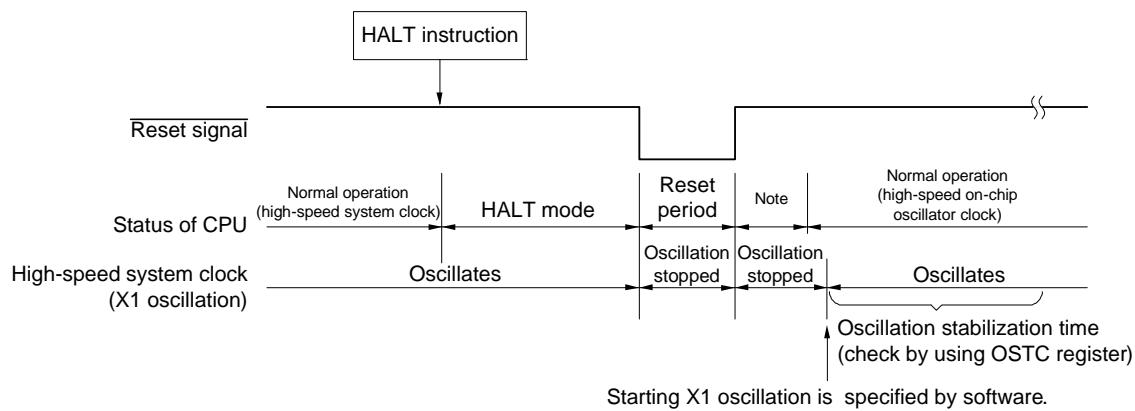
When the reset signal is generated, HALT mode is released, and then, as in the case with a normal reset operation, the program is executed after branching to the reset vector address.

**Figure 23 - 2 HALT Mode Release by Reset (1/2)**

## (1) When high-speed on-chip oscillator clock is used as CPU clock



## (2) When high-speed system clock is used as CPU clock

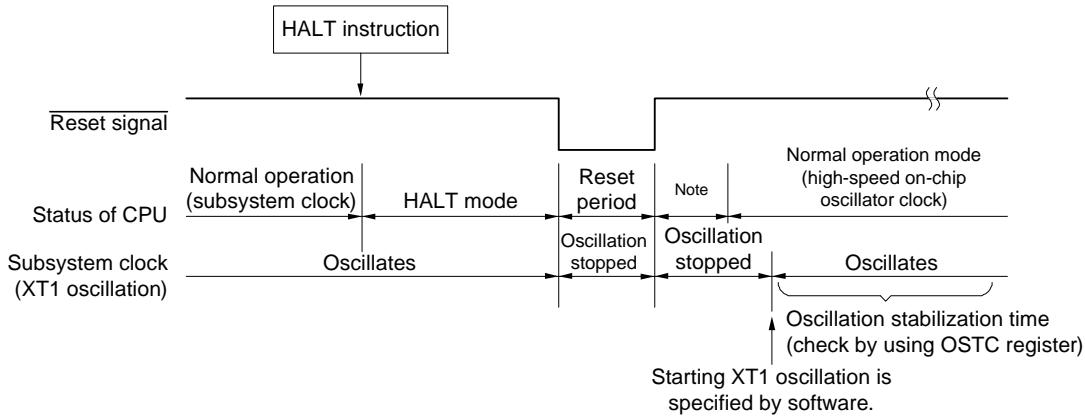
**Note**

For the reset processing time, see **CHAPTER 24 RESET FUNCTION**.

For the reset processing time of the power-on-reset circuit (POR) and voltage detector (LVD), see **CHAPTER 25 POWER-ON-RESET CIRCUIT**.

**Figure 23 - 3 HALT Mode Release by Reset (2/2)**

(3) When subsystem clock is used as CPU clock

**Note**For the reset processing time, see **CHAPTER 24 RESET FUNCTION**.For the reset processing time of the power-on-reset circuit (POR) and voltage detector (LVD), see **CHAPTER 25 POWER-ON-RESET CIRCUIT**.

### 23.3.2 STOP mode

#### (1) STOP mode setting and operating statuses

The STOP mode is set by executing the STOP instruction, and it can be set only when the CPU clock before the setting was the high-speed on-chip oscillator clock, X1 clock, or external main system clock.

**Caution** Because the interrupt request signal is used to clear the STOP mode, if the interrupt mask flag is 0 (the interrupt processing is enabled) and the interrupt request flag is 1 (the interrupt request signal is generated), the STOP mode is immediately cleared if set when the STOP instruction is executed in such a situation.

Accordingly, once the STOP instruction is executed, the system returns to its normal operating mode after the elapse of release time from the STOP mode.

**Remark** 30 to 64-pin products: p = 00; q = 0; m = 0

80, 100-pin products: p = 00, 20; q = 0, 2; m = 0, 1

The operating statuses in the STOP mode are shown below.

**Table 23 - 3 Operating Statuses in STOP Mode**

Item	STOP Mode Setting	When STOP Instruction is Executed While CPU is Operating on Main System Clock							
		When CPU is Operating on High-speed On-chip Oscillator Clock (fIH)	When CPU is Operating on X1 Clock (fx)	When CPU is Operating on External Main System Clock (fEX)					
System clock	Clock supply to the CPU is stopped								
Main system clock	fIH	Stopped							
	fx								
	fEX								
Subsystem clock	fXT	Status before STOP mode was set is retained							
	fEXS								
fIL	<p>Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of subsystem clock supply mode control register (OSMC)</p> <ul style="list-style-type: none"> <li>• WUTMMCK0 = 1: Oscillates</li> <li>• WUTMMCK0 = 0 and WDTON = 0: Stops</li> <li>• WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates</li> <li>• WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops</li> </ul>								
CPU	Operation stopped								
Code flash memory									
Data flash memory									
RAM									
Port (latch)	Status before STOP mode was set is retained								
Timer array unit	Operation disabled								
Real-time clock (RTC)	Operable								
12-bit Interval timer									
Watchdog timer	See <b>CHAPTER 13 WATCHDOG TIMER</b> .								
Timer RJ	<ul style="list-style-type: none"> <li>• Operable in event count mode when TRJIO input with no filer is selected</li> <li>• Operable when the subsystem clock is selected as the count source and RTCLPC in the OSMC register = 0</li> <li>• Operable when the low-speed on-chip oscillator is selected as the count source</li> <li>• Operation is disabled under any conditions other than the above</li> </ul>								
Timer RD	Operation disabled								
Timer RG									
Clock output/buzzer output	Operates when the subsystem clock is selected as the clock source for counting and the RTCLPC bit is 0 (operation is disabled when a clock other than the subsystem clock is selected and the RTCLPC bit is not 0).								
A/D converter	Wakeup operation is enabled (switching to SNOOZE mode)								
D/A converter Note	Operable (status before STOP mode was set is retained)								
Comparator Note	Operable (only when the digital filter is not used and the external input (IVREFn) is selected as the reference voltage of the comparator)								
Serial array unit (SAU)	<p>Wakeup operation is enabled only for CSIP and UARTq (switching to SNOOZE mode)</p> <p>Operation is disabled for anything other than CSIP and UARTq</p>								
Serial interface (IICA)	Wakeup by address match operable								
Data transfer controller (DTC)	DTC activation source receiving operation enabled (switching to SNOOZE mode)								
Event link controller (ELC)	Operable function blocks can be linked								
Power-on-reset function	Operable								
Voltage detection function									
External interrupt									
Key interrupt function									
CRC operation function	High-speed CRC	Operation stopped							
	General-purpose CRC								
Illegal-memory access detection function									
RAM parity error detection function									
RAM guard function									
SFR guard function									

**Note** Only for products with 96 KB or more code flash memory.

(Remarks are listed on the next page.)

**Remark 1.** Operation stopped: Operation is automatically stopped before switching to the STOP mode.

Operation disabled: Operation is stopped before switching to the STOP mode.

f<sub>H</sub>: High-speed on-chip oscillator clock

f<sub>L</sub>: Low-speed on-chip oscillator clock

f<sub>X</sub>: X1 clock

f<sub>EX</sub>: External main system clock

f<sub>XT1</sub>: XT1 clock

f<sub>EXS</sub>: External subsystem clock

**Remark 2.** 30 to 64-pin products:

p = 00; q = 0

80, 100-pin products:

p = 00, 20; q = 0, 2

## (2) STOP mode release

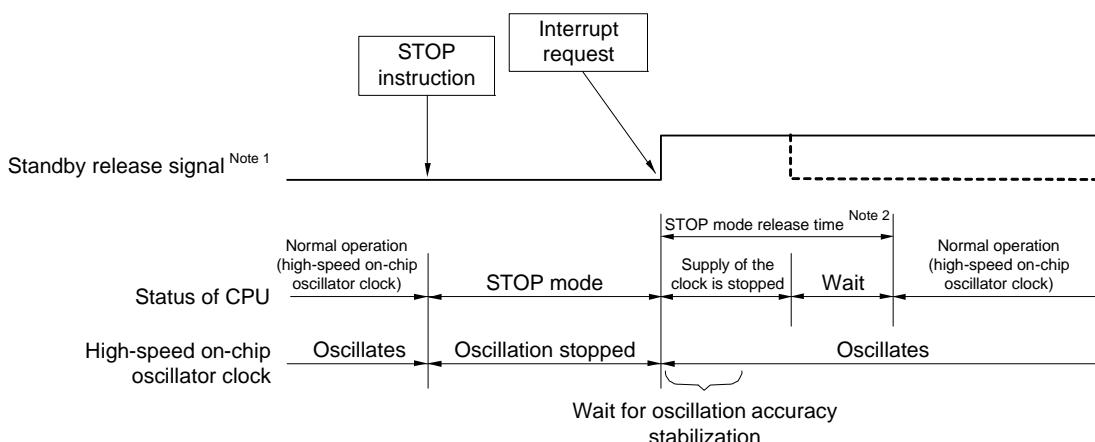
The STOP mode can be released by the following two sources.

### (a) Release by unmasked interrupt request

When an unmasked interrupt request is generated, the STOP mode is released. After the oscillation stabilization time has elapsed, if interrupt acknowledgment is enabled, vectored interrupt servicing is carried out. If interrupt acknowledgment is disabled, the next address instruction is executed.

**Figure 23 - 4 STOP Mode Release by Interrupt Request Generation (1/2)**

### (1) When high-speed on-chip oscillator clock is used as CPU clock



**Note 1.** For details of the standby release signal, see **Figure 21 - 1 Basic Configuration of Interrupt Function**.

**Note 2.** STOP mode release time

Supply of the clock is stopped:

- When FRQSEL4 = 0: 18 µs to 65 µs
- When FRQSEL4 = 1: 18 µs to 135 µs

Wait:

- When vectored interrupt servicing is carried out: 7 clocks
- When vectored interrupt servicing is not carried out: 1 clock

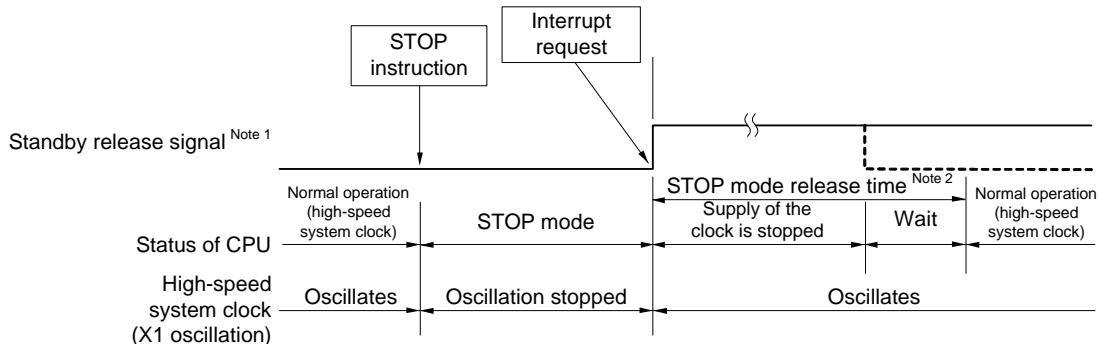
**Caution** To shorten oscillation stabilization time after the STOP mode is released when the CPU operates with the high-speed system clock (X1 oscillation), temporarily switch the CPU clock to the high-speed on-chip oscillator clock before the execution of the STOP instruction.

**Remark 1.** The clock supply stop time varies depending on the temperature conditions and STOP mode period.

**Remark 2.** The broken lines indicate the case when the interrupt request that has released the standby mode is acknowledged.

**Figure 23 - 5 STOP Mode Release by Interrupt Request Generation (2/2)**

(2) When high-speed system clock (X1 oscillation) is used as CPU clock



**Note 1.** For details of the standby release signal, see **Figure 21 - 1 Basic Configuration of Interrupt Function**.

**Note 2.** STOP mode release time

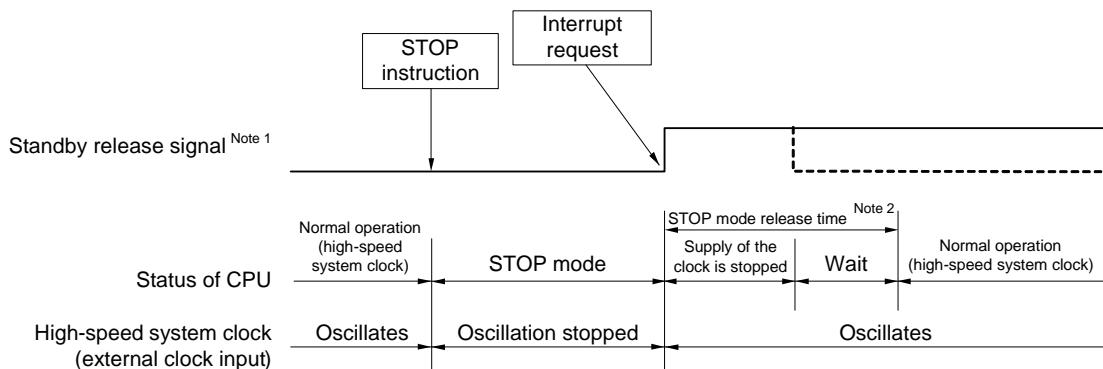
Supply of the clock is stopped:

- When FRQSEL4 = 0: 18 µs to "whichever is longer 65 µs or the oscillation stabilization time (set by OSTS)"
- When FRQSEL4 = 1: 18 µs to "whichever is longer 135 µs or the oscillation stabilization time (set by OSTS)"

Wait:

- When vectored interrupt servicing is carried out: 10 to 11 clocks
- When vectored interrupt servicing is not carried out: 4 to 5 clocks

(3) When high-speed system clock (external clock input) is used as CPU clock



**Note 1.** For details of the standby release signal, see **Figure 21 - 1 Basic Configuration of Interrupt Function**.

**Note 2.** STOP mode release time

Supply of the clock is stopped:

- When FRQSEL4 = 0: 18 µs to 65 µs
- When FRQSEL4 = 1: 18 µs to 135 µs

Wait:

- When vectored interrupt servicing is carried out: 7 clocks
- When vectored interrupt servicing is not carried out: 1 clock

**Caution** To reduce the oscillation stabilization time after release from the STOP mode while CPU operates based on the high-speed system clock (X1 oscillation), switch the clock to the high-speed on-chip oscillator clock temporarily before executing the STOP instruction.

**Remark 1.** The clock supply stop time varies depending on the temperature conditions and STOP mode period.

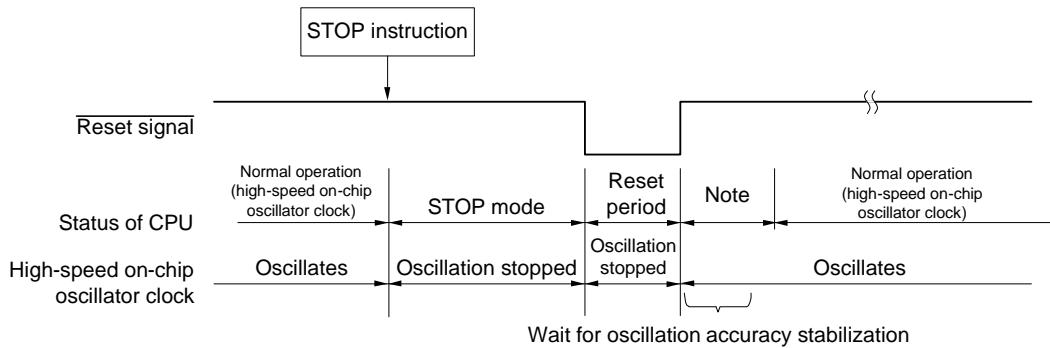
**Remark 2.** The broken lines indicate the case when the interrupt request that has released the standby mode is acknowledged.

## (b) Release by reset signal generation

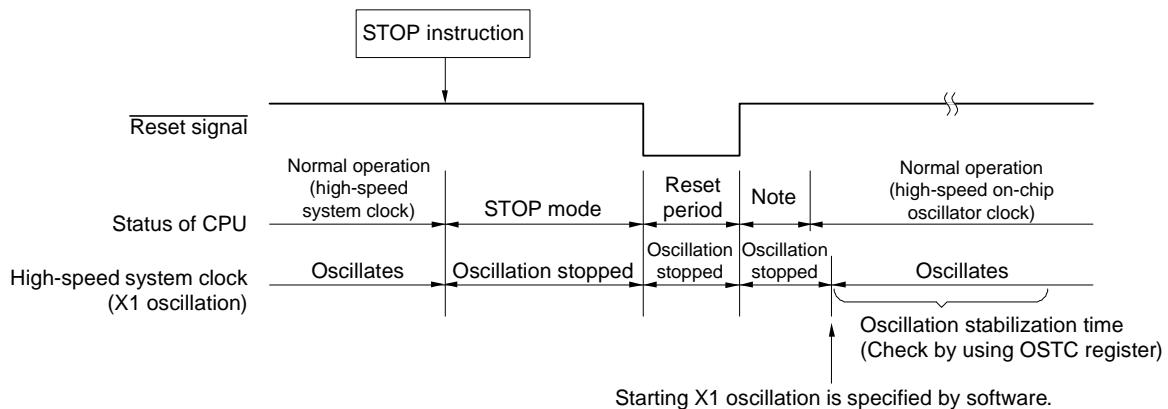
When the reset signal is generated, STOP mode is released, and then, as in the case with a normal reset operation, the program is executed after branching to the reset vector address.

**Figure 23 - 6 STOP Mode Release by Reset**

## (1) When high-speed on-chip oscillator clock is used as CPU clock



## (2) When high-speed system clock is used as CPU clock



**Note** For the reset processing time, see **CHAPTER 24 RESET FUNCTION**.

For the reset processing time of the power-on-reset circuit (POR) and voltage detector (LVD), see **CHAPTER 25 POWER-ON-RESET CIRCUIT**.

### 23.3.3 SNOOZE mode

(1) SNOOZE mode setting and operating statuses

The SNOOZE mode can only be specified for CSIp, the A/D converter, or DTC. The UARTq can be specified only when FRQSEL4 in the option byte 000C2H is 0. Note that this mode can only be specified if the CPU clock is the high-speed on-chip oscillator clock.

When using CSIp or UARTq in the SNOOZE mode, set up serial standby control register m (SSCm) before switching to the STOP mode. For details, see **17.3 Registers Controlling Serial Array Unit**.

When using the A/D converter in the SNOOZE mode, set up A/D converter mode register 2 (ADM2) before switching to the STOP mode. For details, see **14.3 Registers Controlling A/D Converter**.

When DTC transfer is used in SNOOZE mode, before switching to the STOP mode, allow DTC activation by interrupt to be used. During STOP mode, detecting DTC activation by interrupt enables DTC transit to SNOOZE mode, automatically. For details, see **19.3 Registers Controlling DTC**.

**Remark** 30 to 64-pin products: p = 00; q = 0; m = 0

80, 100-pin products: p = 00, 20; q = 0, 2; m = 0, 1

In SNOOZE mode transition, wait status to be only following time.

Transition time from STOP mode to SNOOZE mode

When FRQSEL4 = 0: 18  $\mu$ s to 65  $\mu$ s

When FRQSEL4 = 1: 18  $\mu$ s to 135  $\mu$ s

**Remark** Transition time from STOP mode to SNOOZE mode varies depending on the temperature conditions and the STOP mode period.

Transition time from SNOOZE mode to normal operation:

- When vectored interrupt servicing is carried out:

HS (High-speed main) mode: "4.99  $\mu$ s to 9.44  $\mu$ s" + 7 clocks

LS (Low-speed main) mode: "1.10  $\mu$ s to 5.08  $\mu$ s" + 7 clocks

LV (Low-voltage main) mode: "16.58  $\mu$ s to 25.40  $\mu$ s" + 7 clocks

- When vectored interrupt servicing is not carried out:

HS (High-speed main) mode: "4.99  $\mu$ s to 9.44  $\mu$ s" + 1 clock

LS (Low-speed main) mode: "1.10  $\mu$ s to 5.08  $\mu$ s" + 1 clock

LV (Low-voltage main) mode: "16.58  $\mu$ s to 25.40  $\mu$ s" + 1 clock

The operating statuses in the SNOOZE mode are shown next.

**Table 23 - 4 Operating Statuses in SNOOZE Mode**

Item	STOP Mode Setting	During STOP mode, receiving data signal from CSIP and UARTq, inputting timer trigger signal to A/D converter, and generating DTC activation by interrupt When CPU is Operating on High-speed On-chip Oscillator Clock (fIH)
System clock	Clock supply to the CPU is stopped	
Main system clock	fIH fx fEX	Operation started Stopped
Subsystem clock	fXT fEXS	Use of the status while in the STOP mode continues
fIL		Set by bits 0 (WDSTBYON) and 4 (WDTON) of option byte (000C0H), and WUTMMCK0 bit of subsystem clock supply mode control register (OSMC) <ul style="list-style-type: none"> <li>• WUTMMCK0 = 1: Oscillates</li> <li>• WUTMMCK0 = 0 and WDTON = 0: Stops</li> <li>• WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 1: Oscillates</li> <li>• WUTMMCK0 = 0, WDTON = 1, and WDSTBYON = 0: Stops</li> </ul>
CPU		Operation stopped
Code flash memory		
Data flash memory		
RAM		Operation stopped (Operable while in the DTC is executed)
Port (latch)		Use of the status while in the STOP mode continues
Timer array unit		Operation disabled
Real-time clock (RTC)		Operable
12-bit interval timer		
Watchdog timer		See <b>CHAPTER 13 WATCHDOG TIMER</b> .
Timer RJ		<ul style="list-style-type: none"> <li>• Operable in event count mode when TRJIO input with no filer is selected</li> <li>• Operable when the subsystem clock is selected as the count source and RTCLPC in the OSMC register = 0</li> <li>• Operable when the low-speed on-chip oscillator is selected as the count source</li> <li>• Operation is disabled under any conditions other than the above</li> </ul>
Timer RD		Operation disabled
Timer RG		
Clock output/buzzer output		Operates when the subsystem clock is selected as the clock source for counting and the RTCLPC bit is 0 (operation is disabled when a clock other than the subsystem clock is selected and the RTCLPC bit is not 0).
A/D converter		Operable
D/A converter Note		Operable (Status before SNOOZE mode was set is retained)
Comparator Note		Operable (when digital filter is not used)
Serial array unit (SAU)		Operable only CSIP and UARTq only. Operation disabled other than CSIP and UARTq.
Serial interface (IICA)		Operation disabled
Data transfer controller (DTC)		Operable
Event link controller (ELC)		Operable function blocks can be linked
Power-on-reset function		Operable
Voltage detection function		
External interrupt		
Key interrupt function		
CRC operation function	High-speed CRC General-purpose CRC	Operation stopped Operation disabled
Illegal-memory access detection function		Operable when executing the DTC
RAM parity error detection function		
RAM guard function		
SFR guard function		

**Note** Only for products with 96 KB or more code flash memory.

(Remarks are listed on the next page.)

**Remark 1.** Operation stopped: Operation is automatically stopped before switching to the STOP mode.

Operation disabled: Operation is stopped before switching to the STOP mode.

f<sub>H</sub>: High-speed on-chip oscillator clock

f<sub>L</sub>: Low-speed on-chip oscillator clock

f<sub>X</sub>: X1 clock

f<sub>EX</sub>: External main system clock

f<sub>XT</sub>: XT1 clock

f<sub>EXS</sub>: External subsystem clock

**Remark 2.** 30 to 64-pin products:

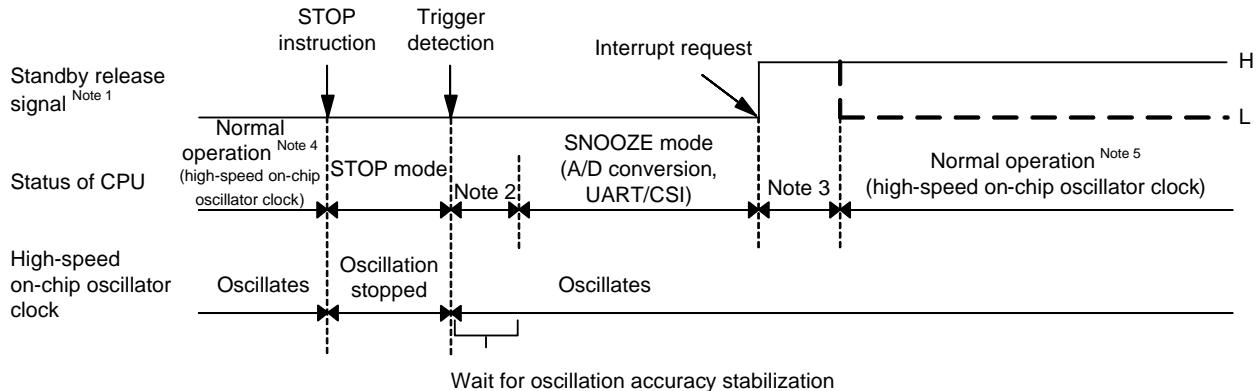
p = 00; q = 0

80, 100-pin products:

p = 00, 20; q = 0, 2

- (2) Timing diagram when the interrupt request signal is generated in the SNOOZE mode

**Figure 23 - 7 When the Interrupt Request Signal is Generated in the SNOOZE Mode**



**Note 1.** For details of the standby release signal, see **Figure 21 - 1**.

**Note 2.** Transition time from STOP mode to SNOOZE mode

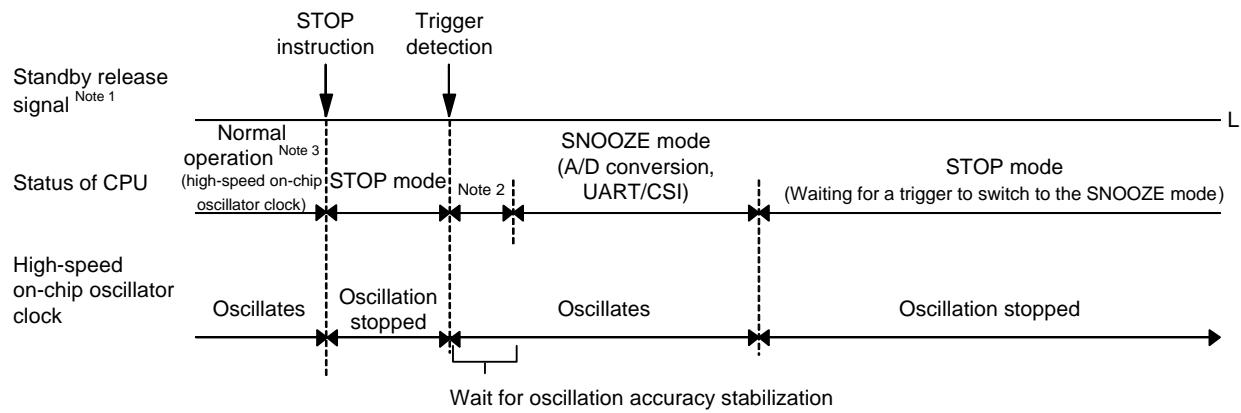
**Note 3.** Transition time from SNOOZE mode to normal operation

**Note 4.** Enable the SNOOZE mode (AWC = 1 or SWC = 1) immediately before switching to the STOP mode.

**Note 5.** Be sure to release the SNOOZE mode (AWC = 0 or SWC = 0) immediately after return to the normal operation.

- (3) Timing diagram when the interrupt request signal is not generated in the SNOOZE mode

**Figure 23 - 8 When the Interrupt Request Signal is not Generated in the SNOOZE Mode**



**Note 1.** For details of the standby release signal, see **Figure 21 - 1**.

**Note 2.** Transition time from STOP mode to SNOOZE mode

**Note 3.** Enable the SNOOZE mode (AWC = 1 or SWC = 1) immediately before switching to the STOP mode.

**Remark** For details of the SNOOZE mode function, see **CHAPTER 14 A/D CONVERTER** and **CHAPTER 17 SERIAL ARRAY UNIT**.

## CHAPTER 24 RESET FUNCTION

The following seven operations are available to generate a reset signal.

- (1) External reset input via RESET pin
- (2) Internal reset by watchdog timer program loop detection
- (3) Internal reset by comparison of supply voltage and detection voltage of power-on-reset (POR) circuit
- (4) Internal reset by comparison of supply voltage of the voltage detector (LVD) and detection voltage
- (5) Internal reset by execution of illegal instruction Note
- (6) Internal reset by RAM parity error
- (7) Internal reset by illegal-memory access

External and internal resets start program execution from the address at 00000H and 00001H when the reset signal is generated.

A reset is effected when a low level is input to the RESET pin, the watchdog timer overflows, or by POR and LVD circuit voltage detection, execution of illegal instruction Note, RAM parity error or illegal-memory access, and each item of hardware is set to the status shown in Table 24 - 1.

**Note** The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution not issued by emulation with the in-circuit emulator or on-chip debug emulator.

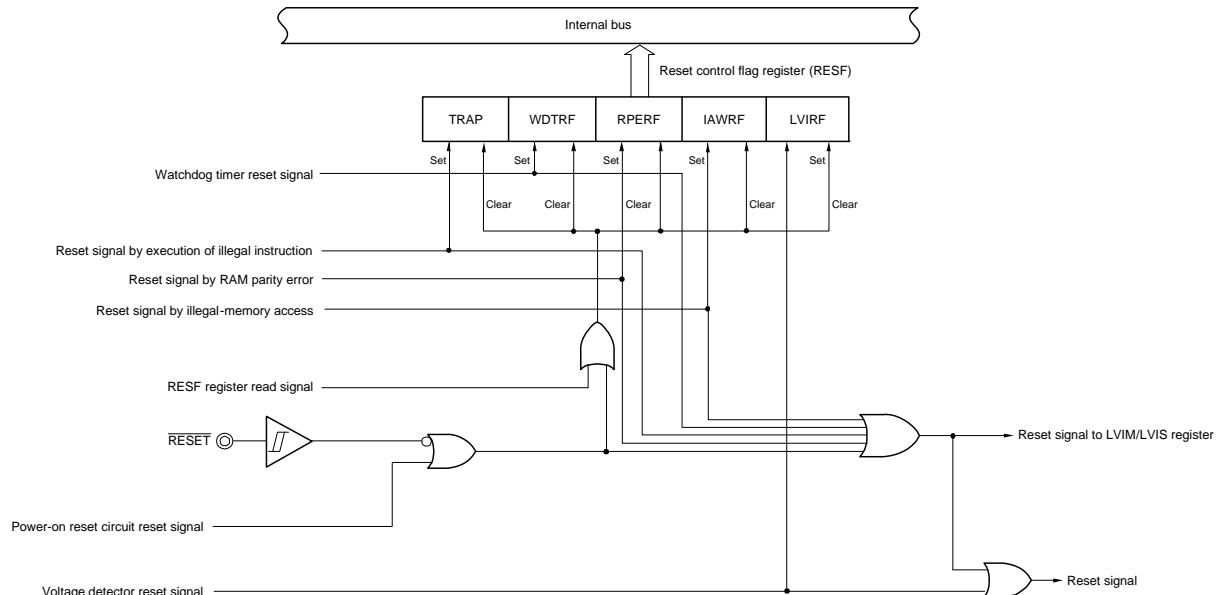
**Caution 1. For an external reset, input a low level for 10  $\mu$ s or more to the RESET pin.**

To perform an external reset upon power application, input a low level to the RESET pin, turn power on, continue to input a low level to the pin for 10  $\mu$ s or more within the operating voltage range shown in 34.4 or 35.4 AC Characteristics, and then input a high level to the pin.

**Caution 2. During reset input, the X1 clock, XT1 clock, high-speed on-chip oscillator clock, and low-speed on-chip oscillator clock oscillating. External main system clock input and external subsystem clock input become invalid.**

**Caution 3. The port pins become the following state because each SFR and 2nd SFR are initialized after reset.**

- P40: High-impedance during the external reset period or reset period by the POR. High level during other types of reset or after receiving a reset signal (connected to the on-chip pull-up resistance).
- P130: Low level during the reset period or after receiving a reset signal.
- Ports other than P40 and P130: High-impedance during the reset period or after receiving a reset signal.

**Figure 24 - 1 Block Diagram of Reset Function**

**Caution** An LVD circuit internal reset does not reset the LVD circuit.

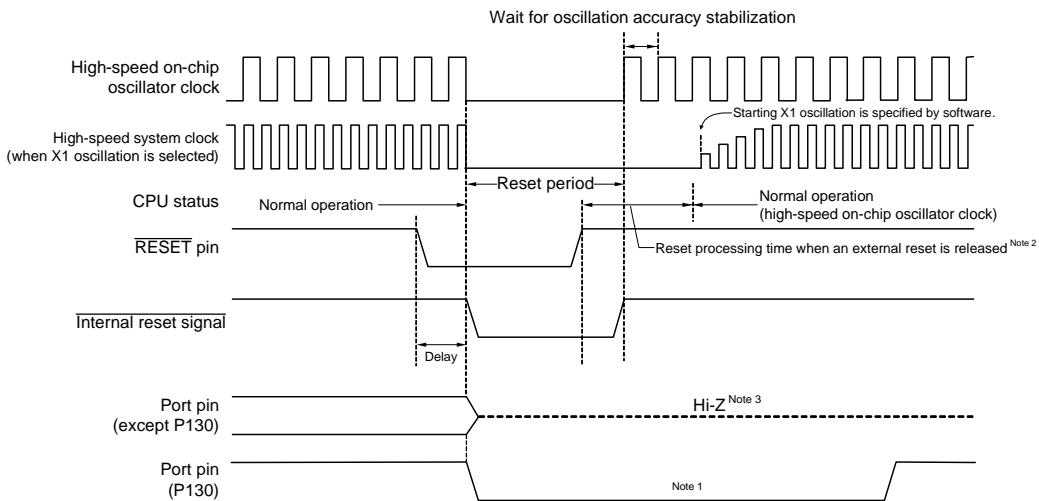
**Remark 1.** LVIM: Voltage detection register

**Remark 2.** LVIS: Voltage detection level register

## 24.1 Timing of Reset Operation

This LSI is reset by input of the low level on the **RESET** pin and released from the reset state by input of the high level on the **RESET** pin. After reset processing, execution of the program with the high-speed on-chip oscillator clock as the operating clock starts.

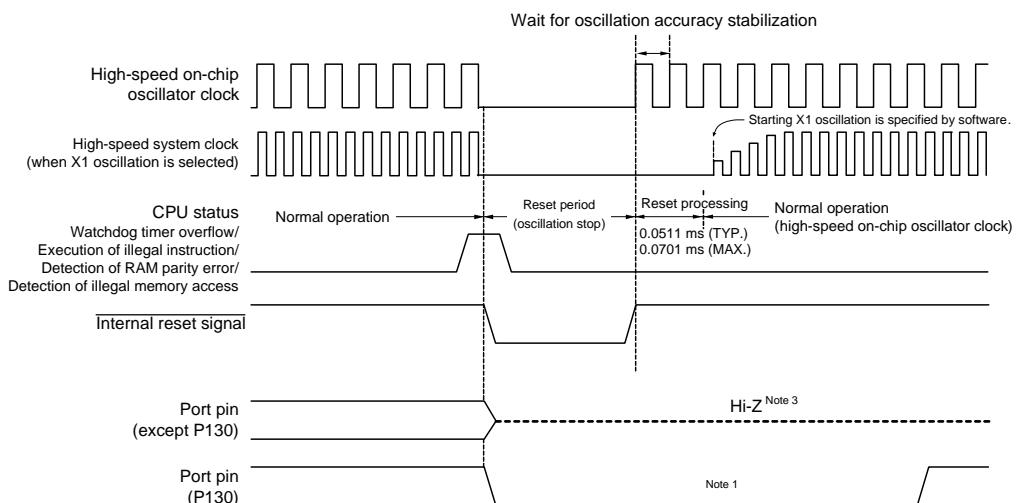
**Figure 24 - 2 Timing of Reset by **RESET** Input**



(Notes and Caution are listed on the next page.)

Release from the reset state is automatic in the case of a reset due to a watchdog timer overflow, execution of an illegal instruction, detection of a RAM parity error, or detection of illegal memory access. After reset processing, program execution starts with the high-speed on-chip oscillator clock as the operating clock.

**Figure 24 - 3 Timing of Reset Due to Watchdog Timer Overflow, Execution of Illegal Instruction, Detection of RAM Parity Error, or Detection of Illegal Memory Access**



(Notes and Caution are listed on the next page.)

**Note 1.** When P130 is set to high-level output before reset is effected, the output signal of P130 can be dummy-output as a reset signal to an external device, because P130 outputs a low level when reset is effected. To release a reset signal to an external device, set P130 to high-level output by software.

**Note 2.** Reset times (times for release from the external reset state)

After the first release of the POR: 0.672 ms (TYP.), 0.832 ms (MAX.) when the LVD is in use.

0.399 ms (TYP.), 0.519 ms (MAX.) when the LVD is off.

After the second release of the POR: 0.531 ms (TYP.), 0.675 ms (MAX.) when the LVD is in use.

0.259 ms (TYP.), 0.362 ms (MAX.) when the LVD is off.

After power is supplied, a voltage stabilization waiting time of about 0.99 ms (TYP.) and up to 2.30 ms (MAX.) is required before reset processing starts after release of the external reset.

**Note 3.** The state of P40 is as follows.

- High-impedance during the external reset period or reset period by the POR.

- High level during other types of reset or after receiving a reset signal (connected to the on-chip pull-up resistance).

Reset by POR and LVD circuit supply voltage detection is automatically released when  $V_{DD} \geq V_{POR}$  or  $V_{DD} \geq V_{LVD}$  after the reset. After reset processing, execution of the program with the high-speed on-chip oscillator clock as the operating clock starts. For details, see **CHAPTER 25 POWER-ON-RESET CIRCUIT** or **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark**  $V_{POR}$ : POR power supply rise detection voltage

$V_{LVD}$ : LVD detection voltage

**Table 24 - 1 Operation Statuses During Reset Period**

Item	During Reset Period	
System clock	Clock supply to the CPU is stopped.	
Main system clock	fiH	Operation stopped
	fx	Operation stopped (the X1 and X2 pins are input port mode)
	fEX	Clock input invalid (the pin is input port mode)
Subsystem clock	fXT	Operation stopped (the XT1 and XT2 pins are input port mode)
	fEXS	Clock input invalid (the pin is input port mode)
fil	Operation stopped	
CPU		
Code flash memory	Operation stopped	
Data flash memory	Operation stopped	
RAM	Operation stopped	
Port (latch)	High impedance Note 2	
Timer array unit	Operation stopped	
Timer RJ		
Timer RD		
Timer RG		
Real-time clock (RTC)		
12-bit Interval timer		
Watchdog timer		
Clock output/buzzer output		
A/D converter		
D/A converter Note 1		
Comparator Note 1		
Serial array unit (SAU)		
Serial interface (IICA)		
Data transfer controller (DTC)		
Power-on-reset function	Detection operation possible	
Voltage detection function	Operation is possible in the case of an LVD reset and stopped in the case of other types of reset.	
External interrupt	Operation stopped	
Key interrupt function		
CRC operation function	High-speed CRC	
	General-purpose CRC	
Illegal-memory access detection function		
RAM parity error detection function		
RAM guard function		
SFR guard function		

**Note 1.** Only for products with 96 KB or more code flash memory.

**Note 2.** P40 and P130 become the following state.

- P40: High-impedance during the external reset period or reset period by the POR. High level during other types of reset (connected to the on-chip pull-up resistance).
- P130: Low level during the reset period

**Remark** fiH: High-speed on-chip oscillator clock

fx: X1 oscillation clock

fEX: External main system clock

fXT: XT1 oscillation clock

fEXS: External subsystem clock

fil: Low-speed on-chip oscillator clock

**Table 24 - 2 Hardware Statuses After Reset Acknowledgment**

Hardware		After Reset Acknowledgment Note
Program counter (PC)		The contents of the reset vector table (00000H, 00001H) are set.
Stack pointer (SP)		Undefined
Program status word (PSW)		06H
RAM	Data memory	Undefined
	General-purpose registers	Undefined

**Note** During reset signal generation or oscillation stabilization time wait, only the PC contents among the hardware statuses become undefined. All other hardware statuses remain unchanged after reset.

**Remark** For the state of the special function register (SFR) after receiving a reset signal, see **3.1.4 Special function register (SFR) area** and **3.1.5 Extended special function register (2nd SFR: 2nd Special Function Register) area**.

## 24.2 Register for Confirming Reset Source

### 24.2.1 Reset control flag register (RESF)

Many internal reset generation sources exist in the RL78 microcontroller. The reset control flag register (RESF) is used to store which source has generated the reset request.

The RESF register can be read by an 8-bit memory manipulation instruction.

RESET input, reset by power-on-reset (POR) circuit, and reading the RESF register clear TRAP, WDTRF, RPERF, IAWRF, and LVIRF flags. Accordingly, to judge the sources of the types of reset indicated by this register, store the value of the RESF register at a desired location in RAM and use the stored value to judge the source of a reset.

**Figure 24 - 4 Format of Reset control flag register (RESF)**

Address: FFFA8H      After reset: Undefined Note 1 R

Symbol	7	6	5	4	3	2	1	0
RESF	TRAP	0	0	WDTRF	0	RPERF	IAWRF	LVIRF
TRAP	Internal reset request by execution of illegal instruction Note 2							
0	Internal reset request is not generated, or the RESF register is cleared.							
1	Internal reset request is generated.							
WDTRF	Internal reset request by watchdog timer (WDT)							
0	Internal reset request is not generated, or the RESF register is cleared.							
1	Internal reset request is generated.							
RPERF	Internal reset request t by RAM parity							
0	Internal reset request is not generated, or the RESF register is cleared.							
1	Internal reset request is generated.							
IAWRF	Internal reset request t by illegal-memory access							
0	Internal reset request is not generated, or the RESF register is cleared.							
1	Internal reset request is generated.							
LVIRF	Internal reset request by voltage detector (LVD)							
0	Internal reset request is not generated, or the RESF register is cleared.							
1	Internal reset request is generated.							

**Note 1.** The value after reset varies depending on the reset source. See **Table 24 - 3**.

**Note 2.** The illegal instruction is generated when instruction code FFH is executed.

Reset by the illegal instruction execution not issued by emulation with the in-circuit emulator or on-chip debug emulator.

**Caution 1. Do not read data by a 1-bit memory manipulation instruction.**

**Caution 2. When enabling RAM parity error resets (RPERDIS = 0), be sure to initialize the used RAM area at data access or the used RAM area + 10 bytes at execution of instruction from the RAM area.**

Reset generation enables RAM parity error resets (RPERDIS = 0). For details, see 27.3.3 RAM parity error detection function.

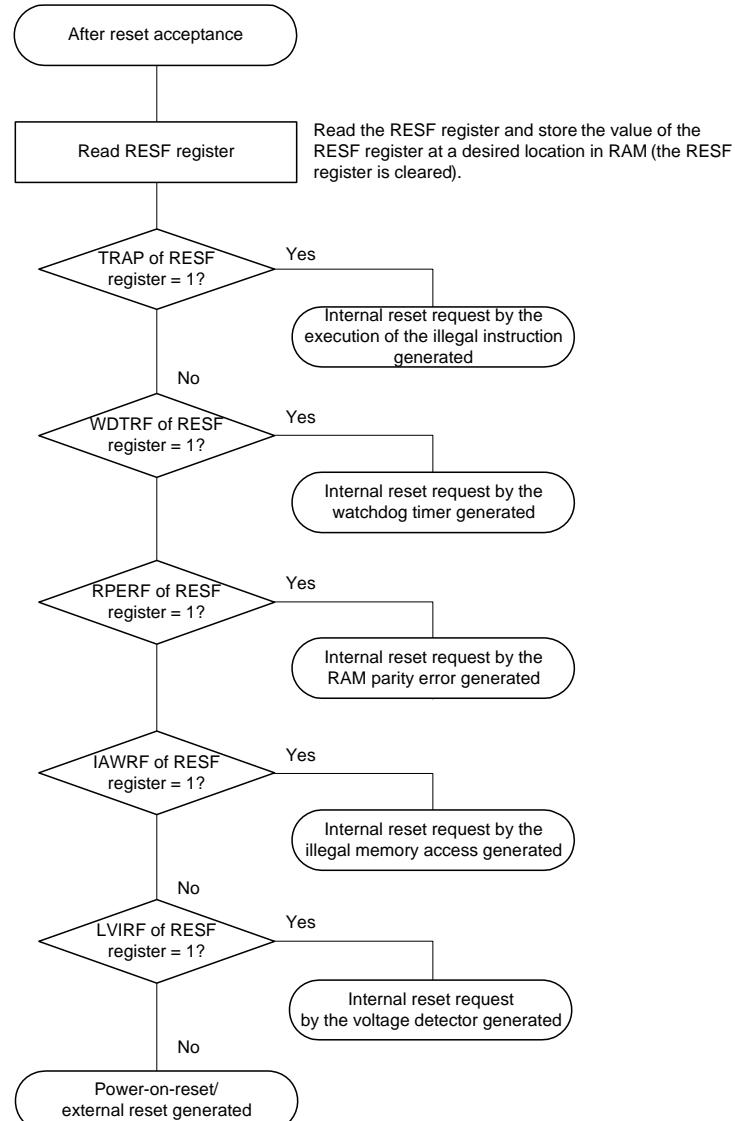
The status of the RESF register when a reset request is generated is shown in Table 24 - 3.

**Table 24 - 3 RESF Register Status When Reset Request Is Generated**

Reset Source Flag	RESET Input	Reset by POR	Reset by Execution of Illegal Instruction	Reset by WDT	Reset by RAM parity error	Reset by illegal- memory access	Reset by LVD
TRAP	Cleared (0)	Cleared (0)	Set (1)	Held	Held	Held	Held
WDTRF				Set (1)			
RPERF				Held	Set (1)		
IAWRF					Held	Set (1)	
LVIRF						Held	Set (1)

Figure 24 - 5 shows an example of procedure for checking reset source.

**Figure 24 - 5 Example of Procedure for Checking Reset Source**



**Caution** The flow described above is an example of the procedure for checking.

## CHAPTER 25 POWER-ON-RESET CIRCUIT

### 25.1 Functions of Power-on-reset Circuit

The power-on-reset circuit (POR) has the following functions.

- Generates internal reset signal at power on.

The reset signal is released when the supply voltage (VDD) exceeds the detection voltage (VPOR). Note that the reset state must be retained until the operating voltage becomes in the range defined in **34.4 or 35.4 AC Characteristics**.

This is done by utilizing the voltage detection circuit or controlling the externally input reset signal.

- Compares supply voltage (VDD) and detection voltage (VPDR), generates internal reset signal when  $VDD < VPDR$ . Note that, after power is supplied, this LSI should be placed in the STOP mode, or in the reset state by utilizing the voltage detection circuit or externally input reset signal, before the operation voltage falls below the range defined in **34.4 or 35.4 AC Characteristics**. When restarting the operation, make sure that the operation voltage has returned within the range of operation.

**Caution** If an internal reset signal is generated in the power-on-reset circuit, the reset control flag register (RESF) is cleared.

**Remark 1.** The RL78 microcontroller incorporates multiple hardware functions that generate an internal reset signal.

A flag that indicates the reset source is located in the reset control flag register (RESF) for when an internal reset signal is generated by the watchdog timer (WDT), voltage-detector (LVD), illegal instruction execution, RAM parity error, or illegal-memory access. The RESF register is not cleared to 00H and the flag is set to 1 when an internal reset signal is generated by the watchdog timer (WDT), voltage-detector (LVD), illegal instruction execution, RAM parity error, or illegal-memory access.

For details of the RESF register, see **CHAPTER 24 RESET FUNCTION**.

**Remark 2.** VPOR: POR power supply rise detection voltage

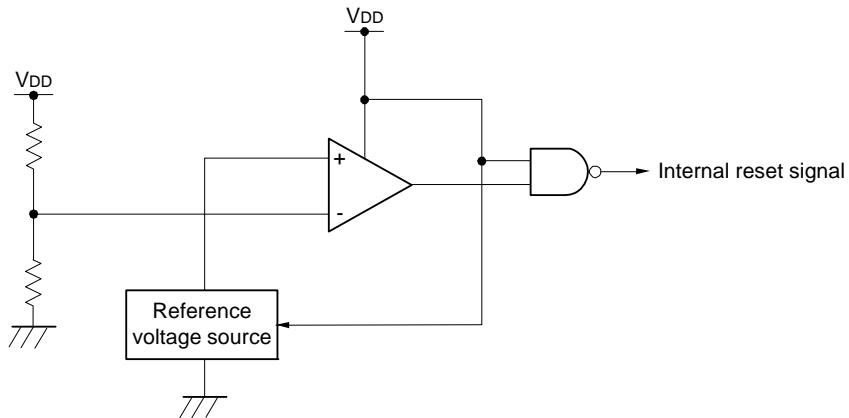
VPDR: POR power supply fall detection voltage

For details, see **34.6.5 or 35.6.5 POR circuit characteristics**.

## 25.2 Configuration of Power-on-reset Circuit

The block diagram of the power-on-reset circuit is shown in Figure 25 - 1.

Figure 25 - 1 Block Diagram of Power-on-reset Circuit

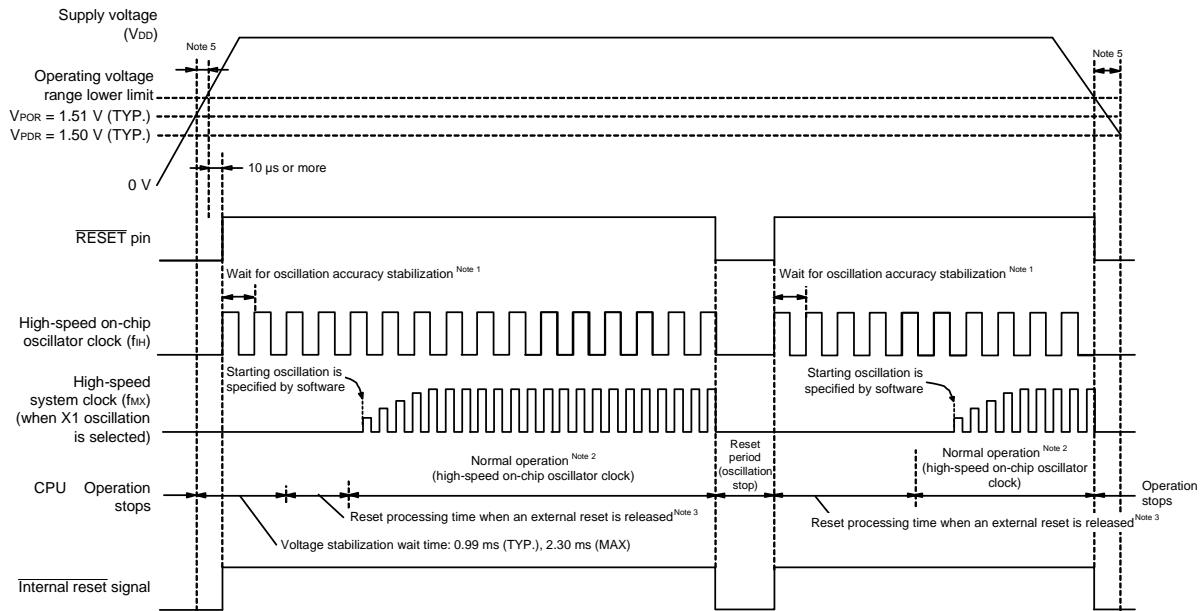


## 25.3 Operation of Power-on-reset Circuit

The timing of generation of the internal reset signal by the power-on-reset circuit and voltage detector is shown next.

**Figure 25 - 2 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (1/3)**

- (1) When using an external reset by the RESET pin



**Note 1.** The internal reset processing time includes the oscillation accuracy stabilization time of the high-speed on-chip oscillator clock.

**Note 2.** The high-speed on-chip oscillator clock and a high-speed system clock or subsystem clock can be selected as the CPU clock. To use the X1 clock, use the oscillation stabilization time counter status register (OSTC) to confirm the lapse of the oscillation stabilization time. To use the XT1 clock, use the timer function for confirmation of the lapse of the stabilization time.

**Note 3.** The time until normal operation starts includes the following reset processing time when the external reset is released (in the first reset processing following release from the POR state) after the RESET signal is driven high (1) as well as the voltage stabilization wait time after  $V_{POR}$  (1.51 V, typ.) is reached.

The reset processing time when the external reset is released is shown below.

In the first reset processing following release from the POR state:

$$\begin{aligned} &0.672\ \text{ms (TYP.)}, 0.832\ \text{ms (MAX.)} \text{ (when the LVD is in use)} \\ &0.399\ \text{ms (TYP.)}, 0.519\ \text{ms (MAX.)} \text{ (when the LVD is off)} \end{aligned}$$

**Note 4.** The reset processing time when the external reset is released in the second reset processing following release from the POR state is shown below.

In the second reset processing following release from the POR state:

$$\begin{aligned} &0.531\ \text{ms (TYP.)}, 0.675\ \text{ms (MAX.)} \text{ (when the LVD is in use)} \\ &0.259\ \text{ms (TYP.)}, 0.362\ \text{ms (MAX.)} \text{ (when the LVD is off)} \end{aligned}$$

**Note 5.** After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in **34.4** or **35.4 AC Characteristics**. This is done by controlling the externally input reset signal.

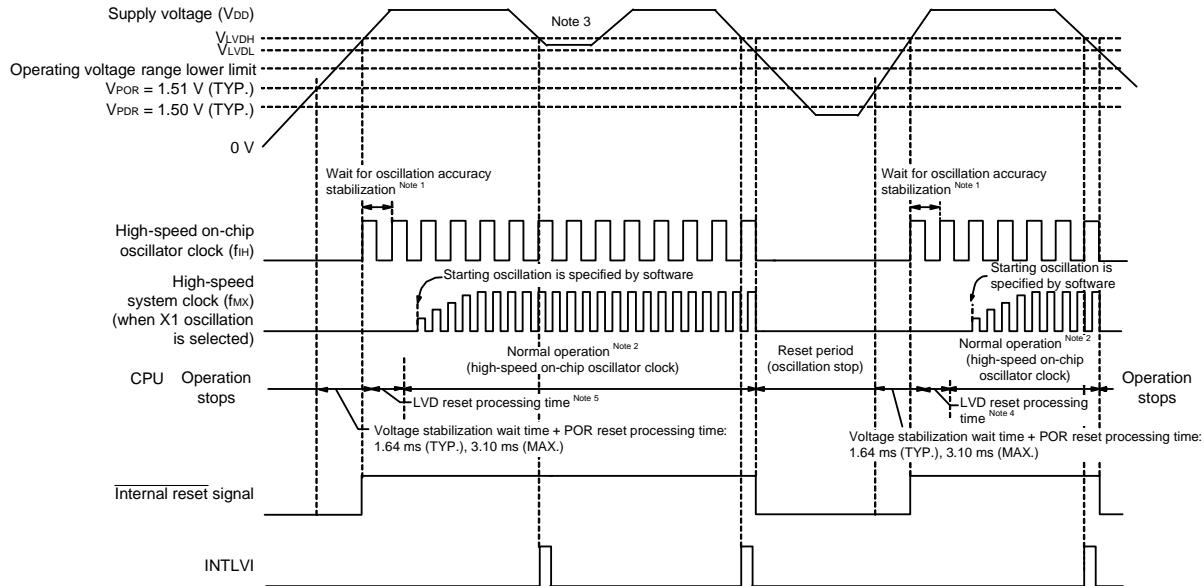
After power supply is turned off, this LSI should be placed in the STOP mode, or in the reset state by utilizing the voltage detection circuit or externally input reset signal, before the voltage falls below the operating range. When restarting the operation, make sure that the operation voltage has returned within the range of operation.

**Remark**  $V_{POR}$ : POR power supply rise detection voltage  
 $V_{PDR}$ : POR power supply fall detection voltage

**Caution** For power-on reset, be sure to use the externally input reset signal on the RESET pin when the LVD is off. For details, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Figure 25 - 3 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (2/3)**

- (2) LVD is interrupt & reset mode (option byte 000C1: LVIMDS1, LVIMDS0 = 1, 0)



**Note 1.** The internal reset processing time includes the oscillation accuracy stabilization time of the high-speed on-chip oscillator clock.

**Note 2.** The high-speed on-chip oscillator clock and a high-speed system clock or subsystem clock can be selected as the CPU clock. To use the X1 clock, use the oscillation stabilization time counter status register (OSTC) to confirm the lapse of the oscillation stabilization time. To use the XT1 clock, use the timer function for confirmation of the lapse of the stabilization time.

**Note 3.** After the interrupt request signal (INTLVI) is generated, the LVILV and LVIMD bits of the voltage detection level register (LVIS) are automatically set to 1. After INTLVI is generated, appropriate settings should be made according to **Figure 26 - 9 Setting Procedure for Operating Voltage Check and Reset** and **Figure 26 - 10 Setting Procedure for Initial Setting of Interrupt and Reset Mode**, taking into consideration that the supply voltage might return to the high voltage detection level (VLVDH) or higher without falling below the low voltage detection level (VLVDL).

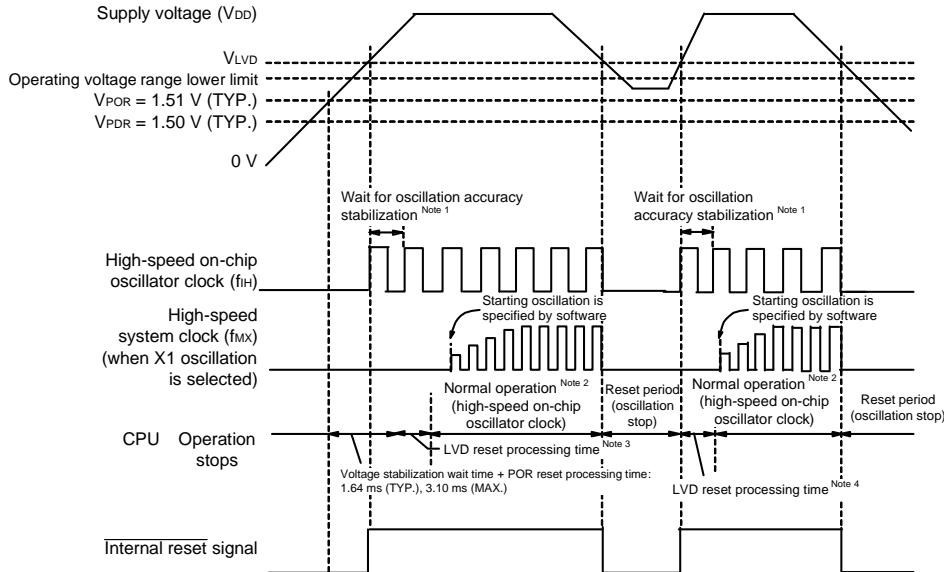
**Note 4.** The time until normal operation starts includes the following LVD reset processing time after the LVD detection level (VLVDH) is reached as well as the voltage stabilization wait + POR reset processing time after the VPOR (1.51 V, TYP.) is reached.

LVD reset processing time: 0 ms to 0.0701 ms (MAX.)

<b>Remark</b>	VLDH, VLVDL:	LVD detection voltage
	VPOR:	POR power supply rise detection voltage
	VPDR:	POR power supply fall detection voltage

**Figure 25 - 4 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (3/3)**

(3) LVD reset mode (option byte 000C1H: LVIMDS1, LVIMDS0 = 1, 1)



**Note 1.** The internal reset processing time includes the oscillation accuracy stabilization time of the high-speed on-chip oscillator clock.

**Note 2.** The high-speed on-chip oscillator clock and a high-speed system clock or subsystem clock can be selected as the CPU clock. To use the X1 clock, use the oscillation stabilization time counter status register (OSTC) to confirm the lapse of the oscillation stabilization time. To use the XT1 clock, use the timer function for confirmation of the lapse of the stabilization time.

**Note 3.** The time until normal operation starts includes the following LVD reset processing time after the LVD detection level (V<sub>LVD</sub>) is reached as well as the voltage stabilization wait + POR reset processing time after the V<sub>POR</sub> (1.51 V, TYP.) is reached.

LVD reset processing time: 0 ms to 0.0701 ms (MAX.)

**Note 4.** When the power supply voltage is below the lower limit for operation and the power supply voltage is then restored after an internal reset is generated only by the voltage detector (LVD), the following LVD reset processing time is required after the LVD detection level (V<sub>LVD</sub>) is reached.

LVD reset processing time: 0.0511 ms (TYP.), 0.0701 ms (MAX.)

**Remark 1.** V<sub>LVDH</sub>, V<sub>LVDL</sub>: LVD detection voltage

V<sub>POR</sub>: POR power supply rise detection voltage

V<sub>PDR</sub>: POR power supply fall detection voltage

**Remark 2.** When the LVD interrupt mode is selected (option byte 000C1H: LVIMD1 = 0, LVIMD0 = 1), the time until normal operation starts after power is turned on is the same as the time specified in Note 3 of Figure 25 - 4 (3).

## CHAPTER 26 VOLTAGE DETECTOR

### 26.1 Functions of Voltage Detector

The operation mode and detection voltages ( $V_{LVDH}$ ,  $V_{LVDL}$ ,  $V_{LVD}$ ) for the voltage detector is set by using the option byte (000C1H). The voltage detector (LVD) has the following functions.

- The LVD circuit compares the supply voltage ( $V_{DD}$ ) with the detection voltage ( $V_{LVDH}$ ,  $V_{LVDL}$ ,  $V_{LVD}$ ), and generates an internal reset or internal interrupt signal.
  - The detection level for the power supply detection voltage ( $V_{LVDH}$ ,  $V_{LVDL}$ ) can be selected by using the option byte as one of 14 levels (for details, see **CHAPTER 29 OPTION BYTE**).
  - Operable in STOP mode.
  - After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in **34.4** or **35.4** AC Characteristics. This is done by utilizing the voltage detector or controlling the externally input reset signal. After the power supply is turned off, this LSI should be placed in the STOP mode, or placed in the reset state by utilizing the voltage detector or controlling the externally input reset signal before the voltage falls below the operating range. The range of operating voltage varies with the setting of the user option byte (000C2H or 010C2H).
- (a) Interrupt & reset mode (option byte LVIMDS1, LVIMDS0 = 1, 0)  
The two detection voltages ( $V_{LVDH}$ ,  $V_{LVDL}$ ) are selected by the option byte 000C1H. The high-voltage detection level ( $V_{LVDH}$ ) is used for releasing resets and generating interrupts. The low-voltage detection level ( $V_{LVDL}$ ) is used for generating resets.
- (b) Reset mode (option byte LVIMDS1, LVIMDS0 = 1, 1)  
The detection voltage ( $V_{LVD}$ ) selected by the option byte 000C1H is used for triggering and ending resets.
- (c) Interrupt mode (option byte LVIMDS1, LVIMDS0 = 0, 1)  
The detection voltage ( $V_{LVD}$ ) selected by the option byte 000C1H is used for generating interrupts/reset release.

The reset and internal interrupt signals are generated in each mode as follows.

Interrupt & reset mode (LVIMDS1, LVIMDS0 = 1, 0)	Reset mode (LVIMDS1, LVIMDS0 = 1, 1)	Interrupt mode (LVIMDS1, LVIMDS0 = 0, 1)
Generates an releases an internal reset by detecting $V_{DD} < V_{LVDH}$ when the operating voltage falls, and releases an internal reset by detecting $V_{DD} < V_{LVDL}$ . Releases an internal reset by detecting $V_{DD} \geq V_{LVDH}$ .	Releases an internal reset by detecting $V_{DD} \geq V_{LVD}$ . Generates an internal reset by detecting $V_{DD} < V_{LVD}$ .	Retains the state of an internal reset by the LVD immediately after a reset until $V_{DD} \geq V_{LVD}$ . Releases the LVD internal reset by detecting $V_{DD} \geq V_{LVD}$ . Generates an interrupt request signal (INTLVI) by detecting $V_{DD} < V_{LVD}$ or $V_{DD} \geq V_{LVD}$ after the LVD internal reset is released.

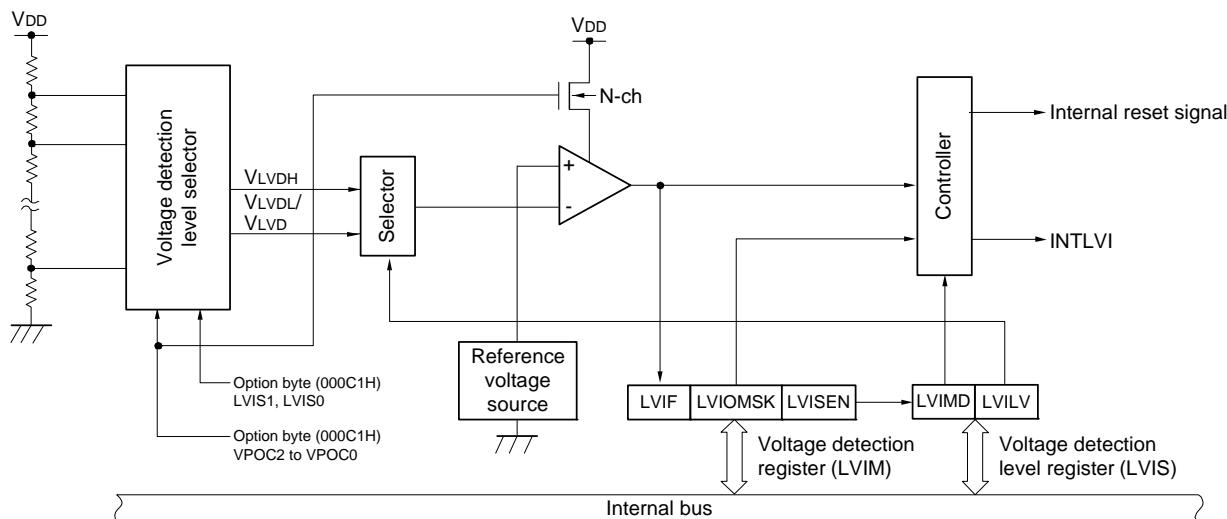
While the voltage detector is operating, whether the supply voltage is more than or less than the detection level can be checked by reading the voltage detection flag (LVIF: bit 0 of the voltage detection register (LVIM)).

Bit 0 (LVIRF) of the reset control flag register (RESF) is set to 1 if reset occurs. For details of the RESF register, see **CHAPTER 24 RESET FUNCTION**.

## 26.2 Configuration of Voltage Detector

The block diagram of the voltage detector is shown in Figure 26 - 1.

**Figure 26 - 1 Block Diagram of Voltage Detector**



## 26.3 Registers Controlling Voltage Detector

The voltage detector is controlled by the following registers.

- Voltage detection register (LVIM)
- Voltage detection level register (LVIS)

### 26.3.1 Voltage detection register (LVIM)

This register is used to specify whether to enable or disable rewriting the voltage detection level register (LVIS), as well as to check the LVD output mask status.

This register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 26 - 2 Format of Voltage detection register (LVIM)**

Address: FFFA9H      After reset: 00H Note 1 R/W Note 2

Symbol      <7>      6      5      4      3      2      <1>      <0>

LVIM	LVISEN Note 3	0	0	0	0	0	LVIOMSK	LVIF
------	---------------	---	---	---	---	---	---------	------

LVISEN Note 3	Specification of whether to enable or disable rewriting the voltage detection level register (LVIS)
0	Disabling of rewriting the LVIS register (LVIOMSK = 0 (Mask of LVD output is invalid))
1	Enabling of rewriting the LVIS register Note 3 (LVIOMSK = 1 (Mask of LVD output is valid))

LVIOMSK	Mask status flag of LVD output
0	Mask of LVD output is invalid
1	Mask of LVD output is valid Note 4

LVIF	Voltage detection flag
0	Supply voltage ( $V_{DD}$ ) $\geq$ detection voltage ( $V_{LVD}$ ), or when LVD is off
1	Supply voltage ( $V_{DD}$ ) $<$ detection voltage ( $V_{LVD}$ )

**Note 1.** The reset value changes depending on the reset source.

If the LVIS register is reset by LVD, it is not reset but holds the current value. In other reset, LVISEN is cleared to 0.

**Note 2.** Bits 0 and 1 are read-only.

**Note 3.** LVISEN and LVIOMSK can only be set in the interrupt & reset mode (option byte LVIMDS1, LVIMDS0 = 1, 0).

Do not change the initial value in other modes.

**Note 4.** LVIOMSK bit is only automatically set to "1" when the interrupt & reset mode is selected (option byte LVIMDS1, LVIMDS0 = 1, 0) and reset or interrupt by LVD is masked.

- Period during LVISEN = 1
- Waiting period from the time when LVD interrupt is generated until LVD detection voltage becomes stable
- Waiting period from the time when the value of LVILV bit changes until LVD detection voltage becomes stable

### 26.3.2 Voltage detection level register (LVIS)

This register selects the voltage detection level.

This register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation input sets this register to 00H/01H/81H Note 1.

**Figure 26 - 3 Format of Voltage detection level register (LVIS)**

Address: FFFAAH After reset:00H/01H/81H Note 1R/W

Symbol	<7>	6	5	4	3	2	1	<0>
LVIS	LVIMD Note 2	0	0	0	0	0	0	LVILV Note 2
LVIMD Note 2		Operation mode of voltage detection						
0		Interrupt mode						
1		Reset mode						
LVILV Note 2		LVD detection level						
0		High-voltage detection level (VLVDH)						
1		Low-voltage detection level (VLVDL or VLVD)						

**Note 1.** The reset value changes depending on the reset source and the setting of the option byte.

This register is not cleared (00H) by LVD reset.

The generation of reset signal other than an LVD reset sets as follows.

- When option byte LVIMDS1, LVIMDS0 = 1, 0: 00H
- When option byte LVIMDS1, LVIMDS0 = 1, 1: 81H
- When option byte LVIMDS1, LVIMDS0 = 0, 1: 01H

**Note 2.** Writing "0" can only be allowed in the interrupt & reset mode (option byte LVIMDS1, LVIMDS0 = 1, 0). Do not set LVIMD and LVILV in other cases. The value is switched automatically when reset or interrupt is generated in the interrupt & reset mode.

**Caution 1. Rewrite the value of the LVIS register according to Figures 26 - 9 and 26 - 10.**

**Caution 2. Specify the LVD operation mode and detection voltage (VLVDH, VLVDL, VLVD) of each mode by using the option byte 000C1H. Figure 26 - 4 shows the format of the user option byte (000C1H/010C1H). For details about the option byte, see CHAPTER 29 OPTION BYTE.**

**Figure 26 - 4 Format of User Option Byte (000C1H/010C1H) (1/2)**Address: 000C1H/010C1HNote

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD setting (interrupt & reset mode)

Detection voltage		Option byte Setting Value						Mode setting	
VLVDH		VLVDL	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	LVIMDS1	
Rising edge	Falling edge	Falling edge						LVIMDS1	LVIMDS0
1.77 V	1.73 V	1.63 V	0	0	0	1	0	1	0
1.88 V	1.84 V					0	1		
2.92 V	2.86 V					0	0		
1.98 V	1.94 V	1.84 V	0	1	1	1	0	0	1
2.09 V	2.04 V					0	1		
3.13 V	3.06 V					0	0		
2.61 V	2.55 V	2.45 V	1	0	1	1	0	0	0
2.71 V	2.65 V					0	1		
3.75 V	3.67 V					0	0		
2.92 V	2.86 V	2.75 V	1	1	1	1	0	0	1
3.02 V	2.96 V					0	1		
4.06 V	3.98 V					0	0		
—		Settings other than the above are prohibited							

- LVD setting (reset mode)

Detection voltage		Option byte Setting Value						Mode setting	
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	LVIMDS1		LVIMDS0
Rising edge	Falling edge						LVIMDS1	LVIMDS0	
1.67 V	1.63 V	0	0	0	1	1	1	1	1
1.77 V	1.73 V				0	1			
1.88 V	1.84 V				1	1			
1.98 V	1.94 V	0	1	1	1	0	0	1	0
2.09 V	2.04 V				0	1			
2.50 V	2.45 V				1	0			
2.61 V	2.55 V	1	0	1	1	0	0	1	0
2.71 V	2.65 V				1	0			
2.81 V	2.75 V				1	0			
2.92 V	2.86 V	1	1	1	1	1	0	1	0
3.02 V	2.96 V				1	1			
3.13 V	3.06 V				0	1			
3.75 V	3.67 V	1	0	0	1	0	0	0	0
4.06 V	3.98 V				1	1			
—		Settings other than the above are prohibited							

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Remark 1.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 2.** The detection voltage is a TYP. value. For details, see **34.6.6 or 35.6.6 LVD circuit characteristics**.

(Cautions are listed on the next page.)

**Figure 26 - 4 Format of User Option Byte (000C1H/010C1H) (2/2)**Address: 000C1H/010C1HNote

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD setting (interrupt mode)

Detection voltage		Option byte Setting Value							
VLVD		0	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	Mode setting	
Rising edge	Falling edge							LVIMDS1	LVIMDS0
1.67 V	1.63 V		0	0	0	1	1	0	1
1.77 V	1.73 V		0	0	1	0	0		
1.88 V	1.84 V		0	1	1	1	1		
1.98 V	1.94 V		0	1	1	0	0		
2.09 V	2.04 V		0	1	0	1	1		
2.50 V	2.45 V		1	0	1	1	1		
2.61 V	2.55 V		1	0	1	0	0		
2.71 V	2.65 V		1	0	0	1	1		
2.81 V	2.75 V		1	1	1	1	1		
2.92 V	2.86 V		1	1	1	0	0		
3.02 V	2.96 V		1	1	0	1	1		
3.13 V	3.06 V		0	1	0	0	0		
3.75 V	3.67 V		1	0	0	0	0		
4.06 V	3.98 V		1	1	0	0	0		
—		Settings other than the above are prohibited							

Detection voltage		Option byte Setting Value							
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	Mode setting		
Rising edge	Falling edge						LVIMDS1	LVIMDS0	
—	—	1	×	×	×	×	×	1	
—		Settings other than the above are prohibited							

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Caution 1. Set bit 4 to 1.**

**Caution 2. After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in 29.4 or 30.4 AC Characteristics. This is done by utilizing the voltage detection circuit or controlling the externally input reset signal. After the power supply is turned off, this LSI should be placed in the STOP mode, or placed in the reset state by utilizing the voltage detection circuit or controlling the externally input reset signal, before the voltage falls below the operating range. The range of operating voltage varies with the setting of the user option byte (000C2H or 010C2H).**

**Remark 1.** x: Don't care

**Remark 2.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 3.** The detection voltage is a TYP. value. For details, see **34.6.6** or **35.6.6 LVD circuit characteristics**.

## 26.4 Operation of Voltage Detector

### 26.4.1 When used as reset mode

Specify the operation mode (the reset mode (LVIMDS1, LVIMDS0 = 1, 1)) and the detection voltage (VLVD) by using the option byte 000C1H.

The operation is started in the following initial setting state when the reset mode is set.

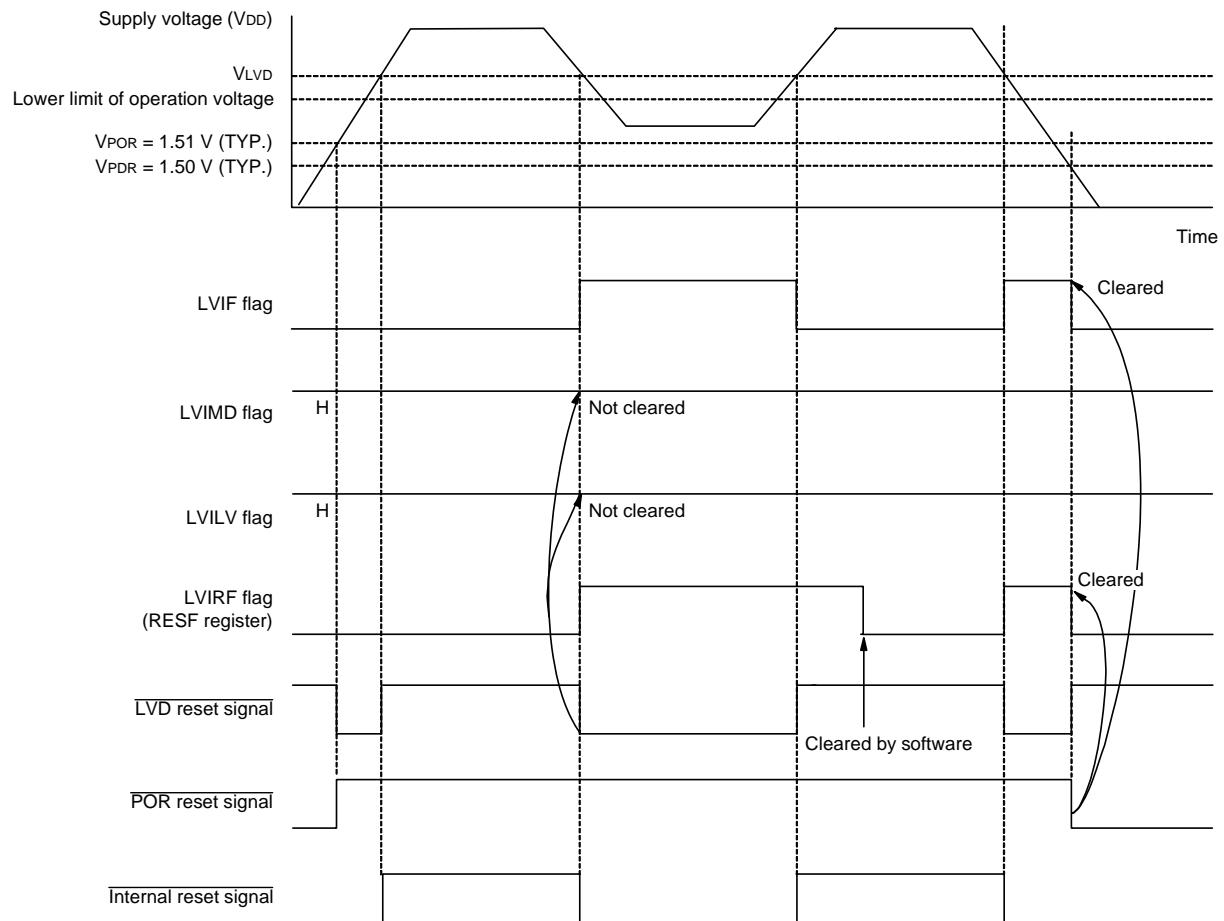
- Bit 7 (LVISEN) of the voltage detection register (LVIM) is set to 0 (disable rewriting of voltage detection level register (LVIS))
- The initial value of the voltage detection level select register (LVIS) is set to 81H.
- Bit 7 (LVIMD) is 1 (reset mode).
- Bit 0 (LVILV) is 1 (low-voltage detection level: VLVD).

- Operation in LVD reset mode

In the reset mode (option byte LVIMDS1, LVIMDS0 = 1, 1), the state of an internal reset by LVD is retained until the supply voltage (VDD) exceeds the voltage detection level (VLVD) after power is supplied. The internal reset is released when the supply voltage (VDD) exceeds the voltage detection level (VLVD).

At the fall of the operating voltage, an internal reset by LVD is generated when the supply voltage (VDD) falls below the voltage detection level (VLVD).

Figure 26 - 5 shows the timing of the internal reset signal generated in the LVD reset mode.

**Figure 26 - 5 Timing of Voltage Detector Internal Reset Signal Generation (Option Byte LVIMDS1, LVIMDS0 = 1, 1)**

**Remark**     $V_{POR}$ : POR power supply rise detection voltage  
 $V_{PDR}$ : POR power supply fall detection voltage

### 26.4.2 When used as interrupt mode

Specify the operation mode (the interrupt mode (LVIMDS1, LVIMDS0 = 0, 1)) and the detection voltage (VLVD) by using the option byte 000C1H.

The operation is started in the following initial setting state when the interrupt mode is set.

- Bit 7 (LVISEN) of the voltage detection register (LVIM) is set to 0 (disable rewriting of voltage detection level register (LVIS))
- The initial value of the voltage detection level select register (LVIS) is set to 01H.
- Bit 7 (LVIMD) is 0 (interrupt mode).
- Bit 0 (LVILV) is 1 (low-voltage detection level: VLVD).

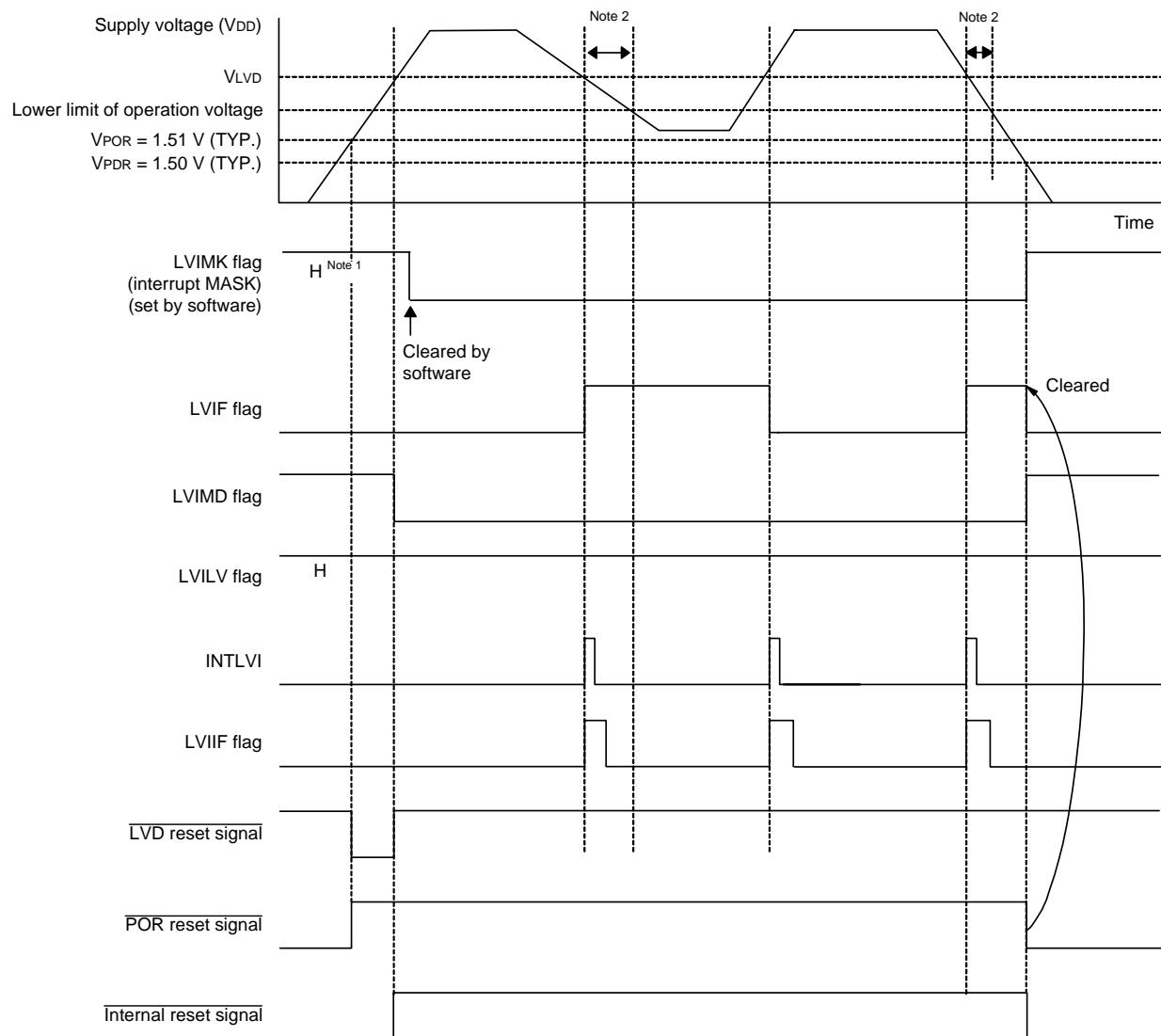
- Operation in LVD interrupt mode

In interrupt mode (LVIMDS1 and LVIMDS0 = 0 and 1 in the option byte), the state of an internal reset by the LVD is retained immediately after a reset until the supply voltage (VDD) exceeds the voltage detection level (VLVD). The LVD internal reset is released when the supply voltage (VDD) exceeds the voltage detection level (VLVD).

After the LVD internal reset is released, an interrupt request signal (INTLVI) by the LVD is generated when the supply voltage (VDD) exceeds the voltage detection level (VLVD). When the voltage falls, this LSI should be placed in the STOP mode, or placed in the reset state by controlling the externally input reset signal, before the voltage falls below the operating voltage range defined in **34.4** or **35.4** AC Characteristics. When restarting the operation, make sure that the operation voltage has returned within the range of operation.

Figure 26 - 6 shows the timing of the interrupt request signal generated in the LVD interrupt mode.

**Figure 26 - 6 Timing of Voltage Detector Internal Interrupt Signal Generation  
(Option Byte LVIMDS1, LVIMDS0 = 0, 1)**



**Remark** VPOR: POR power supply rise detection voltage

VPDR: POR power supply fall detection voltage

### 26.4.3 When used as interrupt and reset mode

Specify the operation mode (the interrupt & reset (LVIMDS1, LVIMDS0 = 1, 0)) and the detection voltage (VLVDH, VLVDL) by using the option byte 000C1H.

The operation is started in the following initial setting state when the interrupt & reset mode is set.

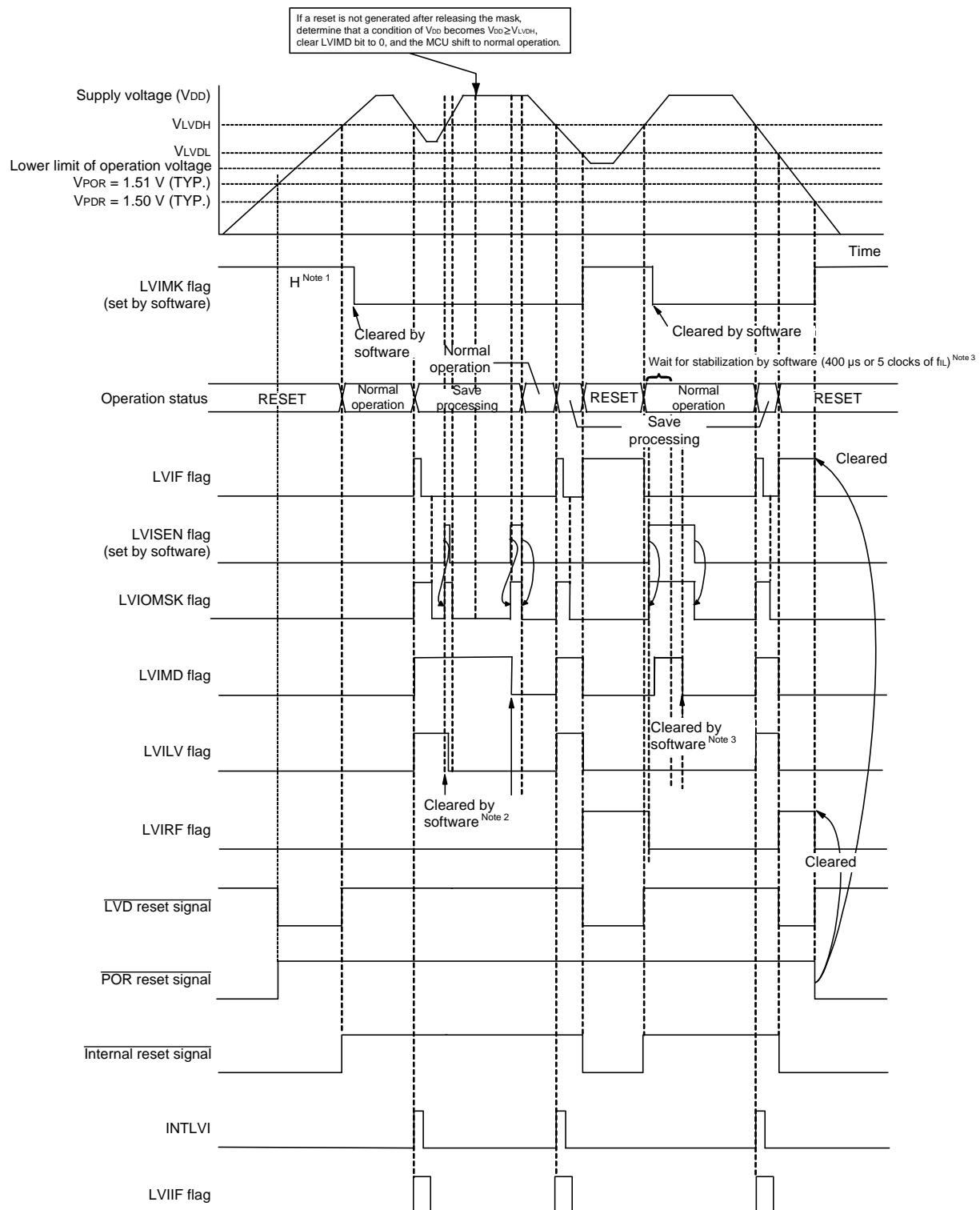
- Bit 7 (LVISEN) of the voltage detection register (LVIM) is set to 0 (disable rewriting of voltage detection level register (LVIS))
- The initial value of the voltage detection level select register (LVIS) is set to 00H.
- Bit 7 (LVIMD) is 0 (interrupt mode).
- Bit 0 (LVILV) is 0 (high-voltage detection level: VLVDH).

- Operation in LVD interrupt & reset mode

In the interrupt & reset mode (option byte LVIMDS1, LVIMDS0 = 1, 0), the state of an internal reset by LVD is retained until the supply voltage (VDD) exceeds the high-voltage detection level (VLVDH) after power is supplied. The internal reset is released when the supply voltage (VDD) exceeds the high-voltage detection level (VLVDH). An interrupt request signal by LVD (INTLVI) is generated and arbitrary save processing is performed when the supply voltage (VDD) falls below the high-voltage detection level (VLVDH). After that, an internal reset by LVD is generated when the supply voltage (VDD) falls below the low-voltage detection level (VLVDL). After INTLVI is generated, an interrupt request signal is not generated even if the supply voltage becomes equal to or higher than the high-voltage detection voltage (VLVDH) without falling below the low-voltage detection voltage (VLVDL). To use the LVD reset & interrupt mode, perform the processing according to **Figure 26 - 9 Setting Procedure for Operating Voltage Check and Reset** and **Figure 26 - 10 Setting Procedure for Initial Setting of Interrupt and Reset Mode**.

Figure 26 - 7 and 26 - 8 show the timing of the internal reset signal and interrupt signal generated in the LVD interrupt & reset mode.

**Figure 26 - 7 Timing of Voltage Detector Reset Signal and Interrupt Signal Generation  
(Option Byte LVIMDS1, LVIMDS0 = 1, 0) (1/2)**

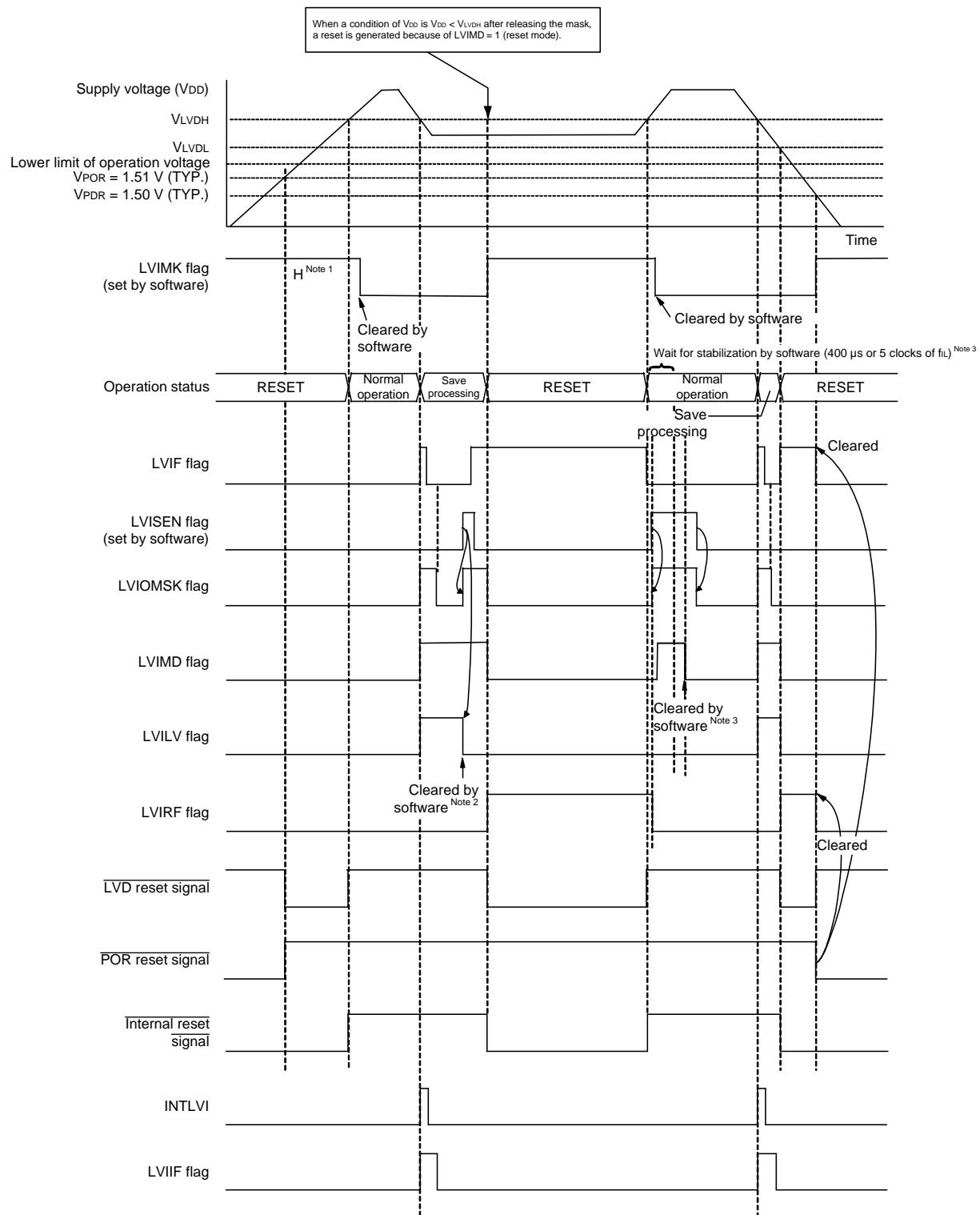


(Notes and Remark are listed on the next page.)

- Note 1.** The LVIMK flag is set to "1" by reset signal generation.
- Note 2.** After an interrupt is generated, perform the processing according to Figure 26 - 9 Setting Procedure for Operating Voltage Check and Reset in interrupt and reset mode.
- Note 3.** After a reset is released, perform the processing according to Figure 26 - 10 Setting Procedure for Initial Setting of Interrupt and Reset Mode in interrupt and reset mode.

**Remark** VPOR: POR power supply rise detection voltage  
VPOR: POR power supply fall detection voltage

**Figure 26 - 8 Timing of Voltage Detector Reset Signal and Interrupt Signal Generation  
(Option Byte LVIMDS1, LVIMDS0 = 1, 0) (2/2)**

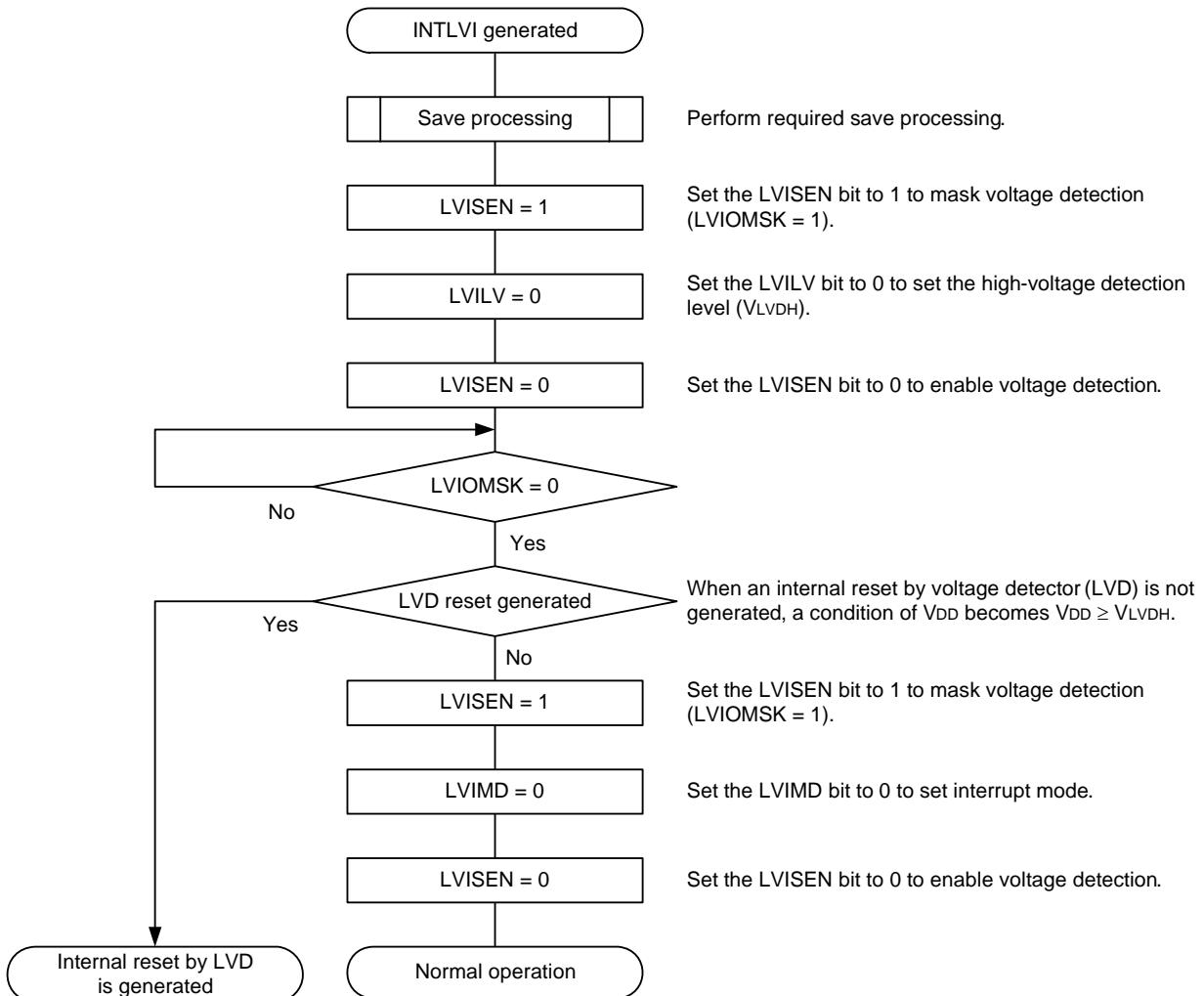


(Notes and Remark are listed on the next page.)

- Note 1.** The LVIMK flag is set to "1" by reset signal generation.
- Note 2.** After an interrupt is generated, perform the processing according to Figure 26 - 9 Setting Procedure for Operating Voltage Check and Reset in interrupt and reset mode.
- Note 3.** After a reset is released, perform the processing according to Figure 26 - 10 Setting Procedure for Initial Setting of Interrupt and Reset Mode in interrupt and reset mode.

**Remark** VPOR: POR power supply rise detection voltage  
VPDR: POR power supply fall detection voltage

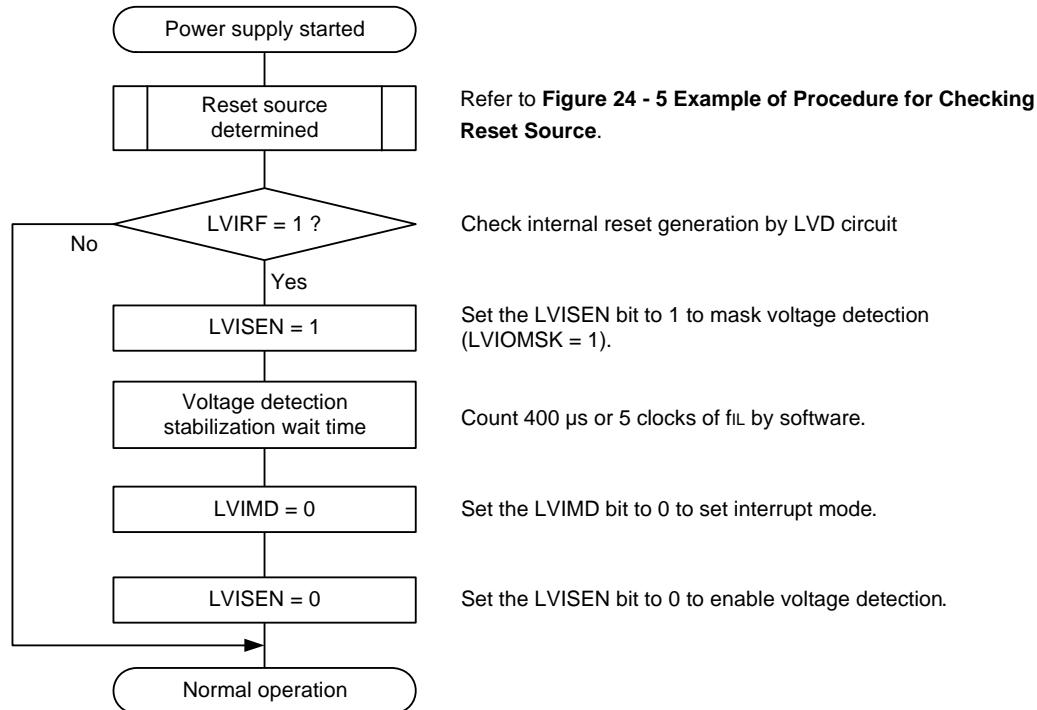
**Figure 26 - 9 Setting Procedure for Operating Voltage Check and Reset**



When setting an interrupt and reset mode (LVIMDS1, LVIMDS0 = 1, 0), voltage detection stabilization wait time for 400  $\mu$ s or 5 clocks of f<sub>L</sub> is necessary after LVD reset is released (LVIRF = 1). After waiting until voltage detection stabilizes, (0) clear the LVIMD bit for initialization. While voltage detection stabilization wait time is being counted and when the LVIMD bit is rewritten, set LVISEN to 1 to mask a reset or interrupt generation by LVD.

Figure 26 - 10 shows the procedure for Setting Procedure for Initial Setting of Interrupt and Reset Mode.

**Figure 26 - 10 Setting Procedure for Initial Setting of Interrupt and Reset Mode**



**Remark** f<sub>L</sub>: Low-speed on-chip oscillator clock frequency

## 26.5 Cautions for Voltage Detector

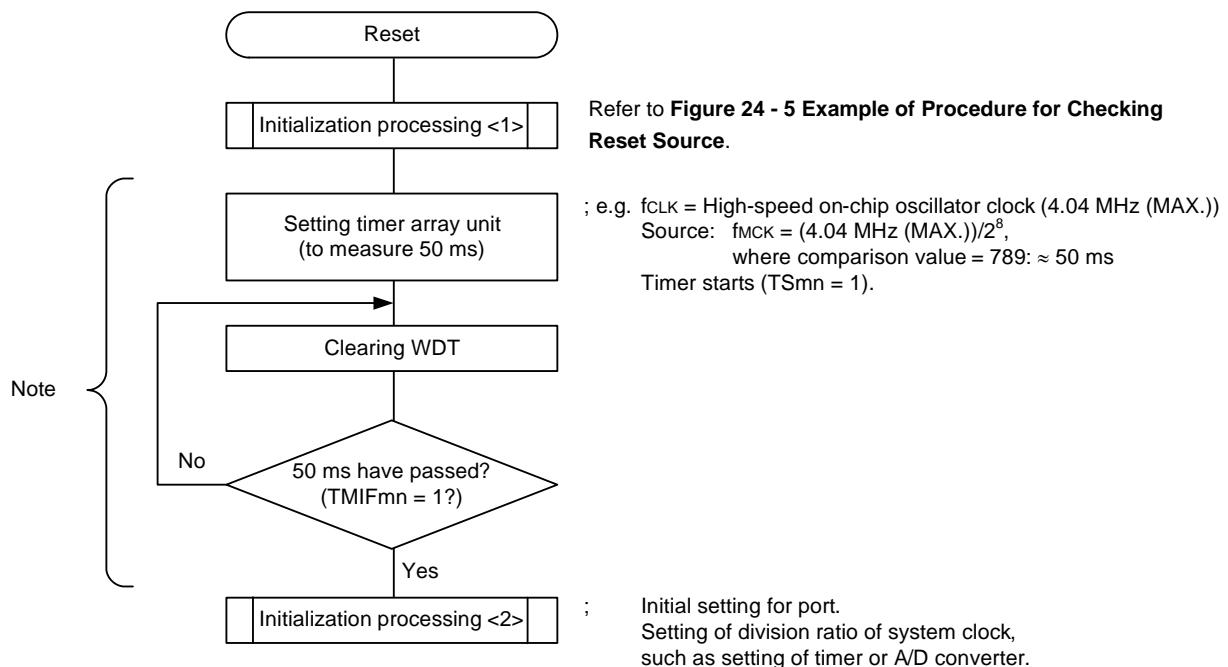
### (1) Voltage fluctuation when power is supplied

In a system where the supply voltage (VDD) fluctuates for a certain period in the vicinity of the LVD detection voltage, the system may be repeatedly reset and released from the reset status. In this case, the time from release of reset to the start of the operation of the microcontroller can be arbitrarily set by taking the following action.

<Action>

After releasing the reset signal, wait for the supply voltage fluctuation period of each system by means of a software counter that uses a timer, and then initialize the ports.

**Figure 26 - 11 Example of Software Processing If Supply Voltage Fluctuation is 50 ms or Less in Vicinity of LVD Detection Voltage**

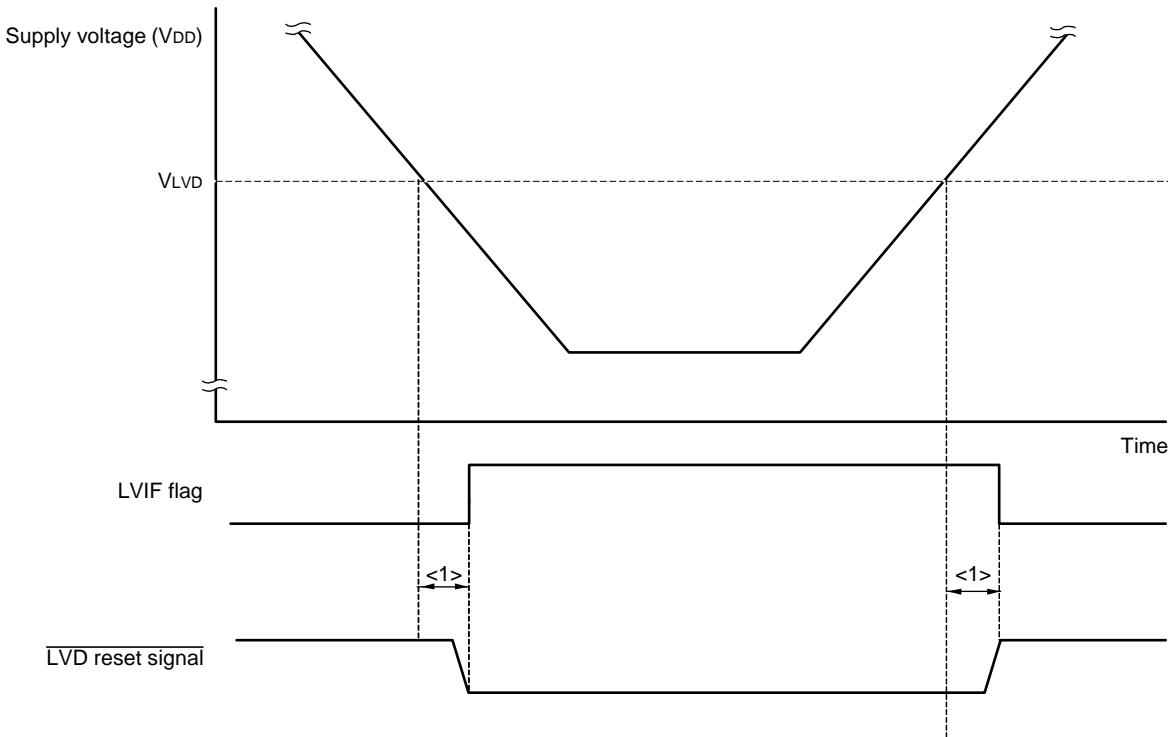


**Note** If reset is generated again during this period, initialization processing <2> is not started.

**Remark** m = 0, 1  
n = 0 to 3

- (4) Delay from the time LVD reset source is generated until the time LVD reset has been generated or released  
 There is some delay from the time supply voltage ( $V_{DD}$ ) < LVD detection voltage ( $V_{LVD}$ ) until the time LVD reset has been generated.  
 In the same way, there is also some delay from the time LVD detection voltage ( $V_{LVD}$ )  $\leq$  supply voltage ( $V_{DD}$ ) until the time LVD reset has been released (see **Figure 26 - 12**).

**Figure 26 - 12 Delay from the time LVD reset source is generated until the time LVD reset has been generated or released**



<1>: Detection delay (300  $\mu$ s (MAX.))

- (3) Power on when LVD is off

Use the external rest input via the RESET pin when the LVD is off.

For an external reset, input a low level for 10  $\mu$ s or more to the RESET pin. To perform an external reset upon power application, input a low level to the RESET pin, turn power on, continue to input a low level to the pin for 10  $\mu$ s or more within the operating voltage range shown in **34.4 or 35.4 AC Characteristics**, and then input a high level to the pin.

- (4) Operating voltage fall when LVD is off or LVD interrupt mode is selected

When the operating voltage falls with the LVD is off or with the LVD interrupt mode is selected, this LSI should be placed in the STOP mode, or placed in the reset state by controlling the externally input reset signal, before the voltage falls below the operating voltage range defined in **34.4 or 35.4 AC Characteristics**. When restarting the operation, make sure that the operation voltage has returned within the range of operation.

## CHAPTER 27 SAFETY FUNCTIONS

### 27.1 Overview of Safety Functions

The following safety functions are provided in the RL78/G14 to comply with the IEC60730 and IEC61508 safety standards.

These functions enable the microcontroller to self-diagnose abnormalities and stop operating if an abnormality is detected.

- (1) Flash memory CRC operation function (high-speed CRC, general-purpose CRC)

This detects data errors in the flash memory by performing CRC operations.

Two CRC functions are provided in the RL78/G14 that can be used according to the application or purpose of use.

- High-speed CRC: The CPU can be stopped and a high-speed check executed on its entire code flash memory area during the initialization routine.

- General CRC: This can be used for checking various data in addition to the code flash memory area while the CPU is running.

- (2) RAM parity error detection function

This detects parity errors when the RAM is read as data.

- (3) RAM guard function

This prevents RAM data from being rewritten when the CPU freezes.

- (4) SFR guard function

This prevents SFRs from being rewritten when the CPU freezes.

- (5) Invalid memory access detection function

This detects illegal accesses to invalid memory areas (such as areas where no memory is allocated and areas to which access is restricted).

- (6) Frequency detection function

This uses the timer array unit to perform a self-check of the CPU/peripheral hardware clock frequency.

- (7) A/D test function

This is used to perform a self-check of A/D converter by performing A/D conversion on the positive internal reference voltage, negative reference voltage, analog input channel (ANI), temperature sensor output, and internal reference voltage output.

- (8) Digital output signal level detection function for I/O pins

When the I/O pins are output mode, the output level of the pin can be read.

**Remark** Refer to the IEC60730/60335 self-test library application notes (R01AN1062, R01AN1296) for the RL78 MCU Series, for more information on usage examples of the safety functions required to comply with the IEC60730 and IEC61508 safety standards.

## 27.2 Registers Used by Safety Functions

The safety functions use the following registers:

Register	Each Function of Safety Function
• Flash memory CRC control register (CRC0CTL)	Flash memory CRC operation function (high-speed CRC)
• Flash memory CRC operation result register (PGCRCL)	
• CRC input register (CRCIN)	CRC operation function (general-purpose CRC)
• CRC data register (CRCD)	
• RAM parity error control register (RPECTL)	RAM parity error detection function
• Invalid memory access detection control register (IAWCTL)	RAM guard function
	SFR guard function
	Invalid memory access detection function
• Timer input select register 0 (TIS0)	Frequency detection function
• A/D test register (ADTES)	A/D test function
• Port mode select register (PMS)	Digital output signal level detection function for I/O pins

The content of each register is described in [27.3 Operation of Safety Functions](#).

## 27.3 Operation of Safety Functions

### 27.3.1 Flash memory CRC operation function (high-speed CRC)

The IEC60730 standard mandates the checking of data in the flash memory, and recommends using CRC to do it. The high-speed CRC provided in the RL78/G14 can be used to check the entire code flash memory area during the initialization routine. The high-speed CRC can be executed only when the program is allocated on the RAM and in the HALT mode of the main system clock.

The high-speed CRC performs an operation by reading 32-bit data per clock from the flash memory while stopping the CPU. This function therefore can finish a check in a shorter time (for example, 512 µs@32 MHz with 64-KB flash memory).

The CRC generator polynomial used complies with “ $X^{16} + X^{12} + X^5 + 1$ ” of CRC-16-CCITT.

The high-speed CRC operates in MSB first order from bit 31 to bit 0.

**Caution** The CRC operation result might differ during on-chip debugging because the monitor program is allocated.

**Remark** The operation result is different between the high-speed CRC and the general CRC, because the general CRC operates in LSB first order.

### 27.3.1.1 Flash memory CRC control register (CRC0CTL)

This register is used to control the operation of the high-speed CRC ALU, as well as to specify the operation range.

The CRC0CTL register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 1 Format of Flash memory CRC control register (CRC0CTL)**

Address: F02F0H		After reset:00H		R/W				
Symbol	<7>	6	5	4	3	2	1	0
CRC0CTL	CRC0EN	0	FEA5	FEA4	FEA3	FEA2	FEA1	FEA0
CRC0EN	Control of high-speed CRC ALU operation							
0	Stop the operation.							
1	Start the operation according to HALT instruction execution.							
FEA5	FEA4	FEA3	FEA2	FEA1	FEA0	High-speed CRC operation range		
0	0	0	0	0	0	00000H to 03FFBH (16 K - 4 bytes)		
0	0	0	0	0	1	00000H to 07FFBH (32 K - 4 bytes)		
0	0	0	0	1	0	00000H to 0BFFBH (48K - 4 bytes)		
0	0	0	0	1	1	00000H to 0FFFFBH (64K - 4 bytes)		
0	0	0	1	0	0	00000H to 13FFBH (80K - 4 bytes)		
0	0	0	1	0	1	00000H to 17FFBH (96K - 4 bytes)		
0	0	0	1	1	0	00000H to 1BFFBH (112K - 4 bytes)		
0	0	0	1	1	1	00000H to 1FFFFBH (128K - 4 bytes)		
0	0	1	0	0	0	00000H to 23FFBH (144K - 4 bytes)		
0	0	1	0	0	1	00000H to 27FFBH (160K - 4 bytes)		
0	0	1	0	1	0	00000H to 2BFFBH (176K - 4 bytes)		
0	0	1	0	1	1	00000H to 2FFFFBH (192K - 4 bytes)		
0	0	1	1	0	0	00000H to 33FFBH (208K - 4 bytes)		
0	0	1	1	0	1	00000H to 37FFBH (224K - 4 bytes)		
0	0	1	1	1	0	00000H to 3BFFBH (240K - 4 bytes)		
0	0	1	1	1	1	00000H to 3FFFFBH (256K - 4 bytes)		
0	1	0	0	0	0	00000H to 43FFBH (272 K-4 bytes)		
0	1	0	0	0	1	00000H to 47FFBH (288 K-4 bytes)		
0	1	0	0	1	0	00000H to 4BFFBH (304 K-4 bytes)		
0	1	0	0	1	1	00000H to 4FFFFBH (320 K-4 bytes)		
0	1	0	1	0	0	00000H to 53FFBH (336 K-4 bytes)		
0	1	0	1	0	1	00000H to 57FFBH (352 K-4 bytes)		
0	1	0	1	1	0	00000H to 5BFFBH (368 K-4 bytes)		
0	1	0	1	1	1	00000H to 5FFFFBH (384 K-4 bytes)		
0	1	1	0	0	0	00000H to 63FFBH (400 K-4 bytes)		
0	1	1	0	0	1	00000H to 67FFBH (416 K-4 bytes)		
0	1	1	0	1	0	00000H to 6BFFBH (432 K-4 bytes)		
0	1	1	0	1	1	00000H to 6FFFFBH (448 K-4 bytes)		
0	1	1	1	0	0	00000H to 73FFBH (464 K-4 bytes)		
0	1	1	1	0	1	00000H to 77FFBH (480 K-4 bytes)		
0	1	1	1	1	0	00000H to 7BFFBH (496 K-4 bytes)		
0	1	1	1	1	1	00000H to 7FFFFBH (512 K-4 bytes)		
Other than the above						Setting prohibited		

**Remark** Input the expected CRC operation result value to be used for comparison in the lowest 4 bytes of the flash memory. Note that the operation range will thereby be reduced by 4 bytes.

### 27.3.1.2 Flash memory CRC operation result register (PGCRCL)

This register is used to store the high-speed CRC operation results.

The PGCRCL register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Figure 27 - 2 Format of Flash memory CRC operation result register (PGCRCL)**

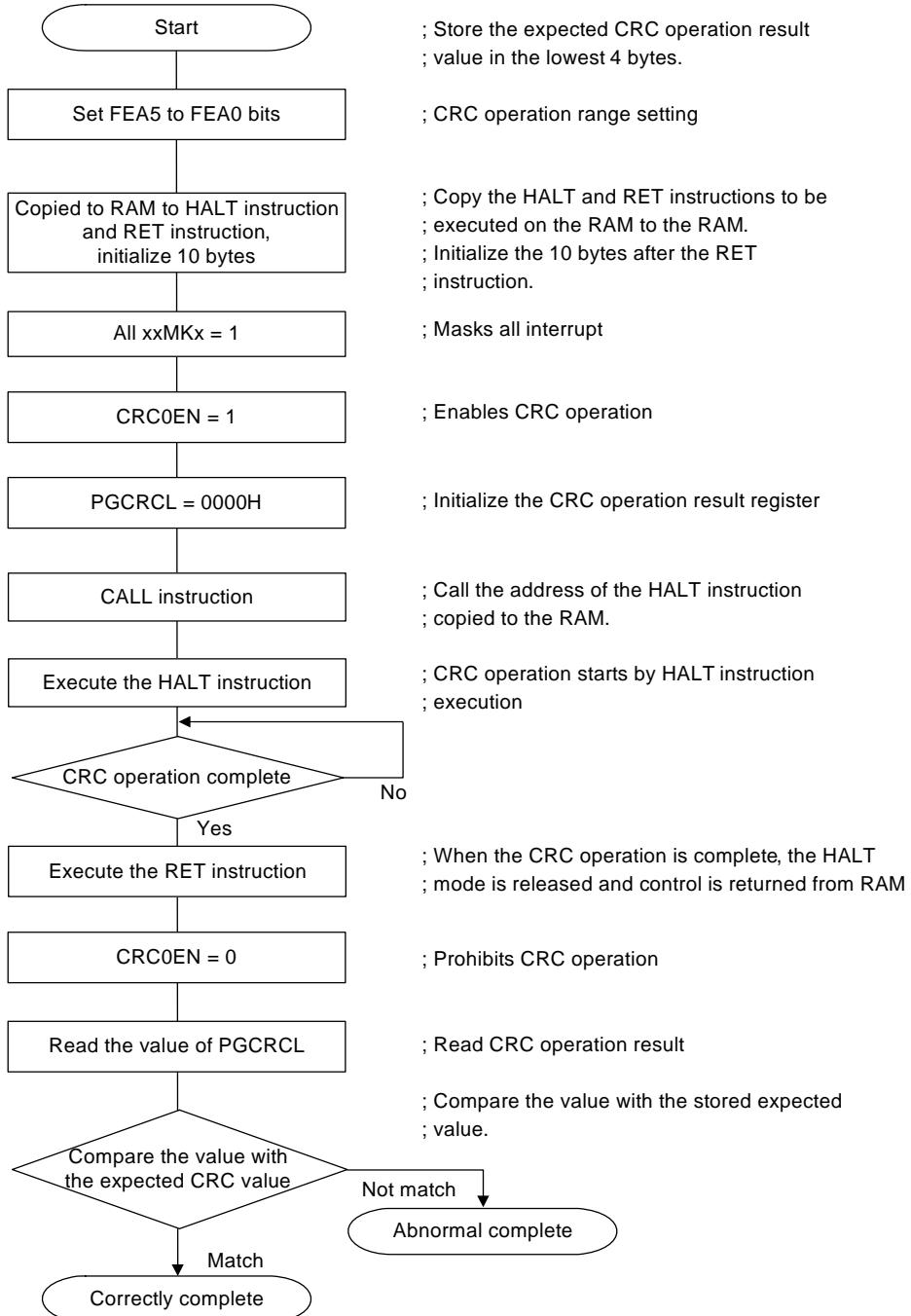
Address: F02F2H      After reset: 0000H      R/W

Symbol	15	14	13	12	11	10	9	8
PGCRCL	PGCRC15	PGCRC14	PGCRC13	PGCRC12	PGCRC11	PGCRC10	PGCRC9	PGCRC8
	7	6	5	4	3	2	1	0
	PGCRC7	PGCRC6	PGCRC5	PGCRC4	PGCRC3	PGCRC2	PGCRC1	PGCRC0
PGCRC15 to 0	High-speed CRC operation results							
0000H to FFFFH	Store the high-speed CRC operation results.							

**Caution** The PGCRCL register can only be written if CRC0EN (bit 7 of the CRC0CTL register) = 1.

Figure 27 - 3 shows the Flowchart of Flash Memory CRC Operation Function (High-speed CRC).

&lt;Operation flow&gt;

**Figure 27 - 3 Flowchart of Flash Memory CRC Operation Function (High-speed CRC)**

**Caution 1. The CRC operation is executed only on the code flash.**

**Caution 2. Store the expected CRC operation value in the area below the operation range in the code flash.**

**Caution 3. The CRC operation is enabled by executing the HALT instruction in the RAM area.**

**Be sure to execute the HALT instruction in RAM area.**

The expected CRC operation value can be calculated by using the integrated development environment CubeSuite+ development environment. Refer to the CubeSuite+ integrated development environment user's manual for details.

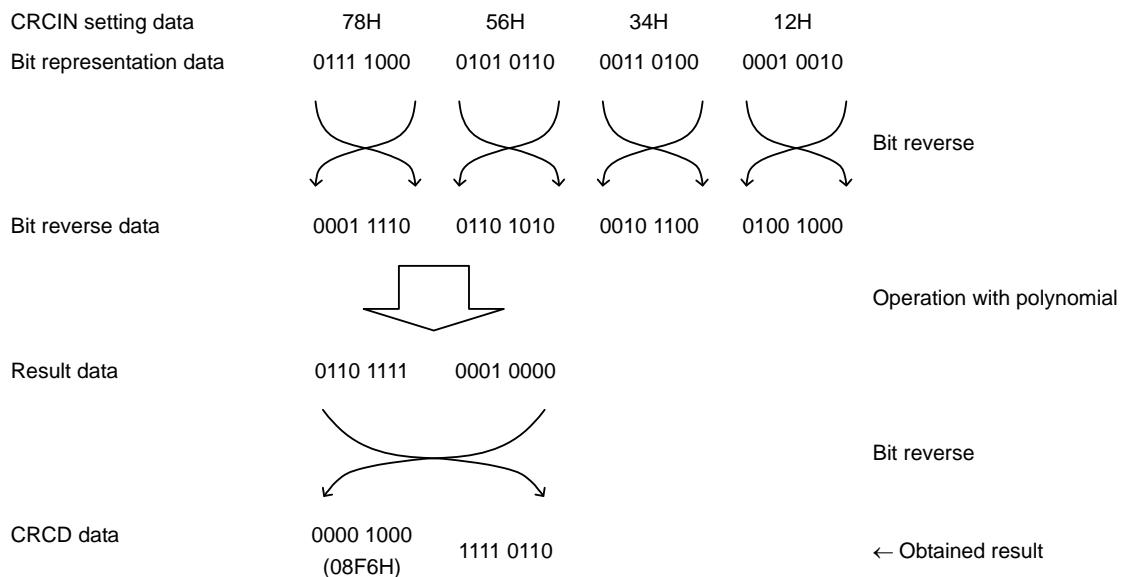
### 27.3.2 CRC operation function (general-purpose CRC)

In order to guarantee safety during operation, the IEC61508 standard mandates the checking of data even while the CPU is operating.

In the RL78/G14, a general CRC operation can be executed as a peripheral function while the CPU is operating. The general CRC can be used for checking various data in addition to the code flash memory area. The data to be checked can be specified by using software (a user-created program). In HALT mode, the CRC operation function can be used only during DTC transfer.

The general CRC operation can be executed in the main system clock operation mode as well as the subsystem clock operation mode.

The CRC generator polynomial used is “ $X^{16} + X^{12} + X^5 + 1$ ” of CRC-16-CCITT. The data to be input is inverted in bit order and then calculated to allow for LSB-first communication. For example, if the data 12345678H is sent from the LSB, values are written to the CRCIN register in the order of 78H, 56H, 34H, and 12H, enabling a value of 08F6H to be obtained from the CRCD register. This is the result obtained by executing a CRC operation on the bit rows shown below, which consist of the data 12345678H inverted in bit order.



**Caution** Because the debugger rewrites the software break setting line to a break instruction during program execution, the CRC operation result differs if a software break is set in the CRC operation target area.

### 27.3.2.1 CRC input register (CRCIN)

CRCIN register is an 8-bit register that is used to set the CRC operation data of general-purpose CRC.

The possible setting range is 00H to FFH.

The CRCIN register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 4 Format of CRC input register (CRCIN)**

Address:FFFAC<sub>H</sub>      After reset:00H      R/W

Symbol      7      6      5      4      3      2      1      0

CRCIN 

Bits 7 to 0	Function
00H to FFH	Data input.

### 27.3.2.2 CRC data register (CRCD)

This register is used to store the general-purpose CRC operation result.

The possible setting range is 0000H to FFFFH.

After 1 clock of CPU/peripheral hardware clock (fCLK) has elapsed from the time CRCIN register is written, the CRC operation result is stored to the CRCD register.

The CRCD register can be set by a 16-bit memory manipulation instruction.

Reset signal generation clears this register to 0000H.

**Figure 27 - 5 Format of CRC data register (CRCD)**

Address: F02FAH      After reset: 0000H      R/W

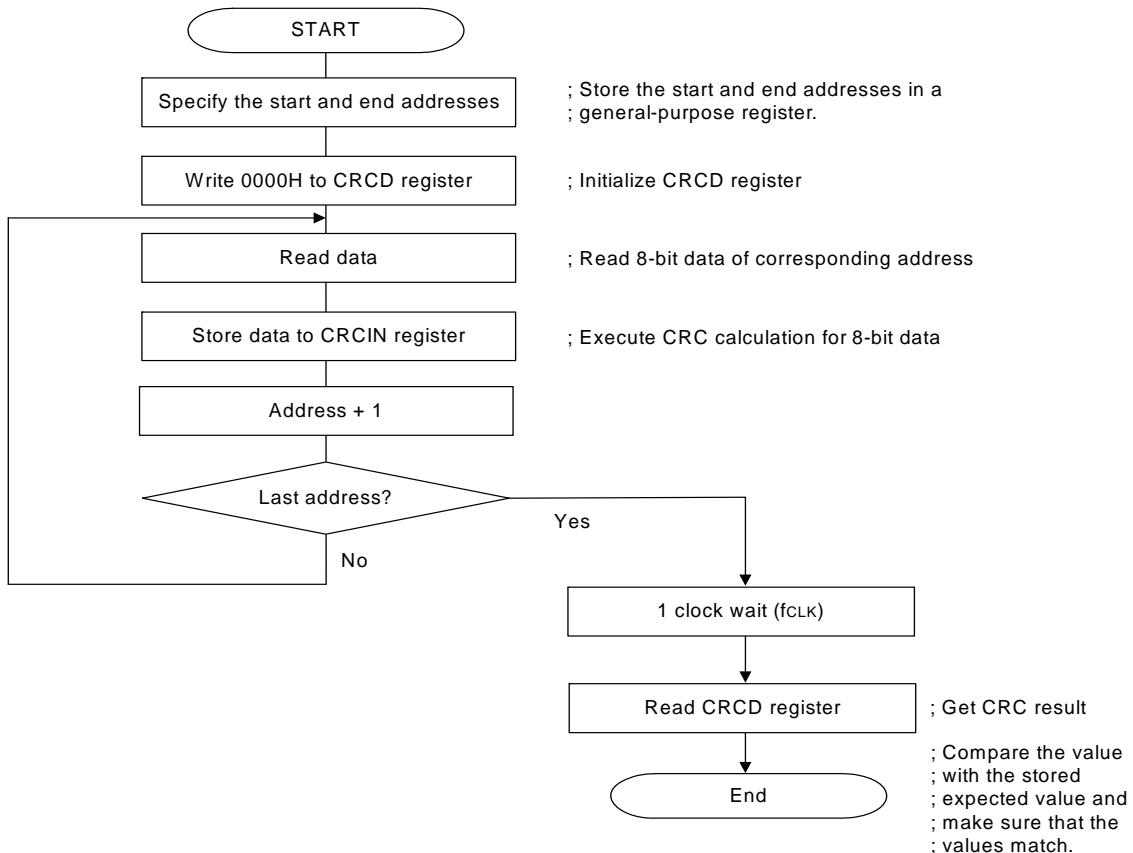
Symbol	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CRCD																

**Caution 1. Read the value written to CRCD register before writing to CRCIN register.**

**Caution 2. If writing and storing operation result to CRCD register conflict, the writing is ignored.**

<Operation flow>

**Figure 27 - 6 CRC Operation Function (General-Purpose CRC)**



### 27.3.3 RAM parity error detection function

The IEC60730 standard mandates the checking of RAM data. A single-bit parity bit is therefore added to all 8-bit data in the RL78/G14's RAM. By using this RAM parity error detection function, the parity bit is appended when data is written, and the parity is checked when the data is read. This function can also be used to trigger a reset when a parity error occurs.

#### 27.3.3.1 RAM parity error control register (RPECTL)

This register is used to control parity error generation check bit and reset generation due to parity errors.

The RPECTL register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 7 Format of RAM parity error control register (RPECTL)**

Address: F00F5H		After reset: 00H		R/W					
Symbol	<7>	6	5	4	3	2	1	<0>	
RPECTL	RPERDIS	0	0	0	0	0	0	RPEF	
RPERDIS		Parity error reset mask flag							
0	Enable parity error resets.								
1	Disable parity error resets.								
RPEF		Parity error status flag							
0	No parity error has occurred.								
1	A parity error has occurred.								

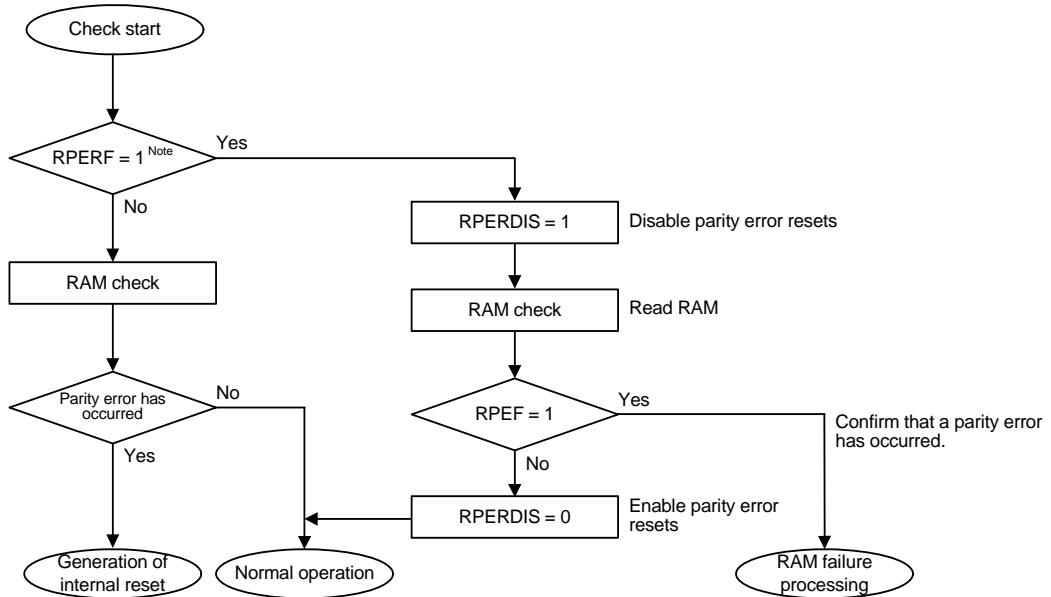
**Caution** The parity bit is appended when data is written, and the parity is checked when the data is read. Therefore, while RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize RAM areas where data access is to proceed before reading data. The RL78's CPU executes look-ahead due to the pipeline operation, the CPU might read an uninitialized RAM area that is allocated beyond the RAM used, which causes a RAM parity error. Therefore, while RAM parity error resets are enabled (RPERDIS = 0), be sure to initialize the RAM area + 10 bytes when instructions are fetched from RAM areas.

**Remark 1.** The parity error reset is enabled by default (RPERDIS = 0).

**Remark 2.** Even if the parity error reset is disabled (RPERDIS = 1), the RPEF flag will be set (1) if a parity error occurs. If the parity error reset is enabled (RPERDIS = 0) while RPEF = 1, a parity error reset occurs when RPERDIS is cleared (0).

**Remark 3.** The RPECTL flag in the RESF register is set (1) by RAM parity errors and cleared (0) by writing 0 to it or by any reset source. When RPEF = 1, the value is retained even if RAM for which no parity error has occurred is read.

**Remark 4.** General-purpose registers are not included in the range of RAM parity error detection.

**Figure 27 - 8 RAM Parity Error Check Flow**

**Note** See **CHAPTER 24 RESET FUNCTION** for details on how to confirm internal resets due to RAM parity errors.

### 27.3.4 RAM guard function

In order to guarantee safety during operation, the IEC61508 standard mandates that important data stored in the RAM be protected, even if the CPU freezes.

This RAM guard function is used to protect data in the specified memory space.

If the RAM guard function is specified, writing to the specified RAM space is disabled, but reading from the space can be carried out as usual.

#### 27.3.4.1 Invalid memory access detection control register (IAWCTL)

This register is used to control the detection of invalid memory access and RAM/SFR guard function.

GRAM1 and GRAM0 bits are used in RAM guard function.

The IAWCTL register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 9 Format of Invalid memory access detection control register (IAWCTL)**

Address: F0078H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
IAWCTL	IAWEN	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC
<hr/>								
GRAM1	GRAM0	RAM guard space <small>Note</small>						
0	0	Disabled. RAM can be written to.						
0	1	The 128 bytes starting at the start RAM address						
1	0	The 256 bytes starting at the start RAM address						
1	1	The 512 bytes starting at the start RAM address						

**Note**      The RAM start address differs depending on the size of the RAM provided with the product.

### 27.3.5 SFR guard function

In order to guarantee safety during operation, the IEC61508 standard mandates that important SFRs be protected from being overwritten, even if the CPU freezes.

This SFR guard function is used to protect data in the control registers used by the port function, interrupt function, clock control function, voltage detection function, and RAM parity error detection function.

If the SFR guard function is specified, writing to the specified SFRs is disabled, but reading from the SFRs can be carried out as usual.

#### 27.3.5.1 Invalid memory access detection control register (IAWCTL)

This register is used to control the detection of invalid memory access and RAM/SFR guard function.

GPORT, GINT and GCSC bits are used in SFR guard function.

The IAWCTL register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 10 Format of Invalid memory access detection control register (IAWCTL)**

Address: F0078H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
IAWCTL	IAWEN	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC
GPORT	Control registers of port function guard							
0	Disabled. Control registers of port function can be read or written to.							
1	Enabled. Writing to control registers of port function is disabled. Reading is enabled. [Guarded SFR] PMxx, PUxx, PIMxx, POMxx, PMCxx, ADPC, PIORx Note							
GINT	Registers of interrupt function guard							
0	Disabled. Registers of interrupt function can be read or written to.							
1	Enabled. Writing to registers of interrupt function is disabled. Reading is enabled. [Guarded SFR] IFxx, MKxx, PRxx, EGPx, EGNx							
GCSC	Control registers of clock control function, voltage detector, and RAM parity error detection function guard							
0	Disabled. Control registers of clock control function, voltage detector and RAM parity error detection function can be read or written to.							
1	Enabled. Writing to control registers of clock control function, voltage detector and RAM parity error detection function is disabled. Reading is enabled. [Guarded SFR] CMC, CSC, OSTS, CKC, PERx, OSMC, LVIM, LVIS, RPECTL							

**Note**      Px (Port register) is not guarded.

### 27.3.6 Invalid memory access detection function

The IEC60730 standard mandates checking that the CPU and interrupts are operating correctly.

The illegal memory access detection function triggers a reset if a memory space specified as access-prohibited is accessed.

The illegal memory access detection function applies to the areas indicated by NG in Figure 27 - 11.

**Figure 27 - 11 Invalid access detection area**

		Read	Write	Fetching instructions (execute)
FFFFFH	Special function register (SFR) 256 byte		OK	NG
FFF00H	General-purpose register 32 byte		OK	OK
FFEFHH				NG
FFEE0H				OK
FFEDFH				NG
zzzzzH	RAM <sup>Note</sup>	OK	NG	
	Mirror			
F1000H	Data flash memory			
F0FFFH	Reserved			
F0800H				
F07FFFH	Extended special function register (2nd SFR) 2 Kbyte		OK	
F0000H				
EFFFFH				
EF000H				OK
EEFFFH				
	Reserved	NG	NG	NG
yyyyyH				
xxxxxH		OK		OK
00000H	Code flash memory <sup>Note</sup>			

**Note** The code flash memory, RAM, and lowest detection address of each product are as follows.

Products	Code Flash Memory (00000H to xxxxH)	RAM (zzzzzH to FFEFFFH)	Lowest Detection Address (yyyyyH) when Reading/Fetching (Executing) Instructions
R5F104xA (x = A to C, E to G)	16384 × 8 bits (00000H to 03FFFF)	2560 × 8 bits (FF500H to FFEFFFH)	10000H
R5F104xC (x = A to C, E to G, J, L)	32768 × 8 bits (00000H to 07FFFF)	4096 × 8 bits (FEF00H to FFEFFFH)	10000H
R5F104xD (x = A to C, E to G, J, L)	49152 × 8 bits (00000H to 0BFFFF)	5632 × 8 bits (FE900H to FFEFFFH)	10000H
R5F104xE (x = A to C, E to G, J, L)	65536 × 8 bits (00000H to 0FFFFF)	5632 × 8 bits (FE900H to FFEFFFH)	10000H
R5F104xF (x = A to C, E to G, J, L, M, P)	98304 × 8 bits (00000H to 17FFFF)	12288 × 8 bits (FCF00H to FFEFFFH)	20000H
R5F104xG (x = A to C, E to G, J, L, M, P)	131072 × 8 bits (00000H to 1FFFFFF)	16384 × 8 bits (FBF00H to FFEFFFH)	20000H
R5F104xH (x = E to G, J, L, M, P)	196608 × 8 bits (00000H to 2FFFFFF)	20480 × 8 bits (FAF00H to FFEFFFH)	30000H
R5F104xJ (x = F, G, J, L, M, P)	262144 × 8 bits (00000H to 3FFFFFF)	24576 × 8 bits (F9F00H to FFEFFFH)	40000H
R5F104xK (x = G, L, M, P)	393216 × 8 bits (00000H to 5FFFFFF)	32768 × 8 bits (F7F00H to FFEFFFH)	60000H
R5F104xL (x = G, L, M, P)	524688 × 8 bits (00000H to 7FFFFFF)	49152 × 8 bits (F3F00H to FFEFFFH)	80000H

### 27.3.6.1 Invalid memory access detection control register (IAWCTL)

This register is used to control the detection of invalid memory access and RAM/SFR guard function.

IAWEN bit is used in invalid memory access detection function.

The IAWCTL register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 12 Format of Invalid memory access detection control register (IAWCTL)**

Address: F0078H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
--------	---	---	---	---	---	---	---	---

IAWCTL	IAWEN Note	0	GRAM1	GRAM0	0	GPORT	GINT	GCSC
IAWEN Note	Control of invalid memory access detection							
0	Disable the detection of invalid memory access.							
1	Enable the detection of invalid memory access.							

**Note** Only writing 1 to the IAWEN bit is enabled, not writing 0 to it after setting it to 1.

**Remark** By specifying WDTON = 1 for the option byte (watchdog timer operation enable), the invalid memory access detection function is enabled even if IAWEN = 0.

### 27.3.7 Frequency detection function

The IEC60730 standard mandates checking that the oscillation frequency is correct.

By using the CPU/peripheral hardware clock frequency ( $f_{CLK}$ ) and measuring the pulse width of the input signal to channel 1 of the timer array unit 0 (TAU0), whether the proportional relationship between the two clock frequencies is correct can be determined.

Note that, however, if one or both clock operations are stopped, the proportional relationship between the clocks cannot be determined.

<Clocks to be compared>

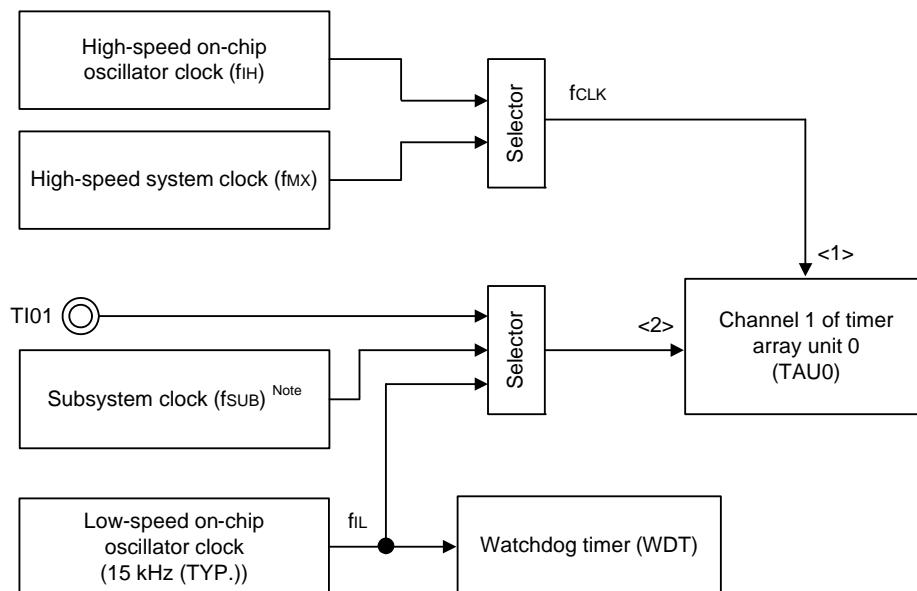
<1> CPU/peripheral hardware clock frequency ( $f_{CLK}$ ):

- High-speed on-chip oscillator clock ( $f_{IH}$ )
- High-speed system clock ( $f_{MX}$ )

<2> Input to channel 1 of the timer array unit 0

- Timer input to channel 1 (TI01)
- Low-speed on-chip oscillator clock ( $f_{IL}$ : 15 kHz (typ.))
- Subsystem clock ( $f_{SUB}$ ) Note

**Figure 27 - 13 Configuration of Frequency Detection Function**



If pulse interval measurement results in an abnormal value, it can be concluded that the clock frequency is abnormal.

For how to execute pulse interval measurement, see **6.8.4 Operation as input pulse interval measurement**.

**Note** Can only be selected in the products incorporating the subsystem clock.

### 27.3.7.1 Timer input select register 0 (TIS0)

The TIS0 register is used to select the timer input of channels 0 and 1 of the timer array unit 0 (TAU0).

The TIS0 register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 14 Format of Timer input select register 0 (TIS0)**

Address: F0074H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
TIS0	0	0	0	TIS04	0	TIS02	TIS01	TIS00

TIS04	Selection of timer input used with channel 0
0	Input signal of timer input pin (TI00)
1	Event input signal from ELC

TIS02	TIS01	TIS00	Selection of timer input used with channel 1
0	0	0	Input signal of timer input pin (TI01)
0	0	1	Event input signal from ELC
0	1	0	Input signal of timer input pin (TI01)
0	1	1	
1	0	0	Low-speed on-chip oscillator clock (f <sub>IL</sub> )
1	0	1	Subsystem clock (f <sub>SUB</sub> )
Other than above			Setting prohibited

### 27.3.8 A/D test function

The IEC60730 standard mandates testing the A/D converter. The A/D test function is used to check whether the A/D converter is operating normally by executing A/D conversions of the positive reference voltage and negative reference voltage of the A/D converter, analog input channel (ANI), temperature sensor output voltage, and internal reference voltage. For details on the checking method, refer to the safety function (A/D test) application note (R01AN0955).

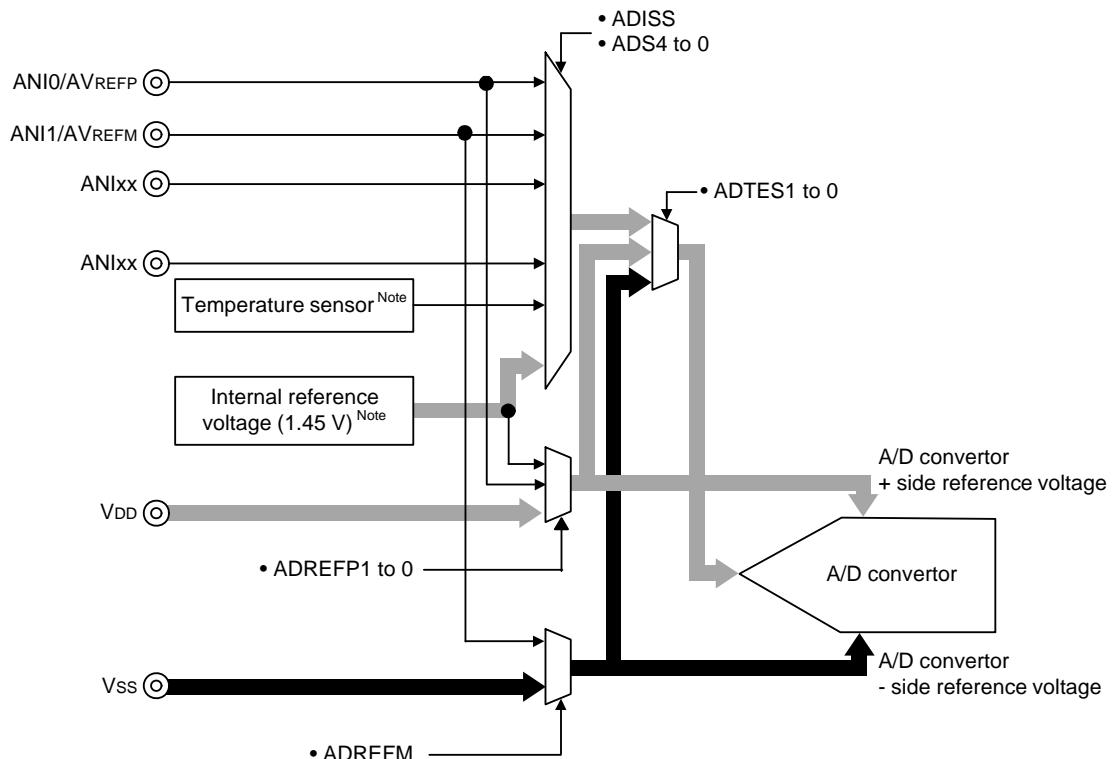
The analog multiplexer can be checked using the following procedure.

- (1) Select the ANIx pin as the target for A/D conversion by setting the ADTES register (ADTES1, ADTES0 = 0, 0).
- (2) Perform A/D conversion for the ANIx pin (conversion result 1-1).
- (3) Select the negative reference voltage of the A/D converter as the target for A/D conversion by setting the ADTES register (ADTES1, ADTES0 = 1, 0).
- (4) Perform A/D conversion of the negative reference voltage of the A/D converter (conversion result 2-1).
- (5) Select the ANIx pin as the target for A/D conversion by setting the ADTES register (ADTES1, ADTES0 = 0, 0).
- (6) Perform A/D conversion for the ANIx pin (conversion result 1-2).
- (7) Select the positive reference voltage of the A/D converter as the target for A/D conversion by setting the ADTES register (ADTES1, ADTES0 = 1, 1).
- (8) Perform A/D conversion of the positive reference voltage of the A/D converter (conversion result 2-2).
- (9) Select the ANIx pin as the target for A/D conversion by setting the ADTES register (ADTES1, ADTES0 = 0, 0).
- (10) Perform A/D conversion for the ANIx pin (conversion result 1-3).
- (11) Make sure that “conversion result 1-1” = “conversion result 1-2” = “conversion result 1-3”.
- (12) Make sure that the A/D conversion results of “conversion result 2-1” are all 0 and those of “conversion result 2-2” are all 1.

Using the procedure above can confirm that the analog multiplexer is selected and all wiring is connected.

**Remark 1.** If the analog input voltage is variable during conversion in steps (1) to (10) above, use another method to check the analog multiplexer.

**Remark 2.** The conversion results might contain an error. Consider an appropriate level of error when comparing the conversion results.

**Figure 27 - 15 Configuration of A/D Test Function**

**Note** Selectable only in HS (high-speed main) mode.

### 27.3.8.1 A/D test register (ADTES)

This register is used to select the A/D converter positive reference voltage, negative reference voltage, analog input channel (ANIx<sub>x</sub>), temperature sensor output voltage, or internal reference voltage (1.45 V) as the target of A/D conversion.

When using the A/D test function, specify the following settings:

- Select the negative reference voltage as the target of A/D conversion when measuring the zero-scale.
- Select the positive reference voltage as the target of A/D conversion when measuring the full-scale.

The ADTES register can be set by an 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 16 Format of A/D test register (ADTES)**

Address: F0013H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADTES	0	0	0	0	0	0	ADTES1	ADTES0

		A/D conversion target
ADTES1	ADTES0	
0	0	ANIx <sub>x</sub> /temperature sensor output Note/internal reference voltage (1.45 V) Note (This is specified using the analog input channel specification register (ADS).)
1	0	Negative reference voltage (selected by the ADREFM bit in the ADM2 register)
1	1	Positive reference voltage (selected by the ADREFP1 and ADREFP0 bits in the ADM2 register) Note
Other than the above		Setting prohibited

**Note** Temperature sensor output voltage/internal reference voltage (1.45 V) can be used only in HS (high-speed main) mode.

### 27.3.8.2 Analog input channel specification register (ADS)

This register specifies the input channel of the analog voltage to be A/D converted.

Set A/D test register (ADTES) to 00H when measuring the ANIx<sub>x</sub>/temperature sensor output /internal reference voltage (1.45 V).

The ADS register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears this register to 00H.

**Figure 27 - 17 Format of Analog input channel specification register (ADS)**

Address: FFF31H      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
ADS	ADISS	0	0	ADS4	ADS3	ADS2	ADS1	ADS0

○ Select mode (ADMD = 0)

ADISS	ADS4	ADS3	ADS2	ADS1	ADS0	Analog input channel	Input source
0	0	0	0	0	0	ANI0	P20/ANI0/AVREFP pin
0	0	0	0	0	1	ANI1	P21/ANI1/AVREFM pin
0	0	0	0	1	0	ANI2	P22/ANI2 pin
0	0	0	0	1	1	ANI3	P23/ANI3 pin
0	0	0	1	0	0	ANI4	P24/ANI4 pin
0	0	0	1	0	1	ANI5	P25/ANI5 pin
0	0	0	1	1	0	ANI6	P26/ANI6 pin
0	0	0	1	1	1	ANI7	P27/ANI7 pin
0	0	1	0	0	0	ANI8	P150/ANI8 pin
0	0	1	0	0	1	ANI9	P151/ANI9 pin
0	0	1	0	1	0	ANI10	P152/ANI10 pin
0	0	1	0	1	1	ANI11	P153/ANI11 pin
0	0	1	1	0	0	ANI12	P154/ANI12 pin
0	0	1	1	0	1	ANI13	P155/ANI13 pin
0	0	1	1	1	0	ANI14	P156/ANI14 pin
0	1	0	0	0	0	ANI16	P03/ANI16 pin Note 1
0	1	0	0	0	1	ANI17	P02/ANI17 pin Note 2
0	1	0	0	1	0	ANI18	P147/ANI18 pin
0	1	0	0	1	1	ANI19	P120/ANI19 pin
0	1	0	1	0	0	ANI20	P100/ANI20 pin
1	0	0	0	0	0	—	Temperature sensor output Notes 3, 4
1	0	0	0	0	1	—	Internal reference voltage output (1.45 V) Note 4
Other than the above						Setting prohibited	

**Note 1.** 30-, 32-pin products: P01/ANI16 pin

**Note 2.** 30-, 32-pin products: P00/ANI17 pin

**Note 3.** If the internal reference voltage (1.45 V) is selected as the reference voltage of comparator 0 or comparator 1 in products with 96 KB or more code flash memory, the temperature sensor output cannot be selected.

**Note 4.** This setting can be used only in HS (high-speed main) mode.

(Cautions are listed on the next page.)

- Caution 1. Be sure to clear bits 5 and 6 to 0.
- Caution 2. For ports that set to analog input using the ADPC and PMC registers, select input mode using port mode register 0, 2, 10, 12, 14, or 15 (PM0, PM2, PM10, PM12, PM14, PM15).
- Caution 3. Do not use the ADS register to set ports that to be set as digital I/O using the A/D port configuration register.
- Caution 4. Do not use the ADS register to set ports that to be set as digital I/O using port mode control register 0, 10, 12, or 14 (PMC0, PMC10, PMC12, PMC14).
- Caution 5. Only rewrite the value of the ADISS bit while conversion is stopped (ADCS = 0, ADCE = 0).
- Caution 6. When using AVREFP as the positive reference voltage of the A/D converter, do not select ANI0 as an A/D conversion channel.
- Caution 7. When using AVREFM as the negative reference voltage of the A/D converter, do not select ANI1 as an A/D conversion channel.
- Caution 8. If ADISS is set to 1, the internal reference voltage output (1.45 V) cannot be used for the positive reference voltage. Also, the first conversion result cannot be used after ADISS is set to 1. For details on the setup flow, see 14.7.4 Setup when temperature sensor output voltage/internal reference voltage is selected.
- Caution 9. Do not set ADISS to 1 when entering HALT mode while in STOP mode or while the CPU operates on the subsystem clock. With ADISS = 1, the current value of the A/D converter reference voltage current (IADREF) listed in 34.3.2 Supply current characteristics is added.

### 27.3.9 Digital output signal level detection function for I/O pins

In the IEC60730, it is required to check that the I/O function correctly operates.

By using the digital output signal level detection function for I/O pins, the digital output level of the pin can be read when the port is set to output mode.

#### 27.3.9.1 Port mode select register (PMS)

This register is used to select the output level from output latch level or pin output level when the pin is output mode in which PMm bit of port mode register (PMm) is 0.

This register can be set by a 1-bit or 8-bit memory manipulation instruction.

Reset signal generation clears these registers to 00H.

**Figure 27 - 18 Format of Port mode select register (PMS)**

Address: F007BH      After reset: 00H      R/W

Symbol	7	6	5	4	3	2	1	0
PMS	0	0	0	0	0	0	0	PMS0
PMS0	Method for selecting output level to be read when pin is output mode							
0	Pmn register value is read.							
1	Digital output level of the pin is read.							

**Caution 1.** While the PMS0 bit in the PMS register is set to 1, do not change the value of the port register (Pxx) using a bit manipulation instruction. To change the value of the port register (Pxx), use an 8-bit data manipulation instruction.

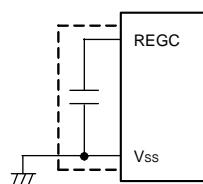
**Caution 2.** When the digital output level of a pin that is held in the high-impedance state by the timer RD pulse output forced cutoff function, the read value is 0.

**Remark**    m = 0 to 8, 10 to 12, 14, 15  
              n = 0 to 7

## CHAPTER 28 REGULATOR

### 28.1 Regulator Overview

The RL78/G14 contains a circuit for operating the device with a constant voltage. At this time, in order to stabilize the regulator output voltage, connect the REGC pin to Vss via a capacitor (0.47 to 1  $\mu$ F). Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.



**Caution** Keep the wiring length as short as possible for the broken-line part in the above figure.

The regulator output voltage, see **Table 28 - 1**.

**Table 28 - 1 Regulator Output Voltage Conditions**

Mode	Output Voltage	Condition
LV (low-voltage main) mode	1.8 V	—
LS (low-speed main) mode		
HS (high-speed main) mode	1.8 V	In STOP mode
		When both the high-speed system clock ( $f_{SUB}$ ) and the high-speed on-chip oscillator clock ( $f_{IH}$ ) are stopped during CPU operation with the subsystem clock ( $f_{XT}$ )
		When both the high-speed system clock ( $f_{SUB}$ ) and the high-speed on-chip oscillator clock ( $f_{IH}$ ) are stopped during the HALT mode when the CPU operation with the subsystem clock ( $f_{XT}$ ) has been set
	2.1 V	Other than above (include during OCD mode)

**Note** When it shifts to the subsystem clock operation or STOP mode during the on-chip debugging, the regulator output voltage is kept at 2.1 V (not decline to 1.8 V).

## CHAPTER 29 OPTION BYTE

### 29.1 Functions of Option Bytes

Addresses 000C0H to 000C3H of the flash memory of the RL78/G14 form an option byte area.

Option bytes consist of user option byte (000C0H to 000C2H) and on-chip debug option byte (000C3H).

Upon power application or resetting and starting, an option byte is automatically referenced and a specified function is set. When using the product, be sure to set the following functions by using the option bytes.

For the bits to which no function is allocated, do not change their initial values.

To use the boot swap operation during self-programming, 000C0H to 000C3H are replaced by 010C0H to 010C3H.

Therefore, set the same values as 000C0H to 000C3H to 010C0H to 010C3H.

**Remark** The option bytes should always be set regardless of whether each function is used.

#### 29.1.1 User option byte (000C0H to 000C2H/010C0H to 010C2H)

- (1) 000C0H/010C0H
  - Setting of watchdog timer operation
    - Enabling or disabling of counter operation
    - Enabling or disabling of counter operation in the HALT or STOP mode
  - Setting of interval time of watchdog timer
  - Setting of window open period of watchdog timer
  - Setting of interval interrupt of watchdog timer
    - Interval interrupt is used or not used

**Caution** Set the same value as 000C0H to 010C0H when the boot swap operation is used because 000C0H is replaced by 010C0H.

- (2) 000C1H/010C1H
  - Setting of LVD operation mode
    - Interrupt & reset mode
    - Reset mode
    - Interrupt mode
    - LVD off (external reset input from the RESET pin is used)
  - Setting of LVD detection level (VLVDH, VLVDL, VLVD)

**Caution 1.** After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in 34.4 or 35.4 AC Characteristics. This is done by utilizing the voltage detection circuit or controlling the externally input reset signal. After the power supply is turned off, this LSI should be placed in the STOP mode, or placed in the reset state by utilizing the voltage detection circuit or controlling the externally input reset signal, before the voltage falls below the operating range. The range of operating voltage varies with the setting of the user option byte (000C2H or 010C2H).

**Caution 2.** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

## (3) 000C2H/010C2H

 Setting of flash operation mode

Make the setting depending on the main system clock frequency ( $f_{MAIN}$ ) and power supply voltage ( $V_{DD}$ ) to be used.

- LV (low-voltage main) mode
- LS (low-speed main) mode
- HS (high-speed main) mode

 Setting of the frequency of the high-speed on-chip oscillator

- Select from 1 MHz to 32 MHz, 48 MHz, and 64 MHz.

**Caution** Set the same value as 000C2H to 010C2H when the boot swap operation is used because 000C2H is replaced by 010C2H.

### 29.1.2 On-chip debug option byte (000C3H/ 010C3H)

 Control of on-chip debug operation

- On-chip debug operation is disabled or enabled.

 Handling of data of flash memory in case of failure in on-chip debug security ID authentication

- Data of flash memory is erased or not erased in case of failure in on-chip debug security ID authentication.

**Caution** Set the same value as 000C3H to 010C3H when the boot swap operation is used because 000C3H is replaced by 010C3H.

## 29.2 Format of User Option Byte

The format of user option byte is shown below.

**Figure 29 - 1 Format of User Option Byte (000C0H/010C0H)**

Address: 000C0H/010C0H Note 1

7	6	5	4	3	2	1	0						
WDTINT	WINDOW1	WINDOW0	WDTON	WDCS2	WDCS1	WDCS0	WDSTBYON						
WDTINT      Use of interval interrupt of watchdog timer													
0	Interval interrupt is not used.												
1	Interval interrupt is generated when 75% + 1/2 f <sub>L</sub> of the overflow time is reached.												
WINDOW1    WINDOW0      Watchdog timer window open period Note 2													
0	0	Setting prohibited											
0	1	50%											
1	0	75%											
1	1	100%											
WDTON      Operation control of watchdog timer counter													
0	Counter operation disabled (counting stopped after reset)												
1	Counter operation enabled (counting started after reset)												
WDCS2    WDCS1    WDCS0      Watchdog timer overflow time (f <sub>L</sub> = 17.25 kHz (MAX.))													
0	0	0	2 <sup>6</sup> /f <sub>L</sub> (3.71 ms)										
0	0	1	2 <sup>7</sup> /f <sub>L</sub> (7.42 ms)										
0	1	0	2 <sup>8</sup> /f <sub>L</sub> (14.84 ms)										
0	1	1	2 <sup>9</sup> /f <sub>L</sub> (29.68 ms)										
1	0	0	2 <sup>11</sup> /f <sub>L</sub> (118.72 ms)										
1	0	1	2 <sup>13</sup> /f <sub>L</sub> (474.90 ms)										
1	1	0	2 <sup>14</sup> /f <sub>L</sub> (949.80 ms)										
1	1	1	2 <sup>16</sup> /f <sub>L</sub> (3799.19 ms)										
WDSTBYON      Operation control of watchdog timer counter (HALT/STOP mode)													
0	Counter operation stopped in HALT/STOP mode Note 2												
1	Counter operation enabled in HALT/STOP mode												

**Note 1.** Set the same value as 000C0H to 010C0H when the boot swap operation is used because 000C0H is replaced by 010C0H.

**Note 2.** The window open period is 100% when WDSTBYON = 0, regardless the value of the WINDOW1 and WINDOW0 bits.

**Remark** f<sub>L</sub>: Low-speed on-chip oscillator clock frequency

**Figure 29 - 2 Format of User Option Byte (000C1H/010C1H) (1/4)**

Address: 000C1H/010C1H Note

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD setting (interrupt & reset mode)

Detection Voltage			Option Byte Setting Value						
VLVDH		VLVDL	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	Mode setting	
Rising edge	Falling edge	Falling edge						LVIMDS1	LVIMDS0
1.77 V	1.73 V	1.63 V	0	0	0	1	0	1	0
1.88 V	1.84 V					0	1		
2.92 V	2.86 V					0	0		
1.98 V	1.94 V					0	1		
2.09 V	2.04 V					0	1		
3.13 V	3.06 V					0	0		
2.61 V	2.55 V					1	0		
2.71 V	2.65 V					1	0		
3.75 V	3.67 V					0	1		
2.92 V	2.86 V					0	0		
3.02 V	2.96 V	2.75 V	1	1	1	1	0	1	0
4.06 V	3.98 V					0	1		
—			Settings other than the above are prohibited						

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Caution** Be sure to set bit 4 to “1”.

**Remark 1.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 2.** The detection voltage is a typical value. For details, see **34.6.6** or **35.6.6 LVD circuit characteristics**.

**Figure 29 - 3 Format of User Option Byte (000C1H/010C1H) (2/4)**

Address: 000C1H/010C1H Note

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD setting (reset mode)

Detection voltage		Option byte Setting Value					Mode setting	
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	LVIMDS1	LVIMDS0
Rising edge	Falling edge							
1.67 V	1.63 V		0	0	1	1	1	1
1.77 V	1.73 V		0	0	1	0		
1.88 V	1.84 V		0	1	1	1		
1.98 V	1.94 V		0	1	1	0		
2.09 V	2.04 V		0	1	0	1		
2.50 V	2.45 V		1	0	1	1		
2.61 V	2.55 V		1	0	1	0		
2.71 V	2.65 V		1	0	0	1		
2.81 V	2.75 V		1	1	1	1		
2.92 V	2.86 V		1	1	1	0		
3.02 V	2.96 V		1	1	0	1		
3.13 V	3.06 V		0	1	0	0		
3.75 V	3.67 V		1	0	0	0		
4.06 V	3.98 V		1	1	0	0		
—		Settings other than the above are prohibited						

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Caution** Be sure to set bit 4 to “1”.

**Remark 1.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 2.** The detection voltage is a typical value. For details, see **34.6.6** or **35.6.6 LVD circuit characteristics**.

**Figure 29 - 4 Format of User Option Byte (000C1H/010C1H) (3/4)**

Address: 000C1H/010C1H Note

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD setting (interrupt mode)

Detection voltage		Option byte Setting Value						Mode setting	
VLVD		0	VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	LVIMDS1	LVIMDS0
Rising edge	Falling edge								
1.67 V	1.63 V								
1.77 V	1.73 V								
1.88 V	1.84 V								
1.98 V	1.94 V								
2.09 V	2.04 V								
2.50 V	2.45 V								
2.61 V	2.55 V								
2.71 V	2.65 V								
2.81 V	2.75 V								
2.92 V	2.86 V								
3.02 V	2.96 V								
3.13 V	3.06 V								
3.75 V	3.67 V								
4.06 V	3.98 V								
—		Settings other than the above are prohibited							

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Caution** Be sure to set bit 4 to “1”.

**Remark 1.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 2.** The detection voltage is a typical value. For details, see **34.6.6** or **35.6.6 LVD circuit characteristics**.

**Figure 29 - 5 Format of User Option Byte (000C1H/010C1H) (4/4)**

Address: 000C1H/010C1H Note

7	6	5	4	3	2	1	0
VPOC2	VPOC1	VPOC0	1	LVIS1	LVIS0	LVIMDS1	LVIMDS0

- LVD off setting (external reset input from the RESET pin is used)

Detection voltage		Option byte Setting Value						
VLVD		VPOC2	VPOC1	VPOC0	LVIS1	LVIS0	Mode setting	
Rising edge	Falling edge						LVIMDS1	LVIMDS0
—	—	1	×	×	×	×	×	1
—		Settings other than the above are prohibited						

**Note** Set the same value as 000C1H to 010C1H when the boot swap operation is used because 000C1H is replaced by 010C1H.

**Caution 1.** Be sure to set bit 4 to “1”.

**Caution 2.** After power is supplied, the reset state must be retained until the operating voltage becomes in the range defined in 34.4 or 35.4 AC Characteristics. This is done by utilizing the voltage detection circuit or controlling the externally input reset signal. After the power supply is turned off, this LSI should be placed in the STOP mode, or placed in the reset state by utilizing the voltage detection circuit or controlling the externally input reset signal, before the voltage falls below the operating range. The range of operating voltage varies with the setting of the user option byte (000C2H or 010C2H).

**Remark 1.** ×: Don't care

**Remark 2.** For details on the LVD circuit, see **CHAPTER 26 VOLTAGE DETECTOR**.

**Remark 3.** The detection voltage is a typical value. For details, see **34.6.6 or 35.6.6 LVD circuit characteristics**.

**Figure 29 - 6 Format of Option Byte (000C2H/010C2H)**

Address: 000C2H/010C2H Note

7            6            5            4            3            2            1            0

CMODE1	CMODE0	1	FRQSEL4	FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0
--------	--------	---	---------	---------	---------	---------	---------

CMODE1	CMODE0	Setting of flash operation mode					
		Operating Frequency Range (f <sub>MAIN</sub> )	Operating Voltage Range (V <sub>DD</sub> )				
0	0	LV (low-voltage main) mode	1 to 4 MHz	1.6 to 5.5 V			
1	0	LS (low-speed main) mode	1 to 8 MHz	1.8 to 5.5 V			
1	1	HS (high-speed main) mode	1 to 16 MHz	2.4 to 5.5 V			
			1 to 32 MHz	2.7 to 5.5 V			
Other than above		Setting prohibited					

FRQSEL4	FRQSEL3	FRQSEL2	FRQSEL1	FRQSEL0	Frequency of the high-speed on-chip oscillator clock	
					f <sub>HOCO</sub>	f <sub>lH</sub>
1	1	0	0	0	64 MHz	32 MHz
1	0	0	0	0	48 MHz	24 MHz
0	1	0	0	0	32 MHz	32 MHz
0	0	0	0	0	24 MHz	24 MHz
0	1	0	0	1	16 MHz	16 MHz
0	0	0	0	1	12 MHz	12 MHz
0	1	0	1	0	8 MHz	8 MHz
0	0	0	1	0	6 MHz	6 MHz
0	1	0	1	1	4 MHz	4 MHz
0	0	0	1	1	3 MHz	3 MHz
0	1	1	0	0	2 MHz	2 MHz
0	1	1	0	1	1 MHz	1 MHz
Other than above				Setting prohibited		

**Note** Set the same value as 000C2H to 010C2H when the boot swap operation is used because 000C2H is replaced by 010C2H.

**Caution 1.** Be sure to set bit 5 to 1.

**Caution 2.** The operating frequency range and operating voltage range depend on each operating mode of the flash memory. See 34.4 or 35.4 AC Characteristics for details.

### 29.3 Format of On-chip Debug Option Byte

The format of on-chip debug option byte is shown below.

**Figure 29 - 7 Format of On-chip Debug Option Byte (000C3H/010C3H)**

Address: 000C3H/010C3H Note

	7	6	5	4	3	2	1	0
OCDENSET	0	0	0	0	1	0	OCDERSD	
OCDENSET		OCDERSD	Control of on-chip debug operation					
0	0	0	Disables on-chip debug operation.					
0	1	1	Setting prohibited					
1	0	0	Enables on-chip debugging. Erases data of flash memory in case of failures in authenticating on-chip debug security ID.					
1	1	1	Enables on-chip debugging. Does not erases data of flash memory in case of failures in authenticating on-chip debug security ID.					

**Note** Set the same value as 000C3H to 010C3H when the boot swap operation is used because 000C3H is replaced by 010C3H.

**Caution** Bits 7 and 0 (OCDENSET and OCDERSD) can only be specified a value.  
Be sure to set 000010B to bits 6 to 1.

**Remark** The value on bits 3 to 1 will be written over when the on-chip debug function is in use and thus it will become unstable after the setting.  
However, be sure to set the default values (0, 1, and 0) to bits 3 to 1 at setting.

## 29.4 Setting of Option Byte

The user option byte and on-chip debug option byte can be set using the assembler linker option, in addition to describing in the source. When doing so, the contents set by using the link option take precedence, even if descriptions exist in the source, as mentioned below.

A software description example of the option byte setting is shown below.

OPT	CSEG	OPT_BYTE	
	DB	36H	; Does not use interval interrupt of watchdog timer, ; Enables watchdog timer operation, ; Window open period of watchdog timer is 50%, ; Overflow time of watchdog timer is $2^9/f_{IL}$ , ; Stops watchdog timer operation during HALT/STOP mode
	DB	1AH	; Select 1.63 V for VLVDL ; Select rising edge 1.77 V, falling edge 1.73 V for VLVDH ; Select the interrupt & reset mode as the LVD operation mode
	DB	2DH	; Select the LV (low-voltage main) mode as the flash operation mode and 1 MHz as the frequency of the high-speed on-chip oscillator clock
	DB	85H	; Enables on-chip debug operation, does not erase flash memory data when security ID authorization fails

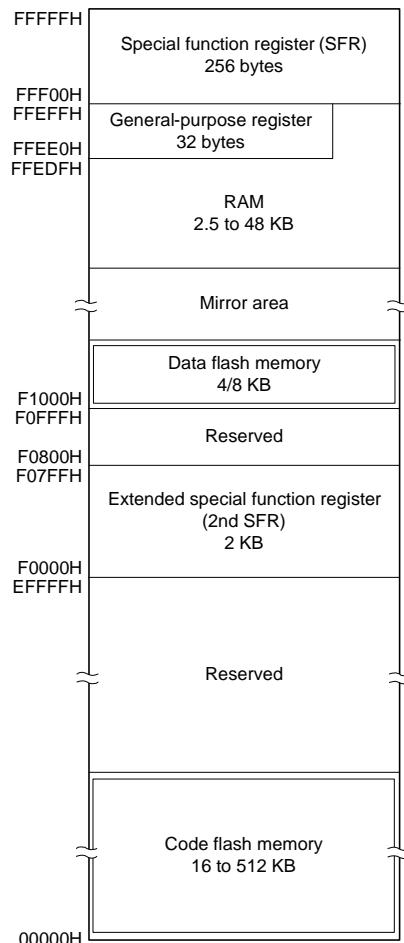
When the boot swap function is used during self-programming, 000C0H to 000C3H is switched to 010C0H to 010C3H. Describe to 010C0H to 010C3H, therefore, the same values as 000C0H to 000C3H as follows.

OPT2	CSEG	AT	010C0H
	DB	36H	; Does not use interval interrupt of watchdog timer, ; Enables watchdog timer operation, ; Window open period of watchdog timer is 50%, ; Overflow time of watchdog timer is $2^9/f_{IL}$ , ; Stops watchdog timer operation during HALT/STOP mode
	DB	1AH	; Select 1.63 V for VLVDL ; Select rising edge 1.77 V, falling edge 1.73 V for VLVDH ; Select the interrupt & reset mode as the LVD operation mode
	DB	2DH	; Select the LV (low main voltage) mode as the flash operation mode and 1 MHz as the frequency of the high-speed on-chip oscillator clock
	DB	85H	; Enables on-chip debug operation, does not erase flash memory data when security ID authorization fails

**Caution** To specify the option byte by using assembly language, use OPT\_BYT as the relocation attribute name of the CSEG pseudo instruction. To specify the option byte to 010C0H to 010C3H in order to use the boot swap function, use the relocation attribute AT to specify an absolute address.

## CHAPTER 30 FLASH MEMORY

The RL78 microcontroller incorporates the flash memory to which a program can be written, erased, and overwritten while mounted on the board. The flash memory includes the “code flash memory”, in which programs can be executed, and the “data flash memory”, an area for storing data.



The following methods for programming the flash memory are available.

The code flash memory can be rewritten to through serial programming using a flash memory programmer or an external device (UART communication), or through self-programming.

- Serial Programming Using Flash Memory Programmer (see **30.1**)

Data can be written to the flash memory on-board or off-board by using a dedicated flash memory programmer.

- Serial Programming Using External Device (that Incorporates UART) (see **30.2**)

Data can be written to the flash memory on-board through UART communication with an external device (microcontroller or ASIC).

- Self-Programming (see **30.6**)

The user application can execute self-programming of the code flash memory by using the flash self-programming library.

The data flash memory can be rewritten to by using the data flash library during user program execution (background operation). For access and writing to the data flash memory, see **30.8 Data Flash**.

### 30.1 Serial Programming Using Flash Memory Programmer

The following dedicated flash memory programmer can be used to write data to the internal flash memory of the RL78 microcontroller.

- PG-FP5, FL-PR5
- E1 on-chip debugging emulator

Data can be written to the flash memory on-board or off-board, by using a dedicated flash memory programmer.

(1) On-board programming

The contents of the flash memory can be rewritten after the RL78 microcontroller has been mounted on the target system. The connectors that connect the dedicated flash memory programmer must be mounted on the target system.

(2) Off-board programming

Data can be written to the flash memory with a dedicated program adapter (FA series) before the RL78 microcontroller is mounted on the target system.

**Remark** FL-PR5 and FA series are products of Naito Densei Machida Mfg. Co., Ltd.

**Table 30 - 1 Wiring Between RL78/G14 and Dedicated Flash Memory Programmer**

Pin Configuration of Dedicated Flash Memory Programmer			Pin Name	Pin No.						
Signal Name		I/O		30-pin	32-pin	36-pin	40-pin	44-pin	48-pin	
PG-FP5, FL-PR5	E1 on-chip debugging emulator			SSOP	WQFN (5 × 5), QFP (7 × 7)	FLGA (4 × 4)	WQFN (6 × 6)	LQFP (10 × 10)	LQFP (7 × 7), WQFN (7 × 7)	
SI/RxD	TOOL0	I/O	Transmit/ receive signal	TOOL0/ P40	5	1	F6	1	2	39
/RESET	RESET	Output	Reset signal	RESET	6	2	E5	2	3	40
V <sub>DD</sub>		I/O	V <sub>DD</sub> voltage generation/ power monitoring	V <sub>DD</sub>	12	8	B6	10	11	48
GND		—	Ground	Vss	11	7	C5	9	10	47
				REGC Note	10	6	D5	8	9	46
FLMD1	EMV <sub>DD</sub>	—	Driving power for TOOL0 pin	V <sub>DD</sub>	12	8	B6	10	11	48

Pin Configuration of Dedicated Flash Memory Programmer			Pin Name	Pin No.						
Signal Name		I/O		52-pin	64-pin		80-pin	100-pin		
PG-FP5, FL-PR5	E1 on-chip debugging emulator			LQFP (10 × 10)	LQFP (14 × 14), LQFP (12 × 12), LQFP (10 × 10)	FLGA (5 × 5)	LQFP (14 × 14), LQFP (12 × 12)	LQFP (14 × 14)	LQFP (14 × 20)	
SI/RxD	TOOL0	I/O	Transmit/ receive signal	TOOL0/ P40	4	5	D6	9	12	89
/RESET	RESET	Output	Reset signal	RESET	5	6	E7	10	13	90
V <sub>DD</sub>		I/O	V <sub>DD</sub> voltage generation/ power monitoring	V <sub>DD</sub>	13	15	B7	19	22	99
GND		—	Ground	Vss	12	13	C7	17	20	97
				EVss	—	14	B8	18	21, 43	98, 20
				REGC Note	11	12	D7	16	19	96
FLMD1	EMV <sub>DD</sub>	—	Driving power for TOOL0 pin	V <sub>DD</sub>	13	—	—	—	—	—
				EV <sub>DD0</sub>	—	16	A8	20	23, 53	100, 30

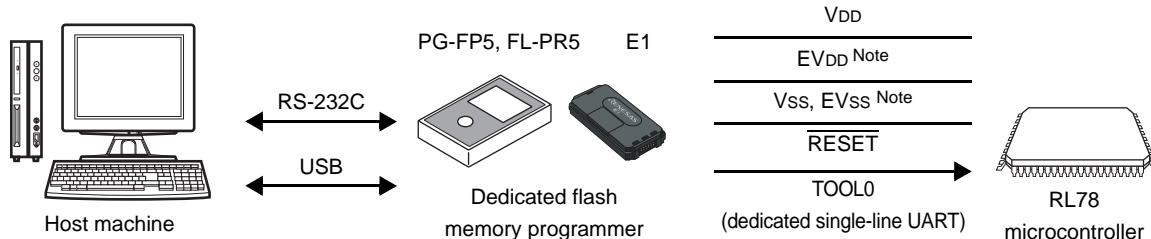
**Note** Connect REGC pin to ground via a capacitor (0.47 to 1  $\mu$ F).

**Remark** Pins that are not indicated in the above table can be left open when using the flash memory programmer for flash programming.

### 30.1.1 Programming Environment

The environment required for writing a program to the flash memory of the RL78 microcontroller is illustrated below.

**Figure 30 - 1 Environment for Writing Program to Flash Memory**



**Note** 64, 80, 100-pin products only.

A host machine that controls the dedicated flash memory programmer is necessary.

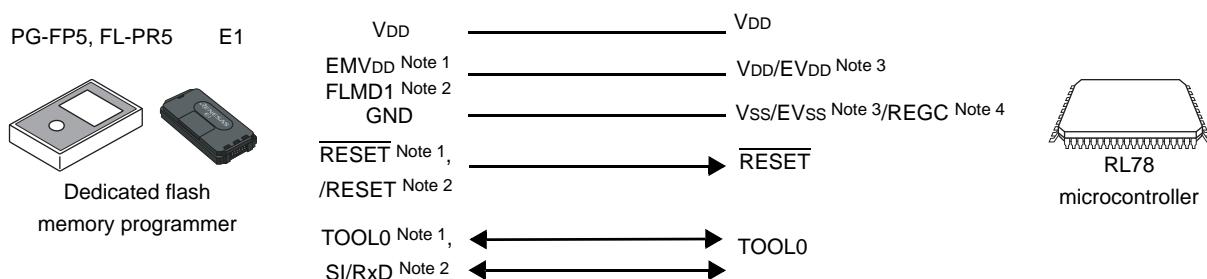
To interface between the dedicated flash memory programmer and the RL78 microcontroller, the TOOL0 pin is used for manipulation such as writing and erasing via a dedicated single-line UART.

### 30.1.2 Communication Mode

Communication between the dedicated flash memory programmer and the RL78 microcontroller is established by serial communication using the TOOL0 pin via a dedicated single-line UART of the RL78 microcontroller.

Transfer rate: 1 M, 500 k, 250 k, 115.2 kbps

**Figure 30 - 2 Communication with Dedicated Flash Memory Programmer**



**Note 1.** When using E1 on-chip debugging emulator.

**Note 2.** When using PG-FP5 or FL-PR5.

**Note 3.** 64, 80, and 100-pin products only.

**Note 4.** Connect REGC pin to ground via a capacitor (0.47 to 1  $\mu$ F).

The dedicated flash memory programmer generates the following signals for the RL78 microcontroller. See the manual of PG-FP5, FL-PR5, or E1 on-chip debugging emulator for details.

**Table 30 - 2 Pin Connection**

Dedicated Flash Memory Programmer			RL78 microcontroller
Signal Name		I/O	Pin Function
PG-FP5, FL-PR5	E1 on-chip debugging emulator		Pin Name Note 2
VDD		I/O	VDD voltage generation/power monitoring
GND		—	Ground
FLMD1	EMVDD	—	Driving power for TOOL0 pin
/RESET	<u>RESET</u>	Output	Reset signal
SI/RxD	TOOL0	I/O	Transmit/receive signal

**Note 1.** Connect REGC pin to ground via a capacitor (0.47 to 1  $\mu$ F).

**Note 2.** Pins to be connected differ with the product. For details, see **Table 30 - 1**.

## 30.2 Serial Programming Using External Device (that Incorporates UART)

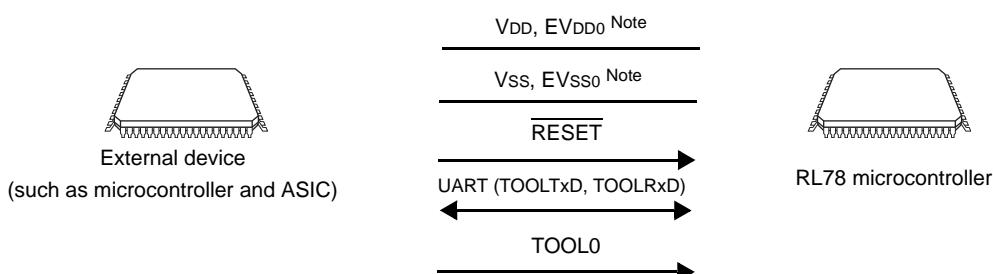
On-board data writing to the internal flash memory is possible by using the RL78 microcontroller and an external device (a microcontroller or ASIC) connected to a UART.

On the development of flash memory programmer by user, refer to the RL78 Microcontrollers (RL78 Protocol A) Programmer Edition Application Note (R01AN0815).

### 30.2.1 Programming Environment

The environment required for writing a program to the flash memory of the RL78 microcontroller is illustrated below.

**Figure 30 - 3 Environment for Writing Program to Flash Memory**



**Note** 64, 80, and 100-pin products only.

Processing to write data to or delete data from the RL78 microcontroller by using an external device is performed on-board. Off-board writing is not possible.

### 30.2.2 Communication Mode

Communication between the external device and the RL78 microcontroller is established by serial communication using the TOOLTxD and TOOLRxD pins via the dedicated UART of the RL78 microcontroller.

Transfer rate: 1 M, 500 k, 250 k, 115.2 kbps

**Figure 30 - 4 Communication with External Device**



**Note 1.** Connect REGC pin to ground via a capacitor (0.47 to 1  $\mu$ F).

**Note 2.** 64, 80, and 100-pin products only.

The external device generates the following signals for the RL78 microcontroller.

**Table 30 - 3 Pin Connection**

External Device			RL78 microcontroller
Signal Name	I/O	Pin Function	Pin Name
Vdd	I/O	Vdd voltage generation/power monitoring	Vdd, EVdd0 Note 2
GND	—	Ground	Vss, EVss0 Note 2, REGC Note 1
RESETOUT	Output	Reset signal output	RESET
RxD	Input	Receive signal	TOOLTxD
TxD	Output	Transmit signal	TOOLRxD
PORT	Output	Mode signal	TOOL0

**Note 1.** Connect REGC pin to ground via a capacitor (0.47 to 1  $\mu$ F).

**Note 2.** 64, 80, and 100-pin products only.

### 30.3 Connection of Pins on Board

To write the flash memory on-board by using the flash memory programmer, connectors that connect the dedicated flash memory programmer must be provided on the target system. First provide a function that selects the normal operation mode or flash memory programming mode on the board.

When the flash memory programming mode is set, all the pins not used for programming the flash memory are in the same status as immediately after reset. Therefore, if the external device does not recognize the state immediately after reset, the pins must be handled as described below.

**Remark** For the flash programming mode, see **30.4.2 Flash memory programming mode**.

#### 30.3.1 P40/TOOL0 pin

In the flash memory programming mode, connect this pin to the dedicated flash memory programmer via an external  $1\text{ k}\Omega$  pull-up resistor.

When this pin is used as the port pin, use that by the following method.

When used as an input pin: Input of low-level is prohibited for  $t_{HD}$  period after external reset release. However, when this pin is used via pull-down resistors, use the  $500\text{ k}\Omega$  or more resistors.

When used as an output pin: When this pin is used via pull-down resistors, use the  $500\text{ k}\Omega$  or more resistors.

**Remark 1.**  $t_{HD}$ : How long to keep the TOOL0 pin at the low level from when the external and internal resets end for setting of the flash memory programming mode (see **34.10** or **35.10 Timing of Entry to Flash Memory Programming Modes**).

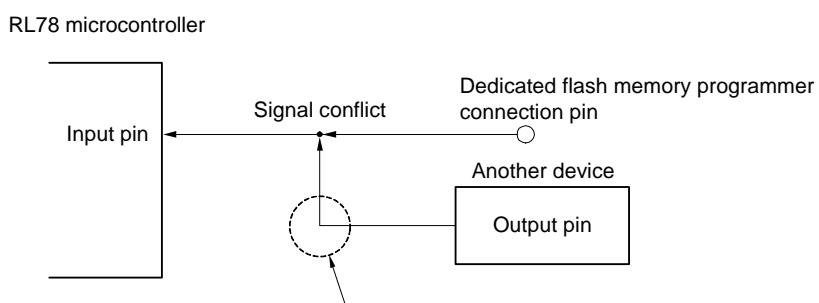
**Remark 2.** The SAU and IICA pins are not used for communication between the RL78 microcontroller and dedicated flash memory programmer, because single-line UART (TOOL0 pin) is used.

#### 30.3.2 RESET pin

Signal conflict will occur if the reset signal of the dedicated flash memory programmer and external device are connected to the RESET pin that is connected to the reset signal generator on the board. To prevent this conflict, isolate the connection with the reset signal generator.

The flash memory will not be correctly programmed if the reset signal is input from the user system while the flash memory programming mode is set. Do not input any signal other than the reset signal of the dedicated flash memory programmer and external device.

**Figure 30 - 5 Signal Conflict (RESET Pin)**



In the flash memory programming mode, a signal output by another device will conflict with the signal output by the dedicated flash memory programmer. Therefore, isolate the signal of another device.

### 30.3.3 Port pins

**Example** When the flash memory programming mode is set, all the pins not used for flash memory programming enter the same status as that immediately after reset. If external devices connected to the ports do not recognize the port status immediately after reset, the port pin must be connected to either VDD or EV<sub>D</sub>0, or VSS or EV<sub>S</sub>0, via a resistor.

### 30.3.4 REGC pin

Connect the REGC pin to GND via a capacitor having excellent characteristics (0.47 to 1  $\mu$ F) in the same manner as during normal operation. Also, use a capacitor with good characteristics, since it is used to stabilize internal voltage.

### 30.3.5 X1 and X2 pins

Connect X1 and X2 in the same status as in the normal operation mode.

**Remark** In the flash memory programming mode, the high-speed on-chip oscillator clock ( $f_{IH}$ ) is used.

### 30.3.6 Power supply

To use the supply voltage output of the flash memory programmer, connect the VDD pin to VDD of the flash memory programmer, and the VSS pin to GND of the flash memory programmer.

To use the on-board supply voltage, connect in compliance with the normal operation mode.

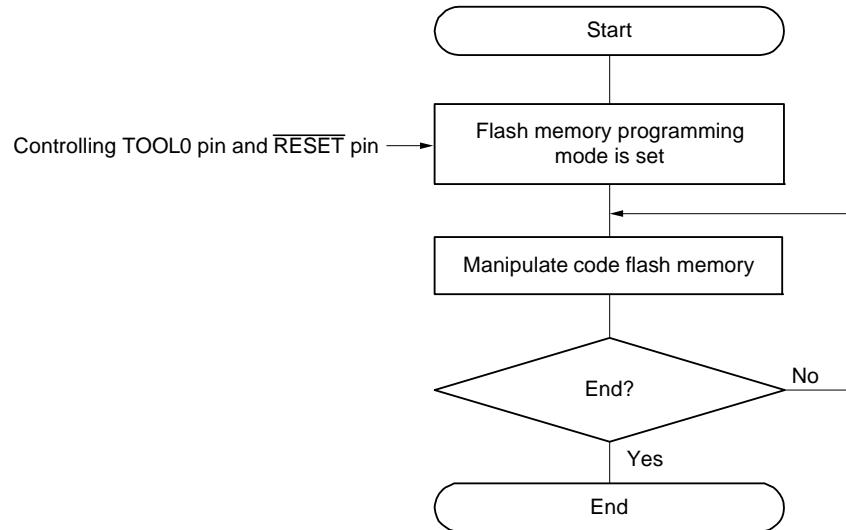
However, when writing to the flash memory by using the flash memory programmer and using the on-board supply voltage, be sure to connect the VDD and VSS pins to VDD and GND of the flash memory programmer to use the power monitor function with the flash memory programmer.

## 30.4 Programming Method

### 30.4.1 Serial programming procedure

The following figure illustrates a flow for rewriting the code flash memory through serial programming.

Figure 30 - 6 Code Flash Memory Manipulation Procedure



### 30.4.2 Flash memory programming mode

To rewrite the contents of the code flash memory through serial programming, specify the flash memory programming mode. To enter the mode, set as follows.

<When serial programming by using the dedicated flash memory programmer>

Connect the RL78 microcontroller to a dedicated flash memory programmer. Communication from the dedicated flash memory programmer is performed to automatically switch to the flash memory programming mode.

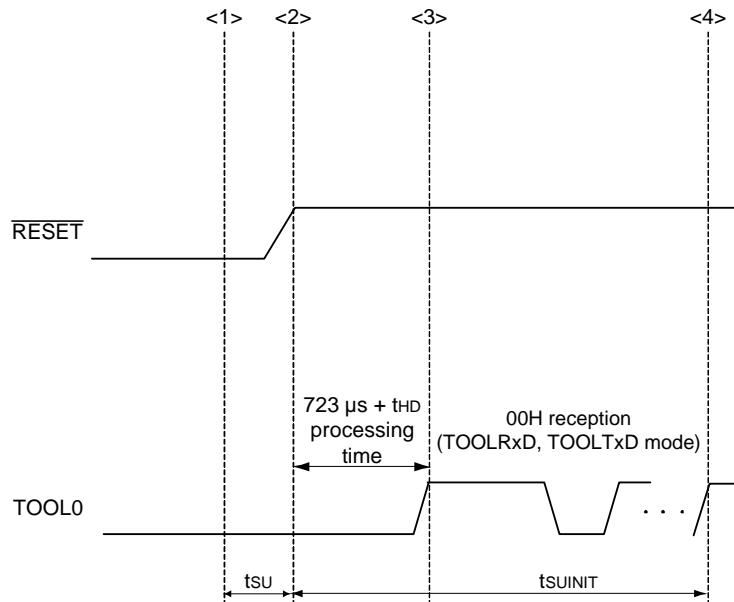
<When serial programming by using an external device>

Set the TOOL0 pin to the low level, and then cancel the reset (see **Table 30 - 4**). After that, enter flash memory programming mode according to the procedures <1> to <4> shown in **Figure 30 - 7**. For details, refer to the **RL78 microcontrollers (RL78 Protocol A) Programmer Edition Application Note (R01AN0815)**.

**Table 30 - 4 Relationship Between TOOL0 Pin and Operation Mode After Reset Release**

TOOL0	Operation Mode
EVDD	Normal operation mode
0 V	Flash memory programming mode

**Figure 30 - 7 Setting of Flash Memory Programming Mode**



<1> The low level is input to the TOOL0 pin.

<2> The external reset ends (POR and LVD reset must end before the external reset ends).

<3> The TOOL0 pin is set to the high level.

<4> Setting of the flash memory programming mode by UART reception and complete the baud rate setting.

<b>Remark</b>	tsUINIT:	The segment shows that it is necessary to finish specifying the initial communication settings within 100 ms from when the external resets end.
	tsU:	How long from when the TOOL0 pin is placed at the low level until a pin reset ends.
	tHD:	How long to keep the TOOL0 pin at the low level from when the external resets end (the flash firmware processing time is excluded).

For details, see **34.10 or 35.10 Timing of Entry to Flash Memory Programming Modes**.

There are two flash memory programming modes: wide voltage mode and full speed mode. The supply voltage value applied to the microcontroller during write operations and the setting information of the user option byte for setting of the flash memory programming mode determine which mode is selected.

When a dedicated flash memory programmer is used for serial programming, setting the voltage on GUI selects the mode automatically.

**Table 30 - 5 Programming Modes and Voltages at Which Data Can Be Written, Erased, or Verified**

Power Supply Voltage (VDD)	User Option Byte Setting for Switching to Flash Memory Programming Mode		Flash Programming Mode
	Flash Operation Mode	Operating Frequency (fCLK)	
2.7 V ≤ VDD ≤ 5.5 V	Blank state		Full speed mode
	HS (high-speed main) mode	1 MHz to 32 MHz	Full speed mode
	LS (low-speed main) mode	1 MHz to 8 MHz	Wide voltage mode
	LV (low-voltage main) mode	1 MHz to 4 MHz	Wide voltage mode
2.4 V ≤ VDD < 2.7 V	Blank state		Full speed mode
	HS (high-speed main) mode	1 MHz to 16 MHz	Full speed mode
	LS (low-speed main) mode	1 MHz to 8 MHz	Wide voltage mode
	LV (low-voltage main) mode	1 MHz to 4 MHz	Wide voltage mode
1.8 V ≤ VDD < 2.4 V	Blank state		Wide voltage mode
	LS (low-speed main) mode	1 MHz to 8 MHz	Wide voltage mode
	LV (low-voltage main) mode	1 MHz to 4 MHz	Wide voltage mode

**Remark 1.** Using both the wide voltage mode and full speed mode imposes no restrictions on writing, erasing, or verification.

**Remark 2.** For details about communication commands, see **30.4.4 Communication commands**.

### 30.4.3 Selecting communication mode

Communication mode of the RL78 microcontroller as follows.

**Table 30 - 6 Communication Modes**

Communication Mode	Standard Setting Note 1				Pins Used
	Port	Speed Note 2	Frequency	Multiply Rate	
1-line mode (when flash memory programmer is used, or when external device is used)	UART	115200 bps, 250000 bps, 500000 bps, 1 Mbps	—	—	TOOL0
Dedicated UART (when external device is used)	UART	115200 bps, 250000 bps, 500000 bps, 1 Mbps	—	—	TOOLTxD, TOOLRxD

**Note 1.** Selection items for Standard settings on GUI of the flash memory programmer.

**Note 2.** Because factors other than the baud rate error, such as the signal waveform slew, also affect UART communication, thoroughly evaluate the slew as well as the baud rate error.

### 30.4.4 Communication commands

The RL78 microcontroller executes serial programming through the commands listed in **Table 30 - 7**.

The signals sent from the dedicated flash memory programmer or external device to the RL78 microcontroller are called commands, and programming functions corresponding to the commands are executed. For details, refer to the **RL78 microcontroller (RL78 Protocol A) Programmer Edition Application Note (R01AN0815)**.

**Table 30 - 7 Flash Memory Control Commands**

Classification	Command Name	Function
Verify	Verify	Compares the contents of a specified area of the flash memory with data transmitted from the programmer.
Erase	Block Erase	Erases a specified area in the flash memory.
Blank check	Block Blank Check	Checks if a specified block in the flash memory has been correctly erased
Write	Programming	Writes data to a specified area in the flash memory <small>Note</small> .
Getting information	Silicon Signature	Gets the RL78 microcontroller information (such as the part number, flash memory configuration, and programming firmware version).
	Checksum	Gets the checksum data for a specified area.
Security	Security Set	Sets security information.
	Security Get	Gets security information.
	Security Release	Release setting of prohibition of writing.
Others	Reset	Used to detect synchronization status of communication.
	Baud Rate Set	Sets baud rate when UART communication mode is selected.

**Note** Confirm that no data has been written to the write area. Because data cannot be erased after block erase is prohibited, do not write data if the data has not been erased.

Product information (such as product name and firmware version) can be obtained by executing the "Silicon Signature" command.

Tables 30 - 8 and 30 - 9 show signature data list and example of signature data list.

**Table 30 - 8 Signature Data List**

Field name	Description	Number of transmit data
Device code	The serial number assigned to the device	3 bytes
Device name	Device name (ASCII code)	10 bytes
Code flash memory area last address	Last address of code flash memory area (Sent from lower address. Example. 00000H to 0FFFFH (64 KB) → FFH, 1FH, 00H)	3 bytes
Data flash memory area last address	Last address of data flash memory area (Sent from lower address. Example. F1000H to F1FFFH (4 KB) → FFH, 1FH, 0FH)	3 bytes
Firmware version	Version information of firmware for programming (Sent from upper address. Example. From Ver. 1.23 → 01H, 02H, 03H)	3 bytes

**Table 30 - 9 Signature Data List**

Field name	Description	Number of transmit data	Data (hexadecimal)
Device code	RL78 protocol A	3 bytes	10 00 06
Device name	R5F104LE	10 bytes	52 = "R" 35 = "5" 46 = "F" 31 = "1" 30 = "0" 34 = "4" 4C = "L" 45 = "E" 20 = " 20 = "
Code flash memory area last address	Code flash memory area 00000H to 0FFFFH (64 KB)	3 bytes	FF FF 00
Data flash memory area last address	Data flash memory area F1000H to F1FFFH (4 KB)	3 bytes	FF 1F 0F
Firmware version	Ver.1.23	3 bytes	01 02 03

### 30.5 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value)

The following shows the processing time for each command (reference value) when PG-FP5 is used as a dedicated flash memory programmer.

**Table 30 - 10 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value)**

PG-FP5 Command	Port: TOOL0 (UART)									
	Speed: 1M bps									
	16 Kbytes	32 Kbytes	48 Kbytes	64 Kbytes	96 Kbytes	128 Kbytes	192 Kbytes	256 Kbytes	384 Kbytes	512 K Kbytes
Erasing	1 s	1 s	1 s	1.5 s	1.5 s	2 s	2 s	2.5 s	3 s	4 s
Writing	1.5 s	1.5 s	2 s	2.5 s	3 s	3.5 s	5 s	6 s	8.5 s	11 s
Verification	1.5 s	1.5 s	2 s	2 s	3 s	3.5 s	4.5 s	5.5 s	8 s	10.5 s
Writing after erasing	1.5 s	2 s	2.5 s	3 s	4 s	4.5 s	6.5 s	8 s	11 s	14.5 s

**Remark** The command processing times (reference values) shown in the table are typical values under the following conditions.

Port: TOOL0 (single-line UART)

Speed: 1,000,000 bps

Mode: Full speed mode (flash operation mode: HS (high speed main) mode)

## 30.6 Self-Programming

The RL78 microcontroller supports a self-programming function that can be used to rewrite the flash memory via a user program. Because this function allows a user application to rewrite the flash memory by using the RL78 microcontroller self-programming library, it can be used to upgrade the program in the field.

**Caution 1.** The self-programming function cannot be used when the CPU operates with the subsystem clock.

**Caution 2.** To prohibit an interrupt during self-programming, in the same way as in the normal operation mode, execute the flash self-programming library in the state where the IE flag is cleared (0) by the DI instruction. To enable an interrupt, clear (0) the interrupt mask flag to accept in the state where the IE flag is set (1) by the EI instruction, and then execute the flash self-programming library.

**Caution 3.** The high-speed on-chip oscillator should be kept operating during self-programming. If it is kept stopped, it should be operated ( $\text{HIOSTOP} = 0$ ). The flash self-programming library should be executed after  $30 \mu\text{s}$  have elapsed when the FRQSEL4 in the user option byte (000C2H) is 0, and after  $80 \mu\text{s}$  have elapsed when the FRQSEL4 is 1.

**Remark 1.** For details of the self-programming function, refer to the **RL78 microcontroller Flash Self-Programming Library Type01 User's Manual (R01US0050)**.

**Remark 2.** For details of the time required to execute self-programming, see the notes on use that accompany the flash self-programming library tool.

The self-programming function has two flash memory programming modes; wide voltage mode and full speed mode. Specify the mode that corresponds to the flash operation mode specified in bits CMODE1 and CMODE0 in option byte 000C2H.

Specify the full speed mode when the HS (high-speed main) mode is specified. Specify the wide voltage mode when the LS (low-speed main) mode or LV (low-voltage main) mode is specified.

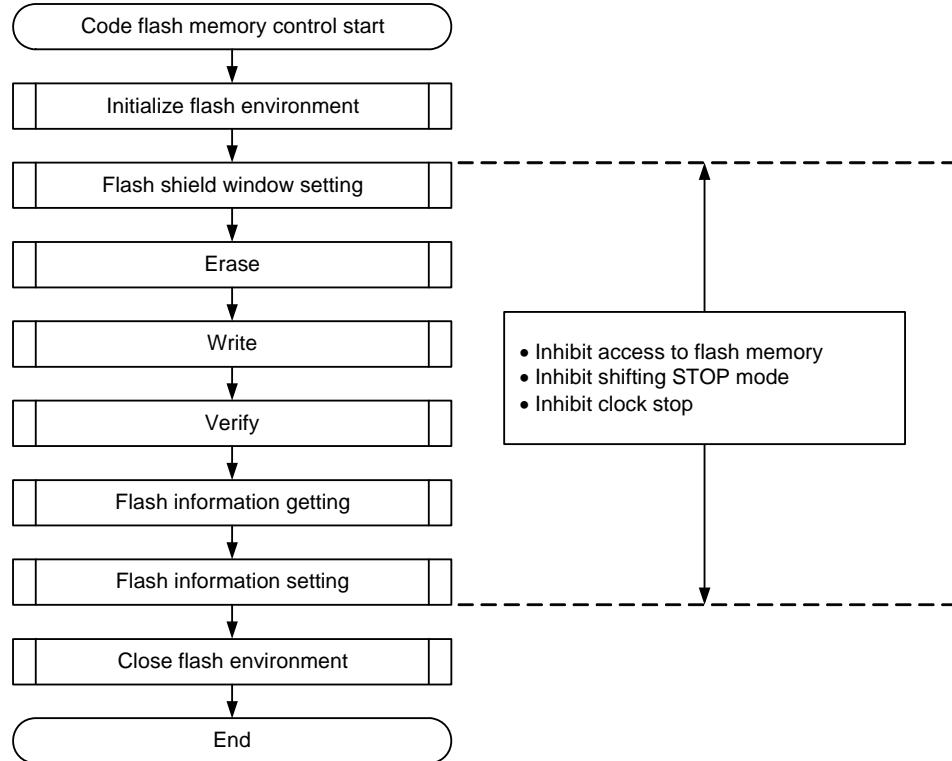
If the argument `fsl_flash_voltage_u08` is 00H when the `FSL_Init` function of the flash self-programming library provided by Renesas Electronics is executed, full speed mode is specified. If the argument is other than 00H, the wide voltage mode is specified.

**Remark** Using both the wide voltage mode and full speed mode imposes no restrictions on writing, erasing, or verification.

### 30.6.1 Self-programming procedure

The following figure illustrates a flow for rewriting the code flash memory by using a flash self-programming library.

Figure 30 - 8 Flow of Self-Programming (Rewriting Flash Memory)



### 30.6.2 Boot swap function

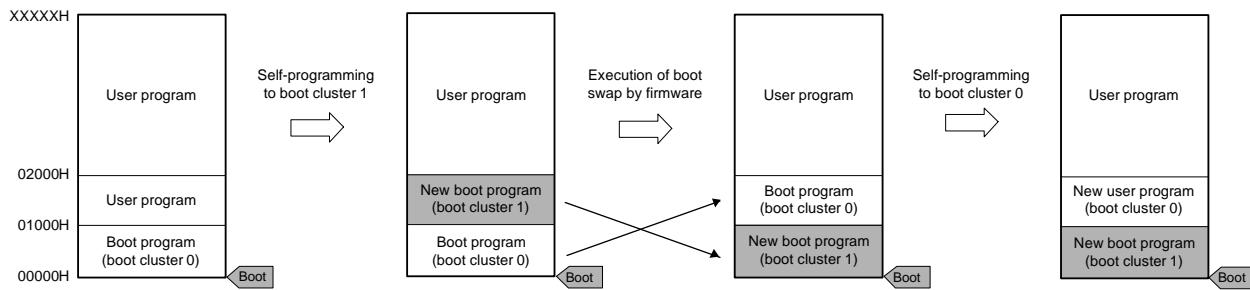
If rewriting the boot area failed by temporary power failure or other reasons, restarting a program by resetting or overwriting is disabled due to data destruction in the boot area.

The boot swap function is used to avoid this problem.

Before erasing boot cluster 0 <sup>Note</sup>, which is a boot program area, by self-programming, write a new boot program to boot cluster 1 in advance. When the program has been correctly written to boot cluster 1, swap this boot cluster 1 and boot cluster 0 by using the set information function of the firmware of the RL78 microcontroller, so that boot cluster 1 is used as a boot area. After that, erase or write the original boot program area, boot cluster 0. As a result, even if a power failure occurs while the area is being rewritten, the program is executed correctly because it is booted from boot cluster 1 to be swapped when the program is reset and started next.

**Note** A boot cluster is a 4 KB area and boot clusters 0 and 1 are swapped by the boot swap function.

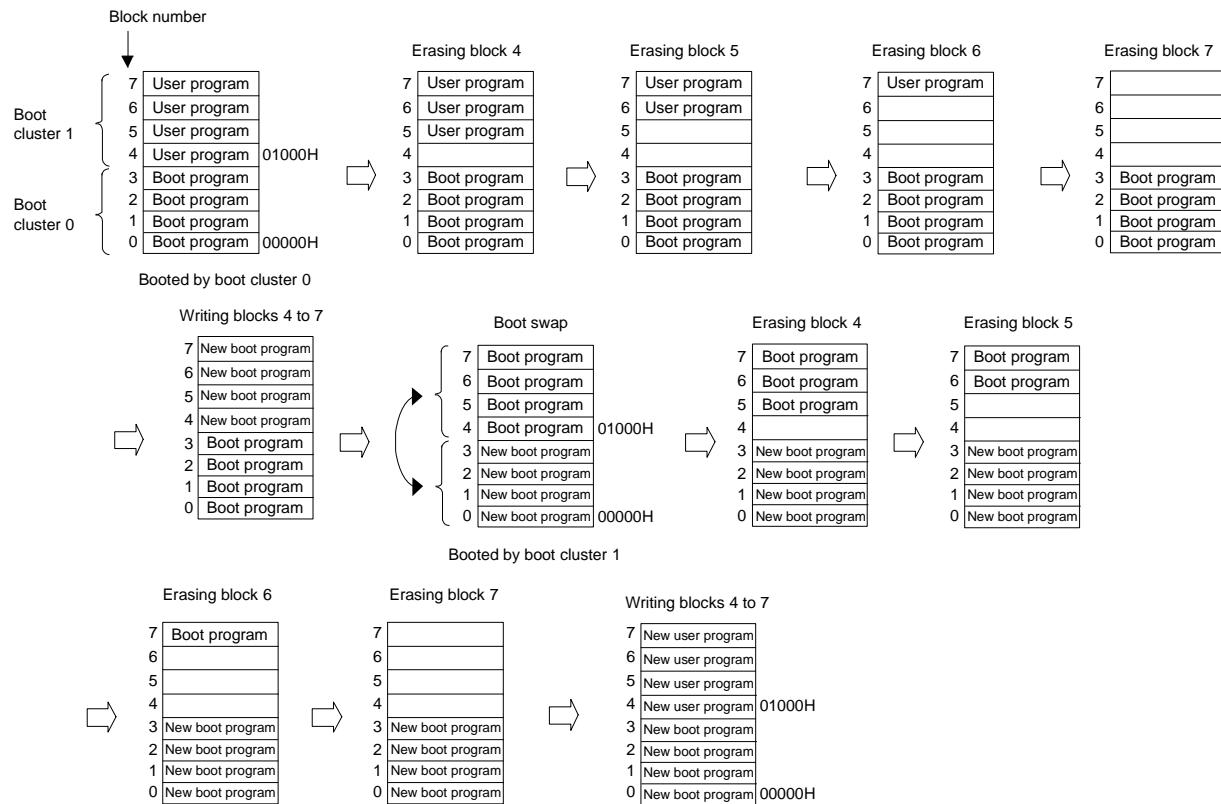
**Figure 30 - 9 Boot Swap Function**



In an example of above figure, it is as follows.

Boot cluster 0: Boot area before boot swap

Boot cluster 1: Boot area after boot swap

**Figure 30 - 10 Example of Executing Boot Swapping**

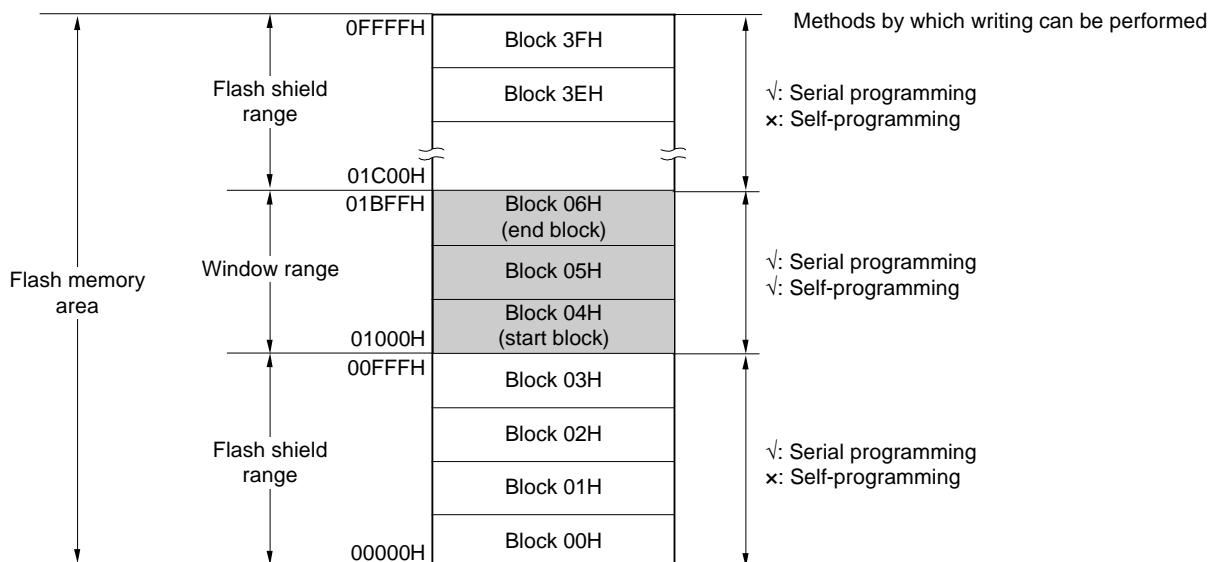
### 30.6.3 Flash shield window function

The flash shield window function is provided as one of the security functions for self-programming. It disables writing to and erasing areas outside the range specified as a window only during self-programming.

The window range can be set by specifying the start and end blocks. The window range can be set or changed during both serial programming and self-programming.

Writing to and erasing areas outside the window range are disabled during self-programming. During serial programming, however, areas outside the range specified as a window can be written and erased.

**Figure 30 - 11 Flash Shield Window Setting Example**  
(Target Devices: R5F104LE, Start Block: 04H, End Block: 06H)



**Caution 1.** If the rewrite-prohibited area of the boot cluster 0 overlaps with the flash shield window range, prohibition to rewrite the boot cluster 0 takes priority.

**Caution 2.** The flash shield window can only be used for the code flash memory (and is not supported for the data flash memory).

**Table 30 - 11 Relationship between Flash Shield Window Function Setting/Change Methods and Commands**

Programming conditions	Window Range Setting/ Change Methods	Execution Commands	
		Block erase	Write
Self-programming	Specify the starting and ending blocks by the flash self-programming library.	Block erasing is enabled only within the window range.	Writing is enabled only within the range of window range.
Serial programming	Specify the starting and ending blocks on GUI of dedicated flash memory programmer, etc.	Block erasing is enabled also outside the window range.	Writing is enabled also outside the window range.

**Remark** See 30.7 Security Settings to prohibit writing/erasing during serial programming.

## 30.7 Security Settings

The RL78 microcontroller supports a security function that prohibits rewriting the user program written to the internal flash memory, so that the program cannot be changed by an unauthorized person.

The operations shown below can be performed using the Security Set command.

- Disabling block erase

Execution of the block erase command for a specific block in the flash memory is prohibited during serial programming. However, blocks can be erased by means of self-programming.

- Disabling write

Execution of the write command for entire blocks in the flash memory is prohibited during serial programming. However, blocks can be written by means of self-programming.

After the setting of prohibition of writing is specified, releasing the setting by the Security Release command is enabled by a reset.

- Disabling rewriting boot cluster 0

Execution of the block erase command and write command on boot cluster 0 (00000H to 00FFFH) in the flash memory is prohibited by this setting.

The block erase, write commands, and rewriting boot cluster 0 are enabled by the default setting when the flash memory is shipped. Security can be set by serial programming and self-programming. Each security setting can be used in combination.

Table 30 - 12 shows the relationship between the erase and write commands when the RL78 microcontroller security function is enabled.

After the security settings are specified, releasing the security settings by the Security Release command is enabled by a reset.

**Caution** **The security function of the flash programmer does not support self-programming.**

**Remark** To prohibit writing and erasing during self-programming, use the flash shield window function (see **30.6.3** for detail).

**Table 30 - 12 Relationship Between Enabling Security Function and Command**

## (1) During serial programming

Valid Security	Executed Command	
	Block Erase	Write
Prohibition of block erase	Blocks cannot be erased.	Can be performed. Note
Prohibition of writing	Blocks can be erased.	Cannot be performed.
Prohibition of rewriting boot cluster 0	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

**Note** Confirm that no data has been written to the write area. Because data cannot be erased after block erase is prohibited, do not write data if the data has not been erased.

## (2) During self-programming

Valid Security	Executed Command	
	Block Erase	Write
Prohibition of block erase	Blocks can be erased.	Can be performed.
Prohibition of writing		
Prohibition of rewriting boot cluster 0	Boot cluster 0 cannot be erased.	Boot cluster 0 cannot be written.

**Remark** To prohibit writing and erasing during self-programming, use the flash shield window function (see 30.6.3 for detail).

**Table 30 - 13 Setting Security in Each Programming Mode**

## (1) During serial programming

Security	Security Setting	How to Disable Security Setting
Prohibition of block erase	Set via GUI of dedicated flash memory programmer, etc.	Cannot be disabled after set.
Prohibition of writing		Set via GUI of dedicated flash memory programmer, etc.
Prohibition of rewriting boot cluster 0		Cannot be disabled after set.

**Caution** Releasing the setting of prohibition of writing is enabled only when the security is not set as the block erase prohibition and the boot cluster 0 rewrite prohibition with code flash memory area and data flash memory area being blanks.

## (2) During self-programming

Security	Security Setting	How to Disable Security Setting
Prohibition of block erase	Set by using flash self-programming library.	Cannot be disabled after set.
Prohibition of writing		Cannot be disabled during self-programming (set via GUI of dedicated flash memory programmer, etc. during serial programming).
Prohibition of rewriting boot cluster 0		Cannot be disabled after set.

## 30.8 Data Flash

### 30.8.1 Data flash overview

An overview of the data flash memory is provided below.

- The user program can rewrite the data flash memory by using the flash data library. For details, refer to RL78 Family Flash Data Library User's Manual.
- The data flash memory can also be rewritten through serial programming using the dedicated flash memory programmer or an external device.
- The data flash can be erased in 1-block (1 KB) units.
- The data flash can be accessed only in 8-bit units.
- The data flash can be directly read by CPU instructions.
- Instructions can be executed from the code flash memory while rewriting the data flash memory (that is, background operation (BGO) is supported).
- Because the data flash memory is an area exclusively used for data, it cannot be used to execute instructions.
- Accessing the data flash memory is not possible while rewriting the code flash memory (during self-programming).
- Manipulating the DFLCTL register is not possible while rewriting the data flash memory.
- Transition to the STOP mode is not possible while rewriting the data flash memory.

**Caution 1. The data flash memory is stopped after a reset is canceled. The data flash control register (DFLCTL) must be set up in order to use the data flash memory.**

**Caution 2. The high-speed on-chip oscillator should be kept operating during data flash rewrite. If it is kept stopped, it should be operated ( $\text{HIOSTOP} = 0$ ). The flash self-programming library should be executed after 30  $\mu\text{s}$  have elapsed when the FRQSEL4 in the user option byte (000C2H) is 0, and after 80  $\mu\text{s}$  have elapsed when the FRQSEL4 is 1.**

**Remark** For the flash programming mode, see [30.6 Self-Programming](#).

### 30.8.2 Register controlling data flash memory

#### 30.8.2.1 Data flash control register (DFLCTL)

This register is used to enable or disable accessing to the data flash.

The DFLCTL register is set by a 1-bit or 8-bit memory manipulation instruction.

Reset input sets this register to 00H.

**Figure 30 - 12 Format of Data flash control register (DFLCTL)**

Address: F0090H      After reset: 00H      R/W

Symbol      7      6      5      4      3      2      1      <0>

DFLCTL	0	0	0	0	0	0	0	DFLEN
DFLEN	Data flash access control							
0	Disables data flash access							
1	Enables data flash access							

**Caution** Manipulating the DFLCTL register is not possible while rewriting the data flash memory.

### 30.8.3 Procedure for accessing data flash memory

The data flash memory is initially stopped after a reset ends and cannot be accessed (read or programmed). To access the memory, perform the following procedure:

<1> Write 1 to bit 0 (DFLEN) of the data flash control register (DFLCTL).

<2> Wait for the setup to finish for software timer, etc.

The time setup takes differs for each main clock mode.

<Setup time for each main clock mode>

- HS (High-speed main): 5  $\mu$ s
- LS (Low-speed main): 720 ns
- LV (Low-voltage main): 10  $\mu$ s

<3> After the wait, the data flash memory can be accessed.

**Caution 1. Accessing the data flash memory is not possible during the setup time.**

**Caution 2. Transition to the STOP mode is not possible during the setup time. To enter the STOP mode during the setup time, clear DFLEN to 0 and then execute the STOP instruction.**

**Caution 3. The high-speed on-chip oscillator should be kept operating during data flash rewrite. If it is kept stopped, it should be operated (HIOSTOP = 0). The flash self-programming library should be executed after 30  $\mu$ s have elapsed when the FRQSEL4 in the user option byte (000C2H) is 0, and after 80  $\mu$ s have elapsed when the FRQSEL4 is 1.**

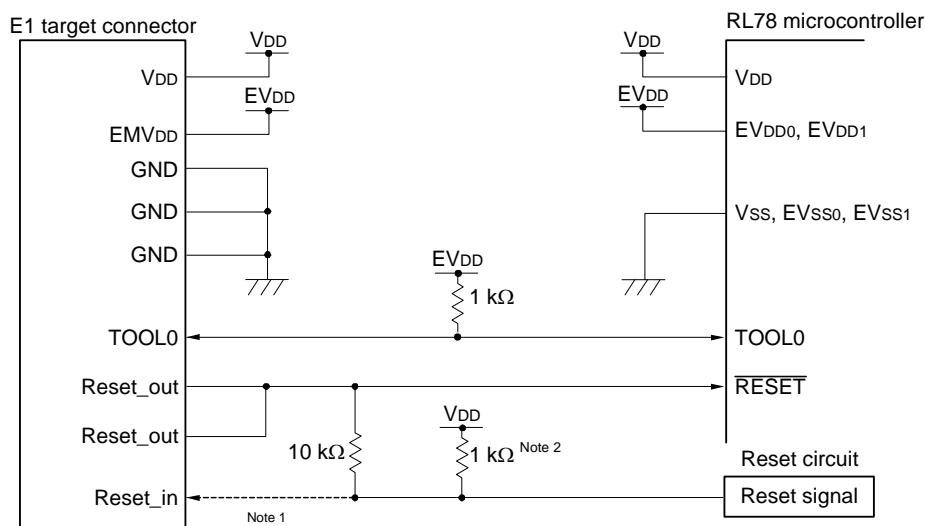
## CHAPTER 31 ON-CHIP DEBUG FUNCTION

### 31.1 Connecting E1 On-chip Debugging Emulator

The RL78 microcontroller uses the VDD, RESET, TOOL0, and Vss pins to communicate with the host machine via an E1 on-chip debugging emulator. Serial communication is performed by using a single-line UART that uses the TOOL0 pin.

**Caution** The RL78 microcontroller has an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.

Figure 31 - 1 Connection Example of E1 On-chip Debugging Emulator



**Note 1.** Connecting the dotted line is not necessary during serial programming.

**Note 2.** If the reset circuit on the target system does not have a buffer and generates a reset signal only with resistors and capacitors, this pull-up resistor is not necessary.

**Caution** This circuit diagram is assumed that the reset signal outputs from an N-ch O.D. buffer (output resistor: 100 Ω or less)

**Remark** With products not provided with an EVDD0, EVDD1, EVss0, or EVss1 pin, replace EVDD0 and EVDD1 with VDD, or replace EVss0 and EVss1 with Vss.

## 31.2 On-Chip Debug Security ID

The RL78 microcontroller has an on-chip debug operation control bit in the flash memory at 000C3H (see **CHAPTER 29 OPTION BYTE**) and an on-chip debug security ID setting area at 000C4H to 000CDH, to prevent third parties from reading memory content.

When the boot swap function is used, also set a value that is the same as that of 010C3H and 010C4H to 010CDH in advance, because 000C3H, 000C4H to 000CDH and 010C3H, and 010C4H to 010CDH are switched.

**Table 31 - 1 On-Chip Debug Security ID**

Address	On-Chip Debug Security ID
000C4H to 000CDH	Any ID code of 10 bytes <small>Note</small>
010C4H to 010CDH	

**Note** The setting FFFFFFFFFFFFFFFFFFFFH for the ID code is not possible.

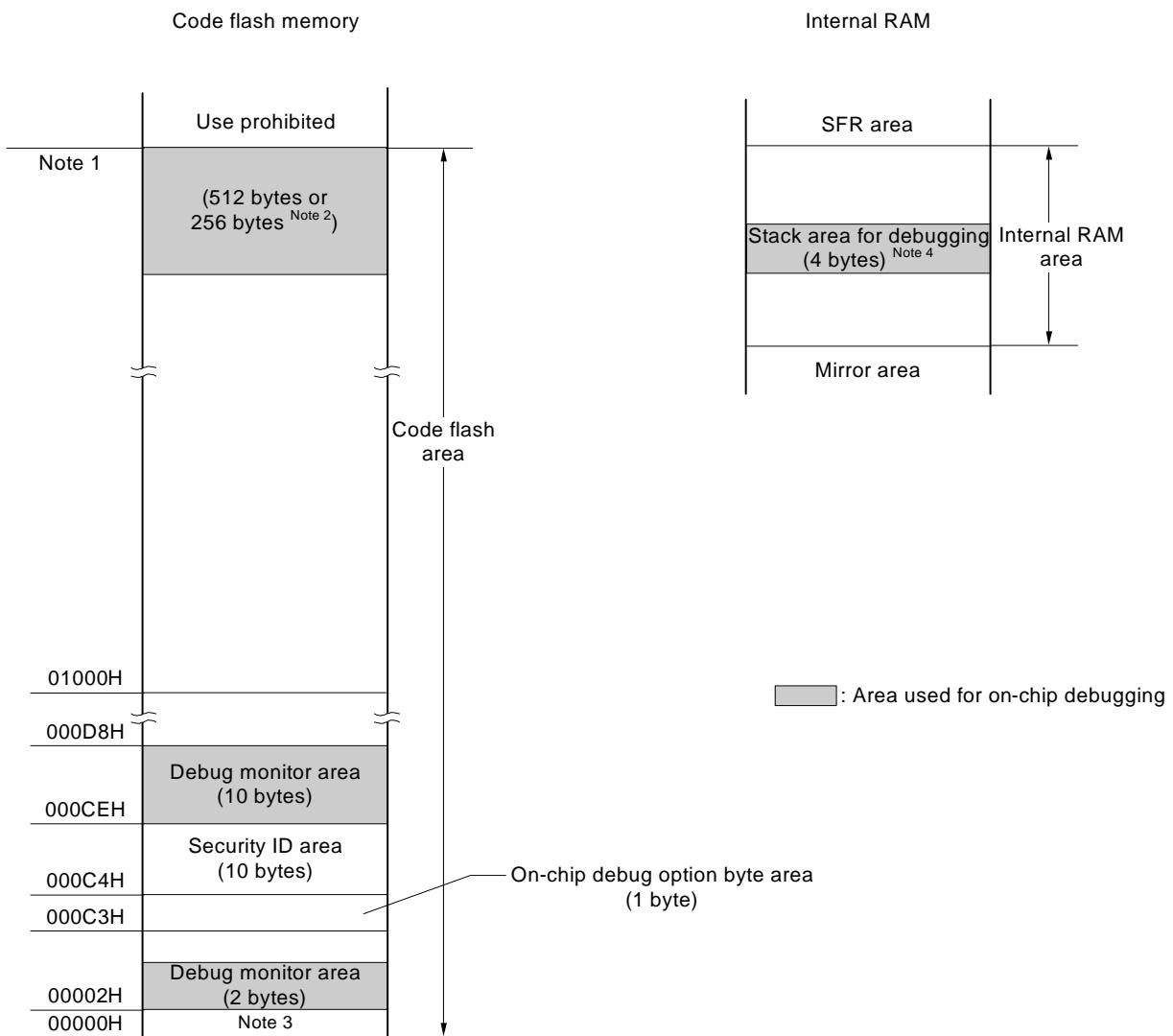
## 31.3 Securing of User Resources

To perform communication between the RL78 microcontroller and E1 on-chip debugging emulator, as well as each debug function, the securing of memory space must be done beforehand.

If Renesas Electronics assembler or compiler is used, the items can be set by using link options.

### (1) Securement of memory space

The shaded portions in Figure 31 - 2 are the areas reserved for placing the debug monitor program, so user programs or data cannot be allocated in these spaces. When using the on-chip debug function, these spaces must be secured so as not to be used by the user program. Moreover, this area must not be rewritten by the user program.

**Figure 31 - 2 Memory Spaces Where Debug Monitor Programs Are Allocated**

**Note 1.** Address differs depending on products as follows.

Products (code flash memory capacity)	Address of <b>Note 1.</b>
R5F104xA (x = A to C, E to G)	03FFFH
R5F104xC (x = A to C, E to G, J, L)	07FFFH
R5F104xD (x = A to C, E to G, J, L)	0BFFFH
R5F104xE (x = A to C, E to G, J, L)	0FFFFH
R5F104xF (x = A to C, E to G, J, L, M, P)	17FFFH
R5F104xG (x = A to C, E to G, J, L, M, P)	1FFFFFFH
R5F104xH (x = E to G, J, L, M, P)	2FFFFFFH
R5F104xJ (x = F, G, J, L, M, P)	3FFFFFFH
R5F104xK (x = G, L, M, P)	5FFFFFFH
R5F104xL (x = G, L, M, P)	7FFFFFFH

**Note 2.** When real-time RAM monitor (RRM) function and dynamic memory modification (DMM) function are not used, it is 256 bytes.

**Note 3.** In debugging, reset vector is rewritten to address allocated to a monitor program.

**Note 4.** Since this area is allocated immediately before the stack area, the address of this area varies depending on the stack increase and decrease. That is, 4 extra bytes are consumed for the stack area used.

When using self-programming, 12 extra bytes are consumed for the stack area used.

## CHAPTER 32 BCD CORRECTION CIRCUIT

### 32.1 BCD Correction Circuit Function

The result of addition/subtraction of the BCD (binary-coded decimal) code and BCD code can be obtained as BCD code with this circuit.

The decimal correction operation result is obtained by performing addition/subtraction having the A register as the operand and then adding/ subtracting the BCD correction result register (BCDADJ).

### 32.2 Registers Used by BCD Correction Circuit

The BCD correction circuit uses the following registers.

- BCD correction result register (BCDADJ)

#### 32.2.1 BCD correction result register (BCDADJ)

The BCDADJ register stores correction values for obtaining the add/subtract result as BCD code through add/subtract instructions using the A register as the operand.

The value read from the BCDADJ register varies depending on the value of the A register when it is read and those of the CY and AC flags.

The BCDADJ register is read by an 8-bit memory manipulation instruction.

Reset input sets this register to undefined.

**Figure 32 - 1 Format of BCD correction result register (BCDADJ)**

Address: F00FEH      After reset: Undefined      R

Symbol	7	6	5	4	3	2	1	0
BCDADJ								

### 32.3 BCD Correction Circuit Operation

The basic operation of the BCD correction circuit is as follows.

- (1) Addition: Calculating the result of adding a BCD code value and another BCD code value by using a BCD code value

- <1> The BCD code value to which addition is performed is stored in the A register.
- <2> By adding the value of the A register and the second operand (value of one more BCD code to be added) as are in binary, the binary operation result is stored in the A register and the correction value is stored in the BCD correction result register (BCDADJ).
- <3> Decimal correction is performed by adding in binary the value of the A register (addition result in binary) and the BCDADJ register (correction value), and the correction result is stored in the A register and CY flag.

**Caution** The value read from the BCDADJ register varies depending on the value of the A register when it is read and those of the CY and AC flags. Therefore, execute the instruction <3> after the instruction <2> instead of executing any other instructions. To perform BCD correction in the interrupt enabled state, saving and restoring the A register is required within the interrupt function. PSW (CY flag and AC flag) is restored by the RETI instruction.

An example is shown below.

Examples 1:  $99 + 89 = 188$

Instruction		A Register	CY Flag	AC Flag	BCDADJ Register
MOV	A, #99H ;<1>	99H	—	—	—
ADD	A, #89H ;<2>	22H	1	1	66H
ADD	A, !BCDADJ ;<3>	88H	1	0	—

Examples 2:  $85 + 15 = 100$

Instruction		A Register	CY Flag	AC Flag	BCDADJ Register
MOV	A, #85H ;<1>	85H	—	—	—
ADD	A, #15H ;<2>	9AH	0	0	66H
ADD	A, !BCDADJ ;<3>	00H	1	1	—

Examples 3:  $80 + 80 = 160$

Instruction		A Register	CY Flag	AC Flag	BCDADJ Register
MOV	A, #80H ;<1>	80H	—	—	—
ADD	A, #80H ;<2>	00H	1	0	60H
ADD	A, !BCDADJ ;<3>	60H	1	0	—

- (2) Subtraction: Calculating the result of subtracting a BCD code value from another BCD code value by using a BCD code value

- <1> The BCD code value from which subtraction is performed is stored in the A register.
- <2> By subtracting the value of the second operand (value of BCD code to be subtracted) from the A register as is in binary, the calculation result in binary is stored in the A register, and the correction value is stored in the BCD correction result register (BCDADJ).
- <3> Decimal correction is performed by subtracting the value of the BCDADJ register (correction value) from the A register (subtraction result in binary) in binary, and the correction result is stored in the A register and CY flag.

**Caution** The value read from the BCDADJ register varies depending on the value of the A register when it is read and those of the CY and AC flags. Therefore, execute the instruction <3> after the instruction <2> instead of executing any other instructions. To perform BCD correction in the interrupt enabled state, saving and restoring the A register is required within the interrupt function. PSW (CY flag and AC flag) is restored by the RETI instruction.

An example is shown below.

Example:  $91 - 52 = 39$

Instruction			A Register	CY Flag	AC Flag	BCDADJ Register
MOV	A, #91H	; <1>	91H	—	—	—
SUB	A, #52H	; <2>	3FH	0	1	06H
SUB	A, !BCDADJ	; <3>	39H	0	0	—

## CHAPTER 33 INSTRUCTION SET

This chapter lists the instructions in the RL78 microcontroller instruction set. For details of each operation and operation code, refer to the separate document **RL78 Family User's Manual Software (R01US0015)**.

## 33.1 Conventions Used in Operation List

### 33.1.1 Operand identifiers and specification methods

Operands are described in the “Operand” column of each instruction in accordance with the description method of the instruction operand identifier (refer to the assembler specifications for details). When there are two or more description methods, select one of them. Alphabetic letters in capitals and the symbols, #, !, !!, \$, \$!, [ ], and ES: are keywords and are described as they are. Each symbol has the following meaning.

- #: Immediate data specification
- !: 16-bit absolute address specification
- !!: 20-bit absolute address specification
- \$: 8-bit relative address specification
- \$!: 16-bit relative address specification
- [ ]: Indirect address specification
- ES:: Extension address specification

In the case of immediate data, describe an appropriate numeric value or a label. When using a label, be sure to describe the #, !, !!, \$, \$!, [ ], and ES: symbols.

For operand register identifiers, r and rp, either function names (X, A, C, etc.) or absolute names (names in parentheses in the table below, R0, R1, R2, etc.) can be used for description.

**Table 33 - 1 Operand Identifiers and Specification Methods**

Identifier	Description Method
r rp sfr sfrp	X (R0), A (R1), C (R2), B (R3), E (R4), D (R5), L (R6), H (R7) AX (RP0), BC (RP1), DE (RP2), HL (RP3) Special-function register symbol (SFR symbol) FFF00H to FFFFFH Special-function register symbols (16-bit manipulatable SFR symbol. Even addresses only <small>Note</small> ) FFF00H to FFFFFH
saddr saddrp	FFE20H to FFF1FH Immediate data or labels FFE20H to FF1FH Immediate data or labels (even addresses only <small>Note</small> )
addr20 addr16 addr5	00000H to FFFFFH Immediate data or labels 0000H to FFFFH Immediate data or labels (only even addresses for 16-bit data transfer instructions <small>Note</small> ) 0080H to 00BFH Immediate data or labels (even addresses only)
word byte bit	16-bit immediate data or label 8-bit immediate data or label 3-bit immediate data or label
RBn	RB0 to RB3

**Note** Bit 0 = 0 when an odd address is specified.

**Remark** The special function registers can be described to operand sfr as symbols. See **Tables 3 - 9 to 3 - 13 Special Function Register (SFR) List** for the symbols of the special function registers. The extended special function registers can be described to operand !addr16 as symbols. See **Tables 3 - 14 to 3 - 22 Extended Special Function Register (2nd SFR) List** for the symbols of the extended special function registers.

### 33.1.2 Description of operation column

The operation when the instruction is executed is shown in the “Operation” column using the following symbols.

**Table 33 - 2 Symbols in “Operation” Column**

Symbol	Function
A	A register; 8-bit accumulator
X	X register
B	B register
C	C register
D	D register
E	E register
H	H register
L	L register
ES	ES register
CS	CS register
AX	AX register pair; 16-bit accumulator
BC	BC register pair
DE	DE register pair
HL	HL register pair
PC	Program counter
SP	Stack pointer
PSW	Program status word
CY	Carry flag
AC	Auxiliary carry flag
Z	Zero flag
RBS	Register bank select flag
IE	Interrupt request enable flag
()	Memory contents indicated by address or register contents in parentheses
XH, XL	16-bit registers: XH = higher 8 bits, XL = lower 8 bits
Xs, XH, XL	20-bit registers: Xs = (bits 19 to 16), XH = (bits 15 to 8), XL = (bits 7 to 0)
^	Logical product (AND)
∨	Logical sum (OR)
⊻	Exclusive logical sum (exclusive OR)
—	Inverted data
addr5	16-bit immediate data (even addresses only in 0080H to 00BFH)
addr16	16-bit immediate data
addr20	20-bit immediate data
jdisp8	Signed 8-bit data (displacement value)
jdisp16	Signed 16-bit data (displacement value)

### 33.1.3 Description of flag operation column

The change of the flag value when the instruction is executed is shown in the “Flag” column using the following symbols.

**Table 33 - 3 Symbols in “Flag” Column**

Symbol	Change of Flag Value
(Blank)	Unchanged
0	Cleared to 0
1	Set to 1
x	Set/cleared according to the result
R	Previously saved value is restored

### 33.1.4 PREFIX instruction

Instructions with “ES:” have a PREFIX operation code as a prefix to extend the accessible data area to the 1 MB space (00000H to FFFFFH), by adding the ES register value to the 64 KB space from F0000H to FFFFFH. When a PREFIX operation code is attached as a prefix to the target instruction, only one instruction immediately after the PREFIX operation code is executed as the addresses with the ES register value added.

A interrupt and DTC transfer are not acknowledged between a PREFIX instruction code and the instruction immediately after.

**Table 33 - 4 Use Example of PREFIX Operation Code**

Instruction	Opcode				
	1	2	3	4	5
MOV !addr16, #byte	CFH		!addr16	#byte	—
MOV ES:!addr16, #byte	11H	CFH		!addr16	#byte
MOV A, [HL]	8BH	—	—	—	—
MOV A, ES: [HL]	11H	8BH	—	—	—

**Caution** Set the ES register value with MOV ES, A, etc., before executing the PREFIX instruction.

## 33.2 Operation List

**Table 33 - 5 Operation List (1/18)**

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	MOV	r, #byte	2	1	—	r ← byte			
		PSW, #byte	3	3	—	PSW ← byte	x	x	x
		CS, #byte	3	1	—	CS ← byte			
		ES, #byte	2	1	—	ES ← byte			
		!addr16, #byte	4	1	—	(addr16) ← byte			
		ES:!addr16, #byte	5	2	—	((ES, addr16) ← byte			
		saddr, #byte	3	1	—	((saddr) ← byte			
		sfr, #byte	3	1	—	sfr ← byte			
		[DE+byte], #byte	3	1	—	((DE + byte) ← byte			
		ES:[DE+byte], #byte	4	2	—	((ES, DE) + byte) ← byte			
		[HL+byte], #byte	3	1	—	((HL + byte) ← byte			
		ES:[HL+byte], #byte	4	2	—	((ES, HL) + byte) ← byte			
		[SP+byte], #byte	3	1	—	((SP + byte) ← byte			
		word[B], #byte	4	1	—	((B + word) ← byte			
		ES:word[B], #byte	5	2	—	((ES, B) + word) ← byte			
		word[C], #byte	4	1	—	((C + word) ← byte			
		ES:word[C], #byte	5	2	—	((ES, C) + word) ← byte			
		word[BC], #byte	4	1	—	((BC + word) ← byte			
		ES:word[BC], #byte	5	2	—	((ES, BC) + word) ← byte			
		A, r Note 3	1	1	—	A ← r			
		r, A Note 3	1	1	—	r ← A			
		A, PSW	2	1	—	A ← PSW			
		PSW, A	2	3	—	PSW ← A	x	x	x
		A, CS	2	1	—	A ← CS			
		CS, A	2	1	—	CS ← A			
		A, ES	2	1	—	A ← ES			
		ES, A	2	1	—	ES ← A			
		A, !addr16	3	1	4	A ← (addr16)			
		A, ES:!addr16	4	2	5	A ← ((ES, addr16))			
		!addr16, A	3	1	—	((addr16) ← A)			
		ES:!addr16, A	4	2	—	((ES, addr16) ← A)			
		A, saddr	2	1	—	A ← (saddr)			
		saddr, A	2	1	—	((saddr) ← A)			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Table 33 - 6 Operation List (2/18)**

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	MOV	A, sfr	2	1	—	A ← sfr			
		sfr, A	2	1	—	sfr ← A			
		A, [DE]	1	1	4	A ← (DE)			
		[DE], A	1	1	—	(DE) ← A			
		A, ES:[DE]	2	2	5	A ← (ES, DE)			
		ES:[DE], A	2	2	—	(ES, DE) ← A			
		A, [HL]	1	1	4	A ← (HL)			
		[HL], A	1	1	—	(HL) ← A			
		A, ES:[HL]	2	2	5	A ← (ES, HL)			
		ES:[HL], A	2	2	—	(ES, HL) ← A			
		A, [DE+byte]	2	1	4	A ← (DE + byte)			
		[DE+byte], A	2	1	—	(DE + byte) ← A			
		A, ES:[DE+byte]	3	2	5	A ← ((ES, DE) + byte)			
		ES:[DE+byte], A	3	2	—	((ES, DE) + byte) ← A			
		A, [HL+byte]	2	1	4	A ← (HL + byte)			
		[HL+byte], A	2	1	—	(HL + byte) ← A			
		A, ES:[HL+byte]	3	2	5	A ← ((ES, HL) + byte)			
		ES:[HL+byte], A	3	2	—	((ES, HL) + byte) ← A			
		A, [SP+byte]	2	1	—	A ← (SP + byte)			
		[SP+byte], A	2	1	—	(SP + byte) ← A			
		A, word[B]	3	1	4	A ← (B + word)			
		word[B], A	3	1	—	(B + word) ← A			
		A, ES:word[B]	4	2	5	A ← ((ES, B) + word)			
		ES:word[B], A	4	2	—	((ES, B) + word) ← A			
		A, word[C]	3	1	4	A ← (C + word)			
		word[C], A	3	1	—	(C + word) ← A			
		A, ES:word[C]	4	2	5	A ← ((ES, C) + word)			
		ES:word[C], A	4	2	—	((ES, C) + word) ← A			
		A, word[BC]	3	1	4	A ← (BC + word)			
		word[BC], A	3	1	—	(BC + word) ← A			
		A, ES:word[BC]	4	2	5	A ← ((ES, BC) + word)			
		ES:word[BC], A	4	2	—	((ES, BC) + word) ← A			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 7 Operation List (3/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	MOV	A, [HL+B]	2	1	4	A $\leftarrow$ (HL + B)			
		[HL+B], A	2	1	—	(HL + B) $\leftarrow$ A			
		A, ES:[HL+B]	3	2	5	A $\leftarrow$ ((ES, HL) + B)			
		ES:[HL+B], A	3	2	—	((ES, HL) + B) $\leftarrow$ A			
		A, [HL+C]	2	1	4	A $\leftarrow$ (HL + C)			
		[HL+C], A	2	1	—	(HL + C) $\leftarrow$ A			
		A, ES:[HL+C]	3	2	5	A $\leftarrow$ ((ES, HL) + C)			
		ES:[HL+C], A	3	2	—	((ES, HL) + C) $\leftarrow$ A			
		X, !addr16	3	1	4	X $\leftarrow$ (addr16)			
		X, ES:!addr16	4	2	5	X $\leftarrow$ (ES, addr16)			
		X, saddr	2	1	—	X $\leftarrow$ (saddr)			
		B, !addr16	3	1	4	B $\leftarrow$ (addr16)			
		B, ES:!addr16	4	2	5	B $\leftarrow$ (ES, addr16)			
		B, saddr	2	1	—	B $\leftarrow$ (saddr)			
		C, !addr16	3	1	4	C $\leftarrow$ (addr16)			
		C, ES:!addr16	4	2	5	C $\leftarrow$ (ES, addr16)			
		C, saddr	2	1	—	C $\leftarrow$ (saddr)			
		ES, saddr	3	1	—	ES $\leftarrow$ (saddr)			
XCH	XCH	A, r Note 3	1 (r = X) 2 (other than r = X)	1	—	A $\longleftrightarrow$ r			
		A, !addr16	4	2	—	A $\longleftrightarrow$ (addr16)			
		A, ES:!addr16	5	3	—	A $\longleftrightarrow$ (ES, addr16)			
		A, saddr	3	2	—	A $\longleftrightarrow$ (saddr)			
		A, sfr	3	2	—	A $\longleftrightarrow$ sfr			
		A, [DE]	2	2	—	A $\longleftrightarrow$ (DE)			
		A, ES:[DE]	3	3	—	A $\longleftrightarrow$ (ES, DE)			
		A, [HL]	2	2	—	A $\longleftrightarrow$ (HL)			
		A, ES:[HL]	3	3	—	A $\longleftrightarrow$ (ES, HL)			
		A, [DE+byte]	3	2	—	A $\longleftrightarrow$ (DE + byte)			
		A, ES:[DE+byte]	4	3	—	A $\longleftrightarrow$ ((ES, DE) + byte)			
		A, [HL+byte]	3	2	—	A $\longleftrightarrow$ (HL + byte)			
		A, ES:[HL+byte]	4	3	—	A $\longleftrightarrow$ ((ES, HL) + byte)			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 8 Operation List (4/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit data transfer	XCH	A, [HL+B]	2	2	—	A $\leftrightarrow$ (HL + B)			
		A, ES:[HL+B]	3	3	—	A $\leftrightarrow$ ((ES, HL) + B)			
		A, [HL+C]	2	2	—	A $\leftrightarrow$ (HL + C)			
		A, ES:[HL+C]	3	3	—	A $\leftrightarrow$ ((ES, HL) + C)			
	ONEB	A	1	1	—	A $\leftarrow$ 01H			
		X	1	1	—	X $\leftarrow$ 01H			
		B	1	1	—	B $\leftarrow$ 01H			
		C	1	1	—	C $\leftarrow$ 01H			
		!addr16	3	1	—	(addr16) $\leftarrow$ 01H			
		ES:!addr16	4	2	—	(ES, addr16) $\leftarrow$ 01H			
		saddr	2	1	—	(saddr) $\leftarrow$ 01H			
	CLRB	A	1	1	—	A $\leftarrow$ 00H			
		X	1	1	—	X $\leftarrow$ 00H			
		B	1	1	—	B $\leftarrow$ 00H			
		C	1	1	—	C $\leftarrow$ 00H			
		!addr16	3	1	—	(addr16) $\leftarrow$ 00H			
		ES:!addr16	4	2	—	(ES,addr16) $\leftarrow$ 00H			
		saddr	2	1	—	(saddr) $\leftarrow$ 00H			
	MOVS	[HL+byte], X	3	1	—	(HL + byte) $\leftarrow$ X	x	x	
		ES:[HL+byte], X	4	2	—	(ES, HL + byte) $\leftarrow$ X	x	x	
16-bit data transfer	MOVW	rp, #word	3	1	—	rp $\leftarrow$ word			
		saddrp, #word	4	1	—	(saddrp) $\leftarrow$ word			
		sfrp, #word	4	1	—	sfrp $\leftarrow$ word			
		AX, rp Note 3	1	1	—	AX $\leftarrow$ rp			
		rp, AX Note 3	1	1	—	rp $\leftarrow$ AX			
		AX, !addr16	3	1	4	AX $\leftarrow$ (addr16)			
		!addr16, AX	3	1	—	(addr16) $\leftarrow$ AX			
		AX, ES:!addr16	4	2	5	AX $\leftarrow$ (ES, addr16)			
		ES:!addr16, AX	4	2	—	(ES, addr16) $\leftarrow$ AX			
		AX, saddrp	2	1	—	AX $\leftarrow$ (saddrp)			
		saddrp, AX	2	1	—	(saddrp) $\leftarrow$ AX			
		AX, sfrp	2	1	—	AX $\leftarrow$ sfrp			
		sfrp, AX	2	1	—	sfrp $\leftarrow$ AX			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except rp = AX

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Table 33 - 9 Operation List (5/18)**

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit data transfer	MOVW	AX, [DE]	1	1	4	AX ← (DE)			
		[DE], AX	1	1	—	(DE) ← AX			
		AX, ES:[DE]	2	2	5	AX ← (ES, DE)			
		ES:[DE], AX	2	2	—	(ES, DE) ← AX			
		AX, [HL]	1	1	4	AX ← (HL)			
		[HL], AX	1	1	—	(HL) ← AX			
		AX, ES:[HL]	2	2	5	AX ← (ES, HL)			
		ES:[HL], AX	2	2	—	(ES, HL) ← AX			
		AX, [DE+byte]	2	1	4	AX ← (DE + byte)			
		[DE+byte], AX	2	1	—	(DE + byte) ← AX			
		AX, ES:[DE+byte]	3	2	5	AX ← ((ES, DE) + byte)			
		ES:[DE+byte], AX	3	2	—	((ES, DE) + byte) ← AX			
		AX, [HL+byte]	2	1	4	AX ← (HL + byte)			
		[HL+byte], AX	2	1	—	(HL + byte) ← AX			
		AX, ES:[HL+byte]	3	2	5	AX ← ((ES, HL) + byte)			
		ES:[HL+byte], AX	3	2	—	((ES, HL) + byte) ← AX			
		AX, [SP+byte]	2	1	—	AX ← (SP + byte)			
		[SP+byte], AX	2	1	—	(SP + byte) ← AX			
		AX, word[B]	3	1	4	AX ← (B + word)			
		word[B], AX	3	1	—	(B + word) ← AX			
		AX, ES:word[B]	4	2	5	AX ← ((ES, B) + word)			
		ES:word[B], AX	4	2	—	((ES, B) + word) ← AX			
		AX, word[C]	3	1	4	AX ← (C + word)			
		word[C], AX	3	1	—	(C + word) ← AX			
		AX, ES:word[C]	4	2	5	AX ← ((ES, C) + word)			
		ES:word[C], AX	4	2	—	((ES, C) + word) ← AX			
		AX, word[BC]	3	1	4	AX ← (BC + word)			
		word[BC], AX	3	1	—	(BC + word) ← AX			
		AX, ES:word[BC]	4	2	5	AX ← ((ES, BC) + word)			
		ES:word[BC], AX	4	2	—	((ES, BC) + word) ← AX			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 10 Operation List (6/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit data transfer	MOVW	BC, !addr16	3	1	4	BC $\leftarrow$ (addr16)			
		BC, ES:!addr16	4	2	5	BC $\leftarrow$ (ES, addr16)			
		DE, !addr16	3	1	4	DE $\leftarrow$ (addr16)			
		DE, ES:!addr16	4	2	5	DE $\leftarrow$ (ES, addr16)			
		HL, !addr16	3	1	4	HL $\leftarrow$ (addr16)			
		HL, ES:!addr16	4	2	5	HL $\leftarrow$ (ES, addr16)			
		BC, saddrp	2	1	—	BC $\leftarrow$ (saddrp)			
		DE, saddrp	2	1	—	DE $\leftarrow$ (saddrp)			
		HL, saddrp	2	1	—	HL $\leftarrow$ (saddrp)			
	XCHW	AX, rp Note 3	1	1	—	AX $\longleftrightarrow$ rp			
	ONEW	AX	1	1	—	AX $\leftarrow$ 0001H			
		BC	1	1	—	BC $\leftarrow$ 0001H			
	CLRW	AX	1	1	—	AX $\leftarrow$ 0000H			
		BC	1	1	—	BC $\leftarrow$ 0000H			
8-bit operation	ADD	A, #byte	2	1	—	A, CY $\leftarrow$ A + byte	x	x	x
		saddr, #byte	3	2	—	(saddr), CY $\leftarrow$ (saddr) + byte	x	x	x
		A, r Note 4	2	1	—	A, CY $\leftarrow$ A + r	x	x	x
		r, A	2	1	—	r, CY $\leftarrow$ r + A	x	x	x
		A, !addr16	3	1	4	A, CY $\leftarrow$ A + (addr16)	x	x	x
		A, ES:!addr16	4	2	5	A, CY $\leftarrow$ A + (ES, addr16)	x	x	x
		A, saddr	2	1	—	A, C $\leftarrow$ A + (saddr)	x	x	x
		A, [HL]	1	1	4	A, CY $\leftarrow$ A + (HL)	x	x	x
		A, ES:[HL]	2	2	5	A, CY $\leftarrow$ A + (ES, HL)	x	x	x
		A, [HL+byte]	2	1	4	A, CY $\leftarrow$ A + (HL + byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A, CY $\leftarrow$ A + ((ES, HL) + byte)	x	x	x
		A, [HL+B]	2	1	4	A, CY $\leftarrow$ A + (HL + B)	x	x	x
		A, ES:[HL+B]	3	2	5	A, CY $\leftarrow$ A + ((ES, HL) + B)	x	x	x
		A, [HL+C]	2	1	4	A, CY $\leftarrow$ A + (HL + C)	x	x	x
		A, ES:[HL+C]	3	2	5	A, CY $\leftarrow$ A + ((ES, HL) + C)	x	x	x

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except rp = AX

**Note 4.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 11 Operation List (7/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	ADDC	A, #byte	2	1	—	A, CY $\leftarrow$ A + byte + CY	x	x	x
		saddr, #byte	3	2	—	(saddr), CY $\leftarrow$ (saddr) + byte + CY	x	x	x
		A, r Note 3	2	1	—	A, CY $\leftarrow$ A + r + CY	x	x	x
		r, A	2	1	—	r, CY $\leftarrow$ r + A + CY	x	x	x
		A, !addr16	3	1	4	A, CY $\leftarrow$ A + (addr16) + CY	x	x	x
		A, ES:!addr16	4	2	5	A, CY $\leftarrow$ A + (ES, addr16) + CY	x	x	x
		A, saddr	2	1	—	A, CY $\leftarrow$ A + (saddr) + CY	x	x	x
		A, [HL]	1	1	4	A, CY $\leftarrow$ A + (HL) + CY	x	x	x
		A, ES:[HL]	2	2	5	A,CY $\leftarrow$ A + (ES, HL) + CY	x	x	x
		A, [HL+byte]	2	1	4	A, CY $\leftarrow$ A + (HL + byte) + CY	x	x	x
		A, ES:[HL+byte]	3	2	5	A,CY $\leftarrow$ A + ((ES, HL) + byte) + CY	x	x	x
		A, [HL+B]	2	1	4	A, CY $\leftarrow$ A + (HL + B) + CY	x	x	x
		A, ES:[HL+B]	3	2	5	A,CY $\leftarrow$ A + ((ES, HL) + B) + CY	x	x	x
		A, [HL+C]	2	1	4	A, CY $\leftarrow$ A + (HL + C) + CY	x	x	x
		A, ES:[HL+C]	3	2	5	A,CY $\leftarrow$ A + ((ES, HL) + C) + CY	x	x	x
	SUB	A, #byte	2	1	—	A, CY $\leftarrow$ A - byte	x	x	x
		saddr, #byte	3	2	—	(saddr), CY $\leftarrow$ (saddr) - byte	x	x	x
		A, r Note 3	2	1	—	A, CY $\leftarrow$ A - r	x	x	x
		r, A	2	1	—	r, CY $\leftarrow$ r - A	x	x	x
		A, !addr16	3	1	4	A, CY $\leftarrow$ A - (addr16)	x	x	x
		A, ES:!addr16	4	2	5	A, CY $\leftarrow$ A - (ES, addr16)	x	x	x
		A, saddr	2	1	—	A, CY $\leftarrow$ A - (saddr)	x	x	x
		A, [HL]	1	1	4	A, CY $\leftarrow$ A - (HL)	x	x	x
		A, ES:[HL]	2	2	5	A,CY $\leftarrow$ A - (ES, HL)	x	x	x
		A, [HL+byte]	2	1	4	A, CY $\leftarrow$ A - (HL + byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A,CY $\leftarrow$ A - ((ES, HL) + byte)	x	x	x
		A, [HL+B]	2	1	4	A, CY $\leftarrow$ A - (HL + B)	x	x	x
		A, ES:[HL+B]	3	2	5	A,CY $\leftarrow$ A - ((ES, HL) + B)	x	x	x
		A, [HL+C]	2	1	4	A, CY $\leftarrow$ A - (HL + C)	x	x	x
		A, ES:[HL+C]	3	2	5	A,CY $\leftarrow$ A - ((ES, HL) + C)	x	x	x

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 12 Operation List (8/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	SUBC	A, #byte	2	1	—	A, CY $\leftarrow$ A - byte - CY	x	x	x
		saddr, #byte	3	2	—	(saddr), CY $\leftarrow$ (saddr) - byte - CY	x	x	x
		A, r Note 3	2	1	—	A, CY $\leftarrow$ A - r - CY	x	x	x
		r, A	2	1	—	r, CY $\leftarrow$ r - A - CY	x	x	x
		A, !addr16	3	1	4	A, CY $\leftarrow$ A - (addr16) - CY	x	x	x
		A, ES:!addr16	4	2	5	A, CY $\leftarrow$ A - (ES, addr16) - CY	x	x	x
		A, saddr	2	1	—	A, CY $\leftarrow$ A - (saddr) - CY	x	x	x
		A, [HL]	1	1	4	A, CY $\leftarrow$ A - (HL) - CY	x	x	x
		A, ES:[HL]	2	2	5	A, CY $\leftarrow$ A - (ES, HL) - CY	x	x	x
		A, [HL+byte]	2	1	4	A, CY $\leftarrow$ A - (HL + byte) - CY	x	x	x
		A, ES:[HL+byte]	3	2	5	A, CY $\leftarrow$ A - ((ES, HL) + byte) - CY	x	x	x
		A, [HL+B]	2	1	4	A, CY $\leftarrow$ A - (HL + B) - CY	x	x	x
		A, ES:[HL+B]	3	2	5	A, CY $\leftarrow$ A - ((ES, HL) + B) - CY	x	x	x
		A, [HL+C]	2	1	4	A, CY $\leftarrow$ A - (HL + C) - CY	x	x	x
		A, ES:[HL+C]	3	2	5	A, CY $\leftarrow$ A - ((ES:HL) + C) - CY	x	x	x
AND	AND	A, #byte	2	1	—	A $\leftarrow$ A $\wedge$ byte	x		
		saddr, #byte	3	2	—	(saddr) $\leftarrow$ (saddr) $\wedge$ byte	x		
		A, r Note 3	2	1	—	A $\leftarrow$ A $\wedge$ r	x		
		r, A	2	1	—	R $\leftarrow$ r $\wedge$ A	x		
		A, !addr16	3	1	4	A $\leftarrow$ A $\wedge$ (addr16)	x		
		A, ES:addr16	4	2	5	A $\leftarrow$ A $\wedge$ (ES:addr16)	x		
		A, saddr	2	1	—	A $\leftarrow$ A $\wedge$ (saddr)	x		
		A, [HL]	1	1	4	A $\leftarrow$ A $\wedge$ (HL)	x		
		A, ES:[HL]	2	2	5	A $\leftarrow$ A $\wedge$ (ES:HL)	x		
		A, [HL+byte]	2	1	4	A $\leftarrow$ A $\wedge$ (HL + byte)	x		
		A, ES:[HL+byte]	3	2	5	A $\leftarrow$ A $\wedge$ ((ES:HL) + byte)	x		
		A, [HL+B]	2	1	4	A $\leftarrow$ A $\wedge$ (HL + B)	x		
		A, ES:[HL+B]	3	2	5	A $\leftarrow$ A $\wedge$ ((ES:HL) + B)	x		
		A, [HL+C]	2	1	4	A $\leftarrow$ A $\wedge$ (HL + C)	x		
		A, ES:[HL+C]	3	2	5	A $\leftarrow$ A $\wedge$ ((ES:HL) + C)	x		

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 13 Operation List (9/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	OR	A, #byte	2	1	—	$A \leftarrow A \vee \text{byte}$	x		
		saddr, #byte	3	2	—	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \text{byte}$	x		
		A, r Note 3	2	1	—	$A \leftarrow A \vee r$	x		
		r, A	2	1	—	$r \leftarrow r \vee A$	x		
		A, !addr16	3	1	4	$A \leftarrow A \vee (\text{addr16})$	x		
		A, ES:!addr16	4	2	5	$A \leftarrow A \vee (\text{ES:addr16})$	x		
		A, saddr	2	1	—	$A \leftarrow A \vee (\text{saddr})$	x		
		A, [HL]	1	1	4	$A \leftarrow A \vee (\text{HL})$	x		
		A, ES:[HL]	2	2	5	$A \leftarrow A \vee (\text{ES:HL})$	x		
		A, [HL+byte]	2	1	4	$A \leftarrow A \vee (\text{HL} + \text{byte})$	x		
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + \text{byte})$	x		
		A, [HL+B]	2	1	4	$A \leftarrow A \vee (\text{HL} + B)$	x		
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + B)$	x		
		A, [HL+C]	2	1	4	$A \leftarrow A \vee (\text{HL} + C)$	x		
		A, ES:[HL+C]	3	2	5	$A \leftarrow A \vee ((\text{ES:HL}) + C)$	x		
	XOR	A, #byte	2	1	—	$A \leftarrow A \vee \neg \text{byte}$	x		
		saddr, #byte	3	2	—	$(\text{saddr}) \leftarrow (\text{saddr}) \vee \neg \text{byte}$	x		
		A, r Note 3	2	1	—	$A \leftarrow A \vee \neg r$	x		
		r, A	2	1	—	$r \leftarrow r \vee \neg A$	x		
		A, !addr16	3	1	4	$A \leftarrow A \vee \neg (\text{addr16})$	x		
		A, ES:!addr16	4	2	5	$A \leftarrow A \vee \neg (\text{ES:addr16})$	x		
		A, saddr	2	1	—	$A \leftarrow A \vee \neg (\text{saddr})$	x		
		A, [HL]	1	1	4	$A \leftarrow A \vee \neg (\text{HL})$	x		
		A, ES:[HL]	2	2	5	$A \leftarrow A \vee \neg (\text{ES:HL})$	x		
		A, [HL+byte]	2	1	4	$A \leftarrow A \vee \neg (\text{HL} + \text{byte})$	x		
		A, ES:[HL+byte]	3	2	5	$A \leftarrow A \vee \neg ((\text{ES:HL}) + \text{byte})$	x		
		A, [HL+B]	2	1	4	$A \leftarrow A \vee \neg (\text{HL} + B)$	x		
		A, ES:[HL+B]	3	2	5	$A \leftarrow A \vee \neg ((\text{ES:HL}) + B)$	x		
		A, [HL+C]	2	1	4	$A \leftarrow A \vee \neg (\text{HL} + C)$	x		
		A, ES:[HL+C]	3	2	5	$A \leftarrow A \vee \neg ((\text{ES:HL}) + C)$	x		

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except  $r = A$

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 14 Operation List (10/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
8-bit operation	CMP	A, #byte	2	1	—	A - byte	x	x	x
		!addr16, #byte	4	1	4	(addr16) - byte	x	x	x
		ES:!addr16, #byte	5	2	5	(ES:addr16) - byte	x	x	x
		saddr, #byte	3	1	—	(saddr) - byte	x	x	x
		A, r Note 3	2	1	—	A - r	x	x	x
		r, A	2	1	—	r - A	x	x	x
		A, !addr16	3	1	4	A - (addr16)	x	x	x
		A, ES:!addr16	4	2	5	A - (ES:addr16)	x	x	x
		A, saddr	2	1	—	A - (saddr)	x	x	x
		A, [HL]	1	1	4	A - (HL)	x	x	x
		A, ES:[HL]	2	2	5	A - (ES:HL)	x	x	x
		A, [HL+byte]	2	1	4	A - (HL + byte)	x	x	x
		A, ES:[HL+byte]	3	2	5	A - ((ES:HL) + byte)	x	x	x
		A, [HL+B]	2	1	4	A - (HL + B)	x	x	x
		A, ES:[HL+B]	3	2	5	A - ((ES:HL) + B)	x	x	x
		A, [HL+C]	2	1	4	A - (HL + C)	x	x	x
		A, ES:[HL+C]	3	2	5	A - ((ES:HL) + C)	x	x	x
	CMPO	A	1	1	—	A - 00H	x	0	0
		X	1	1	—	X - 00H	x	0	0
		B	1	1	—	B - 00H	x	0	0
		C	1	1	—	C - 00H	x	0	0
		!addr16	3	1	4	(addr16) - 00H	x	0	0
		ES:!addr16	4	2	5	(ES:addr16) - 00H	x	0	0
		saddr	2	1	—	(saddr) - 00H	x	0	0
	CMPS	X, [HL+byte]	3	1	4	X - (HL + byte)	x	x	x
		X, ES:[HL+byte]	4	2	5	X - ((ES:HL) + byte)	x	x	x

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** Except r = A

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 15 Operation List (11/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
16-bit operation	ADDW	AX, #word	3	1	—	AX, CY ← AX + word	×	×	×
		AX, AX	1	1	—	AX, CY ← AX + AX	×	×	×
		AX, BC	1	1	—	AX, CY ← AX + BC	×	×	×
		AX, DE	1	1	—	AX, CY ← AX + DE	×	×	×
		AX, HL	1	1	—	AX, CY ← AX + HL	×	×	×
		AX, !addr16	3	1	4	AX, CY ← AX + (addr16)	×	×	×
		AX, ES:!addr16	4	2	5	AX, CY ← AX + (ES:addr16)	×	×	×
		AX, saddrp	2	1	—	AX, CY ← AX + (saddrp)	×	×	×
		AX, [HL+byte]	3	1	4	AX, CY ← AX + (HL + byte)	×	×	×
		AX, ES: [HL+byte]	4	2	5	AX, CY ← AX + ((ES:HL) + byte)	×	×	×
	SUBW	AX, #word	3	1	—	AX, CY ← AX - word	×	×	×
		AX, BC	1	1	—	AX, CY ← AX - BC	×	×	×
		AX, DE	1	1	—	AX, CY ← AX - DE	×	×	×
		AX, HL	1	1	—	AX, CY ← AX - HL	×	×	×
		AX, !addr16	3	1	4	AX, CY ← AX - (addr16)	×	×	×
		AX, ES:!addr16	4	2	5	AX, CY ← AX - (ES:addr16)	×	×	×
		AX, saddrp	2	1	—	AX, CY ← AX - (saddrp)	×	×	×
		AX, [HL+byte]	3	1	4	AX, CY ← AX - (HL + byte)	×	×	×
		AX, ES: [HL+byte]	4	2	5	AX, CY ← AX - ((ES:HL) + byte)	×	×	×
		MULU	X	1	1	—	AX ← A × X		

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Table 33 - 16 Operation List (12/18)**

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Multiply, Divide, Multiply & accumulate	MULU	X	1	1	—	AX $\leftarrow$ A $\times$ X			
	MULHU		3	2	—	BCAX $\leftarrow$ AX $\times$ BC (unsigned)			
	MULH		3	2	—	BCAX $\leftarrow$ AX $\times$ BC (signed)			
	DIVHU		3	9	—	AX (quotient), DE (remainder) $\leftarrow$ AX $\div$ DE (unsigned)			
	DIVWU		3	17	—	BCAX (quotient), HLDE (remainder) $\leftarrow$ BCAX $\div$ HLDE (unsigned)			
	MACHU		3	3	—	MACR $\leftarrow$ MACR + AX $\times$ BC (unsigned)	x	x	
	MACH		3	3	—	MACR $\leftarrow$ MACR + AX $\times$ BC(signed)	x	x	

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Caution** **Disable interrupts when executing the DIVHU or DIVWU instruction in an interrupt servicing routine.**

Alternatively, unless they are executed in the RAM area, note that execution of a DIVHU or DIVWU instruction is possible even with interrupts enabled as long as a NOP instruction is added immediately after the DIVHU or DIVWU instruction in the assembly language source code. The following compilers automatically add a NOP instruction immediately after any DIVHU or DIVWU instruction output during the build process.

- V. 1.71 and later versions of the CA78K0R (Renesas Electronics compiler), for both C and assembly language source code
- Service pack 1.40.6 and later versions of the EWRL78 (IAR compiler), for C language source code
- GNURL78 (KPIT compiler), for C language source code

**Remark 1.** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Remark 2.** MACR indicates the multiplication and accumulation register (MACRH, MACRL).

Table 33 - 17 Operation List (13/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Increment/ decrement	INC	r	1	1	—	$r \leftarrow r + 1$	×	×	
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) + 1$	×	×	
		ES:laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) + 1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr) + 1$	×	×	
		[HL+byte]	3	2	—	$(HL + byte) \leftarrow (HL + byte) + 1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL) + byte) \leftarrow ((ES:HL) + byte) + 1$	×	×	
	DEC	r	1	1	—	$r \leftarrow r - 1$	×	×	
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) - 1$	×	×	
		ES:laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) - 1$	×	×	
		saddr	2	2	—	$(saddr) \leftarrow (saddr) - 1$	×	×	
		[HL+byte]	3	2	—	$(HL + byte) \leftarrow (HL + byte) - 1$	×	×	
		ES: [HL+byte]	4	3	—	$((ES:HL) + byte) \leftarrow ((ES:HL) + byte) - 1$	×	×	
Shift	INCW	rp	1	1	—	$rp \leftarrow rp + 1$			
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) + 1$			
		ES:laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) + 1$			
		saddrp	2	2	—	$(saddrp) \leftarrow (saddrp) + 1$			
		[HL+byte]	3	2	—	$(HL + byte) \leftarrow (HL + byte) + 1$			
		ES: [HL+byte]	4	3	—	$((ES:HL) + byte) \leftarrow ((ES:HL) + byte) + 1$			
	DECW	rp	1	1	—	$rp \leftarrow rp - 1$			
		laddr16	3	2	—	$(addr16) \leftarrow (addr16) - 1$			
		ES:laddr16	4	3	—	$(ES, addr16) \leftarrow (ES, addr16) - 1$			
		saddrp	2	2	—	$(saddrp) \leftarrow (saddrp) - 1$			
		[HL+byte]	3	2	—	$(HL + byte) \leftarrow (HL + byte) - 1$			
		ES: [HL+byte]	4	3	—	$((ES:HL) + byte) \leftarrow ((ES:HL) + byte) - 1$			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark 1.** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Remark 2.** cnt indicates the bit shift count.

Table 33 - 18 Operation List (14/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Rotate	ROR	A, 1	2	1	—	(CY, A <sub>7</sub> ← A <sub>0</sub> , A <sub>m-1</sub> ← A <sub>m</sub> ) × 1			×
	ROL	A, 1	2	1	—	(CY, A <sub>0</sub> ← A <sub>7</sub> , A <sub>m+1</sub> ← A <sub>m</sub> ) × 1			×
	RORC	A, 1	2	1	—	(CY ← A <sub>0</sub> , A <sub>7</sub> ← CY, A <sub>m-1</sub> ← A <sub>m</sub> ) × 1			×
	ROLC	A, 1	2	1	—	(CY ← A <sub>7</sub> , A <sub>0</sub> ← CY, A <sub>m+1</sub> ← A <sub>m</sub> ) × 1			×
	ROLWC	AX, 1	2	1	—	(CY ← AX <sub>15</sub> , AX <sub>0</sub> ← CY, AX <sub>m+1</sub> ← AX <sub>m</sub> ) × 1			×
		BC, 1	2	1	—	(CY ← BC <sub>15</sub> , BC <sub>0</sub> ← CY, BC <sub>m+1</sub> ← BC <sub>m</sub> ) × 1			×
Bit manipulate	MOV1	CY, A.bit	2	1	—	CY ← A.bit			×
		A.bit, CY	2	1	—	A.bit ← CY			
		CY, PSW.bit	3	1	—	CY ← PSW.bit			×
		PSW.bit, CY	3	4	—	PSW.bit ← CY		×	×
		CY, saddr.bit	3	1	—	CY ← (saddr).bit			×
		saddr.bit, CY	3	2	—	(saddr).bit ← CY			
		CY, sfr.bit	3	1	—	CY ← sfr.bit			×
		sfr.bit, CY	3	2	—	sfr.bit ← CY			
		CY,[HL].bit	2	1	4	CY ← (HL).bit			×
		[HL].bit, CY	2	2	—	(HL).bit ← CY			
	AND1	CY, ES:[HL].bit	3	2	5	CY ← (ES, HL).bit			×
		ES:[HL].bit, CY	3	3	—	(ES, HL).bit ← CY			
		CY, A.bit	2	1	—	CY ← CY ∧ A.bit			×
		CY, PSW.bit	3	1	—	CY ← CY ∧ PSW.bit			×
		CY, saddr.bit	3	1	—	CY ← CY ∧ (saddr).bit			×
		CY, sfr.bit	3	1	—	CY ← CY ∧ sfr.bit			×
	OR1	CY,[HL].bit	2	1	4	CY ← CY ∨ (HL).bit			×
		CY, ES:[HL].bit	3	2	5	CY ← CY ∨ (ES, HL).bit			×
		CY, A.bit	2	1	—	CY ← CY ∨ A.bit			×
		CY, PSW.bit	3	1	—	CY ← CY ∨ ∨ PSW.bit			×
		CY, saddr.bit	3	1	—	CY ← CY ∨ (saddr).bit			×
		CY, sfr.bit	3	1	—	CY ← CY ∨ sfr.bit			×
		CY,[HL].bit	2	1	4	CY ← CY ∨ (HL).bit			×
		CY, ES:[HL].bit	3	2	5	CY ← CY ∨ (ES, HL).bit			×

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 19 Operation List (15/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Bit manipulate	XOR1	CY, A.bit	2	1	—	CY $\leftarrow$ CY $\vee$ bit			x
		CY, PSW.bit	3	1	—	CY $\leftarrow$ CY $\vee$ PSW.bit			x
		CY, saddr.bit	3	1	—	CY $\leftarrow$ CY $\vee$ (saddr).bit			x
		CY, sfr.bit	3	1	—	CY $\leftarrow$ CY $\vee$ sfr.bit			x
		CY, [HL].bit	2	1	4	CY $\leftarrow$ CY $\vee$ (HL).bit			x
		CY, ES:[HL].bit	3	2	5	CY $\leftarrow$ CY $\vee$ (ES, HL).bit			x
	SET1	A.bit	2	1	—	A.bit $\leftarrow$ 1			
		PSW.bit	3	4	—	PSW.bit $\leftarrow$ 1		x	x
		!addr16.bit	4	2	—	(addr16).bit $\leftarrow$ 1			
		ES:!addr16.bit	5	3	—	(ES, addr16).bit $\leftarrow$ 1			
		saddr.bit	3	2	—	(saddr).bit $\leftarrow$ 1			
		sfr.bit	3	2	—	sfr.bit $\leftarrow$ 1			
		[HL].bit	2	2	—	(HL).bit $\leftarrow$ 1			
		ES:[HL].bit	3	3	—	(ES, HL).bit $\leftarrow$ 1			
	CLR1	A.bit	2	1	—	A.bit $\leftarrow$ 0			
		PSW.bit	3	4	—	PSW.bit $\leftarrow$ 0		x	x
		!addr16.bit	4	2	—	(addr16).bit $\leftarrow$ 0			
		ES:!addr16.bit	5	3	—	(ES, addr16).bit $\leftarrow$ 0			
		saddr.bit	3	2	—	(saddr).bit $\leftarrow$ 0			
		sfr.bit	3	2	—	sfr.bit $\leftarrow$ 0			
		[HL].bit	2	2	—	(HL).bit $\leftarrow$ 0			
	SET1	CY	2	1	—	CY $\leftarrow$ 1			1
	CLR1	CY	2	1	—	CY $\leftarrow$ 0			0
	NOT1	CY	2	1	—	CY $\leftarrow$ $\overline{CY}$			x

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

**Table 33 - 20 Operation List (16/18)**

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Call/return	CALL	rp	2	3	—	(SP - 2) $\leftarrow$ (PC + 2)s, (SP - 3) $\leftarrow$ (PC + 2)H, (SP - 4) $\leftarrow$ (PC + 2)L, PC $\leftarrow$ CS, rp, SP $\leftarrow$ SP - 4			
		\$!addr20	3	3	—	(SP - 2) $\leftarrow$ (PC + 3)s, (SP - 3) $\leftarrow$ (PC + 3)H, (SP - 4) $\leftarrow$ (PC + 3)L, PC $\leftarrow$ PC + 3 + jdisp16, SP $\leftarrow$ SP - 4			
		!addr16	3	3	—	(SP - 2) $\leftarrow$ (PC + 3)s, (SP - 3) $\leftarrow$ (PC + 3)H, (SP - 4) $\leftarrow$ (PC + 3)L, PC $\leftarrow$ 0000, addr16, SP $\leftarrow$ SP - 4			
		!!addr20	4	3	—	(SP - 2) $\leftarrow$ (PC + 4)s, (SP - 3) $\leftarrow$ (PC + 4)H, (SP - 4) $\leftarrow$ (PC + 4)L, PC $\leftarrow$ addr20, SP $\leftarrow$ SP - 4			
	CALLT	[addr5]	2	5	—	(SP - 2) $\leftarrow$ (PC + 2)s, (SP - 3) $\leftarrow$ (PC + 2)H, (SP - 4) $\leftarrow$ (PC + 2)L, PCs $\leftarrow$ 0000, PC <sub>H</sub> $\leftarrow$ (0000, addr5 + 1), PC <sub>L</sub> $\leftarrow$ (0000, addr5), SP $\leftarrow$ SP - 4			
	BRK	—	2	5	—	(SP - 1) $\leftarrow$ PSW, (SP - 2) $\leftarrow$ (PC + 2)s, (SP - 3) $\leftarrow$ (PC + 2)H, (SP - 4) $\leftarrow$ (PC + 2)L, PCs $\leftarrow$ 0000, PC <sub>H</sub> $\leftarrow$ (0007FH), PC <sub>L</sub> $\leftarrow$ (0007EH), SP $\leftarrow$ SP - 4, IE $\leftarrow$ 0			
	RET	—	1	6	—	PC <sub>L</sub> $\leftarrow$ (SP), PC <sub>H</sub> $\leftarrow$ (SP + 1), PCs $\leftarrow$ (SP + 2), SP $\leftarrow$ SP + 4			
	RETI	—	2	6	—	PC <sub>L</sub> $\leftarrow$ (SP), PC <sub>H</sub> $\leftarrow$ (SP + 1), PCs $\leftarrow$ (SP + 2), PSW $\leftarrow$ (SP + 3), SP $\leftarrow$ SP + 4	R	R	R
	RETB	—	2	6	—	PC <sub>L</sub> $\leftarrow$ (SP), PC <sub>H</sub> $\leftarrow$ (SP + 1), PCs $\leftarrow$ (SP + 2), PSW $\leftarrow$ (SP + 3), SP $\leftarrow$ SP + 4	R	R	R

**Note 1.** Number of CPU clocks (f<sub>CLK</sub>) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (f<sub>CLK</sub>) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 21 Operation List (17/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Stack manipulate	PUSH	PSW	2	1	—	(SP - 1) ← PSW, (SP - 2) ← 00H, SP ← SP - 2			
		rp	1	1	—	(SP - 1) ← rpH, (SP - 2) ← rpL, SP ← SP - 2			
	POP	PSW	2	3	—	PSW ← (SP + 1), SP ← SP + 2	R	R	R
		rp	1	1	—	rpL ← (SP), rpH ← (SP + 1), SP ← SP + 2			
	MOVW	SP, #word	4	1	—	SP ← word			
		SP, AX	2	1	—	SP ← AX			
		AX, SP	2	1	—	AX ← SP			
		HL, SP	3	1	—	HL ← SP			
		BC, SP	3	1	—	BC ← SP			
		DE, SP	3	1	—	DE ← SP			
		ADDW	SP, #byte	2	1	—	SP ← SP + byte		
	SUBW	SP, #byte	2	1	—	SP ← SP - byte			
Unconditional branch	BR	AX	2	3	—	PC ← CS, AX			
		\$addr20	2	3	—	PC ← PC + 2 + jdisp8			
		\$!addr20	3	3	—	PC ← PC + 3 + jdisp16			
		!addr16	3	3	—	PC ← 0000, addr16			
		!addr20	4	3	—	PC ← addr20			
Conditional branch	BC	\$addr20	2	2/4 Note 3	—	PC ← PC + 2 + jdisp8 if CY = 1			
	BNC	\$addr20	2	2/4 Note 3	—	PC ← PC + 2 + jdisp8 if CY = 0			
	BZ	\$addr20	2	2/4 Note 3	—	PC ← PC + 2 + jdisp8 if Z = 1			
	BNZ	\$addr20	2	2/4 Note 3	—	PC ← PC + 2 + jdisp8 if Z = 0			
	BH	\$addr20	3	2/4 Note 3	—	PC ← PC + 3 + jdisp8 if (Z ∨ CY) = 0			
	BNH	\$addr20	3	2/4 Note 3	—	PC ← PC + 3 + jdisp8 if (Z ∨ CY) = 1			
	BT	saddr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if (saddr).bit = 1			
		sfr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if sfr.bit = 1			
		A.bit, \$addr20	3	3/5 Note 3	—	PC ← PC + 3 + jdisp8 if A.bit = 1			
		PSW.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if PSW.bit = 1			
		[HL].bit, \$addr20	3	3/5 Note 3	6/7	PC ← PC + 3 + jdisp8 if (HL).bit = 1			
	ES:[HL].bit, \$addr20	4	4/6 Note 3	7/8	—	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 1			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** This indicates the number of clocks “when condition is not met/when condition is met”.

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

Table 33 - 22 Operation List (18/18)

Instruction Group	Mnemonic	Operands	Bytes	Clocks		Clocks	Flag		
				Note 1	Note 2		Z	AC	CY
Conditional branch	BF	saddr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if (saddr).bit = 0			
		sfr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if sfr.bit = 0			
		A.bit, \$addr20	3	3/5 Note 3	—	PC ← PC + 3 + jdisp8 if A.bit = 0			
		PSW.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if PSW.bit = 0			
		[HL].bit, \$addr20	3	3/5 Note 3	6/7	PC ← PC + 3 + jdisp8 if (HL).bit = 0			
		ES:[HL].bit, \$addr20	4	4/6 Note 3	7/8	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 0			
	BTCLR	saddr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if (saddr).bit = 1 then reset (saddr).bit			
		sfr.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if sfr.bit = 1 then reset sfr.bit			
		A.bit, \$addr20	3	3/5 Note 3	—	PC ← PC + 3 + jdisp8 if A.bit = 1 then reset A.bit			
		PSW.bit, \$addr20	4	3/5 Note 3	—	PC ← PC + 4 + jdisp8 if PSW.bit = 1 then reset PSW.bit	x	x	x
		[HL].bit, \$addr20	3	3/5 Note 3	—	PC ← PC + 3 + jdisp8 if (HL).bit = 1 then reset (HL).bit			
		ES:[HL].bit, \$addr20	4	4/6 Note 3	—	PC ← PC + 4 + jdisp8 if (ES, HL).bit = 1 then reset (ES, HL).bit			
Conditional skip	SKC	—	2	1	—	Next instruction skip if CY = 1			
	SKNC	—	2	1	—	Next instruction skip if CY = 0			
	SKZ	—	2	1	—	Next instruction skip if Z = 1			
	SKNZ	—	2	1	—	Next instruction skip if Z = 0			
	SKH	—	2	1	—	Next instruction skip if (Z ∨ CY) = 0			
	SKNH	—	2	1	—	Next instruction skip if (Z ∨ CY) = 1			
CPU control	SEL Note 4	RBn	2	1	—	RBS[1:0] ← n			
	NOP	—	1	1	—	No Operation			
	EI	—	3	4	—	IE ← 1 (Enable Interrupt)			
	DI	—	3	4	—	IE ← 0 (Disable Interrupt)			
	HALT	—	2	3	—	Set HALT Mode			
	STOP	—	2	3	—	Set STOP Mode			

**Note 1.** Number of CPU clocks (fCLK) when the internal RAM area, SFR area, or extended SFR area is accessed, or when no data is accessed.

**Note 2.** Number of CPU clocks (fCLK) when the code flash memory is accessed, or when the data flash memory is accessed by an 8-bit instruction.

**Note 3.** This indicates the number of clocks “when condition is not met/when condition is met”.

**Note 4.** n indicates the number of register banks (n = 0 to 3)

**Remark** Number of clock is when program exists in the internal ROM (flash memory) area. If fetching the instruction from the internal RAM area, the number becomes double number plus 3 clocks at a maximum.

## CHAPTER 34 ELECTRICAL SPECIFICATIONS (TA = -40 to +85°C)

This chapter describes the following electrical specifications.

Target products A: Consumer applications TA = -40 to +85°C

R5F104xxAxx

D: Industrial applications TA = -40 to +85°C

R5F104xxDxx

G: Industrial applications when TA = -40 to +105°C products is used in the range of TA = -40 to +85°C

R5F104xxGxx

**Caution 1. The RL78 microcontrollers have an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.**

**Caution 2. With products not provided with an EVDD0, EVDD1, EVSS0, or EVSS1 pin, replace EVDD0 and EVDD1 with VDD, or replace EVSS0 and EVSS1 with Vss.**

**Caution 3. The pins mounted depend on the product. Refer to 2.1 Port Functions to 2.2.1 Functions for each product.**

### 34.1 Absolute Maximum Ratings

Absolute Maximum Ratings (1/2)				
Parameter	Symbols	Conditions	Ratings	Unit
Supply voltage	VDD		-0.5 to +6.5	V
	EVDD0, EVDD1	EVDD0 = EVDD1	-0.5 to +6.5	V
	EVSS0, EVSS1	EVSS0 = EVSS1	-0.5 to +0.3	V
REGC pin input voltage	VIREGC	REGC	-0.3 to +2.8 and -0.3 to VDD +0.3 Note 1	V
Input voltage	VI1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	-0.3 to EVDD0 +0.3 and -0.3 to VDD +0.3 Note 2	V
	VI2	P60 to P63 (N-ch open-drain)	-0.3 to +6.5	V
	VI3	P20 to P27, P121 to P124, P137, P150 to P156, EXCLK, EXCLKS, RESET	-0.3 to VDD +0.3 Note 2	V
Output voltage	VO1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	-0.3 to EVDD0 +0.3 and -0.3 to VDD +0.3 Note 2	V
	VO2	P20 to P27, P150 to P156	-0.3 to VDD +0.3 Note 2	V
Analog input voltage	VAI1	ANI16 to ANI20	-0.3 to EVDD0 +0.3 and -0.3 to AVREF(+) +0.3 Notes 2, 3	V
	VAI2	ANI0 to ANI14	-0.3 to VDD +0.3 and -0.3 to AVREF(+) +0.3 Notes 2, 3	V

**Note 1.** Connect the REGC pin to Vss via a capacitor (0.47 to 1  $\mu$ F). This value regulates the absolute maximum rating of the REGC pin. Do not use this pin with voltage applied to it.

**Note 2.** Must be 6.5 V or lower.

**Note 3.** Do not exceed AVREF (+) + 0.3 V in case of A/D conversion target pin.

**Caution** Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

**Remark 1.** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

**Remark 2.** AVREF (+): + side reference voltage of the A/D converter.

**Remark 3.** Vss: Reference voltage

**Absolute Maximum Ratings**

(2/2)

Parameter	Symbols	Conditions		Ratings	Unit	
Output current, high	IOH1	Per pin	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	-40	mA	
		Total of all pins -170 mA	P00 to P04, P40 to P47, P102, P120, P130, P140 to P145	-70	mA	
			P05, P06, P10 to P17, P30, P31, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147	-100	mA	
	IOL2	Per pin	P20 to P27, P150 to P156	-0.5	mA	
		Total of all pins		-2	mA	
	IOL1	Per pin	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	40	mA	
		Total of all pins 170 mA	P00 to P04, P40 to P47, P102, P120, P130, P140 to P145	70	mA	
			P05, P06, P10 to P17, P30, P31, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147	100	mA	
		Per pin	P20 to P27, P150 to P156	1	mA	
		Total of all pins		5	mA	
Operating ambient temperature	TA	In normal operation mode		-40 to +85	°C	
Storage temperature	Tstg					

**Caution** Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

## 34.2 Oscillator Characteristics

### 34.2.1 X1, XT1 characteristics

(TA = -40 to +85°C, 1.6 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Resonator	Resonator	Conditions	MIN.	TYP.	MAX.	Unit
X1 clock oscillation frequency (fx) Note	Ceramic resonator/ crystal resonator	2.7 V ≤ VDD ≤ 5.5 V	1.0		20.0	MHz
		2.4 V ≤ VDD < 2.7 V	1.0		16.0	
		1.8 V ≤ VDD < 2.4 V	1.0		8.0	
		1.6 V ≤ VDD < 1.8 V	1.0		4.0	
XT1 clock oscillation frequency (fxT) Note	Crystal resonator		32	32.768	35	kHz

**Note** Indicates only permissible oscillator frequency ranges. Refer to **AC Characteristics** for instruction execution time.  
Request evaluation by the manufacturer of the oscillator circuit mounted on a board to check the oscillator characteristics.

**Caution** Since the CPU is started by the high-speed on-chip oscillator clock after a reset release, check the X1 clock oscillation stabilization time using the oscillation stabilization time counter status register (OSTC) by the user. Determine the oscillation stabilization time of the OSTC register and the oscillation stabilization time select register (OSTS) after sufficiently evaluating the oscillation stabilization time with the resonator to be used.

**Remark** When using the X1 oscillator and XT1 oscillator, refer to **5.4 System Clock Oscillator**.

### 34.2.2 On-chip oscillator characteristics

(TA = -40 to +85°C, 1.6 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Oscillators	Parameters	Conditions		MIN.	TYP.	MAX.	Unit
High-speed on-chip oscillator clock frequency Notes 1, 2	fIH			1		32	MHz
High-speed on-chip oscillator clock frequency accuracy		-20 to +85°C	1.8 V ≤ VDD ≤ 5.5 V	-1.0		+1.0	%
			1.6 V ≤ VDD < 1.8 V	-5.0		+5.0	%
		-40 to -20°C	1.8 V ≤ VDD < 5.5 V	-1.5		+1.5	%
			1.6 V ≤ VDD < 1.8 V	-5.5		+5.5	%
Low-speed on-chip oscillator clock frequency	fIL				15		kHz
Low-speed on-chip oscillator clock frequency accuracy				-15		+15	%

**Note 1.** High-speed on-chip oscillator frequency is selected with bits 0 to 4 of the option byte (000C2H) and bits 0 to 2 of the HOCODIV register.

**Note 2.** This only indicates the oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

### 34.3 DC Characteristics

#### 34.3.1 Pin characteristics

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (1/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, high Note 1	IOH1	Per pin for P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	1.6 V ≤ EVDD0 ≤ 5.5 V			-10.0 Note 2	mA
		Total of P00 to P04, P40 to P47, P102, P120, P130, P140 to P145 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V			-55.0	mA
			2.7 V ≤ EVDD0 < 4.0 V			-10.0	mA
			1.8 V ≤ EVDD0 < 2.7 V			-5.0	mA
			1.6 V ≤ EVDD0 < 1.8 V			-2.5	mA
	Total of P05, P06, P10 to P17, P30, P31, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V				-80.0	mA
			2.7 V ≤ EVDD0 < 4.0 V			-19.0	mA
			1.8 V ≤ EVDD0 < 2.7 V			-10.0	mA
			1.6 V ≤ EVDD0 < 1.8 V			-5.0	mA
	Total of all pins (When duty ≤ 70% Note 3)	1.6 V ≤ EVDD0 ≤ 5.5 V				-135.0 Note 4	mA
		Per pin for P20 to P27, P150 to P156	1.6 V ≤ VDD ≤ 5.5 V			-0.1 Note 2	mA
		Total of all pins (When duty ≤ 70% Note 3)	1.6 V ≤ VDD ≤ 5.5 V			-1.5	mA

**Note 1.** Value of current at which the device operation is guaranteed even if the current flows from the EVDD0, EVDD1, VDD pins to an output pin.

**Note 2.** Do not exceed the total current value.

**Note 3.** Specification under conditions where the duty factor ≤ 70%.

The output current value that has changed to the duty factor > 70% the duty ratio can be calculated with the following expression (when changing the duty factor from 70% to n%).

- Total output current of pins =  $(IOH \times 0.7)/(n \times 0.01)$

<Example> Where n = 80% and IOH = -10.0 mA

$$\text{Total output current of pins} = (-10.0 \times 0.7)/(80 \times 0.01) \approx -8.7 \text{ mA}$$

However, the current that is allowed to flow into one pin does not vary depending on the duty factor.

A current higher than the absolute maximum rating must not flow into one pin.

**Note 4.** -100 mA for industrial applications (R5F104xxDxx, R5F104xxGxx).

**Caution** P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, and P142 to P144 do not output high level in N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (2/5)

Items	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Output current, low Note 1	IOL1	Per pin for P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147			20.0 Note 2	mA
		Per pin for P60 to P63			15.0 Note 2	mA
		Total of P00 to P04, P40 to P47, P102, P120, P130, P140 to P145 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V		70.0	mA
			2.7 V ≤ EVDD0 < 4.0 V		15.0	mA
			1.8 V ≤ EVDD0 < 2.7 V		9.0	mA
			1.6 V ≤ EVDD0 < 1.8 V		4.5	mA
		Total of P05, P06, P10 to P17, P30, P31, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V		80.0	mA
			2.7 V ≤ EVDD0 < 4.0 V		35.0	mA
			1.8 V ≤ EVDD0 < 2.7 V		20.0	mA
			1.6 V ≤ EVDD0 < 1.8 V		10.0	mA
		Total of all pins (When duty ≤ 70% Note 3)			150.0	mA
	IOL2	Per pin for P20 to P27, P150 to P156			0.4 Note 2	mA
		Total of all pins (When duty ≤ 70% Note 3)	1.6 V ≤ VDD ≤ 5.5 V		5.0	mA

**Note 1.** Value of current at which the device operation is guaranteed even if the current flows from an output pin to the EVSS0, EVSS1, and Vss pins.

**Note 2.** Do not exceed the total current value.

**Note 3.** Specification under conditions where the duty factor ≤ 70%.

The output current value that has changed to the duty factor > 70% the duty ratio can be calculated with the following expression (when changing the duty factor from 70% to n%).

- Total output current of pins =  $(I_{OL} \times 0.7)/(n \times 0.01)$

<Example> Where n = 80% and IOL = 10.0 mA

$$\text{Total output current of pins} = (10.0 \times 0.7)/(80 \times 0.01) \approx 8.7 \text{ mA}$$

However, the current that is allowed to flow into one pin does not vary depending on the duty factor.

A current higher than the absolute maximum rating must not flow into one pin.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V) (3/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage, high	VIH1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	Normal input buffer	0.8 EV <sub>DD0</sub>		EV <sub>DD0</sub>	V
	VIH2	P01, P03, P04, P10, P14 to P17, P30, P43, P44, P50, P53 to P55, P80, P81, P142, P143	TTL input buffer 4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	2.2		EV <sub>DD0</sub>	V
			TTL input buffer 3.3 V ≤ EV <sub>DD0</sub> < 4.0 V	2.0		EV <sub>DD0</sub>	V
			TTL input buffer 1.6 V ≤ EV <sub>DD0</sub> < 3.3 V	1.5		EV <sub>DD0</sub>	V
	VIH3	P20 to P27, P150 to P156		0.7 V <sub>DD</sub>		V <sub>DD</sub>	V
	VIH4	P60 to P63		0.7 EV <sub>DD0</sub>		6.0	V
	VIH5	P121 to P124, P137, EXCLK, EXCLKS, RESET		0.8 V <sub>DD</sub>		V <sub>DD</sub>	V
Input voltage, low	VIL1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	Normal input buffer	0		0.2 EV <sub>DD0</sub>	V
	VIL2	P01, P03, P04, P10, P14 to P17, P30, P43, P44, P50, P53 to P55, P80, P81, P142, P143	TTL input buffer 4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0		0.8	V
			TTL input buffer 3.3 V ≤ EV <sub>DD0</sub> < 4.0 V	0		0.5	V
			TTL input buffer 1.6 V ≤ EV <sub>DD0</sub> < 3.3 V	0		0.32	V
	VIL3	P20 to P27, P150 to P156		0		0.3 V <sub>DD</sub>	V
	VIL4	P60 to P63		0		0.3 EV <sub>DD0</sub>	V
	VIL5	P121 to P124, P137, EXCLK, EXCLKS, RESET		0		0.2 V <sub>DD</sub>	V

**Caution** The maximum value of VIH of pins P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, and P142 to P144 is EV<sub>DD0</sub>, even in the N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V) (4/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output voltage, high	V <sub>OH1</sub>	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OH1</sub> = -10.0 mA	EV <sub>DD0</sub> - 1.5			V
			4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OH1</sub> = -3.0 mA	EV <sub>DD0</sub> - 0.7			V
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OH1</sub> = -1.5 mA	EV <sub>DD0</sub> - 0.5			V
			1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, I <sub>OH1</sub> = -1.0 mA	EV <sub>DD0</sub> - 0.5			V
	V <sub>OH2</sub>	P20 to P27, P150 to P156	1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V, I <sub>OH2</sub> = -100 μA	V <sub>DD</sub> - 0.5			V
Output voltage, low	V <sub>OL1</sub>	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 20.0 mA			1.3	V
			4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 8.5 mA			0.7	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 3.0 mA			0.6	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 1.5 mA			0.4	V
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 0.6 mA			0.4	V
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 0.3 mA			0.4	V
	V <sub>OL2</sub>	P20 to P27, P150 to P156	1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V, I <sub>OL2</sub> = 400 μA			0.4	V
	V <sub>OL3</sub>	P60 to P63	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 15.0 mA			2.0	V
			4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 5.0 mA			0.4	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 3.0 mA			0.4	V
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 2.0 mA			0.4	V
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 1.0 mA			0.4	V

**Caution** P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, P142 to P144 do not output high level in N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (5/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input leakage current, high	ILIH1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	VI = EVDD0			1	µA
	ILIH2	P20 to P27, P137, P150 to P156, <u>RESET</u>	VI = VDD			1	µA
	ILIH3	P121 to P124 (X1, X2, EXCLK, XT1, XT2, EXCLKS)	VI = VDD	In input port or external clock input		1	µA
Input leakage current, low	ILIL1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147		In resonator connection		10	µA
		P20 to P27, P137, P150 to P156, <u>RESET</u>	VI = Vss			-1	µA
		P121 to P124 (X1, X2, EXCLK, XT1, XT2, EXCLKS)	VI = Vss	In input port or external clock input		-1	µA
On-chip pull-up resistance	Ru	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147		In resonator connection		-10	µA
			VI = EVSS0, In input port		10	20	100

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

### 34.3.2 Supply current characteristics

#### (1) Flash ROM: 16 to 64 KB of 30- to 64-pin products

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions					MIN.	TYP.	MAX.	Unit
Supply current Note 1	IDD1	Operating mode	HS (high-speed main mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.4		mA
						V <sub>DD</sub> = 3.0 V		2.4		
		HS (high-speed main mode Note 5	f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.1			mA
						V <sub>DD</sub> = 3.0 V		2.1		
			f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.1	8.7		
						V <sub>DD</sub> = 3.0 V		5.1	8.7	
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.8	8.1		
						V <sub>DD</sub> = 3.0 V		4.8	8.1	
			f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.0	6.9		
						V <sub>DD</sub> = 3.0 V		4.0	6.9	
			f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		3.8	6.3		
						V <sub>DD</sub> = 3.0 V		3.8	6.3	
		LS (low-speed main mode Note 5	f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		2.8	4.6		mA
						V <sub>DD</sub> = 3.0 V		2.8	4.6	
		LV (low-voltage main mode Note 5	f <sub>HOCO</sub> = 4 MHz, f <sub>IH</sub> = 4 MHz Note 3	Normal operation	V <sub>DD</sub> = 3.0 V		1.3	2.0		mA
						V <sub>DD</sub> = 2.0 V		1.3	2.0	
		HS (high-speed main mode Note 5	f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		3.3	5.3		mA
					Resonator connection		3.4	5.5		
			f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		3.3	5.3		
					Resonator connection		3.4	5.5		
			f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		2.0	3.1		
					Resonator connection		2.1	3.2		
		LS (low-speed main mode Note 5	f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		2.0	3.1		
					Resonator connection		2.1	3.2		
			f <sub>MX</sub> = 8 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		1.2	1.9		mA
					Resonator connection		1.2	2.0		
		Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		4.7	6.1		μA
					Resonator connection		4.7	6.1		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		4.7	6.1		
					Resonator connection		4.7	6.1		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		4.8	6.7		
					Resonator connection		4.8	6.7		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		4.8	7.5		
					Resonator connection		4.8	7.5		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input		5.4	8.9		
					Resonator connection		5.4	8.9		

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub> and EV<sub>DD0</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub> or V<sub>SS</sub>, EV<sub>SS0</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz

LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz

LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>H</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is TA = 25°C

## (1) Flash ROM: 16 to 64 KB of 30- to 64-pin products

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	IDD2 Note 2	HALT mode	HS (high-speed main) mode Note 7	fHO CO = 64 MHz, fIH = 32 MHz Note 4	VDD = 5.0 V		0.80	3.09	mA
					VDD = 3.0 V		0.80	3.09	
				fHO CO = 32 MHz, fIH = 32 MHz Note 4	VDD = 5.0 V		0.49	2.40	
					VDD = 3.0 V		0.49	2.40	
				fHO CO = 48 MHz, fIH = 24 MHz Note 4	VDD = 5.0 V		0.62	2.40	
					VDD = 3.0 V		0.62	2.40	
				fHO CO = 24 MHz, fIH = 24 MHz Note 4	VDD = 5.0 V		0.4	1.83	
					VDD = 3.0 V		0.4	1.83	
				fHO CO = 16 MHz, fIH = 16 MHz Note 4	VDD = 5.0 V		0.37	1.38	
					VDD = 3.0 V		0.37	1.38	
			LS (low-speed main) mode Note 7	fHO CO = 8 MHz, fIH = 8 MHz Note 4	VDD = 3.0 V		260	710	μA
					VDD = 2.0 V		260	710	
			LV (low-voltage main) mode Note 7	fHO CO = 4 MHz, fIH = 4 MHz Note 4	VDD = 3.0 V		420	700	μA
					VDD = 2.0 V		420	700	
			HS (high-speed main) mode Note 7	fMX = 20 MHz Note 3, VDD = 5.0 V	Square wave input		0.28	1.55	mA
					Resonator connection		0.40	1.74	
				fMX = 20 MHz Note 3, VDD = 3.0 V	Square wave input		0.28	1.55	
					Resonator connection		0.40	1.74	
				fMX = 10 MHz Note 3, VDD = 5.0 V	Square wave input		0.19	0.86	
					Resonator connection		0.25	0.93	
				fMX = 10 MHz Note 3, VDD = 3.0 V	Square wave input		0.19	0.86	
					Resonator connection		0.25	0.93	
			LS (low-speed main) mode Note 7	fMX = 8 MHz Note 3, VDD = 3.0 V	Square wave input		95	550	μA
					Resonator connection		140	590	
				fMX = 8 MHz Note 3, VDD = 2.0 V	Square wave input		95	550	
					Resonator connection		140	590	
			Subsystem clock operation	fSUB = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.25	0.57	μA
					Resonator connection		0.44	0.76	
				fSUB = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.30	0.57	
					Resonator connection		0.49	0.76	
				fSUB = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.36	1.17	
					Resonator connection		0.59	1.36	
				fSUB = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.49	1.97	
					Resonator connection		0.72	2.16	
				fSUB = 32.768 kHz Note 5, TA = +85°C	Square wave input		0.97	3.37	
					Resonator connection		1.16	3.56	
			STOP mode Note 8	TA = -40°C			0.18	0.51	μA
				TA = +25°C			0.24	0.51	
				TA = +50°C			0.29	1.10	
				TA = +70°C			0.41	1.90	
				TA = +85°C			0.90	3.30	

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub> and EV<sub>DD0</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub> or V<sub>SS</sub>, EV<sub>SS0</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz  
LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz  
LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

## (2) Flash ROM: 96 to 256 KB of 30- to 100-pin products

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions					MIN.	TYP.	MAX.	Unit	
Supply current Note 1	I <sub>DD1</sub>	Operating mode	HS (high-speed main) mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.6		mA	
						V <sub>DD</sub> = 3.0 V		2.6			
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.3			
						V <sub>DD</sub> = 3.0 V		2.3			
		HS (high-speed main) mode Note 5		f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.4	10.2	mA	
						V <sub>DD</sub> = 3.0 V		5.4	10.2		
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.0	9.6		
						V <sub>DD</sub> = 3.0 V		5.0	9.6		
				f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.2	7.8		
						V <sub>DD</sub> = 3.0 V		4.2	7.8		
		LS (low-speed main) mode Note 5		f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.0	7.4	mA	
						V <sub>DD</sub> = 3.0 V		4.0	7.4		
				f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		3.0	5.3		
		LV (low-voltage main) mode Note 5				V <sub>DD</sub> = 3.0 V		3.0	5.3	mA	
				f <sub>HOCO</sub> = 8 MHz, f <sub>IH</sub> = 8 MHz Note 3	Normal operation	V <sub>DD</sub> = 2.0 V		1.4	2.3		
		HS (high-speed main) mode Note 5		f <sub>HOCO</sub> = 4 MHz, f <sub>IH</sub> = 4 MHz Note 3	Normal operation	V <sub>DD</sub> = 3.0 V		1.3	1.9	mA	
						V <sub>DD</sub> = 2.0 V		1.3	1.9		
				f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		3.4	6.2		
						Resonator connection		3.6	6.4		
				f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		3.4	6.2		
						Resonator connection		3.6	6.4		
		LS (low-speed main) mode Note 5		f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		2.1	3.6	mA	
						Resonator connection		2.2	3.7		
				f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		2.1	3.6		
						Resonator connection		2.2	3.7		
		Subsystem clock operation		f <sub>MX</sub> = 8 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		1.2	2.2	μA	
						Resonator connection		1.2	2.3		
				f <sub>MX</sub> = 8 MHz Note 2, V <sub>DD</sub> = 2.0 V	Normal operation	Square wave input		1.2	2.2		
						Resonator connection		1.2	2.3		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		4.9	7.1	μA	
						Resonator connection		4.9	7.1		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		4.9	7.1		
						Resonator connection		4.9	7.1		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		5.1	8.8		
						Resonator connection		5.1	8.8		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		5.5	10.5		
						Resonator connection		5.5	10.5		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input		6.5	14.5		
						Resonator connection		6.5	14.5		

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the 12-bit interval timer and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz

LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz

LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>iH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is TA = 25°C

## (2) Flash ROM: 96 to 256 KB of 30- to 100-pin products

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	I <sub>DD2</sub> Note 2	HALT mode mode Note 7	HS (high-speed main) mode Note 7	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.79	3.32	mA
					V <sub>DD</sub> = 3.0 V		0.79	3.32	
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.49	2.63	
					V <sub>DD</sub> = 3.0 V		0.49	2.63	
				f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.62	2.57	
					V <sub>DD</sub> = 3.0 V		0.62	2.57	
				f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.4	2.00	
					V <sub>DD</sub> = 3.0 V		0.4	2.00	
				f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.38	1.49	
					V <sub>DD</sub> = 3.0 V		0.38	1.49	
			LS (low-speed main) mode Note 7	f <sub>HOCO</sub> = 8 MHz, f <sub>IH</sub> = 8 MHz Note 4	V <sub>DD</sub> = 3.0 V		250	800	μA
					V <sub>DD</sub> = 2.0 V		250	800	
			LV (low-voltage main) mode Note 7	f <sub>HOCO</sub> = 4 MHz, f <sub>IH</sub> = 4 MHz Note 4	V <sub>DD</sub> = 3.0 V		420	755	μA
					V <sub>DD</sub> = 2.0 V		420	755	
			HS (high-speed main) mode Note 7	f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.30	1.63	mA
					Resonator connection		0.40	1.85	
				f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.30	1.63	
					Resonator connection		0.40	1.85	
				f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.20	0.89	
					Resonator connection		0.25	0.97	
				f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.20	0.89	
					Resonator connection		0.25	0.97	
			LS (low-speed main) mode Note 7	f <sub>MX</sub> = 8 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		110	580	μA
					Resonator connection		140	630	
				f <sub>MX</sub> = 8 MHz Note 3, V <sub>DD</sub> = 2.0 V	Square wave input		110	580	
					Resonator connection		140	630	
			Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.28	0.66	μA
					Resonator connection		0.47	0.85	
				f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.34	0.66	
					Resonator connection		0.53	0.85	
				f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.37	2.35	
					Resonator connection		0.56	2.54	
				f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.61	4.08	
					Resonator connection		0.80	4.27	
				f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +85°C	Square wave input		1.55	8.09	
					Resonator connection		1.74	8.28	
		STOP mode Note 8	TA = -40°C				0.19	0.57	μA
			TA = +25°C				0.25	0.57	
			TA = +50°C				0.33	2.26	
			TA = +70°C				0.52	3.99	
			TA = +85°C				1.46	8.00	

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz  
LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz  
LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

## (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(1/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	IDD1	Operating mode	HS (high-speed main) mode Note 5	fHO CO = 64 MHz, fIH = 32 MHz Note 3	Basic operation	VDD = 5.0 V		2.9	mA
						VDD = 3.0 V		2.9	
				fHO CO = 32 MHz, fIH = 32 MHz Note 3	Basic operation	VDD = 5.0 V		2.5	
						VDD = 3.0 V		2.5	
		HS (high-speed main) mode Note 5	fHO CO = 64 MHz, fIH = 32 MHz Note 3	Normal operation	VDD = 5.0 V		6.0	11.2	mA
						VDD = 3.0 V		6.0	
			fHO CO = 32 MHz, fIH = 32 MHz Note 3	Normal operation	VDD = 5.0 V		5.5	10.6	
						VDD = 3.0 V		5.5	
			fHO CO = 48 MHz, fIH = 24 MHz Note 3	Normal operation	VDD = 5.0 V		4.7	8.6	
						VDD = 3.0 V		4.7	
		LS (low-speed main) mode Note 5	fHO CO = 24 MHz, fIH = 24 MHz Note 3	Normal operation	VDD = 5.0 V		4.4	8.2	mA
						VDD = 3.0 V		4.4	
			fHO CO = 16 MHz, fIH = 16 MHz Note 3	Normal operation	VDD = 5.0 V		3.3	5.9	
		LV (low-voltage main) mode Note 5				VDD = 3.0 V		3.3	5.9
			fHO CO = 8 MHz, fIH = 8 MHz Note 3	Normal operation	VDD = 3.0 V		1.5	2.5	mA
		HS (high-speed main) mode Note 5	fMX = 4 MHz, fIH = 4 MHz Note 3	Normal operation	VDD = 2.0 V		1.5	2.5	
					VDD = 2.0 V		1.5	2.5	mA
		HS (high-speed main) mode Note 5	fMX = 20 MHz Note 2, VDD = 5.0 V	Normal operation	Square wave input		3.7	6.8	mA
					Resonator connection		3.9	7.0	
			fMX = 20 MHz Note 2, VDD = 3.0 V	Normal operation	Square wave input		3.7	6.8	
					Resonator connection		3.9	7.0	
			fMX = 10 MHz Note 2, VDD = 5.0 V	Normal operation	Square wave input		2.3	4.1	
					Resonator connection		2.3	4.2	
		LS (low-speed main) mode Note 5	fMX = 10 MHz Note 2, VDD = 3.0 V	Normal operation	Square wave input		2.3	4.1	mA
					Resonator connection		2.3	4.2	
			fMX = 8 MHz Note 2, VDD = 3.0 V	Normal operation	Square wave input		1.4	2.4	
					Resonator connection		1.4	2.5	
		Subsystem clock operation	fMX = 8 MHz Note 2, VDD = 2.0 V	Normal operation	Square wave input		1.4	2.4	μA
					Resonator connection		1.4	2.5	
			fSUB = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		5.2		
					Resonator connection		5.2		
			fSUB = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		5.3	7.7	
					Resonator connection		5.3	7.7	
		fSUB = 32.768 kHz Note 4 TA = +50°C	fSUB = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		5.5	10.6	μA
					Resonator connection		5.5	10.6	
			fSUB = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		5.9	13.2	
					Resonator connection		6.0	13.2	
		fSUB = 32.768 kHz Note 4 TA = +85°C	fSUB = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input		6.8	17.5	μA
					Resonator connection		6.9	17.5	

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the 12-bit interval timer and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz

LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz

LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>iH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is TA = 25°C

## (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	I <sub>DD2</sub> Note 2	HALT mode HS (high-speed main) mode Note 7	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.93	3.32		mA
				V <sub>DD</sub> = 3.0 V		0.93	3.32		
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.5	2.63		
				V <sub>DD</sub> = 3.0 V		0.5	2.63		
			f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.72	2.60		
				V <sub>DD</sub> = 3.0 V		0.72	2.60		
			f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.42	2.03		
				V <sub>DD</sub> = 3.0 V		0.42	2.03		
			f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.39	1.50		
				V <sub>DD</sub> = 3.0 V		0.39	1.50		
		LS (low-speed main) mode Note 7	f <sub>HOCO</sub> = 8 MHz, f <sub>IH</sub> = 8 MHz Note 4	V <sub>DD</sub> = 3.0 V		270	800		μA
				V <sub>DD</sub> = 2.0 V		270	800		
		LV (low-voltage main) mode Note 7	f <sub>HOCO</sub> = 4 MHz, f <sub>IH</sub> = 4 MHz Note 4	V <sub>DD</sub> = 3.0 V		450	755		μA
				V <sub>DD</sub> = 2.0 V		450	755		
		HS (high-speed main) mode Note 7	f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.31	1.69		mA
				Resonator connection		0.41	1.91		
			f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.31	1.69		
				Resonator connection		0.41	1.91		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.21	0.94		
				Resonator connection		0.26	1.02		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.21	0.94		
				Resonator connection		0.26	1.02		
			f <sub>MX</sub> = 8 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		110	610		μA
				Resonator connection		150	660		
		LS (low-speed main) mode Note 7	f <sub>MX</sub> = 8 MHz Note 3, V <sub>DD</sub> = 2.0 V	Square wave input		110	610		
				Resonator connection		150	660		
		Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.31			μA
				Resonator connection		0.50			
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.38	0.76		
				Resonator connection		0.57	0.95		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.47	3.59		
				Resonator connection		0.70	3.78		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.80	6.20		
				Resonator connection		1.00	6.39		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +85°C	Square wave input		1.65	10.56		
				Resonator connection		1.84	10.75		
		I <sub>DD3</sub> Note 6	STOP mode Note 8	TA = -40°C		0.19			μA
				TA = +25°C		0.30	0.59		
				TA = +50°C		0.41	3.42		
				TA = +70°C		0.80	6.03		
				TA = +85°C		1.53	10.39		

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz  
LS (low-speed main) mode: 1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 8 MHz  
LV (low-voltage main) mode: 1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 4 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>H</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

**(4) Peripheral Functions (Common to all products)**

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Low-speed on-chip oscillator operating current	I <sub>FIL</sub> Note 1				0.20		µA
RTC operating current	I <sub>RTC</sub> Notes 1, 2, 3				0.02		µA
12-bit interval timer operating current	I <sub>IT</sub> Notes 1, 2, 4				0.02		µA
Watchdog timer operating current	I <sub>WDT</sub> Notes 1, 2, 5	f <sub>L</sub> = 15 kHz			0.22		µA
A/D converter operating current	I <sub>AADC</sub> Notes 1, 6	When conversion at maximum speed	Normal mode, AVREFP = VDD = 5.0 V		1.3	1.7	mA
			Low voltage mode, AVREFP = VDD = 3.0 V		0.5	0.7	mA
A/D converter reference voltage current	I <sub>ADREF</sub> Note 1				75.0		µA
Temperature sensor operating current	I <sub>TMPS</sub> Note 1				75.0		µA
D/A converter operating current	I <sub>DAC</sub> Notes 1, 11, 13	Per D/A converter channel				1.5	mA
Comparator operating current	I <sub>CMP</sub> Notes 1, 12, 13	V <sub>DD</sub> = 5.0 V, Regulator output voltage = 2.1 V	Window mode		12.5		µA
			Comparator high-speed mode		6.5		µA
			Comparator low-speed mode		1.7		µA
		V <sub>DD</sub> = 5.0 V, Regulator output voltage = 1.8 V	Window mode		8.0		µA
			Comparator high-speed mode		4.0		µA
			Comparator low-speed mode		1.3		µA
LVD operating current	I <sub>LVD</sub> Notes 1, 7				0.08		µA
Self-programming operating current	I <sub>FSPI</sub> Notes 1, 9				2.50	12.20	mA
BGO operating current	I <sub>BGO</sub> Notes 1, 8				2.50	12.20	mA
SNOOZE operating current	I <sub>SNOZ</sub> Note 1	ADC operation	The mode is performed Note 10		0.50	0.60	mA
			The A/D conversion operations are performed, Low voltage mode, AVREFP = VDD = 3.0 V		1.20	1.44	
		CSI/UART operation			0.70	0.84	
		DTC operation			3.10		

**Note 1.** Current flowing to V<sub>DD</sub>.**Note 2.** When high speed on-chip oscillator and high-speed system clock are stopped.**Note 3.** Current flowing only to the real-time clock (RTC) (excluding the operating current of the low-speed on-chip oscillator and the XT1 oscillator). The supply current of the RL78 microcontrollers is the sum of the values of either I<sub>DD1</sub> or I<sub>DD2</sub>, and I<sub>RTC</sub>, when the real-time clock operates in operation mode or HALT mode. When the low-speed on-chip oscillator is selected, I<sub>FIL</sub> should be added. I<sub>DD2</sub> subsystem clock operation includes the operational current of the real-time clock.**Note 4.** Current flowing only to the 12-bit interval timer (excluding the operating current of the low-speed on-chip oscillator and the XT1 oscillator). The supply current of the RL78 microcontrollers is the sum of the values of either I<sub>DD1</sub> or I<sub>DD2</sub>, and I<sub>IT</sub>, when the 12-bit interval timer operates in operation mode or HALT mode. When the low-speed on-chip oscillator is selected, I<sub>FIL</sub> should be added.

- Note 5.** Current flowing only to the watchdog timer (including the operating current of the low-speed on-chip oscillator). The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2 or IDD3 and I<sub>WDT</sub> when the watchdog timer is in operation.
- Note 6.** Current flowing only to the A/D converter. The supply current of the RL78 microcontrollers is the sum of IDD1 or IDD2 and I<sub>AADC</sub> when the A/D converter operates in an operation mode or the HALT mode.
- Note 7.** Current flowing only to the LVD circuit. The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2 or IDD3 and I<sub>LVD</sub> when the LVD circuit is in operation.
- Note 8.** Current flowing during programming of the data flash.
- Note 9.** Current flowing during self-programming.
- Note 10.** For shift time to the SNOOZE mode, see **23.3.3 SNOOZE mode**.
- Note 11.** Current flowing only to the D/A converter. The supply current of the RL78 microcontrollers is the sum of IDD1 or IDD2 and I<sub>DAC</sub> when the D/A converter operates in an operation mode or the HALT mode.
- Note 12.** Current flowing only to the comparator circuit. The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2, or IDD3 and I<sub>CMP</sub> when the comparator circuit is in operation.
- Note 13.** A comparator and D/A converter are provided in products with 96 KB or more code flash memory.

**Remark 1.** f<sub>IL</sub>: Low-speed on-chip oscillator clock frequency

**Remark 2.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 3.** f<sub>CLK</sub>: CPU/peripheral hardware clock frequency

**Remark 4.** Temperature condition of the TYP. value is TA = 25°C

### 34.4 AC Characteristics

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(1/2)

Items	Symbol	Conditions			MIN.	TYP.	MAX.	Unit
Instruction cycle (minimum instruction execution time)	TCY	Main system clock (f <sub>MAIN</sub> ) operation	HS (high-speed main) mode	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.03125		1	μs
			LS (low-speed main) mode	2.4 V ≤ V <sub>DD</sub> < 2.7 V	0.0625		1	μs
			LV (low-voltage main) mode	1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.125		1	μs
			Subsystem clock (f <sub>SUB</sub> ) operation	1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V	28.5	30.5	31.3	μs
		In the self-programming mode	HS (high-speed main) mode	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.03125		1	μs
			LS (low-speed main) mode	2.4 V ≤ V <sub>DD</sub> < 2.7 V	0.0625		1	μs
			LV (low-voltage main) mode	1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.125		1	μs
				1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.25		1	μs
External system clock frequency	f <sub>EX</sub>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			1.0		20.0	MHz
		2.4 V ≤ V <sub>DD</sub> ≤ 2.7 V			1.0		16.0	MHz
		1.8 V ≤ V <sub>DD</sub> < 2.4 V			1.0		8.0	MHz
		1.6 V ≤ V <sub>DD</sub> < 1.8 V			1.0		4.0	MHz
	f <sub>EXS</sub>				32		35	kHz
External system clock input high-level width, low-level width	t <sub>EXH</sub> , t <sub>EXL</sub>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			24			ns
		2.4 V ≤ V <sub>DD</sub> ≤ 2.7 V			30			ns
		1.8 V ≤ V <sub>DD</sub> < 2.4 V			60			ns
		1.6 V ≤ V <sub>DD</sub> < 1.8 V			120			ns
	t <sub>EXHS</sub> , t <sub>EXLS</sub>				13.7			μs
TI00 to TI03, TI10 to TI13 input high-level width, low-level width	t <sub>TIH</sub> , t <sub>TIL</sub>				1/f <sub>MCK</sub> + 10 Note			ns
Timer RJ input cycle	fc	TRJIO	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		100			ns
			1.8 V ≤ EV <sub>D0</sub> < 2.7 V		300			ns
			1.6 V ≤ EV <sub>D0</sub> < 1.8 V		500			ns
Timer RJ input high-level width, low-level width	t <sub>TJIH</sub> , t <sub>TJIL</sub>	TRJIO	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		40			ns
			1.8 V ≤ EV <sub>D0</sub> < 2.7 V		120			ns
			1.6 V ≤ EV <sub>D0</sub> < 1.8 V		200			ns

**Note** The following conditions are required for low voltage interface when EV<sub>D0</sub> < V<sub>DD</sub>1.8 V ≤ EV<sub>D0</sub> < 2.7 V: MIN. 125 ns1.6 V ≤ EV<sub>D0</sub> < 1.8 V: MIN. 250 ns**Remark** fmck: Timer array unit operation clock frequency

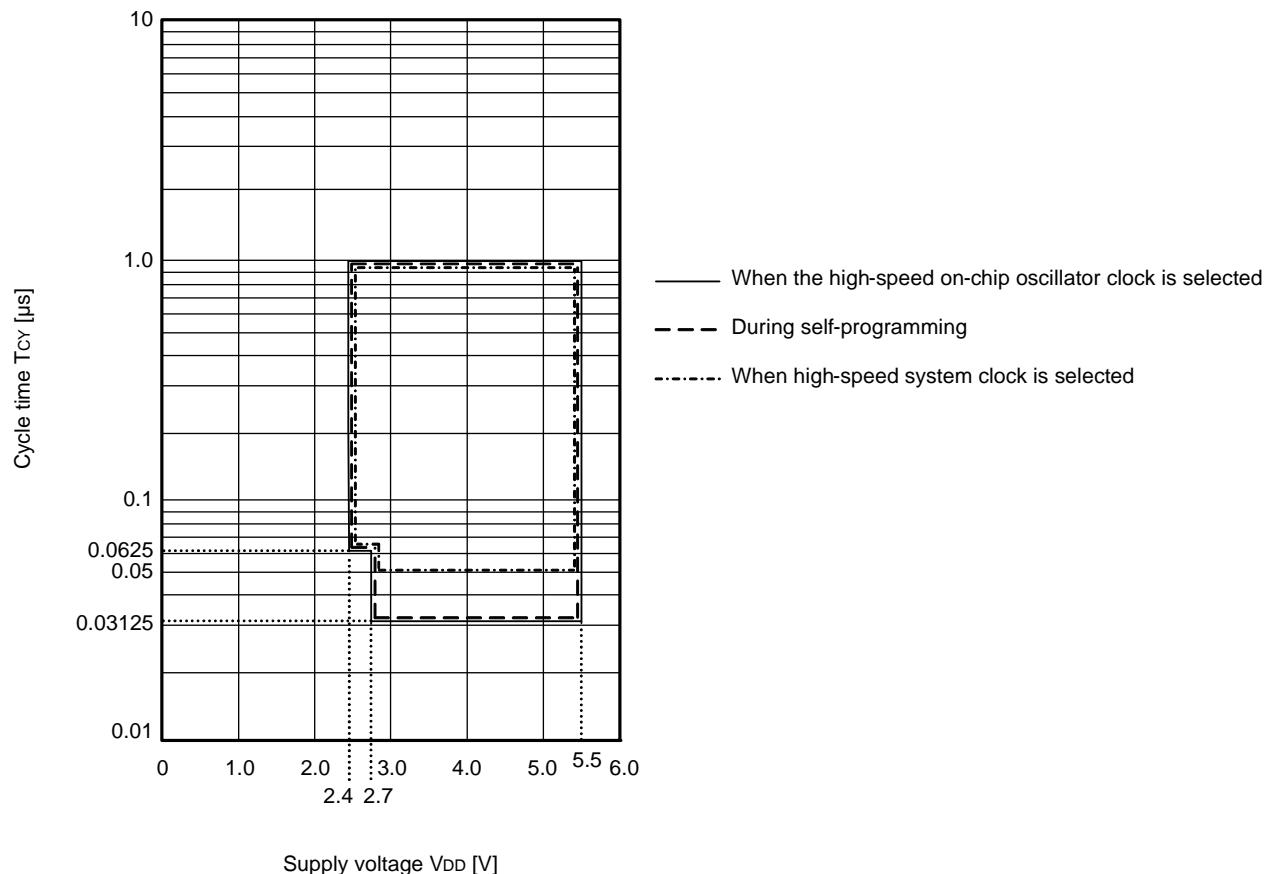
(Operation clock to be set by the CKSmn bit of timer mode register mn (TMRmn). m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3))

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (2/2)

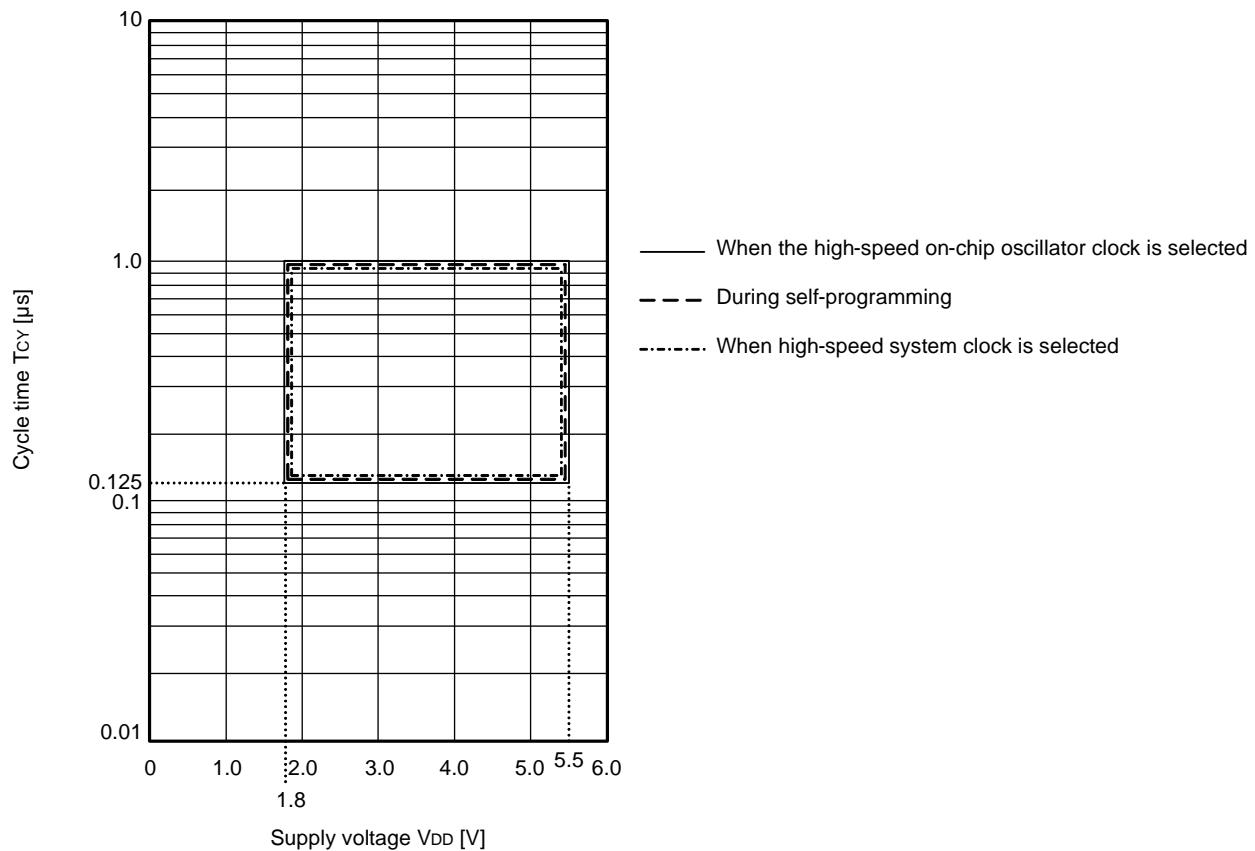
Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Timer RD input high-level width, low-level width	tTDIH, tTDIL	TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1, TRDIODC0, TRDIODC1, TRDIOD0, TRDIOD1		3/fCLK			ns
Timer RD forced cutoff signal input low-level width	tTDSIL	P130/INTP0	2MHz < fCLK ≤ 32 MHz	1			μs
			fCLK ≤ 2 MHz	1/fCLK + 1			
Timer RG input high-level width, low-level width	tTGIH, tTGIL	TRGIOA, TRGIOB		2.5/fCLK			ns
TO00 to TO03, TO10 to TO13, TRJIO0, TRJOO, TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1, TRDIODC0, TRDIODC1, TRDIOD0, TRDIOD1, TRGIOA, TRGIOB output frequency	fro	HS (high-speed main) mode	4.0 V ≤ EVDD0 ≤ 5.5 V			16	MHz
			2.7 V ≤ EVDD0 < 4.0 V			8	MHz
			1.8 V ≤ EVDD0 < 2.7 V			4	MHz
			1.6 V ≤ EVDD0 < 1.8 V			2	MHz
		LS (low-speed main) mode	1.8 V ≤ EVDD0 ≤ 5.5 V			4	MHz
			1.6 V ≤ EVDD0 < 1.8 V			2	MHz
		LV (low-voltage main) mode	1.6 V ≤ EVDD0 ≤ 5.5 V			2	MHz
		HS (high-speed main) mode	4.0 V ≤ EVDD0 ≤ 5.5 V			16	MHz
			2.7 V ≤ EVDD0 < 4.0 V			8	MHz
			1.8 V ≤ EVDD0 < 2.7 V			4	MHz
			1.6 V ≤ EVDD0 < 1.8 V			2	MHz
		LS (low-speed main) mode	1.8 V ≤ EVDD0 ≤ 5.5 V			4	MHz
			1.6 V ≤ EVDD0 < 1.8 V			2	MHz
		LV (low-voltage main) mode	1.8 V ≤ EVDD0 ≤ 5.5 V			4	MHz
			1.6 V ≤ EVDD0 < 1.8 V			2	MHz
Interrupt input high-level width, low-level width	tINTH, tINTL	INTP0	1.6 V ≤ VDD ≤ 5.5 V	1			μs
		INTP1 to INTP11	1.6 V ≤ EVDD0 ≤ 5.5 V	1			μs
Key interrupt input low-level width	tKR	KR0 to KR7	1.8 V ≤ EVDD0 ≤ 5.5 V	250			ns
			1.6 V ≤ EVDD0 < 1.8 V	1			μs
RESET low-level width	trSL			10			μs

## Minimum Instruction Execution Time during Main System Clock Operation

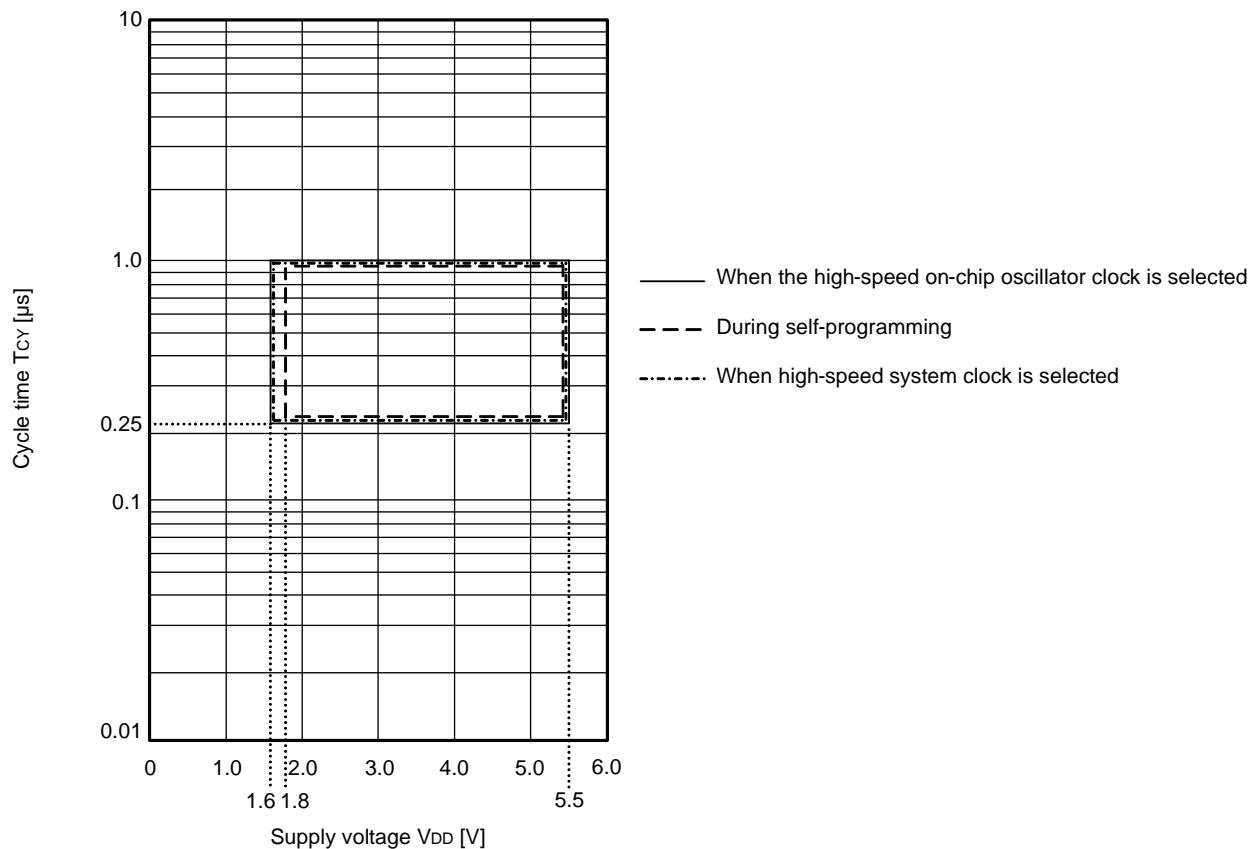
TCY vs VDD (HS (high-speed main) mode)



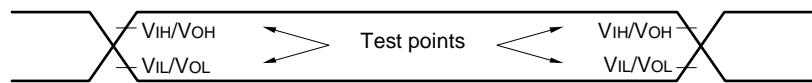
TCY vs VDD (LS (low-speed main) mode)



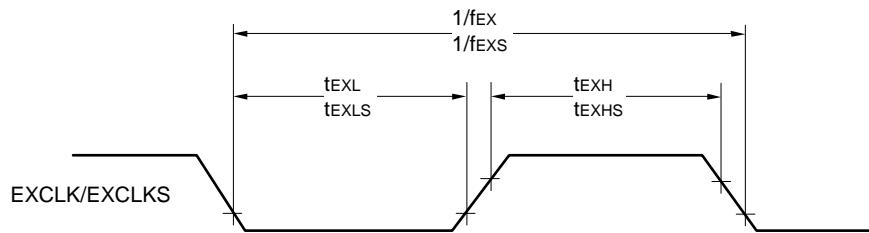
TCY vs VDD (LV (low-voltage main) mode)



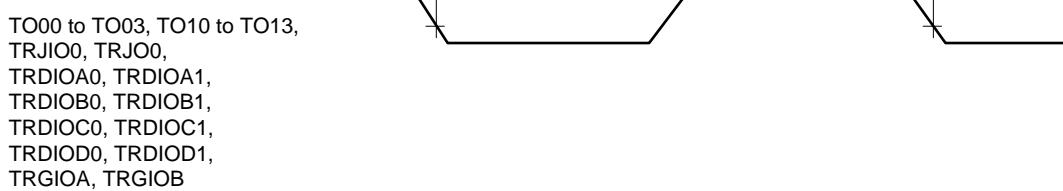
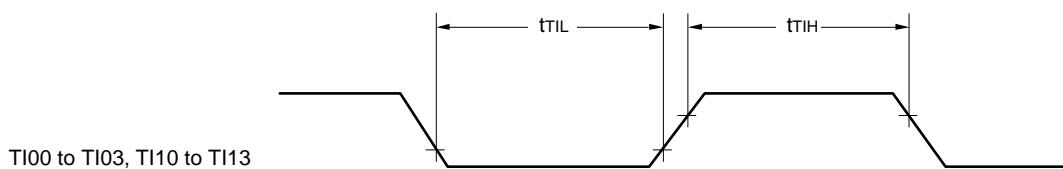
## AC Timing Test Points

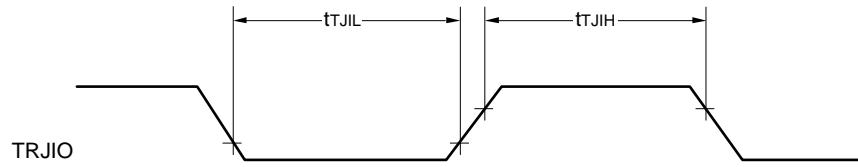


## External System Clock Timing

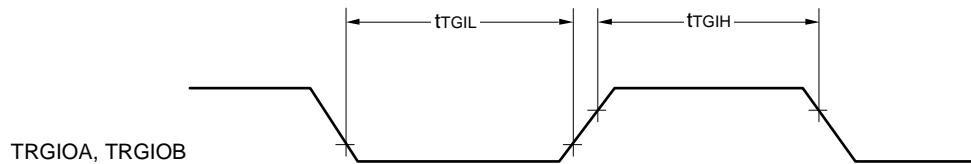
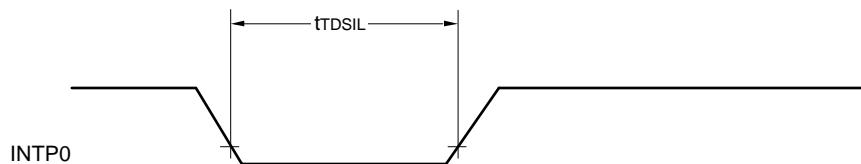


## TI/TO Timing

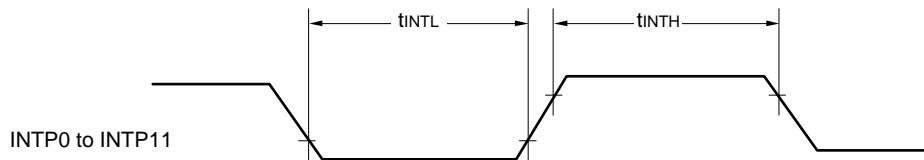




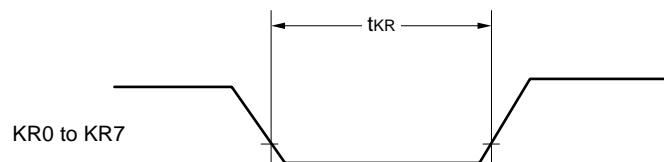
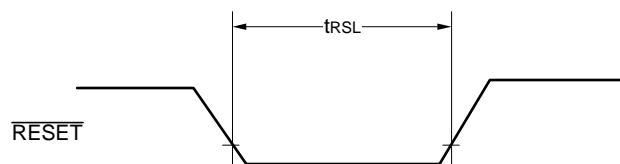
TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1,  
TRDIOC0, TRDIOC1, TRDIOD0, TRDIOD1



## Interrupt Request Input Timing

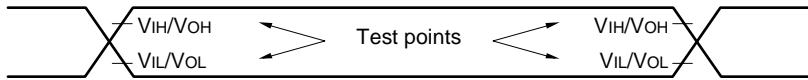


## Key Interrupt Input Timing

RESET Input Timing

## 34.5 Peripheral Functions Characteristics

AC Timing Test Points



### 34.5.1 Serial array unit

#### (1) During communication at same potential (UART mode)

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

Parameter	Symbol	Conditions	HS (high-speed main) Mode		LS (low-speed main) Mode		LV (low-voltage main) Mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Transfer rate Note 1		2.4 V ≤ EV <sub>D0</sub> ≤ 5.5 V		fMCK/6 Note 2		fMCK/6		fMCK/6	bps
		Theoretical value of the maximum transfer rate fMCK = fCLK Note 3		5.3		1.3		0.6	Mbps
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		fMCK/6 Note 2		fMCK/6		fMCK/6	bps
		Theoretical value of the maximum transfer rate fMCK = fCLK Note 3		5.3		1.3		0.6	Mbps
		1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		fMCK/6 Note 2		fMCK/6 Note 2		fMCK/6	bps
		Theoretical value of the maximum transfer rate fMCK = fCLK Note 3		5.3		1.3		0.6	Mbps
		1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—		fMCK/6 Note 2		fMCK/6	bps	
		Theoretical value of the maximum transfer rate fMCK = fCLK Note 3	—			1.3		0.6	Mbps

**Note 1.** Transfer rate in the SNOOZE mode is 4800 bps only.

However, the SNOOZE mode cannot be used when FRQSEL4 = 1.

**Note 2.** The following conditions are required for low voltage interface when EV<sub>D0</sub> < V<sub>D</sub>.

2.4 V ≤ EV<sub>D0</sub> < 2.7 V: MAX. 2.6 Mbps

1.8 V ≤ EV<sub>D0</sub> < 2.4 V: MAX. 1.3 Mbps

1.6 V ≤ EV<sub>D0</sub> < 1.8 V: MAX. 0.6 Mbps

**Note 3.** The maximum operating frequencies of the CPU/peripheral hardware clock (fCLK) are:

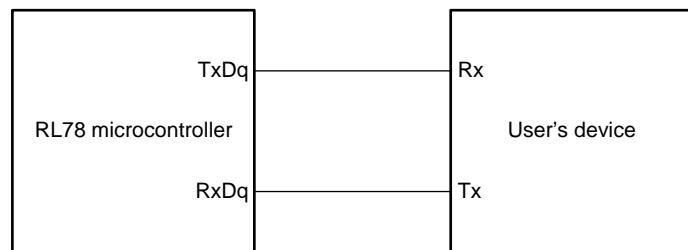
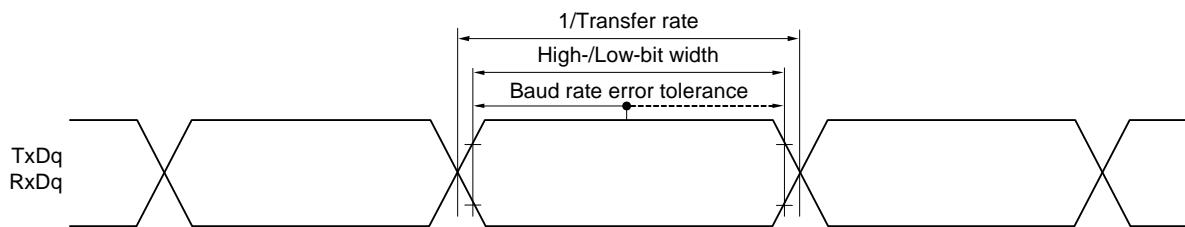
HS (high-speed main) mode: 32 MHz (2.7 V ≤ V<sub>D</sub> ≤ 5.5 V)

16 MHz (2.4 V ≤ V<sub>D</sub> ≤ 5.5 V)

LS (low-speed main) mode: 8 MHz (1.8 V ≤ V<sub>D</sub> ≤ 5.5 V)

LV (low-voltage main) mode: 4 MHz (1.6 V ≤ V<sub>D</sub> ≤ 5.5 V)

**Caution** Select the normal input buffer for the RxDq pin and the normal output mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**UART mode connection diagram (during communication at same potential)****UART mode bit width (during communication at same potential) (reference)**

**Remark 1.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)

**Remark 2.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

(2) During communication at same potential (CSI mode) (master mode, SCKp... internal clock output, corresponding CSI00 only)

(TA = -40 to +85°C, 2.7 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions		HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time	tkCY1	tkCY1 ≥ 2/fCLK	4.0 V ≤ EVDD0 ≤ 5.5 V	62.5		250		500		ns
			2.7 V ≤ EVDD0 ≤ 5.5 V	83.3		250		500		ns
SCKp high-/low-level width	tkH1, tkL1	4.0 V ≤ EVDD0 ≤ 5.5 V 2.7 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 7		tkCY1/2 - 50		tkCY1/2 - 50		ns	
			tkCY1/2 - 10		tkCY1/2 - 50		tkCY1/2 - 50		ns	
Slp setup time (to SCKp↑) Note 1	tsIK1	4.0 V ≤ EVDD0 ≤ 5.5 V 2.7 V ≤ EVDD0 ≤ 5.5 V	23		110		110		ns	
			33		110		110		ns	
Slp hold time (from SCKp↑) Note 2	tksI1	2.7 V ≤ EVDD0 ≤ 5.5 V	10		10		10		ns	
Delay time from SCKp↓ to SOp output Note 3	tkso1	C = 20 pF Note 4		10		10		10		ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes “to SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 2.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes “from SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes “from SCKp↑” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 4.** C is the load capacitance of the SCKp and SOp output lines.

**Caution** Select the normal input buffer for the Slp pin and the normal output mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark 1.** This value is valid only when CSI00's peripheral I/O redirect function is not used.

**Remark 2.** p: CSI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0),  
g: PIM and POM numbers (g = 1)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00))

## (3) During communication at same potential (CSI mode) (master mode, SCKp... internal clock output)

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time	tkCY1	tkCY1 ≥ 4/fCLK 2.7 V ≤ EVDD0 ≤ 5.5 V 2.4 V ≤ EVDD0 ≤ 5.5 V 1.8 V ≤ EVDD0 ≤ 5.5 V 1.7 V ≤ EVDD0 ≤ 5.5 V 1.6 V ≤ EVDD0 ≤ 5.5 V	125		500		1000		ns
			250		500		1000		ns
			500		500		1000		ns
			1000		1000		1000		ns
			—		1000		1000		ns
SCKp high-/low-level width	tkH1, tkL1	4.0 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 12		tkCY1/2 - 50		tkCY1/2 - 50		ns
		2.7 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 18		tkCY1/2 - 50		tkCY1/2 - 50		ns
		2.4 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 38		tkCY1/2 - 50		tkCY1/2 - 50		ns
		1.8 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 50		tkCY1/2 - 50		tkCY1/2 - 50		ns
		1.7 V ≤ EVDD0 ≤ 5.5 V	tkCY1/2 - 100		tkCY1/2 - 100		tkCY1/2 - 100		ns
		1.6 V ≤ EVDD0 ≤ 5.5 V	—		tkCY1/2 - 100		tkCY1/2 - 100		ns
Slp setup time (to SCKp↑) Note 1	tsIK1	4.0 V ≤ EVDD0 ≤ 5.5 V	44		110		110		ns
		2.7 V ≤ EVDD0 ≤ 5.5 V	44		110		110		ns
		2.4 V ≤ EVDD0 ≤ 5.5 V	75		110		110		ns
		1.8 V ≤ EVDD0 ≤ 5.5 V	110		110		110		ns
		1.7 V ≤ EVDD0 ≤ 5.5 V	220		220		220		ns
		1.6 V ≤ EVDD0 ≤ 5.5 V	—		220		220		ns
Slp hold time (from SCKp↑) Note 2	tksI1	1.7 V ≤ EVDD0 ≤ 5.5 V	19		19		19		ns
		1.6 V ≤ EVDD0 ≤ 5.5 V	—		19		19		ns
Delay time from SCKp↓ to SOp output Note 3	tksO1	1.7 V ≤ EVDD0 ≤ 5.5 V C = 30 pF Note 4		25		25		25	ns
		1.6 V ≤ EVDD0 ≤ 5.5 V C = 30 pF Note 4		—		25		25	ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes "to SCKp↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 2.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes "from SCKp↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes "from SCKp↑" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 4.** C is the load capacitance of the SCKp and SOp output lines.

**Caution** Select the normal input buffer for the Slp pin and the normal output mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), g: PIM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

**(4) During communication at same potential (CSI mode) (slave mode, SCKp... external clock input)**

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(1/2)

Parameter	Symbol	Conditions		HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time Note 5	tkCY2	4.0 V ≤ EVDD0 ≤ 5.5 V	20 MHz < fmck	8/fmck	—	—	—	—	—	ns
			fmck ≤ 20 MHz	6/fmck	—	6/fmck	—	6/fmck	—	ns
		2.7 V ≤ EVDD0 ≤ 5.5 V	16 MHz < fmck	8/fmck	—	—	—	—	—	ns
			fmck ≤ 16 MHz	6/fmck	—	6/fmck	—	6/fmck	—	ns
		2.4 V ≤ EVDD0 ≤ 5.5 V		6/fmck and 500	—	6/fmck and 500	—	6/fmck and 500	—	ns
		1.8 V ≤ EVDD0 ≤ 5.5 V		6/fmck and 750	—	6/fmck and 750	—	6/fmck and 750	—	ns
		1.7 V ≤ EVDD0 ≤ 5.5 V		6/fmck and 1500	—	6/fmck and 1500	—	6/fmck and 1500	—	ns
		1.6 V ≤ EVDD0 ≤ 5.5 V		—	—	6/fmck and 1500	—	6/fmck and 1500	—	ns
SCKp high-/low-level width	tKH2, tKL2	4.0 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 7	—	tkCY2/2 - 7	—	tkCY2/2 - 7	—	ns
		2.7 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 8	—	tkCY2/2 - 8	—	tkCY2/2 - 8	—	ns
		1.8 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 18	—	tkCY2/2 - 18	—	tkCY2/2 - 18	—	ns
		1.7 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 66	—	tkCY2/2 - 66	—	tkCY2/2 - 66	—	ns
		1.6 V ≤ EVDD0 ≤ 5.5 V		—	—	tkCY2/2 - 66	—	tkCY2/2 - 66	—	ns
Slp setup time (to SCKp↑) Note 1	tsIK2	2.7 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 20	—	1/fmck + 30	—	1/fmck + 30	—	ns
		1.8 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 30	—	1/fmck + 30	—	1/fmck + 30	—	ns
		1.7 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 40	—	1/fmck + 40	—	1/fmck + 40	—	ns
		1.6 V ≤ EVDD0 ≤ 5.5 V		—	—	1/fmck + 40	—	1/fmck + 40	—	ns
Slp hold time (from SCKp↑) Note 2	tksI2	1.8 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 31	—	1/fmck + 31	—	1/fmck + 31	—	ns
		1.7 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 250	—	1/fmck + 250	—	1/fmck + 250	—	ns
		1.6 V ≤ EVDD0 ≤ 5.5 V		—	—	1/fmck + 250	—	1/fmck + 250	—	ns
Delay time from SCKp↓ to SOp output Note 3	tksO2	C = 30 pF Note 4	2.7 V ≤ EVDD0 ≤ 5.5 V	—	2/fmck + 44	—	2/fmck + 110	—	2/fmck + 110	ns
			2.4 V ≤ EVDD0 ≤ 5.5 V	—	2/fmck + 75	—	2/fmck + 110	—	2/fmck + 110	ns
			1.8 V ≤ EVDD0 ≤ 5.5 V	—	2/fmck + 100	—	2/fmck + 110	—	2/fmck + 110	ns
			1.7 V ≤ EVDD0 ≤ 5.5 V	—	2/fmck + 220	—	2/fmck + 220	—	2/fmck + 220	ns
			1.6 V ≤ EVDD0 ≤ 5.5 V	—	—	—	2/fmck + 220	—	2/fmck + 220	ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes “to SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 2.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes “from SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes “from SCKp↑” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 4.** C is the load capacitance of the SOp output lines.

**Note 5.** The maximum transfer rate when using the SNOOZE mode is 1 Mbps.

**Caution** Select the normal input buffer for the Slp pin and SCKp pin and the normal output mode for the SOp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), m: Unit number (m = 0, 1),  
n: Channel number (n = 0 to 3), g: PIM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** fmck: Serial array unit operation clock frequency  
(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number,  
n: Channel number (mn = 00 to 03, 10 to 13))

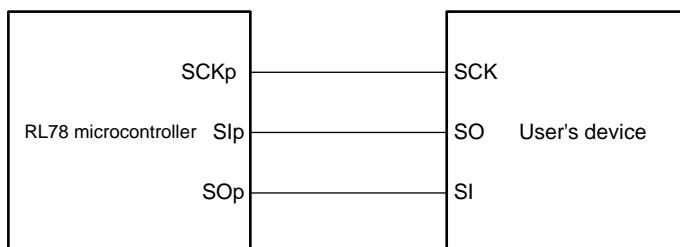
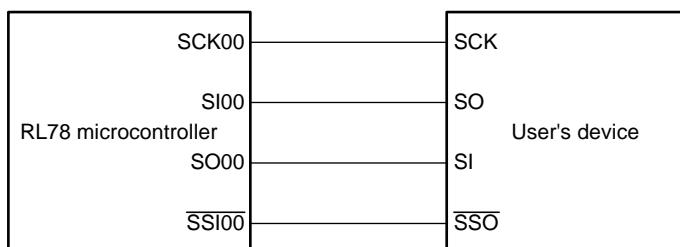
**(4) During communication at same potential (CSI mode) (slave mode, SCKp... external clock input)**(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SSI00 setup time	t <sub>SSIK</sub>	DAP <sub>mn</sub> = 0	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	120		120		120	ns
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	200		200		200	ns
			1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	400		400		400	ns
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		400		400	ns
		DAP <sub>mn</sub> = 1	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 120		1/fMCK + 120		1/fMCK + 120	ns
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 200		1/fMCK + 200		1/fMCK + 200	ns
			1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 400		1/fMCK + 400		1/fMCK + 400	ns
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		1/fMCK + 400		1/fMCK + 400	ns
SSI00 hold time	t <sub>kssi</sub>	DAP <sub>mn</sub> = 0	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 120		1/fMCK + 120		1/fMCK + 120	ns
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 200		1/fMCK + 200		1/fMCK + 200	ns
			1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 400		1/fMCK + 400		1/fMCK + 400	ns
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		1/fMCK + 400		1/fMCK + 400	ns
		DAP <sub>mn</sub> = 1	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	120		120		120	ns
			1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	200		200		200	ns
			1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	400		400		400	ns
			1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		400		400	ns

**Caution** Select the normal input buffer for the Slp pin and SCKp pin and the normal output mode for the SO<sub>p</sub> pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark** p: CSI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0), g: PIM number (g = 3, 5)

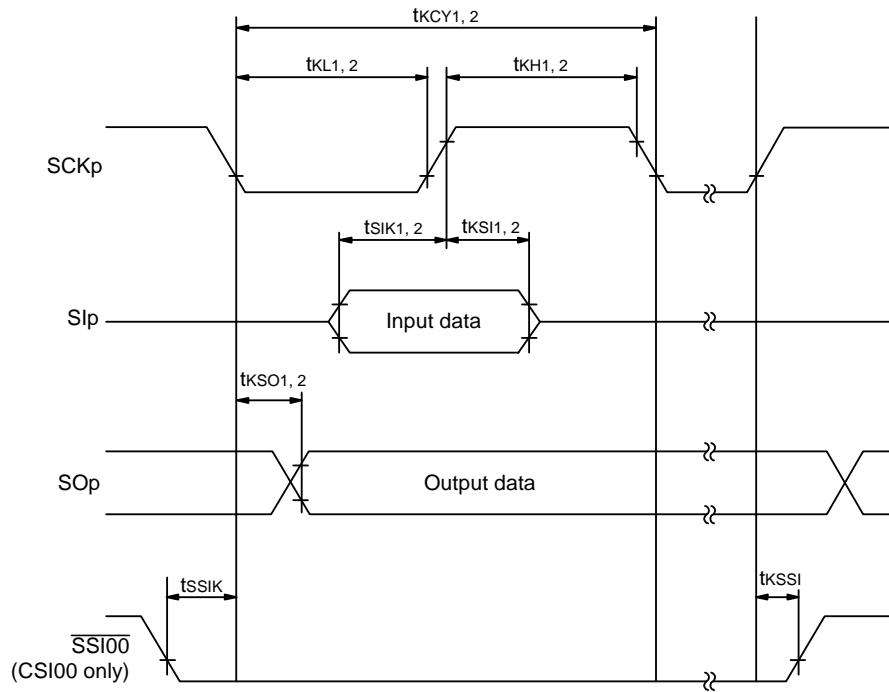
**CSI mode connection diagram (during communication at same potential)****CSI mode connection diagram (during communication at same potential)  
(Slave Transmission of slave select input function (CSI00))**

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)

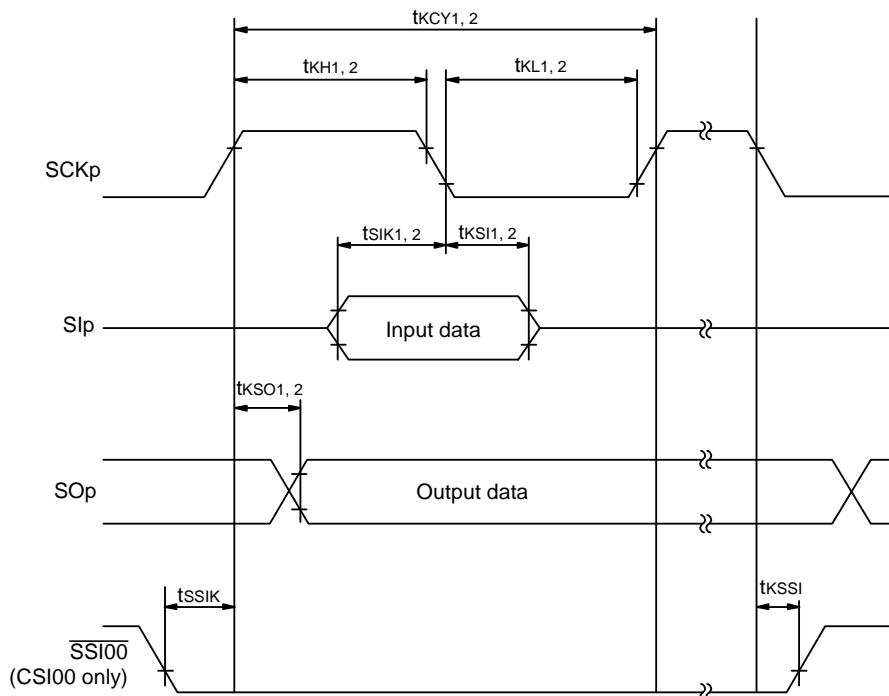
**Remark 2.** m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13)

**CSI mode serial transfer timing (during communication at same potential)**

(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)

**CSI mode serial transfer timing (during communication at same potential)**

(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)**Remark 2.** m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13)

(5) During communication at same potential (simplified I<sup>2</sup>C mode)(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCL <sub>r</sub> clock frequency	f <sub>SCL</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		1000 Note 1		400 Note 1		400 Note 1	kHz
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ		400 Note 1		400 Note 1		400 Note 1	kHz
		1.8 V ≤ EV <sub>DD0</sub> < 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ		300 Note 1		300 Note 1		300 Note 1	kHz
		1.7 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ		250 Note 1		250 Note 1		250 Note 1	kHz
		1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ		—		250 Note 1		250 Note 1	kHz
Hold time when SCL <sub>r</sub> = "L"	t <sub>LOW</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	475		1150		1150		ns
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	1150		1150		1150		ns
		1.8 V ≤ EV <sub>DD0</sub> < 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1550		1550		1550		ns
		1.7 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1850		1850		1850		ns
		1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	—		1850		1850		ns
Hold time when SCL <sub>r</sub> = "H"	t <sub>HIGH</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	475		1150		1150		ns
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	1150		1150		1150		ns
		1.8 V ≤ EV <sub>DD0</sub> < 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1550		1550		1550		ns
		1.7 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1850		1850		1850		ns
		1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	—		1850		1850		ns

(Notes and Caution are listed on the next page, and Remarks are listed on the page after the next page.)

(5) During communication at same potential (simplified I<sup>2</sup>C mode)(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

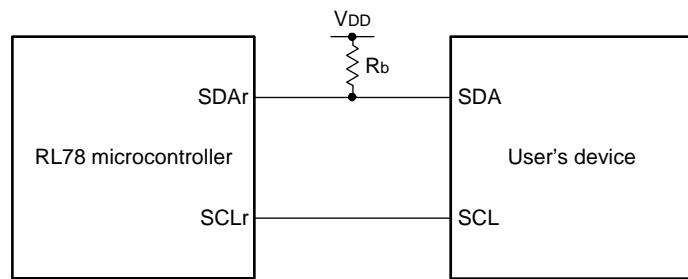
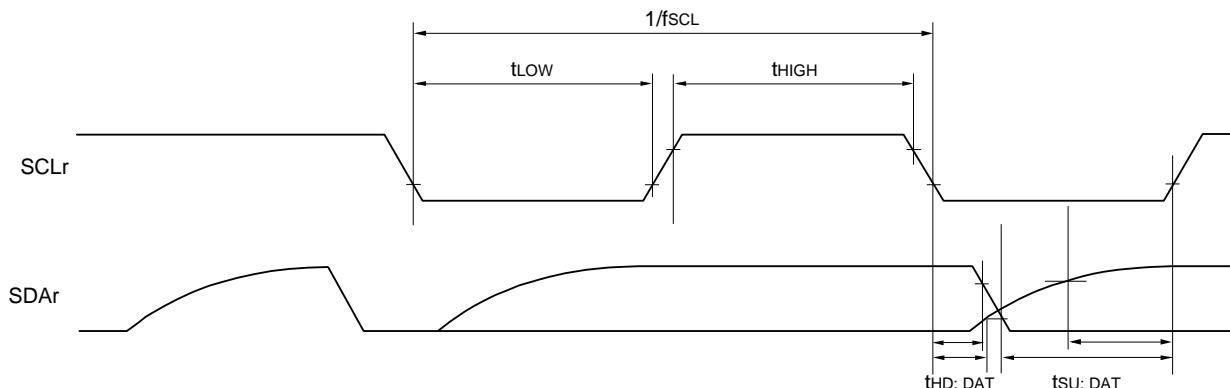
(2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Data setup time (reception)	tsU: DAT	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/fMCK + 85 Note 2		1/fMCK + 145 Note 2		1/fMCK + 145 Note 2		ns
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	1/fMCK + 145 Note 2		1/fMCK + 145 Note 2		1/fMCK + 145 Note 2		ns
		1.8 V ≤ EV <sub>DD0</sub> < 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1/fMCK + 230 Note 2		1/fMCK + 230 Note 2		1/fMCK + 230 Note 2		ns
		1.7 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	1/fMCK + 290 Note 2		1/fMCK + 290 Note 2		1/fMCK + 290 Note 2		ns
		1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	—		1/fMCK + 290 Note 2		1/fMCK + 290 Note 2		ns
Data hold time (transmission)	tHD: DAT	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	305	0	305	0	305	ns
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	0	355	0	355	0	355	ns
		1.8 V ≤ EV <sub>DD0</sub> < 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	0	405	0	405	0	405	ns
		1.7 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	0	405	0	405	0	405	ns
		1.6 V ≤ EV <sub>DD0</sub> < 1.8 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5 kΩ	—		0	405	0	405	ns

**Note 1.** The value must also be equal to or less than fMCK/4.**Note 2.** Set the fMCK value to keep the hold time of SCL<sub>r</sub> = "L" and SCL<sub>r</sub> = "H".

**Caution** Select the normal input buffer and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SDAr pin and the normal output mode for the SCL<sub>r</sub> pin by using port input mode register g (PIMg) and port output mode register h (POMh).

(Remarks are listed on the next page.)

**Simplified I<sup>2</sup>C mode connection diagram (during communication at same potential)****Simplified I<sup>2</sup>C mode serial transfer timing (during communication at same potential)**

**Remark 1.**  $R_b[\Omega]$ : Communication line (SDAr) pull-up resistance,  $C_b[F]$ : Communication line (SDAr, SCLr) load capacitance

**Remark 2.** r: IIC number ( $r = 00, 01, 10, 11, 20, 21, 30, 31$ ), g: PIM number ( $g = 0, 1, 3$  to  $5, 14$ ),

h: POM number ( $h = 0, 1, 3$  to  $5, 7, 14$ )

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number ( $m = 0, 1$ ),

n: Channel number ( $n = 0$  to  $3$ ), mn = 00 to 03, 10 to 13)

## (6) Communication at different potential (1.8 V, 2.5 V, 3 V) (UART mode)

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Transfer rate		reception	4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V		fMCK/6 Note 1		fMCK/6 Note 1		fMCK/6 Note 1 bps
			Theoretical value of the maximum transfer rate fMCK = fCLK Note 4		5.3		1.3		0.6 Mbps
			2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V		fMCK/6 Note 1		fMCK/6 Note 1		fMCK/6 Note 1 bps
			Theoretical value of the maximum transfer rate fMCK = fCLK Note 4		5.3		1.3		0.6 Mbps
			1.8 V ≤ EV <sub>D0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V		fMCK/6 Notes 1, 2, 3		fMCK/6 Notes 1, 2		fMCK/6 Notes 1, 2 bps
			Theoretical value of the maximum transfer rate fMCK = fCLK Note 4		5.3		1.3		0.6 Mbps

**Note 1.** Transfer rate in the SNOOZE mode is 4800 bps only.

However, the SNOOZE mode cannot be used when FRQSEL4 = 1.

**Note 2.** Use it with EV<sub>D0</sub> ≥ V<sub>b</sub>.**Note 3.** The following conditions are required for low voltage interface when EV<sub>D0</sub> < V<sub>DD</sub>.2.4 V ≤ EV<sub>D0</sub> < 2.7 V: MAX. 2.6 Mbps1.8 V ≤ EV<sub>D0</sub> < 2.4 V: MAX. 1.3 Mbps**Note 4.** The maximum operating frequencies of the CPU/peripheral hardware clock (fCLK) are:HS (high-speed main) mode: 32 MHz (2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V)16 MHz (2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V)LS (low-speed main) mode: 8 MHz (1.8 V ≤ V<sub>DD</sub> ≤ 5.5 V)LV (low-voltage main) mode: 4 MHz (1.6 V ≤ V<sub>DD</sub> ≤ 5.5 V)**Caution** Select the TTL input buffer for the RxDq pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>D0</sub> tolerance (for the 64- to 100-pin products)) mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.**Remark 1.** V<sub>b</sub> [V]: Communication line voltage**Remark 2.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)**Remark 3.** fMCK: Serial array unit operation clock frequency(Operation clock to be set by the CKS<sub>Mn</sub> bit of serial mode register mn (SMR<sub>Mn</sub>). m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13)**Remark 4.** UART2 cannot communicate at different potential when bit 1 (PIOR01) of peripheral I/O redirection register 0 (PIOR0) is 1.

## (6) Communication at different potential (1.8 V, 2.5 V, 3 V) (UART mode)

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Transfer rate	transmission	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V		Note 1		Note 1		Note 1	bps
		Theoretical value of the maximum transfer rate Cb = 50 pF, Rb = 1.4 kΩ, Vb = 2.7 V		2.8 Note 2		2.8 Note 2		2.8 Note 2	Mbps
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V		Note 3		Note 3		Note 3	bps
		Theoretical value of the maximum transfer rate Cb = 50 pF, Rb = 2.7 kΩ, Vb = 2.3 V		1.2 Note 4		1.2 Note 4		1.2 Note 4	Mbps
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V		Notes 5, 6		Notes 5, 6		Notes 5, 6	bps
		Theoretical value of the maximum transfer rate Cb = 50 pF, Rb = 5.5 kΩ, Vb = 1.6 V		0.43 Note 7		0.43 Note 7		0.43 Note 7	Mbps

**Note 1.** The smaller maximum transfer rate derived by using fmck/6 or the following expression is the valid maximum transfer rate.  
Expression for calculating the transfer rate when 4.0 V ≤ EVDD0 ≤ 5.5 V and 2.7 V ≤ Vb ≤ 4.0 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 2.** This value as an example is calculated when the conditions described in the "Conditions" column are met.

Refer to **Note 1** above to calculate the maximum transfer rate under conditions of the customer.

**Note 3.** The smaller maximum transfer rate derived by using fmck/6 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 2.7 V ≤ EVDD0 < 4.0 V and 2.3 V ≤ Vb ≤ 2.7 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 4.** This value as an example is calculated when the conditions described in the "Conditions" column are met.

Refer to **Note 3** above to calculate the maximum transfer rate under conditions of the customer.

**Note 5.** Use it with EVDD0 ≥ Vb.

**Note 6.** The smaller maximum transfer rate derived by using fmck/6 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 1.8 V ≤ EVDD0 < 3.3 V and 1.6 V ≤ Vb ≤ 2.0 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\} \times 3} \text{ [bps]}$$

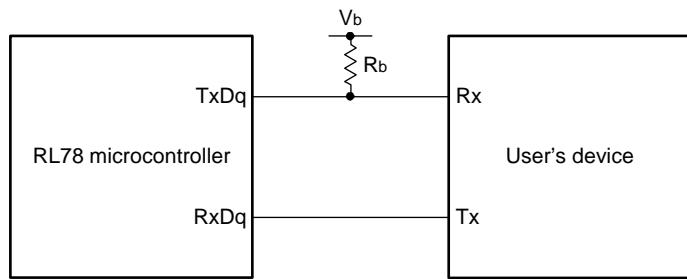
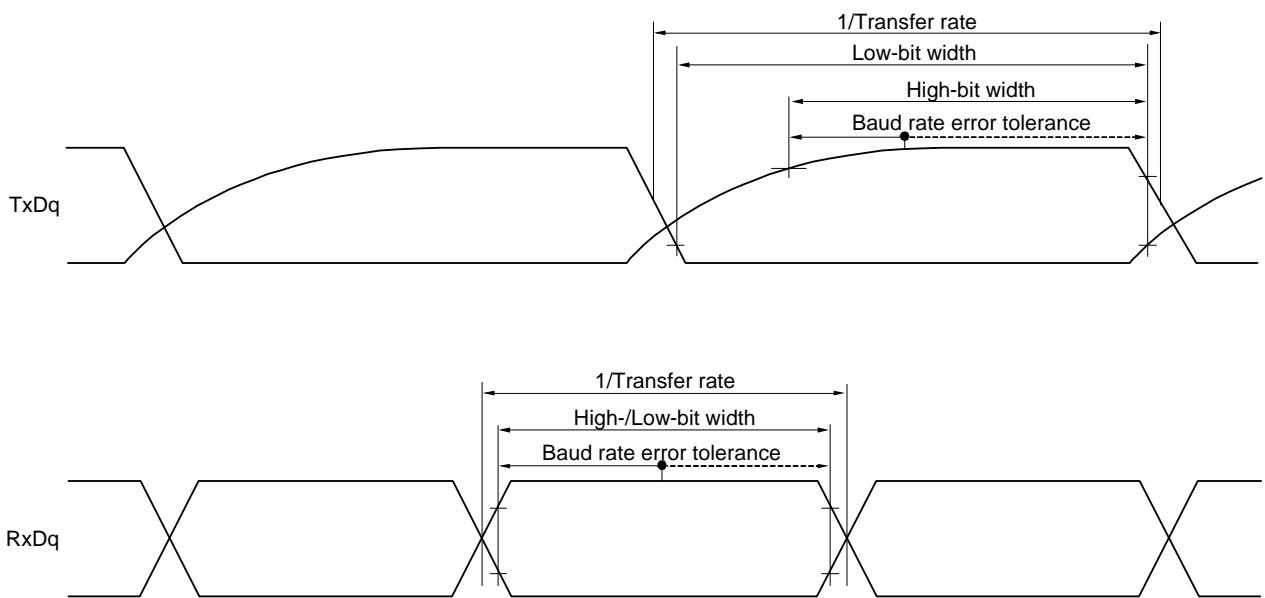
$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 7.** This value as an example is calculated when the conditions described in the “Conditions” column are met.  
Refer to **Note 6** above to calculate the maximum transfer rate under conditions of the customer.

**Caution** Select the TTL input buffer for the RxDq pin and the N-ch open drain output (VDD tolerance (for the 30- to 52-pin products)/EVDD tolerance (for the 64- to 100-pin products)) mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg). For VIH and Vil, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**UART mode connection diagram (during communication at different potential)****UART mode bit width (during communication at different potential) (reference)**

**Remark 1.**  $R_b[\Omega]$ : Communication line (TxDq) pull-up resistance,

$C_b[F]$ : Communication line (TxDq) load capacitance,  $V_b[V]$ : Communication line voltage

**Remark 2.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn)).

m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

**Remark 4.** UART2 cannot communicate at different potential when bit 1 (PIOR01) of peripheral I/O redirection register 0 (PIOR0) is 1.

**(7) Communication at different potential (2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output, corresponding CSI00 only)**

(TA = -40 to +85°C, 2.7 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V) (1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time	tkCY1	tkCY1 ≥ 2/fCLK 4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	200		1150		1150		ns
			2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	300		1150		1150	ns
SCKp high-level width	tkH1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	tkCY1/2 - 50		tkCY1/2 - 50		tkCY1/2 - 50		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	tkCY1/2 - 120		tkCY1/2 - 120		tkCY1/2 - 120		ns
SCKp low-level width	tkL1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	tkCY1/2 - 7		tkCY1/2 - 50		tkCY1/2 - 50		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	tkCY1/2 - 10		tkCY1/2 - 50		tkCY1/2 - 50		ns
Slp setup time (to SCKp↑) Note 1	tSIK1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	58		479		479		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	121		479		479		ns
Slp hold time (from SCKp↑) Note 1	tKSI1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	10		10		10		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	10		10		10		ns
Delay time from SCKp↓ to SO <sub>p</sub> output Note 1	tKS01	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ		60		60		60	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ		130		130		130	ns

(Notes, Caution, and Remarks are listed on the next page.)

**(7) Communication at different potential (2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output, corresponding CSI00 only)**

(TA = -40 to +85°C, 2.7 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V) (2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Slp setup time (to SCKp↓) Note 2	t <sub>SIK1</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	23		110		110		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	33		110		110		ns
Slp hold time (from SCKp↓) Note 2	t <sub>SKI1</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ	10		10		10		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ	10		10		10		ns
Delay time from SCKp↑ to SOp output Note 2	t <sub>KS01</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 1.4 kΩ		10		10		10	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 20 pF, R <sub>b</sub> = 2.7 kΩ		10		10		10	ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.

**Note 2.** When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Caution** Select the TTL input buffer for the Slp pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

**Remark 1.** R<sub>b</sub>[Ω]: Communication line (SCKp, SOp) pull-up resistance, C<sub>b</sub>[F]: Communication line (SCKp, SOp) load capacitance, V<sub>b</sub>[V]: Communication line voltage

**Remark 2.** p: CSI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0), g: PIM and POM number (g = 3, 5)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00))

**Remark 4.** This value is valid only when CSI00's peripheral I/O redirect function is not used.

**(8) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

(TA = -40 to +85°C, 1.8 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V) (1/3)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time	tkCY1	$4.0 \text{ V} \leq \text{EV}_{\text{DD}0} \leq 5.5 \text{ V}$ , $2.7 \text{ V} \leq V_b \leq 4.0 \text{ V}$ , $C_b = 30 \text{ pF}$ , $R_b = 1.4 \text{ k}\Omega$	300		1150		1150		ns
			500		1150		1150		ns
			1150		1150		1150		ns
SCKp high-level width	tKH1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	tkCY1/2 - 75		tkCY1/2 - 75		tkCY1/2 - 75		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	tkCY1/2 - 170		tkCY1/2 - 170		tkCY1/2 - 170		ns
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	tkCY1/2 - 458		tkCY1/2 - 458		tkCY1/2 - 458		ns
SCKp low-level width	tKL1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	tkCY1/2 - 12		tkCY1/2 - 50		tkCY1/2 - 50		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	tkCY1/2 - 18		tkCY1/2 - 50		tkCY1/2 - 50		ns
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	tkCY1/2 - 50		tkCY1/2 - 50		tkCY1/2 - 50		ns

**Note** Use it with EV<sub>DD0</sub> ≥ V<sub>b</sub>.

**Caution** Select the TTL input buffer for the S<sub>I</sub>p pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the S<sub>O</sub>p pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed two pages after the next page.)

**(8) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

(TA = -40 to +85°C, 1.8 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (2/3)

Parameter	Symbol	Conditions	HS (high-speed main mode)		LS (low-speed main mode)		LV (low-voltage main mode)		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Slp setup time (to SCKp↑) Note 1	tsIK1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	81		479		479		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	177		479		479		ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ	479		479		479		ns
Slp hold time (from SCKp↑) Note 1	tksI1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	19		19		19		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	19		19		19		ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ	19		19		19		ns
Delay time from SCKp↓ to SOp output Note 1	tksO1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ		100		100		100	ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ		195		195		195	ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ		483		483		483	ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.

**Note 2.** Use it with EVDD0 ≥ Vb.

**Caution** Select the TTL input buffer for the Slp pin and the N-ch open drain output (VDD tolerance (for the 30- to 52-pin products)/EVDD tolerance (for the 64- to 100-pin products)) mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For VIH and Vil, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the page after the next page.)

**(8) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

(TA = -40 to +85°C, 1.8 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V) (3/3)

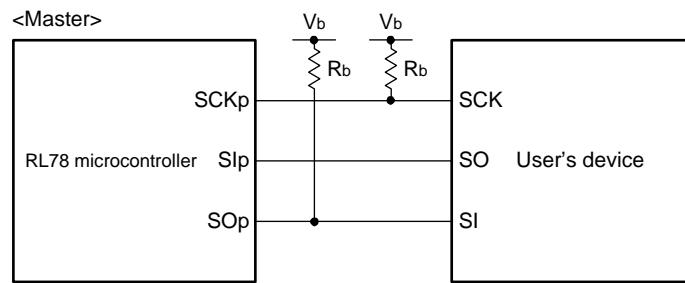
Parameter	Symbol	Conditions	HS (high-speed main mode)		LS (low-speed main mode)		LV (low-voltage main mode)		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Slp setup time (to SCKp↓) Note 1	tsIK1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	44		110		110		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	44		110		110		ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ	110		110		110		ns
Slp hold time (from SCKp↓) Note 1	tksI1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ	19		19		19		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ	19		19		19		ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ	19		19		19		ns
Delay time from SCKp↑ to SOp output Note 1	tksO1	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ		25		25		25	ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ		25		25		25	ns
		1.8 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V Note 2, Cb = 30 pF, Rb = 5.5 kΩ		25		25		25	ns

**Note 1.** When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 2.** Use it with EVDD0 ≥ Vb.

**Caution** Select the TTL input buffer for the Slp pin and the N-ch open drain output (VDD tolerance (for the 30- to 52-pin products)/EVDD tolerance (for the 64- to 100-pin products)) mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For VIH and Vil, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**CSI mode connection diagram (during communication at different potential)**

**Remark 1.**  $R_b[\Omega]$ : Communication line (SCKp, SOp) pull-up resistance,  $C_b[F]$ : Communication line (SCKp, SOp) load capacitance,  $V_b[V]$ : Communication line voltage

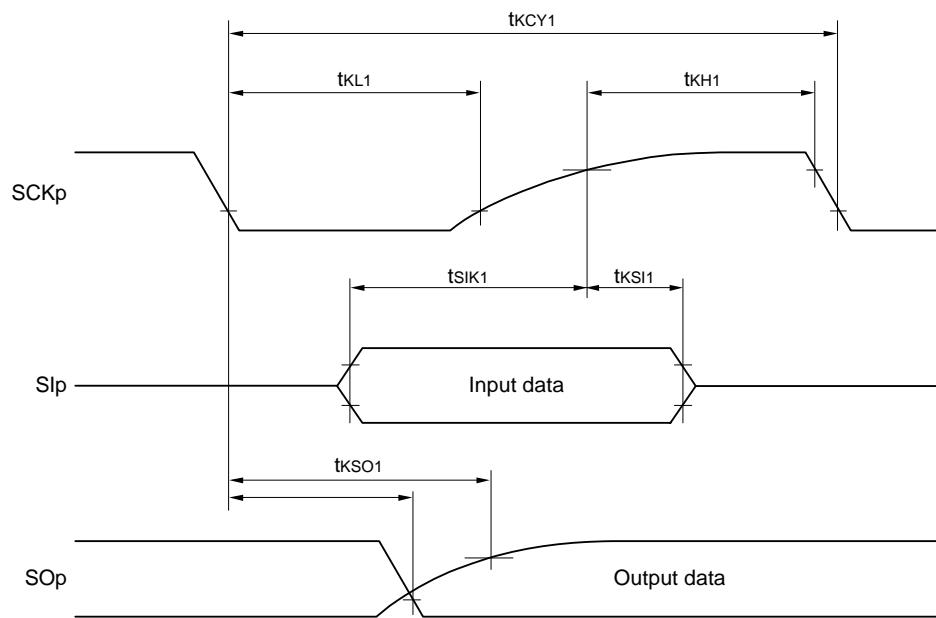
**Remark 2.** p: CSI number ( $p = 00, 01, 10, 20, 30, 31$ ), m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), g: PIM and POM number ( $g = 0, 1, 3$  to  $5, 14$ )

**Remark 3.** fmck: Serial array unit operation clock frequency

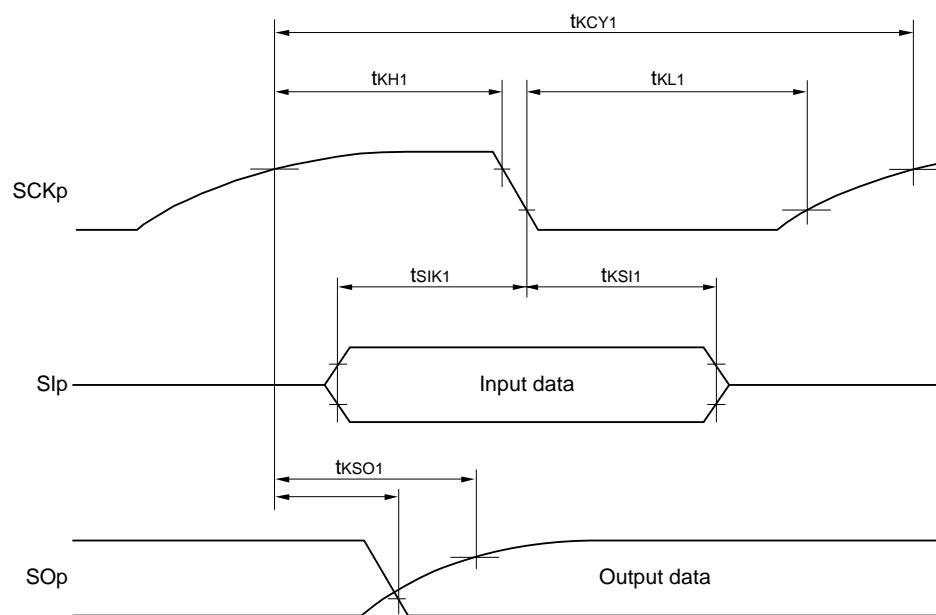
(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number ( $mn = 00$ ))

**Remark 4.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

**CSI mode serial transfer timing (master mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)**



**CSI mode serial transfer timing (master mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)**



**Remark 1.** p: CSI number ( $p = 00, 01, 10, 20, 30, 31$ ), m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ),  
g: PIM and POM number ( $g = 0, 1, 3$  to  $5, 14$ )

**Remark 2.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

**(9) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (slave mode, SCKp... external clock input)**

(TA = -40 to +85°C, 1.8 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

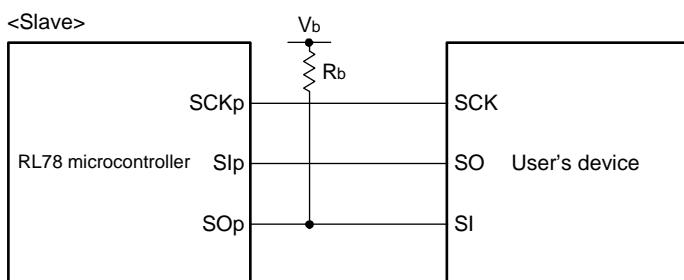
Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCKp cycle time Note 1	tkCY2	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V	24 MHz < f <sub>MCK</sub>	14/f <sub>MCK</sub>	—	—	—	—	ns
			20 MHz < f <sub>MCK</sub> ≤ 24 MHz	12/f <sub>MCK</sub>	—	—	—	—	ns
			8 MHz < f <sub>MCK</sub> ≤ 20 MHz	10/f <sub>MCK</sub>	—	—	—	—	ns
			4 MHz < f <sub>MCK</sub> ≤ 8 MHz	8/f <sub>MCK</sub>	16/f <sub>MCK</sub>	—	—	—	ns
			f <sub>MCK</sub> ≤ 4 MHz	6/f <sub>MCK</sub>	10/f <sub>MCK</sub>	—	10/f <sub>MCK</sub>	—	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V	24 MHz < f <sub>MCK</sub>	20/f <sub>MCK</sub>	—	—	—	—	ns
			20 MHz < f <sub>MCK</sub> ≤ 24 MHz	16/f <sub>MCK</sub>	—	—	—	—	ns
			16 MHz < f <sub>MCK</sub> ≤ 20 MHz	14/f <sub>MCK</sub>	—	—	—	—	ns
			8 MHz < f <sub>MCK</sub> ≤ 16 MHz	12/f <sub>MCK</sub>	—	—	—	—	ns
			4 MHz < f <sub>MCK</sub> ≤ 8 MHz	8/f <sub>MCK</sub>	16/f <sub>MCK</sub>	—	—	—	ns
			f <sub>MCK</sub> ≤ 4 MHz	6/f <sub>MCK</sub>	10/f <sub>MCK</sub>	—	10/f <sub>MCK</sub>	—	ns
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2	24 MHz < f <sub>MCK</sub>	48/f <sub>MCK</sub>	—	—	—	—	ns
			20 MHz < f <sub>MCK</sub> ≤ 24 MHz	36/f <sub>MCK</sub>	—	—	—	—	ns
			16 MHz < f <sub>MCK</sub> ≤ 20 MHz	32/f <sub>MCK</sub>	—	—	—	—	ns
			8 MHz < f <sub>MCK</sub> ≤ 16 MHz	26/f <sub>MCK</sub>	—	—	—	—	ns
			4 MHz < f <sub>MCK</sub> ≤ 8 MHz	16/f <sub>MCK</sub>	16/f <sub>MCK</sub>	—	—	—	ns
			f <sub>MCK</sub> ≤ 4 MHz	10/f <sub>MCK</sub>	10/f <sub>MCK</sub>	—	10/f <sub>MCK</sub>	—	ns
SCKp high-/low-level width	tKH2, tKL2	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V	tkCY2/2 - 12	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	ns	
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V	tkCY2/2 - 18	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	ns	
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	tkCY2/2 - 50	ns	
Slp setup time (to SCKp↑) Note 3	tsIK2	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V	1/f <sub>MCK</sub> + 20	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	ns	
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V	1/f <sub>MCK</sub> + 20	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	ns	
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	1/f <sub>MCK</sub> + 30	ns	
Slp hold time (from SCKp↑) Note 4	tksI2		1/f <sub>MCK</sub> + 31	1/f <sub>MCK</sub> + 31	1/f <sub>MCK</sub> + 31	1/f <sub>MCK</sub> + 31	1/f <sub>MCK</sub> + 31	ns	
Delay time from SCKp↓ to SO <sub>p</sub> output Note 5	tksO2	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ		2/f <sub>MCK</sub> + 120	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	ns	
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ		2/f <sub>MCK</sub> + 214	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	ns	
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ		2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	2/f <sub>MCK</sub> + 573	ns	

(Notes, Caution, and Remarks are listed on the next page.)

- Note 1.** Transfer rate in the SNOOZE mode: MAX. 1 Mbps
- Note 2.** Use it with EV<sub>DD0</sub> ≥ V<sub>b</sub>.
- Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes “to SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.
- Note 4.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes “from SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.
- Note 5.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes “from SCKp↑” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Caution** Select the TTL input buffer for the Slp pin and SCKp pin, and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SOp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

#### CSI mode connection diagram (during communication at different potential)



**Remark 1.** R<sub>b</sub>[Ω]: Communication line (SOp) pull-up resistance, C<sub>b</sub>[F]: Communication line (SOp) load capacitance, V<sub>b</sub>[V]: Communication line voltage

**Remark 2.** p: CSI number (p = 00, 01, 10, 20, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), g: PIM and POM number (g = 0, 1, 3 to 5, 14)

**Remark 3.** fmck: Serial array unit operation clock frequency

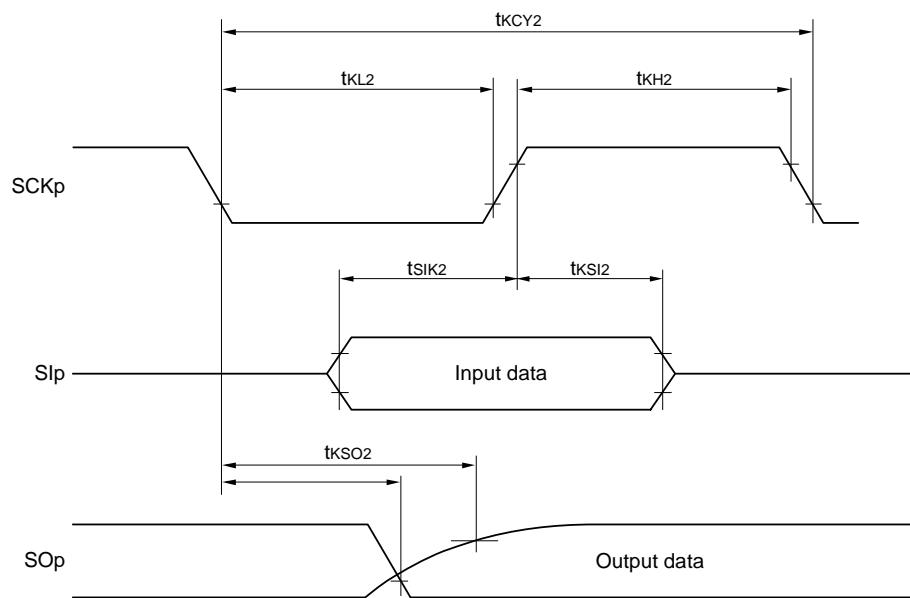
(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn)).

m: Unit number, n: Channel number (mn = 00, 01, 02, 10, 12, 13))

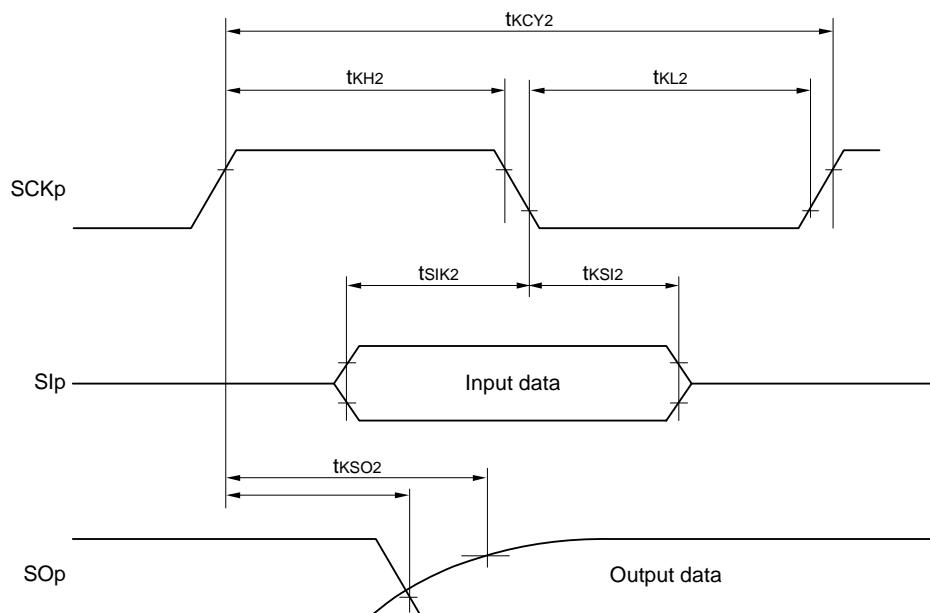
**Remark 4.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

Also, communication at different potential cannot be performed during clock synchronous serial communication with the slave select function.

**CSI mode serial transfer timing (slave mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)**



**CSI mode serial transfer timing (slave mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)**



**Remark 1.** p: CSI number (p = 00, 01, 10, 20, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3),  
g: PIM and POM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

Also, communication at different potential cannot be performed during clock synchronous serial communication with the slave select function.

**(10) Communication at different potential (1.8 V, 2.5 V, 3 V) (simplified I<sup>2</sup>C mode)**(TA = -40 to +85°C, 1.8 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCL <sub>r</sub> clock frequency	f <sub>SCL</sub>	4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		1000 Note 1		300 Note 1		300 Note 1	kHz
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		1000 Note 1		300 Note 1		300 Note 1	kHz
		4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ		400 Note 1		300 Note 1		300 Note 1	kHz
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ		400 Note 1		300 Note 1		300 Note 1	kHz
		1.8 V ≤ EV <sub>D0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ		300 Note 1		300 Note 1		300 Note 1	kHz
Hold time when SCL <sub>r</sub> = "L"	t <sub>LOW</sub>	4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	475		1550		1550		ns
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	475		1550		1550		ns
		4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	1150		1550		1550		ns
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	1150		1550		1550		ns
		1.8 V ≤ EV <sub>D0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	1550		1550		1550		ns
Hold time when SCL <sub>r</sub> = "H"	t <sub>HIGH</sub>	4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	245		610		610		ns
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	200		610		610		ns
		4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	675		610		610		ns
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	600		610		610		ns
		1.8 V ≤ EV <sub>D0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	610		610		610		ns

**(10) Communication at different potential (1.8 V, 2.5 V, 3 V) (simplified I<sup>2</sup>C mode)**(TA = -40 to +85°C, 1.8 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

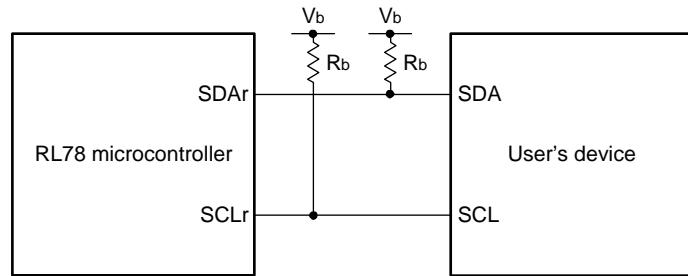
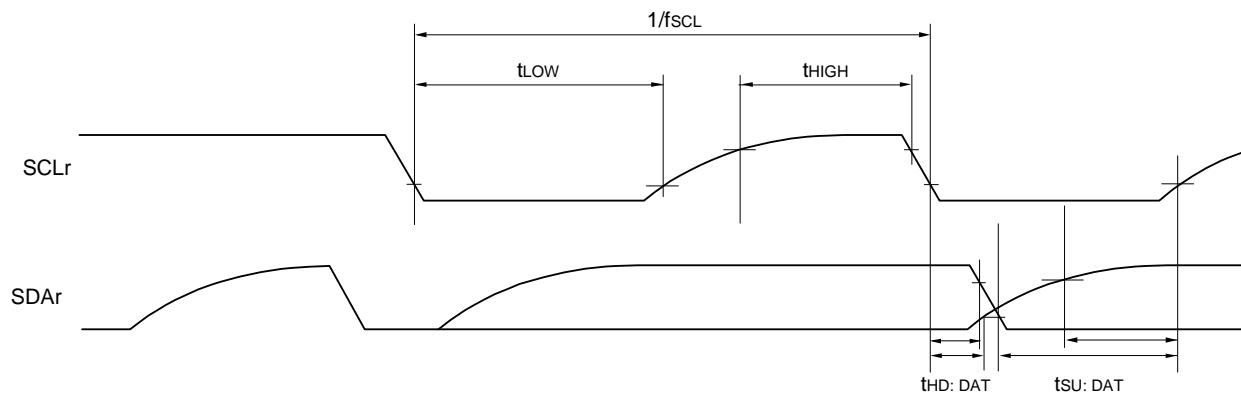
(2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Data setup time (reception)	t <sub>SU:DAT</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/fMCK + 135 Note 3		1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/fMCK + 135 Note 3		1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		ns
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		1/fMCK + 190 Note 3		ns
Data hold time (transmission)	t <sub>HD:DAT</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	305	0	305	0	305	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	305	0	305	0	305	ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	0	355	0	355	0	355	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	0	355	0	355	0	355	ns
		1.8 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V Note 2, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	0	405	0	405	0	405	ns

**Note 1.** The value must also be equal to or less than fMCK/4.**Note 2.** Use it with EV<sub>DD0</sub> ≥ V<sub>b</sub>.**Note 3.** Set the fMCK value to keep the hold time of SCL<sub>r</sub> = "L" and SCL<sub>r</sub> = "H".

**Caution** Select the TTL input buffer and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SD<sub>Ar</sub> pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SCL<sub>r</sub> pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**Simplified I<sup>2</sup>C mode connection diagram (during communication at different potential)****Simplified I<sup>2</sup>C mode serial transfer timing (during communication at different potential)**

**Remark 1.** R<sub>b</sub>[Ω]: Communication line (SDAr, SCLR) pull-up resistance, C<sub>b</sub>[F]: Communication line (SDAr, SCLR) load capacitance, V<sub>b</sub>[V]: Communication line voltage

**Remark 2.** r: IIC number (r = 00, 01, 10, 11, 20, 30, 31), g: PIM, POM number (g = 0, 1, 3 to 5, 14)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), mn = 00, 01, 02, 10, 12, 13)

### 34.5.2 Serial interface IICA

#### (1) I<sup>2</sup>C standard mode

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main mode)		LS (low-speed main mode)		LV (low-voltage main mode)		Unit	
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.		
SCLA0 clock frequency	f <sub>SCL</sub>	Standard mode: f <sub>CLK</sub> ≥ 1 MHz	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	0	100	0	100	0	100	kHz
			1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	0	100	0	100	0	100	kHz
			1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	0	100	0	100	0	100	kHz
			1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—	—	0	100	0	100	kHz
Setup time of restart condition	t <sub>SU: STA</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—	—	4.7	—	4.7	—	μs	
Hold time Note 1	t <sub>HD: STA</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—	—	4.0	—	4.0	—	μs	
Hold time when SCLA0 = "L"	t <sub>LOW</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.7	—	4.7	—	4.7	—	μs	
		1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—	—	4.7	—	4.7	—	μs	
Hold time when SCLA0 = "H"	t <sub>HIGH</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	4.0	—	4.0	—	4.0	—	μs	
		1.6 V ≤ EV <sub>D0</sub> ≤ 5.5 V	—	—	4.0	—	4.0	—	μs	

(Notes, Caution, and Remark are listed on the next page.)

(1) I<sup>2</sup>C standard mode(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
			MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
Data setup time (reception)	tsU: DAT	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	250		250		250		ns
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	250		250		250		ns
		1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	250		250		250		ns
		1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		250		250		ns
Data hold time (transmission) Note 2	tHD: DAT	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0	3.45	0	3.45	0	3.45	μs
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0	3.45	0	3.45	0	3.45	μs
		1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0	3.45	0	3.45	0	3.45	μs
		1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		0	3.45	0	3.45	μs
Setup time of stop condition	tsU: STO	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.0		4.0		4.0		μs
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.0		4.0		4.0		μs
		1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.0		4.0		4.0		μs
		1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		4.0		4.0		μs
Bus-free time	tBUF	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.7		4.7		4.7		μs
		1.8 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.7		4.7		4.7		μs
		1.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	4.7		4.7		4.7		μs
		1.6 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	—		4.7		4.7		μs

**Note 1.** The first clock pulse is generated after this period when the start/restart condition is detected.**Note 2.** The maximum value (MAX.) of tHD: DAT is during normal transfer and a wait state is inserted in the ACK (acknowledge) timing.**Caution** The values in the above table are applied even when bit 2 (PIOR02) in the peripheral I/O redirection register 0 (PIOR0) is 1. At this time, the pin characteristics (I<sub>OH1</sub>, I<sub>OL1</sub>, V<sub>OH1</sub>, V<sub>OL1</sub>) must satisfy the values in the redirect destination.**Remark** The maximum value of C<sub>b</sub> (communication line capacitance) and the value of R<sub>b</sub> (communication line pull-up resistor) at that time in each mode are as follows.Standard mode: C<sub>b</sub> = 400 pF, R<sub>b</sub> = 2.7 kΩ

(2) I<sup>2</sup>C fast mode(TA = -40 to +85°C, 1.6 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

Parameter	Symbol	Conditions		HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 clock frequency	fsCL	Fast mode: f <sub>CLK</sub> ≥ 3.5 MHz	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V	0	400	0	400	0	400	kHz
			1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V	0	400	0	400	0	400	kHz
Setup time of restart condition	tsU: STA	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
Hold time Note 1	t <sub>HD</sub> : STA	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
Hold time when SCLA0 = "L"	t <sub>LOW</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		1.3		1.3		1.3		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		1.3		1.3		1.3		μs
Hold time when SCLA0 = "H"	t <sub>HIGH</sub>	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
Data setup time (reception)	tsU: DAT	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		100		100		100		ns
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		100		100		100		ns
Data hold time (transmission) Note 2	t <sub>HD</sub> : DAT	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0	0.9	0	0.9	0	0.9	μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0	0.9	0	0.9	0	0.9	μs
Setup time of stop condition	tsU: STO	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		0.6		0.6		0.6		μs
Bus-free time	tbUF	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		1.3		1.3		1.3		μs
		1.8 V ≤ EV <sub>D0</sub> ≤ 5.5 V		1.3		1.3		1.3		μs

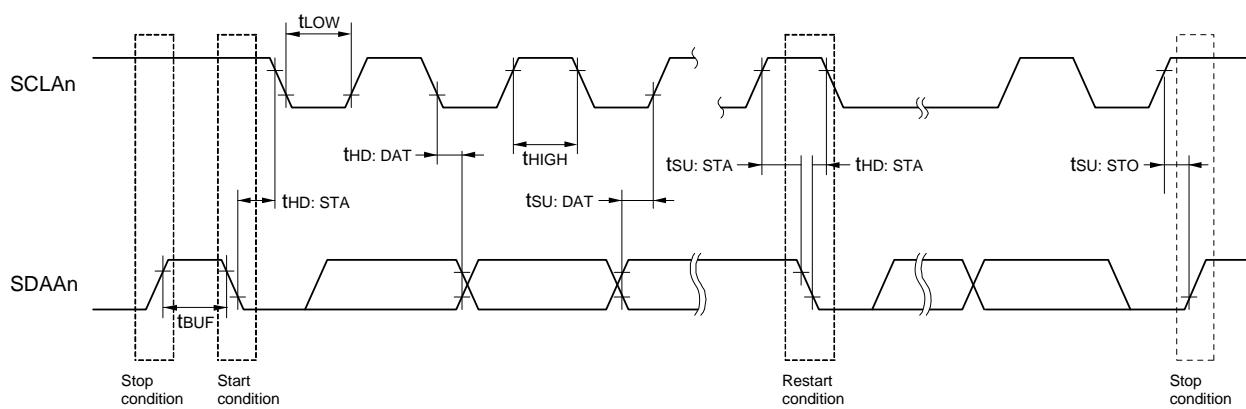
**Note 1.** The first clock pulse is generated after this period when the start/restart condition is detected.**Note 2.** The maximum value (MAX.) of t<sub>HD</sub>: DAT is during normal transfer and a wait state is inserted in the ACK (acknowledge) timing.**Caution** The values in the above table are applied even when bit 2 (PIOR02) in the peripheral I/O redirection register 0 (PIOR0) is 1. At this time, the pin characteristics (I<sub>OH1</sub>, I<sub>OL1</sub>, V<sub>OH1</sub>, V<sub>OL1</sub>) must satisfy the values in the redirect destination.**Remark** The maximum value of C<sub>b</sub> (communication line capacitance) and the value of R<sub>b</sub> (communication line pull-up resistor) at that time in each mode are as follows.Fast mode: C<sub>b</sub> = 320 pF, R<sub>b</sub> = 1.1 kΩ

(3) I<sup>2</sup>C fast mode plus(TA = -40 to +85°C, 1.6 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions		HS (high-speed main) mode		LS (low-speed main) mode		LV (low-voltage main) mode		Unit
				MIN.	MAX.	MIN.	MAX.	MIN.	MAX.	
SCLA0 clock frequency	f <sub>SCL</sub>	Fast mode plus: f <sub>CLK</sub> ≥ 10 MHz	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0	1000	—	—	—	—	kHz
Setup time of restart condition	t <sub>SU: STA</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.26		—	—	—	—	μs
Hold time Note 1	t <sub>HD: STA</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.26		—	—	—	—	μs
Hold time when SCLA0 = "L"	t <sub>LOW</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.5		—	—	—	—	μs
Hold time when SCLA0 = "H"	t <sub>HIGH</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.26		—	—	—	—	μs
Data setup time (reception)	t <sub>SU: DAT</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		50		—	—	—	—	ns
Data hold time (transmission)	t <sub>HD: DAT</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0	0.45	—	—	—	—	μs
Setup time of stop condition	t <sub>SU: STO</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.26		—	—	—	—	μs
Bus-free time	t <sub>BUF</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		0.5		—	—	—	—	μs

**Note 1.** The first clock pulse is generated after this period when the start/restart condition is detected.**Note 2.** The maximum value (MAX.) of t<sub>HD: DAT</sub> is during normal transfer and a wait state is inserted in the ACK (acknowledge) timing.**Caution** The values in the above table are applied even when bit 2 (PIOR02) in the peripheral I/O redirection register 0 (PIOR0) is 1. At this time, the pin characteristics (I<sub>OH1</sub>, I<sub>OL1</sub>, V<sub>OH1</sub>, V<sub>OL1</sub>) must satisfy the values in the redirect destination.**Note 3.** The maximum value of C<sub>b</sub> (communication line capacitance) and the value of R<sub>b</sub> (communication line pull-up resistor) at that time in each mode are as follows.Fast mode plus: C<sub>b</sub> = 120 pF, R<sub>b</sub> = 1.1 kΩ

## IICA serial transfer timing

**Remark** n = 0, 1

## 34.6 Analog Characteristics

### 34.6.1 A/D converter characteristics

#### Classification of A/D converter characteristics

Input channel \ Reference Voltage	Reference voltage (+) = AVREFP Reference voltage (-) = AVREFM	Reference voltage (+) = VDD Reference voltage (-) = Vss	Reference voltage (+) = VBGR Reference voltage (-) = AVREFM
AN10 to AN14	Refer to 34.6.1 (1).	Refer to 34.6.1 (3).	Refer to 34.6.1 (4). —
AN16 to ANI20	Refer to 34.6.1 (2).		
Internal reference voltage Temperature sensor output voltage	Refer to 34.6.1 (1).		

- (1) When reference voltage (+) = AVREFP/ANI0 (ADREFP1 = 0, ADREFP0 = 1), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI2 to ANI14, internal reference voltage, and temperature sensor output voltage

(TA = -40 to +85°C, 1.6 V ≤ AVREFP ≤ VDD ≤ 5.5 V, Vss = 0 V, Reference voltage (+) = AVREFP, Reference voltage (-) = AVREFM = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution	RES		8		10	bit
Overall error Note 1	AINL	10-bit resolution AVREFP = VDD Note 3	1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4	1.2 1.2	±3.5 ±7.0	LSB
Conversion time	tconv	10-bit resolution Target pin: ANI2 to ANI14	3.6 V ≤ VDD ≤ 5.5 V 2.7 V ≤ VDD ≤ 5.5 V 1.8 V ≤ VDD ≤ 5.5 V 1.6 V ≤ VDD ≤ 5.5 V	2.125 3.1875 17 57	39 39 39 95	μs
			3.6 V ≤ VDD ≤ 5.5 V 2.7 V ≤ VDD ≤ 5.5 V 2.4 V ≤ VDD ≤ 5.5 V	2.375 3.5625 17	39 39 39	μs
			10-bit resolution Target pin: Internal reference voltage, and temperature sensor output voltage (HS (high-speed main) mode)			μs
			1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±0.25 ±0.50	%FSR
	Ezs	10-bit resolution AVREFP = VDD Note 3	1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±0.25 ±0.50	%FSR
			1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±0.25 ±0.50	%FSR
Full-scale error Notes 1, 2	Efs	10-bit resolution AVREFP = VDD Note 3	1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±0.25 ±0.50	%FSR
			1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±0.25 ±0.50	%FSR
Integral linearity error Note 1	ILE	10-bit resolution AVREFP = VDD Note 3	1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±2.5 ±5.0	LSB
			1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±1.5 ±2.0	LSB
Differential linearity error Note 1	DLE	10-bit resolution AVREFP = VDD Note 3	1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±1.5 ±2.0	LSB
			1.8 V ≤ AVREFP ≤ 5.5 V 1.6 V ≤ AVREFP ≤ 5.5 V Note 4		±1.5 ±2.0	LSB
Analog input voltage	VAIN	ANI2 to ANI14	0		AVREFP	V
		Internal reference voltage (2.4 V ≤ VDD ≤ 5.5 V, HS (high-speed main) mode)			VBGR Note 5	V
		Temperature sensor output voltage (2.4 V ≤ VDD ≤ 5.5 V, HS (high-speed main) mode)			VTMPS25 Note 5	V

Note 1. Excludes quantization error (±1/2 LSB).

Note 2. This value is indicated as a ratio (%FSR) to the full-scale value.

Note 3. When AVREFP < VDD, the MAX. values are as follows.

Overall error: Add ±1.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.05%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±0.5 LSB to the MAX. value when AVREFP = VDD.

Note 4. Values when the conversion time is set to 57 μs (min.) and 95 μs (max.).

Note 5. Refer to 34.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

- (2) When reference voltage (+) = AVREFP/ANI0 (ADREFP1 = 0, ADREFP0 = 1), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI16 to ANI20

(TA = -40 to +85°C, 1.6 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, 1.6 V ≤ AVREFP ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V, Reference voltage (+) = AVREFP, Reference voltage (-) = AVREFM = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		10	bit
Overall error Note 1	AINL	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	1.8 V ≤ AVREFP ≤ 5.5 V		1.2	±5.0	LSB
			1.6 V ≤ AVREFP ≤ 5.5 V Note 5		1.2	±8.5	LSB
Conversion time	tCONV	10-bit resolution Target ANI pin: ANI16 to ANI20	3.6 V ≤ VDD ≤ 5.5 V	2.125		39	μs
			2.7 V ≤ VDD ≤ 5.5 V	3.1875		39	μs
			1.8 V ≤ VDD ≤ 5.5 V	17		39	μs
			1.6 V ≤ VDD ≤ 5.5 V	57		95	μs
Zero-scale error Notes 1, 2	Ezs	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	1.8 V ≤ AVREFP ≤ 5.5 V			±0.35	%FSR
			1.6 V ≤ AVREFP ≤ 5.5 V Note 5			±0.60	%FSR
Full-scale error Notes 1, 2	Efs	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	1.8 V ≤ AVREFP ≤ 5.5 V			±0.35	%FSR
			1.6 V ≤ AVREFP ≤ 5.5 V Note 5			±0.60	%FSR
Integral linearity error Note 1	ILE	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	1.8 V ≤ AVREFP ≤ 5.5 V			±3.5	LSB
			1.6 V ≤ AVREFP ≤ 5.5 V Note 5			±6.0	LSB
Differential linearity error Note 1	DLE	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	1.8 V ≤ AVREFP ≤ 5.5 V			±2.0	LSB
			1.6 V ≤ AVREFP ≤ 5.5 V Note 5			±2.5	LSB
Analog input voltage	VAIN	ANI16 to ANI20		0		AVREFP and EVDD0	V

**Note 1.** Excludes quantization error (±1/2 LSB).

**Note 2.** This value is indicated as a ratio (%FSR) to the full-scale value.

**Note 3.** When EVDD0 ≤ AVREFP ≤ VDD, the MAX. values are as follows.

Overall error: Add ±1.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.05%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±0.5 LSB to the MAX. value when AVREFP = VDD.

**Note 4.** When AVREFP < EVDD0 ≤ VDD, the MAX. values are as follows.

Overall error: Add ±4.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.20%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±2.0 LSB to the MAX. value when AVREFP = VDD.

**Note 5.** When the conversion time is set to 57 μs (min.) and 95 μs (max.).

- (3) When reference voltage (+) = V<sub>DD</sub> (ADREFP1 = 0, ADREFP0 = 0), reference voltage (-) = V<sub>SS</sub> (ADREFM = 0), target pin: ANI0 to ANI14, ANI16 to ANI20, internal reference voltage, and temperature sensor output voltage

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V, Reference voltage (+) = V<sub>DD</sub>, Reference voltage (-) = V<sub>SS</sub>)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution	RES		8		10	bit
Overall error Note 1	A <sub>INL</sub>	10-bit resolution 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V Note 3		1.2	±7.0	LSB
Conversion time	t <sub>CONV</sub>	10-bit resolution Target pin: ANI0 to ANI14, ANI16 to ANI20 3.6 V ≤ V <sub>DD</sub> ≤ 5.5 V 2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V 1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V 10-bit resolution Target pin: internal reference voltage, and temperature sensor output voltage (HS (high-speed main) mode) 3.6 V ≤ V <sub>DD</sub> ≤ 5.5 V 2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V 2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V	2.125 3.1875 17 57 2.375 3.5625 17		39 39 39 95 39 39 39	μs μs μs μs μs μs μs
Zero-scale error Notes 1, 2	E <sub>SZ</sub>	10-bit resolution 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V Note 3			±0.60 ±0.85	%FSR
Full-scale error Notes 1, 2	E <sub>Fs</sub>	10-bit resolution 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V Note 3			±0.60 ±0.85	%FSR
Integral linearity error Note 1	I <sub>LE</sub>	10-bit resolution 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V Note 3			±4.0 ±6.5	LSB
Differential linearity error Note 1	D <sub>LE</sub>	10-bit resolution 1.6 V ≤ V <sub>DD</sub> ≤ 5.5 V Note 3			±2.0 ±2.5	LSB
Analog input voltage	V <sub>A<sup>IN</sup></sub>	ANI0 to ANI14 ANI16 to ANI20 Internal reference voltage (2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode) Temperature sensor output voltage (2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode)	0 0 V <sub>BGR</sub> Note 4 V <sub>TMPS25</sub> Note 4		V <sub>DD</sub> EV <sub>VDD0</sub> V V	V

Note 1. Excludes quantization error (±1/2 LSB).

Note 2. This value is indicated as a ratio (% FSR) to the full-scale value.

Note 3. When the conversion time is set to 57 μs (min.) and 95 μs (max.).

Note 4. Refer to 34.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

- (4) When reference voltage (+) = Internal reference voltage (ADREFP1 = 1, ADREFP0 = 0), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI0, ANI2 to ANI14, ANI16 to ANI20

(TA = -40 to +85°C, 2.4 V ≤ VDD ≤ 5.5 V, 1.6 V ≤ EVDD = EVDD1 ≤ VDD, Vss = EVSS0 = EVSS1 = 0 V, Reference voltage (+) = VBGR Note 3, Reference voltage (-) = AVREFM = 0 V Note 4, HS (high-speed main) mode)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		bit	
Conversion time	tCONV	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V	17		39	μs
Zero-scale error Notes 1, 2	Ezs	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±0.60	% FSR
Integral linearity error Note 1	ILE	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±2.0	LSB
Differential linearity error Note 1	DLE	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±1.0	LSB
Analog input voltage	VAIN			0		VBGR Note 3	V

**Note 1.** Excludes quantization error (±1/2 LSB).

**Note 2.** This value is indicated as a ratio (% FSR) to the full-scale value.

**Note 3.** Refer to 34.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

**Note 4.** When reference voltage (-) = Vss, the MAX. values are as follows.

Zero-scale error: Add ±0.35%FSR to the MAX. value when reference voltage (-) = AVREFM.

Integral linearity error: Add ±0.5 LSB to the MAX. value when reference voltage (-) = AVREFM.

Differential linearity error: Add ±0.2 LSB to the MAX. value when reference voltage (-) = AVREFM.

### 34.6.2 Temperature sensor characteristics/internal reference voltage characteristic

(TA = -40 to +85°C, 2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V, HS (high-speed main) mode)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Temperature sensor output voltage	V <sub>TMPS25</sub>	Setting ADS register = 80H, TA = +25°C		1.05		V
Internal reference voltage	V <sub>BGR</sub>	Setting ADS register = 81H	1.38	1.45	1.5	V
Temperature coefficient	F <sub>VTMPS</sub>	Temperature sensor that depends on the temperature		-3.6		mV/°C
Operation stabilization wait time	t <sub>AMP</sub>		5			μs

### 34.6.3 D/A converter characteristics

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>SS0</sub> = EV<sub>SS1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES					8	bit
Overall error	AINL	R <sub>load</sub> = 4 MΩ	1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V			±2.5	LSB
		R <sub>load</sub> = 8 MΩ	1.8 V ≤ V <sub>DD</sub> ≤ 5.5 V			±2.5	LSB
Settling time	t <sub>SET</sub>	C <sub>load</sub> = 20 pF	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			3	μs
			1.6 V ≤ V <sub>DD</sub> < 2.7 V			6	μs

### 34.6.4 Comparator

(TA = -40 to +85°C, 1.6 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage range	I <sub>Vref</sub>			0		EV <sub>VDD0</sub> - 1.4	V
	I <sub>Vcmp</sub>			-0.3		EV <sub>VDD0</sub> + 0.3	V
Output delay	t <sub>d</sub>	V <sub>DD</sub> = 3.0 V Input slew rate > 50 mV/μs	Comparator high-speed mode, standard mode			1.2	μs
			Comparator high-speed mode, window mode			2.0	μs
			Comparator low-speed mode, standard mode		3.0	5.0	μs
High-electric-potential reference voltage	V <sub>Tw+</sub>	Comparator high-speed mode, window mode			0.76 V <sub>DD</sub>		V
Low-electric-potential reference voltage	V <sub>Tw-</sub>	Comparator high-speed mode, window mode			0.24 V <sub>DD</sub>		V
Operation stabilization wait time	t <sub>CMP</sub>			100			μs
Internal reference voltage Note	V <sub>BGR</sub>	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode		1.38	1.45	1.50	V

**Note** Not usable in LS (low-speed main) mode, LV (low-voltage main) mode, sub-clock operation, or STOP mode.

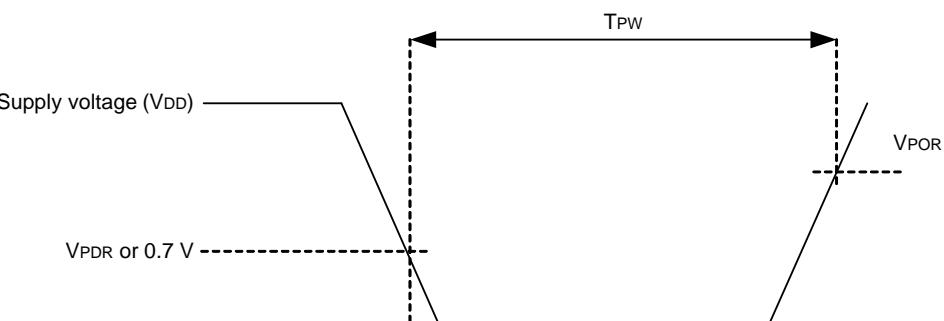
### 34.6.5 POR circuit characteristics

(TA = -40 to +85°C, V<sub>SS</sub> = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Power on/down reset threshold	V <sub>POR</sub>	Voltage threshold on V <sub>DD</sub> rising	1.47	1.51	1.55	V
	V <sub>PDR</sub>	Voltage threshold on V <sub>DD</sub> falling Note 1	1.46	1.50	1.54	V
Minimum pulse width Note 2	T <sub>PW</sub>		300			μs

**Note 1.** However, when the operating voltage falls while the LVD is off, enter STOP mode, or enable the reset status using the external reset pin before the voltage falls below the operating voltage range shown in 34.4 AC Characteristics.

**Note 2.** Minimum time required for a POR reset when V<sub>DD</sub> exceeds below V<sub>PDR</sub>. This is also the minimum time required for a POR reset from when V<sub>DD</sub> exceeds below 0.7 V to when V<sub>DD</sub> exceeds V<sub>POR</sub> while STOP mode is entered or the main system clock is stopped through setting bit 0 (HIOSTOP) and bit 7 (MSTOP) in the clock operation status control register (CSC).



### 34.6.6 LVD circuit characteristics

#### (1) Reset Mode and Interrupt Mode

(TA = -40 to +85°C, VPDR ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter		Symbol	Conditions	MIN.	TYP.	MAX.	Unit	
Voltage detection threshold	Supply voltage level	VLVD0	Rising edge	3.98	4.06	4.14	V	
			Falling edge	3.90	3.98	4.06	V	
		VLVD1	Rising edge	3.68	3.75	3.82	V	
			Falling edge	3.60	3.67	3.74	V	
		VLVD2	Rising edge	3.07	3.13	3.19	V	
			Falling edge	3.00	3.06	3.12	V	
		VLVD3	Rising edge	2.96	3.02	3.08	V	
			Falling edge	2.90	2.96	3.02	V	
		VLVD4	Rising edge	2.86	2.92	2.97	V	
			Falling edge	2.80	2.86	2.91	V	
		VLVD5	Rising edge	2.76	2.81	2.87	V	
			Falling edge	2.70	2.75	2.81	V	
		VLVD6	Rising edge	2.66	2.71	2.76	V	
			Falling edge	2.60	2.65	2.70	V	
		VLVD7	Rising edge	2.56	2.61	2.66	V	
			Falling edge	2.50	2.55	2.60	V	
		VLVD8	Rising edge	2.45	2.50	2.55	V	
			Falling edge	2.40	2.45	2.50	V	
		VLVD9	Rising edge	2.05	2.09	2.13	V	
			Falling edge	2.00	2.04	2.08	V	
		VLVD10	Rising edge	1.94	1.98	2.02	V	
			Falling edge	1.90	1.94	1.98	V	
		VLVD11	Rising edge	1.84	1.88	1.91	V	
			Falling edge	1.80	1.84	1.87	V	
		VLVD12	Rising edge	1.74	1.77	1.81	V	
			Falling edge	1.70	1.73	1.77	V	
		VLVD13	Rising edge	1.64	1.67	1.70	V	
			Falling edge	1.60	1.63	1.66	V	
Minimum pulse width		tLW		300			μs	
Detection delay time						300	μs	

**(2) Interrupt & Reset Mode**

(TA = -40 to +85°C, VPDR ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Voltage detection threshold	VLVDA0	VPOC2, VPOC1, VPOCO = 0, 0, 0, falling reset voltage		1.60	1.63	1.66	V
	VLVDA1	LVIS1, LVIS0 = 1, 0	Rising release reset voltage	1.74	1.77	1.81	V
			Falling interrupt voltage	1.70	1.73	1.77	V
	VLVDA2	LVIS1, LVIS0 = 0, 1	Rising release reset voltage	1.84	1.88	1.91	V
			Falling interrupt voltage	1.80	1.84	1.87	V
	VLVDA3	LVIS1, LVIS0 = 0, 0	Rising release reset voltage	2.86	2.92	2.97	V
			Falling interrupt voltage	2.80	2.86	2.91	V
	VLVDB0	VPOC2, VPOC1, VPOCO = 0, 0, 1, falling reset voltage		1.80	1.84	1.87	V
	VLVDB1	LVIS1, LVIS0 = 1, 0	Rising release reset voltage	1.94	1.98	2.02	V
			Falling interrupt voltage	1.90	1.94	1.98	V
	VLVDB2	LVIS1, LVIS0 = 0, 1	Rising release reset voltage	2.05	2.09	2.13	V
			Falling interrupt voltage	2.00	2.04	2.08	V
	VLVDB3	LVIS1, LVIS0 = 0, 0	Rising release reset voltage	3.07	3.13	3.19	V
			Falling interrupt voltage	3.00	3.06	3.12	V
	VLVDC0	VPOC2, VPOC1, VPOCO = 0, 1, 0, falling reset voltage		2.40	2.45	2.50	V
	VLVDC1	LVIS1, LVIS0 = 1, 0	Rising release reset voltage	2.56	2.61	2.66	V
			Falling interrupt voltage	2.50	2.55	2.60	V
	VLVDC2	LVIS1, LVIS0 = 0, 1	Rising release reset voltage	2.66	2.71	2.76	V
			Falling interrupt voltage	2.60	2.65	2.70	V
	VLVDC3	LVIS1, LVIS0 = 0, 0	Rising release reset voltage	3.68	3.75	3.82	V
			Falling interrupt voltage	3.60	3.67	3.74	V
	VLVDD0	VPOC2, VPOC1, VPOCO = 0, 1, 1, falling reset voltage		2.70	2.75	2.81	V
	VLVDD1	LVIS1, LVIS0 = 1, 0	Rising release reset voltage	2.86	2.92	2.97	V
			Falling interrupt voltage	2.80	2.86	2.91	V
	VLVDD2	LVIS1, LVIS0 = 0, 1	Rising release reset voltage	2.96	3.02	3.08	V
			Falling interrupt voltage	2.90	2.96	3.02	V
	VLVDD3	LVIS1, LVIS0 = 0, 0	Rising release reset voltage	3.98	4.06	4.14	V
			Falling interrupt voltage	3.90	3.98	4.06	V

**34.6.7 Power supply voltage rising slope characteristics**

(TA = -40 to +85°C, Vss = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Power supply voltage rising slope	SVDD				54	V/ms

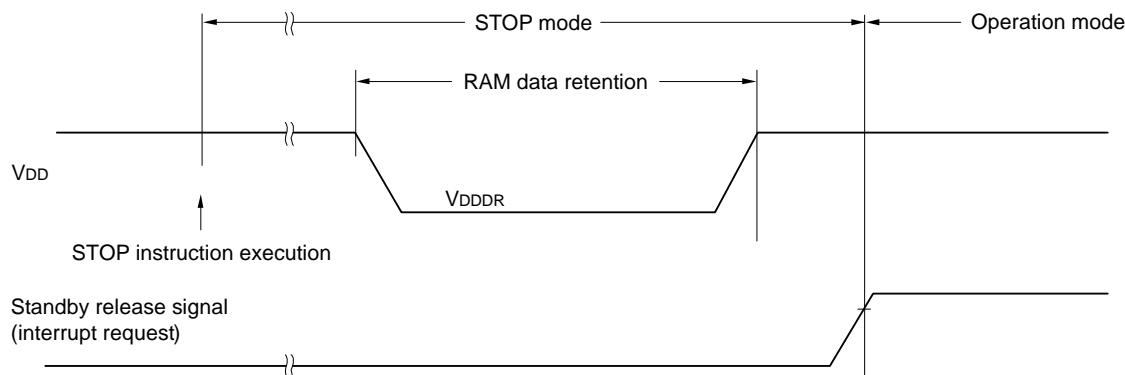
**Caution** Make sure to keep the internal reset state by the LVD circuit or an external reset until Vdd reaches the operating voltage range shown in 34.4 AC Characteristics.

### 34.7 RAM Data Retention Characteristics

(TA = -40 to +85°C, Vss = 0V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention supply voltage	VDDDR		1.46 Note		5.5	V

**Note** The value depends on the POR detection voltage. When the voltage drops, the RAM data is retained before a POR reset is effected, but RAM data is not retained when a POR reset is effected.



### 34.8 Flash Memory Programming Characteristics

(TA = -40 to +85°C, 1.8 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
System clock frequency	fCLK	1.8 V ≤ VDD ≤ 5.5 V	1		32	MHz
Number of code flash rewrites	Cerwr	Retained for 20 years TA = 85°C	1,000			Times
Notes 1, 2, 3		Retained for 1 year TA = 25°C		1,000,000		
Number of data flash rewrites		Retained for 5 years TA = 85°C	100,000			
Notes 1, 2, 3		Retained for 20 years TA = 85°C	10,000			

**Note 1.** 1 erase + 1 write after the erase is regarded as 1 rewrite. The retaining years are until next rewrite after the rewrite.

**Note 2.** When using flash memory programmer and Renesas Electronics self-programming library

**Note 3.** These are the characteristics of the flash memory and the results obtained from reliability testing by Renesas Electronics Corporation.

### 34.9 Dedicated Flash Memory Programmer Communication (UART)

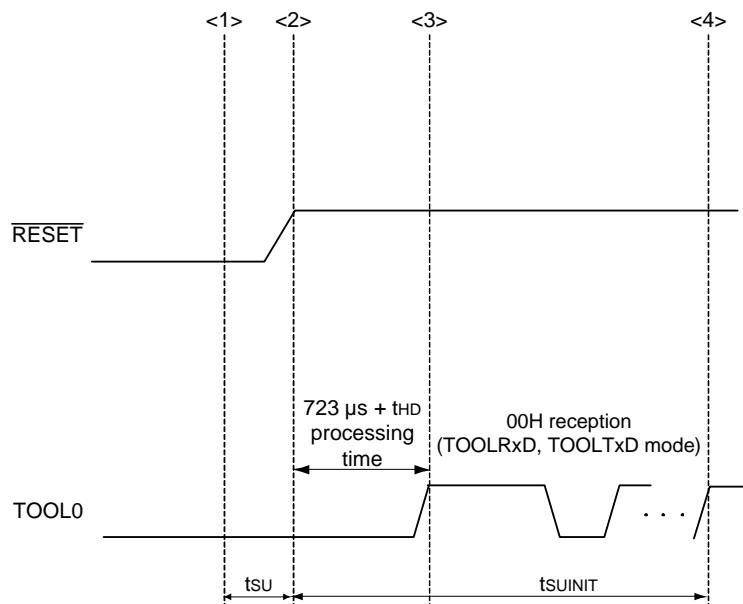
(TA = -40 to +85°C, 1.8 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		During serial programming	115,200		1,000,000	bps

### 34.10 Timing of Entry to Flash Memory Programming Modes

(TA = -40 to +85°C, 1.8 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
How long from when an external reset ends until the initial communication settings are specified	tsUINIT	POR and LVD reset must end before the external reset ends.			100	ms
How long from when the TOOL0 pin is placed at the low level until an external reset ends	tsU	POR and LVD reset must end before the external reset ends.	10			μs
How long the TOOL0 pin must be kept at the low level after an external reset ends (excluding the processing time of the firmware to control the flash memory)	tHD	POR and LVD reset must end before the external reset ends.	1			ms



<1> The low level is input to the TOOL0 pin.

<2> The external reset ends (POR and LVD reset must end before the external reset ends).

<3> The TOOL0 pin is set to the high level.

<4> Setting of the flash memory programming mode by UART reception and complete the baud rate setting.

**Remark** tsUINIT: The segment shows that it is necessary to finish specifying the initial communication settings within 100 ms from when the external resets end.

tsU: How long from when the TOOL0 pin is placed at the low level until a pin reset ends

tHD: How long to keep the TOOL0 pin at the low level from when the external resets end  
(excluding the processing time of the firmware to control the flash memory)

## CHAPTER 35 ELECTRICAL SPECIFICATIONS (G: INDUSTRIAL APPLICATIONS TA = -40 to +105°C)

This chapter describes the following electrical specifications.

Target products G: Industrial applications TA = -40 to +105°C

R5F104xxGxx

**Caution 1.** The RL78 microcontrollers have an on-chip debug function, which is provided for development and evaluation. Do not use the on-chip debug function in products designated for mass production, because the guaranteed number of rewritable times of the flash memory may be exceeded when this function is used, and product reliability therefore cannot be guaranteed. Renesas Electronics is not liable for problems occurring when the on-chip debug function is used.

**Caution 2.** With products not provided with an EVDD0, EVDD1, EVSS0, or EVSS1 pin, replace EVDD0 and EVDD1 with VDD, or replace EVSS0 and EVSS1 with Vss.

**Caution 3.** The pins mounted depend on the product. Refer to 2.1 Port Functions to 2.2.1 Functions for each product.

**Caution 4.** Please contact Renesas Electronics sales office for derating of operation under TA = +85 to +105°C. Derating is the systematic reduction of load for the sake of improved reliability.

**Remark** When RL78/G14 is used in the range of TA = -40 to +85°C, see **CHAPTER 34 ELECTRICAL SPECIFICATIONS (TA = -40 to +85°C)**.

Operation of products rated "G: Industrial applications (TA = -40 to + 105°C)" at ambient operating temperatures above 85°C differs from that of products rated "A: Consumer applications" and "D: Industrial applications" in the ways listed below.

Parameter	A: Consumer applications, D: Industrial applications	G: Industrial applications
Operating ambient temperature	TA = -40 to +85°C	TA = -40 to +105°C
Operating mode Operating voltage range	HS (high-speed main) mode: 2.7 V ≤ VDD ≤ 5.5 V@1 MHz to 32 MHz 2.4 V ≤ VDD ≤ 5.5 V@1 MHz to 16 MHz LS (low-speed main) mode: 1.8 V ≤ VDD ≤ 5.5 V@1 MHz to 8 MHz LV (low-voltage main) mode: 1.6 V ≤ VDD ≤ 5.5 V@1 MHz to 4 MHz	HS (high-speed main) mode only: 2.7 V ≤ VDD ≤ 5.5 V@1 MHz to 32 MHz 2.4 V ≤ VDD ≤ 5.5 V@1 MHz to 16 MHz
High-speed on-chip oscillator clock accuracy	1.8 V ≤ VDD ≤ 5.5 V: ±1.0% @ TA = -20 to +85°C ±1.5% @ TA = -40 to -20°C 1.6 V ≤ VDD < 1.8 V: ±5.0% @ TA = -20 to +85°C ±5.5% @ TA = -40 to -20°C	2.4 V ≤ VDD ≤ 5.5 V: ±2.0% @ TA = +85 to +105°C ±1.0% @ TA = -20 to +85°C ±1.5% @ TA = -40 to -20°C
Serial array unit	UART CSI: fCLK/2 (16 Mbps supported), fCLK/4 Simplified I <sup>2</sup> C communication	UART CSI: fCLK/4 Simplified I <sup>2</sup> C communication
I <sup>2</sup> CA	Standard mode Fast mode Fast mode plus	Standard mode Fast mode
Voltage detector	• Rising: 1.67 V to 4.06 V (14 stages) • Falling: 1.63 V to 3.98 V (14 stages)	• Rising: 2.61 V to 4.06 V (8 stages) • Falling: 2.55 V to 3.98 V (8 stages)

**Remark** The electrical characteristics of products rated "G: Industrial applications (TA = -40 to + 105°C)" at ambient operating temperatures above 85°C differ from those of products rated "A: Consumer applications" and "D: Industrial applications". For details, refer to 35.1 to 35.10.

## 35.1 Absolute Maximum Ratings

**Absolute Maximum Ratings** (1/2)

Parameter	Symbols	Conditions	Ratings	Unit
Supply voltage	VDD		-0.5 to +6.5	V
	EVDD0, EVDD1	EVDD0 = EVDD1	-0.5 to +6.5	V
	EVSS0, EVSS1	EVSS0 = EVSS1	-0.5 to +0.3	V
REGC pin input voltage	VIREGC	REGC	-0.3 to +2.8 and -0.3 to VDD +0.3 Note 1	V
Input voltage	VI1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	-0.3 to EVDD0 +0.3 and -0.3 to VDD +0.3 Note 2	V
	VI2	P60 to P63 (N-ch open-drain)	-0.3 to +6.5	V
	VI3	P20 to P27, P121 to P124, P137, P150 to P156, EXCLK, EXCLKS, RESET	-0.3 to VDD +0.3 Note 2	V
Output voltage	VO1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	-0.3 to EVDD0 +0.3 and -0.3 to VDD +0.3 Note 2	V
	VO2	P20 to P27, P150 to P156	-0.3 to VDD +0.3 Note 2	V
Analog input voltage	VAI1	ANI16 to ANI20	-0.3 to EVDD0 +0.3 and -0.3 to AVREF(+) +0.3 Notes 2, 3	V
	VAI2	ANI0 to ANI14	-0.3 to VDD +0.3 and -0.3 to AVREF(+) +0.3 Notes 2, 3	V

**Note 1.** Connect the REGC pin to Vss via a capacitor (0.47 to 1  $\mu$ F). This value regulates the absolute maximum rating of the REGC pin. Do not use this pin with voltage applied to it.

**Note 2.** Must be 6.5 V or lower.

**Note 3.** Do not exceed AVREF (+) + 0.3 V in case of A/D conversion target pin.

**Caution** Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

**Remark 1.** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

**Remark 2.** AVREF (+): + side reference voltage of the A/D converter.

**Remark 3.** Vss: Reference voltage

**Absolute Maximum Ratings**

(2/2)

Parameter	Symbols	Conditions		Ratings	Unit	
Output current, high	I <sub>OH1</sub>	Per pin	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	-40	mA	
		Total of all pins -170 mA	P00 to P04, P40 to P47, P102, P120, P130, P140 to P145	-70	mA	
			P05, P06, P10 to P17, P30, P31, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147	-100	mA	
	I <sub>OH2</sub>	Per pin	P20 to P27, P150 to P156	-0.5	mA	
		Total of all pins		-2	mA	
	I <sub>OL1</sub>	Per pin	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	40	mA	
		Total of all pins 170 mA	P00 to P04, P40 to P47, P102, P120, P130, P140 to P145	70	mA	
			P05, P06, P10 to P17, P30, P31, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147	100	mA	
		I <sub>OL2</sub>	Per pin	P20 to P27, P150 to P156	1	mA
			Total of all pins		5	mA
Operating ambient temperature	TA	In normal operation mode		-40 to +105	°C	
Storage temperature	T <sub>stg</sub>					

**Caution** Product quality may suffer if the absolute maximum rating is exceeded even momentarily for any parameter. That is, the absolute maximum ratings are rated values at which the product is on the verge of suffering physical damage, and therefore the product must be used under conditions that ensure that the absolute maximum ratings are not exceeded.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

## 35.2 Oscillator Characteristics

### 35.2.1 X1, XT1 characteristics

(TA = -40 to +105°C, 2.4 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Resonator	Resonator	Conditions	MIN.	TYP.	MAX.	Unit
X1 clock oscillation frequency (fx) Note	Ceramic resonator/ crystal resonator	2.7 V ≤ VDD ≤ 5.5 V	1.0		20.0	MHz
		2.4 V ≤ VDD < 2.7 V	1.0		16.0	
XT1 clock oscillation frequency (fx <sub>T</sub> ) Note	Crystal resonator		32	32.768	35	kHz

**Note** Indicates only permissible oscillator frequency ranges. Refer to **AC Characteristics** for instruction execution time.  
Request evaluation by the manufacturer of the oscillator circuit mounted on a board to check the oscillator characteristics.

**Caution** Since the CPU is started by the high-speed on-chip oscillator clock after a reset release, check the X1 clock oscillation stabilization time using the oscillation stabilization time counter status register (OSTC) by the user. Determine the oscillation stabilization time of the OSTC register and the oscillation stabilization time select register (OSTS) after sufficiently evaluating the oscillation stabilization time with the resonator to be used.

**Remark** When using the X1 oscillator and XT1 oscillator, refer to **5.4 System Clock Oscillator**.

### 35.2.2 On-chip oscillator characteristics

(TA = -40 to +105°C, 2.4 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Oscillators	Parameters	Conditions		MIN.	TYP.	MAX.	Unit
High-speed on-chip oscillator clock frequency Notes 1, 2	f <sub>H</sub>			1		32	MHz
High-speed on-chip oscillator clock frequency accuracy		-20 to +85°C	2.4 V ≤ VDD ≤ 5.5 V	-1.0		+1.0	%
		-40 to -20°C	2.4 V ≤ VDD ≤ 5.5 V	-1.5		+1.5	%
		+85 to +105°C	2.4 V ≤ VDD ≤ 5.5 V	-2.0		+2.0	%
Low-speed on-chip oscillator clock frequency	f <sub>L</sub>			15			kHz
Low-speed on-chip oscillator clock frequency accuracy				-15		+15	%

**Note 1.** High-speed on-chip oscillator frequency is selected with bits 0 to 4 of the option byte (000C2H) and bits 0 to 2 of the HOCODIV register.

**Note 2.** This only indicates the oscillator characteristics. Refer to **AC Characteristics** for instruction execution time.

### 35.3 DC Characteristics

#### 35.3.1 Pin characteristics

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(1/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output current, high Note 1	IOH1	Per pin for P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	2.4 V ≤ EVDD0 ≤ 5.5 V			-3.0 Note 2	mA
		Total of P00 to P04, P40 to P47, P102, P120, P130, P140 to P145 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V			-30.0	mA
			2.7 V ≤ EVDD0 < 4.0 V			-10.0	mA
			2.4 V ≤ EVDD0 < 2.7 V			-5.0	mA
		Total of P05, P06, P10 to P17, P30, P31, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V			-30.0	mA
			2.7 V ≤ EVDD0 < 4.0 V			-19.0	mA
			2.4 V ≤ EVDD0 < 2.7 V			-10.0	mA
		Total of all pins (When duty ≤ 70% Note 3)	2.4 V ≤ EVDD0 ≤ 5.5 V			-60.0	mA
	IOH2	Per pin for P20 to P27, P150 to P156	2.4 V ≤ VDD ≤ 5.5 V			-0.1 Note 2	mA
		Total of all pins (When duty ≤ 70% Note 3)	2.4 V ≤ VDD ≤ 5.5 V			-1.5	mA

**Note 1.** Value of current at which the device operation is guaranteed even if the current flows from the EVDD0, EVDD1, VDD pins to an output pin.

**Note 2.** Do not exceed the total current value.

**Note 3.** Specification under conditions where the duty factor ≤ 70%.

The output current value that has changed to the duty factor > 70% the duty ratio can be calculated with the following expression (when changing the duty factor from 70% to n%).

- Total output current of pins =  $(IOH \times 0.7)/(n \times 0.01)$

<Example> Where n = 80% and IOH = -10.0 mA

$$\text{Total output current of pins} = (-10.0 \times 0.7)/(80 \times 0.01) \approx -8.7 \text{ mA}$$

However, the current that is allowed to flow into one pin does not vary depending on the duty factor.

A current higher than the absolute maximum rating must not flow into one pin.

**Caution** P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, and P142 to P144 do not output high level in N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, VSS = EVSS0 = EVSS1 = 0 V)

(2/5)

Items	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Output current, low Note 1	IOL1	Per pin for P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147			8.5 Note 2	mA
		Per pin for P60 to P63			15.0 Note 2	mA
		Total of P00 to P04, P40 to P47, P102, P120, P130, P140 to P145 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V 2.7 V ≤ EVDD0 < 4.0 V 2.4 V ≤ EVDD0 < 2.7 V		40.0 15.0 9.0	mA mA mA
		Total of P05, P06, P10 to P17, P30, P31, P50 to P57, P60 to P67, P70 to P77, P80 to P87, P100, P101, P110, P111, P146, P147 (When duty ≤ 70% Note 3)	4.0 V ≤ EVDD0 ≤ 5.5 V 2.7 V ≤ EVDD0 < 4.0 V 2.4 V ≤ EVDD0 < 2.7 V		40.0 35.0 20.0	mA mA mA
		Total of all pins (When duty ≤ 70% Note 3)			80.0	mA
		Per pin for P20 to P27, P150 to P156			0.4 Note 2	mA
		Total of all pins (When duty ≤ 70% Note 3)	2.4 V ≤ VDD ≤ 5.5 V		5.0	mA

**Note 1.** Value of current at which the device operation is guaranteed even if the current flows from an output pin to the EVSS0, EVSS1, and VSS pins.

**Note 2.** Do not exceed the total current value.

**Note 3.** Specification under conditions where the duty factor ≤ 70%.

The output current value that has changed to the duty factor > 70% the duty ratio can be calculated with the following expression (when changing the duty factor from 70% to n%).

- Total output current of pins =  $(I_{OL} \times 0.7)/(n \times 0.01)$

<Example> Where n = 80% and IOL = 10.0 mA

$$\text{Total output current of pins} = (10.0 \times 0.7)/(80 \times 0.01) \approx 8.7 \text{ mA}$$

However, the current that is allowed to flow into one pin does not vary depending on the duty factor.

A current higher than the absolute maximum rating must not flow into one pin.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(3/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage, high	VIH1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	Normal input buffer	0.8 EV <sub>DD0</sub>		EV <sub>DD0</sub>	V
	VIH2	P01, P03, P04, P10, P14 to P17, P30, P43, P44, P50, P53 to P55, P80, P81, P142, P143	TTL input buffer 4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	2.2		EV <sub>DD0</sub>	V
			TTL input buffer 3.3 V ≤ EV <sub>DD0</sub> < 4.0 V	2.0		EV <sub>DD0</sub>	V
			TTL input buffer 2.4 V ≤ EV <sub>DD0</sub> < 3.3 V	1.5		EV <sub>DD0</sub>	V
	VIH3	P20 to P27, P150 to P156		0.7 V <sub>DD</sub>		V <sub>DD</sub>	V
	VIH4	P60 to P63		0.7 EV <sub>DD0</sub>		6.0	V
	VIH5	P121 to P124, P137, EXCLK, EXCLKS, RESET		0.8 V <sub>DD</sub>		V <sub>DD</sub>	V
Input voltage, low	VIL1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	Normal input buffer	0		0.2 EV <sub>DD0</sub>	V
	VIL2	P01, P03, P04, P10, P14 to P17, P30, P43, P44, P50, P53 to P55, P80, P81, P142, P143	TTL input buffer 4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	0		0.8	V
			TTL input buffer 3.3 V ≤ EV <sub>DD0</sub> < 4.0 V	0		0.5	V
			TTL input buffer 2.4 V ≤ EV <sub>DD0</sub> < 3.3 V	0		0.32	V
	VIL3	P20 to P27, P150 to P156		0		0.3 V <sub>DD</sub>	V
	VIL4	P60 to P63		0		0.3 EV <sub>DD0</sub>	V
	VIL5	P121 to P124, P137, EXCLK, EXCLKS, RESET		0		0.2 V <sub>DD</sub>	V

**Caution** The maximum value of VIH of pins P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, and P142 to P144 is EV<sub>DD0</sub>, even in the N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(4/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Output voltage, high	V <sub>OH1</sub>	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OH1</sub> = -3.0 mA	EV <sub>DD0</sub> - 0.7			V
				EV <sub>DD0</sub> - 0.6			V
				EV <sub>DD0</sub> - 0.5			V
	V <sub>OH2</sub>	P20 to P27, P150 to P156	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, I <sub>OH2</sub> = -100 μA	V <sub>DD</sub> - 0.5			V
Output voltage, low	V <sub>OL1</sub>	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P130, P140 to P147	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 8.5 mA			0.7	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 3.0 mA			0.6	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 1.5 mA			0.4	V
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL1</sub> = 0.6 mA			0.4	V
	V <sub>OL2</sub>	P20 to P27, P150 to P156	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, I <sub>OL2</sub> = 400 μA			0.4	V
	V <sub>OL3</sub>	P60 to P63	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 15.0 mA			2.0	V
			4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 5.0 mA			0.4	V
			2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 3.0 mA			0.4	V
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, I <sub>OL3</sub> = 2.0 mA			0.4	V

**Caution** P00, P02 to P04, P10, P11, P13 to P15, P17, P30, P43 to P45, P50 to P55, P71, P74, P80 to P82, P142 to P144 do not output high level in N-ch open-drain mode.

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, VSS = EVSS0 = EVSS1 = 0 V)

(5/5)

Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit	
Input leakage current, high	ILIH1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	VI = EVDD0			1	µA	
	ILIH2	P20 to P27, P137, P150 to P156, <u>RESET</u>	VI = VDD			1	µA	
	ILIH3	P121 to P124 (X1, X2, EXCLK, XT1, XT2, EXCLKS)	VI = VDD	In input port or external clock input		1	µA	
				In resonator connection		10	µA	
Input leakage current, low	ILIL1	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	VI = EVSS0			-1	µA	
	ILIL2	P20 to P27, P137, P150 to P156, <u>RESET</u>	VI = VSS			-1	µA	
	ILIL3	P121 to P124 (X1, X2, EXCLK, XT1, XT2, EXCLKS)	VI = VSS	In input port or external clock input		-1	µA	
				In resonator connection		-10	µA	
On-chip pull-up resistance	Ru	P00 to P06, P10 to P17, P30, P31, P40 to P47, P50 to P57, P64 to P67, P70 to P77, P80 to P87, P100 to P102, P110, P111, P120, P140 to P147	VI = EVSS0, In input port		10	20	100	kΩ

**Remark** Unless specified otherwise, the characteristics of alternate-function pins are the same as those of the port pins.

### 35.3.2 Supply current characteristics

#### (1) Flash ROM: 16 to 64 KB of 30- to 64-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions					MIN.	TYP.	MAX.	Unit
Supply current Note 1	IDD1	Operating mode	HS (high-speed main mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.4		mA
						V <sub>DD</sub> = 3.0 V		2.4		
		HS (high-speed main mode Note 5	f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.1			mA
						V <sub>DD</sub> = 3.0 V		2.1		
			f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.1	9.3		mA
						V <sub>DD</sub> = 3.0 V		5.1	9.3	
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.8	8.7		mA
						V <sub>DD</sub> = 3.0 V		4.8	8.7	
			f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.0	7.3		mA
						V <sub>DD</sub> = 3.0 V		4.0	7.3	
			f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		3.8	6.7		mA
						V <sub>DD</sub> = 3.0 V		3.8	6.7	
			f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		2.8	4.9		mA
						V <sub>DD</sub> = 3.0 V		2.8	4.9	
		HS (high-speed main mode Note 5	fm <sub>X</sub> = 20 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		3.3	5.7		mA
					Resonator connection		3.4	5.8		
			fm <sub>X</sub> = 20 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		3.3	5.7		
					Resonator connection		3.4	5.8		
			fm <sub>X</sub> = 10 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		2.0	3.4		
					Resonator connection		2.1	3.5		
			fm <sub>X</sub> = 10 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		2.0	3.4		
					Resonator connection		2.1	3.5		
		Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		4.7	6.1		μA
					Resonator connection		4.7	6.1		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		4.7	6.1		
					Resonator connection		4.7	6.1		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		4.8	6.7		
					Resonator connection		4.8	6.7		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		4.8	7.5		
					Resonator connection		4.8	7.5		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input			5.4	8.9		
				Resonator connection		5.4	8.9			
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +105°C	Normal operation	Square wave input			7.2	21.0		
				Resonator connection		7.3	21.1			

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub> and EV<sub>DD0</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub> or V<sub>SS</sub>, EV<sub>SS0</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is TA = 25°C

## (1) Flash ROM: 16 to 64 KB of 30- to 64-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>D0</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	IDD2 Note 2	HALT mode	HS (high-speed main) mode Note 7	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.80	4.36	mA
					V <sub>DD</sub> = 3.0 V		0.80	4.36	
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.49	3.67	
					V <sub>DD</sub> = 3.0 V		0.49	3.67	
				f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.62	3.42	
					V <sub>DD</sub> = 3.0 V		0.62	3.42	
				f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.4	2.85	
					V <sub>DD</sub> = 3.0 V		0.4	2.85	
				f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.37	2.08	
					V <sub>DD</sub> = 3.0 V		0.37	2.08	
		HS (high-speed main) mode Note 7	f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.28	2.45	mA	
				Resonator connection		0.40	2.57		
			f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.28	2.45		
				Resonator connection		0.40	2.57		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.19	1.28		
				Resonator connection		0.25	1.36		
		Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.25	0.57	μA	
				Resonator connection		0.44	0.76		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.30	0.57		
				Resonator connection		0.49	0.76		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.36	1.17		
				Resonator connection		0.59	1.36		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.49	1.97		
				Resonator connection		0.72	2.16		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +85°C	Square wave input		0.97	3.37		
				Resonator connection		1.16	3.56		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +105°C	Square wave input		3.20	17.10		
				Resonator connection		3.40	17.50		
	IDD3 Note 6	STOP mode Note 8	TA = -40°C			0.18	0.51	μA	
			TA = +25°C			0.24	0.51		
			TA = +50°C			0.29	1.10		
			TA = +70°C			0.41	1.90		
			TA = +85°C			0.90	3.30		
			TA = +105°C			3.10	17.00		

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub> and EV<sub>DD0</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub> or V<sub>SS</sub>, EV<sub>SS0</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

## (2) Flash ROM: 96 to 256 KB of 30- to 100-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions					MIN.	TYP.	MAX.	Unit
Supply current Note 1	I <sub>DD1</sub>	Operating mode HS (high-speed main) mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.6			mA
					V <sub>DD</sub> = 3.0 V		2.6			
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.3			
					V <sub>DD</sub> = 3.0 V		2.3			
		HS (high-speed main) mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.4	10.9		mA
					V <sub>DD</sub> = 3.0 V		5.4	10.9		
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.0	10.3		
					V <sub>DD</sub> = 3.0 V		5.0	10.3		
			f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.2	8.2		
					V <sub>DD</sub> = 3.0 V		4.2	8.2		
			f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.0	7.8		
					V <sub>DD</sub> = 3.0 V		4.0	7.8		
		HS (high-speed main) mode Note 5	f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		3.0	5.6		mA
					V <sub>DD</sub> = 3.0 V		3.0	5.6		
			f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		3.4	6.6		
					Resonator connection		3.6	6.7		
			f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		3.4	6.6		
					Resonator connection		3.6	6.7		
			f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		2.1	3.9		
					Resonator connection		2.2	4.0		
		Subsystem clock operation	f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		2.1	3.9		μA
					Resonator connection		2.2	4.0		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		4.9	7.1		
					Resonator connection		4.9	7.1		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		4.9	7.1		
					Resonator connection		4.9	7.1		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		5.1	8.8		μA
					Resonator connection		5.1	8.8		
			f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		5.5	10.5		
					Resonator connection		5.5	10.5		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input		6.5	14.5		μA
					Resonator connection		6.5	14.5		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +105°C	f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +105°C	Normal operation	Square wave input		13.0	58.0		μA
					Resonator connection		13.0	58.0		

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the 12-bit interval timer and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V@1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V@1 MHz to 16 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>H</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is T<sub>A</sub> = 25°C

## (2) Flash ROM: 96 to 256 KB of 30- to 100-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	I <sub>DD2</sub> Note 2	HALT mode HS (high-speed main) mode Note 7	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.79	4.86	mA	
				V <sub>DD</sub> = 3.0 V		0.79	4.86		
			f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.49	4.17		
				V <sub>DD</sub> = 3.0 V		0.49	4.17		
			f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.62	3.82		
				V <sub>DD</sub> = 3.0 V		0.62	3.82		
			f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.4	3.25		
				V <sub>DD</sub> = 3.0 V		0.4	3.25		
			f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.38	2.28		
				V <sub>DD</sub> = 3.0 V		0.38	2.28		
		HS (high-speed main) mode Note 7	f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.30	2.65	mA	
				Resonator connection		0.40	2.77		
			f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.30	2.65		
				Resonator connection		0.40	2.77		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.20	1.36		
				Resonator connection		0.25	1.46		
		Subsystem clock operation	f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.20	1.36	μA	
				Resonator connection		0.25	1.46		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.28	0.66		
				Resonator connection		0.47	0.85		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.34	0.66		
				Resonator connection		0.53	0.85		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.37	2.35		
				Resonator connection		0.56	2.54		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.61	4.08		
				Resonator connection		0.80	4.27		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +85°C	Square wave input		1.55	8.09		
				Resonator connection		1.74	8.28		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +105°C	Square wave input		6.00	51.00		
				Resonator connection		6.00	51.00		
I <sub>DD3</sub> Note 6	STOP mode Note 8	TA = -40°C				0.19	0.57	μA	
		TA = +25°C				0.25	0.57		
		TA = +50°C				0.33	2.26		
		TA = +70°C				0.52	3.99		
		TA = +85°C				1.46	8.00		
		TA = +105°C				5.50	50.00		

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

## (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions					MIN.	TYP.	MAX.	Unit	
Supply current Note 1	I <sub>DD1</sub>	Operating mode	HS (high-speed main) mode Note 5	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.9		mA	
						V <sub>DD</sub> = 3.0 V		2.9			
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Basic operation	V <sub>DD</sub> = 5.0 V		2.5			
						V <sub>DD</sub> = 3.0 V		2.5			
		HS (high-speed main) mode Note 5		f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		6.0	11.2	mA	
						V <sub>DD</sub> = 3.0 V		6.0	11.2		
				f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		5.5	10.6		
						V <sub>DD</sub> = 3.0 V		5.5	10.6		
				f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.7	8.6		
						V <sub>DD</sub> = 3.0 V		4.7	8.6		
				f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		4.4	8.2		
						V <sub>DD</sub> = 3.0 V		4.4	8.2		
		HS (high-speed main) mode Note 5		f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 3	Normal operation	V <sub>DD</sub> = 5.0 V		3.3	5.9	mA	
						V <sub>DD</sub> = 3.0 V		3.3	5.9		
				f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		3.7	6.8		
						Resonator connection		3.9	7.0		
				f <sub>MX</sub> = 20 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		3.7	6.8		
						Resonator connection		3.9	7.0		
				f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 5.0 V	Normal operation	Square wave input		2.3	4.1		
						Resonator connection		2.3	4.2		
		Subsystem clock operation		f <sub>MX</sub> = 10 MHz Note 2, V <sub>DD</sub> = 3.0 V	Normal operation	Square wave input		2.3	4.1	μA	
						Resonator connection		2.3	4.2		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = -40°C	Normal operation	Square wave input		5.2	7.7		
						Resonator connection		5.2	7.7		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +25°C	Normal operation	Square wave input		5.3	7.7		
						Resonator connection		5.3	7.7		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +50°C	Normal operation	Square wave input		5.5	10.6	μA	
						Resonator connection		5.5	10.6		
				f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +70°C	Normal operation	Square wave input		5.9	13.2		
						Resonator connection		6.0	13.2		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +85°C	Normal operation	Square wave input		6.8	17.5	μA	
						Resonator connection		6.9	17.5		
		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +105°C		f <sub>SUB</sub> = 32.768 kHz Note 4 TA = +105°C	Normal operation	Square wave input		15.5	77.8	μA	
						Resonator connection		15.5	77.8		

(Notes and Remarks are listed on the next page.)

**Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.

**Note 2.** When high-speed on-chip oscillator and subsystem clock are stopped.

**Note 3.** When high-speed system clock and subsystem clock are stopped.

**Note 4.** When high-speed on-chip oscillator and high-speed system clock are stopped. When AMPHS1 = 1 (Ultra-low power consumption oscillation). However, not including the current flowing into the 12-bit interval timer and watchdog timer.

**Note 5.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.

HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V@1 MHz to 32 MHz

2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V@1 MHz to 16 MHz

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>H</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation, temperature condition of the TYP. value is T<sub>A</sub> = 25°C

## (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions				MIN.	TYP.	MAX.	Unit
Supply current Note 1	I <sub>DD2</sub> Note 2	HALT mode HS (high-speed main) mode Note 7	f <sub>HOCO</sub> = 64 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.93	5.16	mA	
				V <sub>DD</sub> = 3.0 V		0.93	5.16		
		f <sub>HOCO</sub> = 32 MHz, f <sub>IH</sub> = 32 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.5	4.47			
				V <sub>DD</sub> = 3.0 V		0.5	4.47		
		f <sub>HOCO</sub> = 48 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.72	4.08			
				V <sub>DD</sub> = 3.0 V		0.72	4.08		
		f <sub>HOCO</sub> = 24 MHz, f <sub>IH</sub> = 24 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.42	3.51			
				V <sub>DD</sub> = 3.0 V		0.42	3.51		
		f <sub>HOCO</sub> = 16 MHz, f <sub>IH</sub> = 16 MHz Note 4	V <sub>DD</sub> = 5.0 V		0.39	2.38			
				V <sub>DD</sub> = 3.0 V		0.39	2.38		
		HS (high-speed main) mode Note 7	f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.31	2.83	mA	
				Resonator connection		0.41	2.92		
			f <sub>MX</sub> = 20 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.31	2.83		
				Resonator connection		0.41	2.92		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 5.0 V	Square wave input		0.21	1.46		
				Resonator connection		0.26	1.57		
			f <sub>MX</sub> = 10 MHz Note 3, V <sub>DD</sub> = 3.0 V	Square wave input		0.21	1.46		
				Resonator connection		0.26	1.57		
		Subsystem clock operation	f <sub>SUB</sub> = 32.768 kHz Note 5, TA = -40°C	Square wave input		0.31	0.76	μA	
				Resonator connection		0.50	0.95		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +25°C	Square wave input		0.38	0.76		
				Resonator connection		0.57	0.95		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +50°C	Square wave input		0.47	3.59		
				Resonator connection		0.70	3.78		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +70°C	Square wave input		0.80	6.20		
				Resonator connection		1.00	6.39		
			f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +85°C	Square wave input		1.65	10.56		
				Resonator connection		1.84	10.75		
		I <sub>DD3</sub> Note 6	f <sub>SUB</sub> = 32.768 kHz Note 5, TA = +105°C	Square wave input		8.00	65.7	μA	
				Resonator connection		8.00	65.7		
			TA = -40°C			0.19	0.63		
			TA = +25°C			0.30	0.63		
			TA = +50°C			0.41	3.47		
			TA = +70°C			0.80	6.08		
		STOP mode Note 8	TA = +85°C			1.53	10.44		
			TA = +105°C			6.50	67.14		

(Notes and Remarks are listed on the next page.)

- Note 1.** Total current flowing into V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, including the input leakage current flowing when the level of the input pin is fixed to V<sub>DD</sub>, EV<sub>DD0</sub>, and EV<sub>DD1</sub>, or V<sub>SS</sub>, EV<sub>VSS0</sub>, and EV<sub>VSS1</sub>. The values below the MAX. column include the peripheral operation current. However, not including the current flowing into the A/D converter, D/A converter, comparator, LVD circuit, I/O port, and on-chip pull-up/pull-down resistors and the current flowing during data flash rewrite.
- Note 2.** During HALT instruction execution by flash memory.
- Note 3.** When high-speed on-chip oscillator and subsystem clock are stopped.
- Note 4.** When high-speed system clock and subsystem clock are stopped.
- Note 5.** When high-speed on-chip oscillator and high-speed system clock are stopped. When RTCLPC = 1 and setting ultra-low current consumption (AMPHS1 = 1). The current flowing into the RTC is included. However, not including the current flowing into the 12-bit interval timer and watchdog timer.
- Note 6.** Not including the current flowing into the RTC, 12-bit interval timer, and watchdog timer.
- Note 7.** Relationship between operation voltage width, operation frequency of CPU and operation mode is as below.  
HS (high-speed main) mode: 2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 32 MHz  
2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V @ 1 MHz to 16 MHz
- Note 8.** Regarding the value for current to operate the subsystem clock in STOP mode, refer to that in HALT mode.

**Remark 1.** f<sub>MX</sub>: High-speed system clock frequency (X1 clock oscillation frequency or external main system clock frequency)

**Remark 2.** f<sub>HOCO</sub>: High-speed on-chip oscillator clock frequency (64 MHz max.)

**Remark 3.** f<sub>IH</sub>: High-speed on-chip oscillator clock frequency (32 MHz max.)

**Remark 4.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 5.** Except subsystem clock operation and STOP mode, temperature condition of the TYP. value is TA = 25°C

#### (4) Peripheral Functions (Common to all products)

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Low-speed on-chip oscillator operating current	I <sub>FIL</sub> Note 1				0.20		µA
RTC operating current	I <sub>RTC</sub> Notes 1, 2, 3				0.02		µA
12-bit interval timer operating current	I <sub>IT</sub> Notes 1, 2, 4				0.02		µA
Watchdog timer operating current	I <sub>WDT</sub> Notes 1, 2, 5	f <sub>L</sub> = 15 kHz			0.22		µA
A/D converter operating current	I <sub>AADC</sub> Notes 1, 6	When conversion at maximum speed	Normal mode, AV <sub>REFP</sub> = V <sub>DD</sub> = 5.0 V		1.3	1.7	mA
			Low voltage mode, AV <sub>REFP</sub> = V <sub>DD</sub> = 3.0 V		0.5	0.7	mA
A/D converter reference voltage current	I <sub>AADREF</sub> Note 1				75.0		µA
Temperature sensor operating current	I <sub>TMPS</sub> Note 1				75.0		µA
D/A converter operating current	I <sub>DAAC</sub> Notes 1, 11, 13	Per D/A converter channel				1.5	mA
Comparator operating current	I <sub>CMP</sub> Notes 1, 12, 13	V <sub>DD</sub> = 5.0 V, Regulator output voltage = 2.1 V	Window mode		12.5		µA
			Comparator high-speed mode		6.5		µA
			Comparator low-speed mode		1.7		µA
		V <sub>DD</sub> = 5.0 V, Regulator output voltage = 1.8 V	Window mode		8.0		µA
			Comparator high-speed mode		4.0		µA
			Comparator low-speed mode		1.3		µA
LVD operating current	I <sub>LVD</sub> Notes 1, 7				0.08		µA
Self-programming operating current	I <sub>FSPI</sub> Notes 1, 9				2.50	12.20	mA
BGO operating current	I <sub>BGO</sub> Notes 1, 8				2.50	12.20	mA
SNOOZE operating current	I <sub>SNOZ</sub> Note 1	ADC operation	The mode is performed Note 10		0.50	1.10	mA
			The A/D conversion operations are performed, Low voltage mode, AV <sub>REFP</sub> = V <sub>DD</sub> = 3.0 V		1.20	2.04	
		CSI/UART operation			0.70	1.54	
		DTC operation			3.10		

**Note 1.** Current flowing to V<sub>DD</sub>.

**Note 2.** When high speed on-chip oscillator and high-speed system clock are stopped.

**Note 3.** Current flowing only to the real-time clock (RTC) (excluding the operating current of the low-speed on-chip oscillator and the XT1 oscillator). The supply current of the RL78 microcontrollers is the sum of the values of either I<sub>DD1</sub> or I<sub>DD2</sub>, and I<sub>RTC</sub>, when the real-time clock operates in operation mode or HALT mode. When the low-speed on-chip oscillator is selected, I<sub>FIL</sub> should be added. I<sub>DD2</sub> subsystem clock operation includes the operational current of the real-time clock.

**Note 4.** Current flowing only to the 12-bit interval timer (excluding the operating current of the low-speed on-chip oscillator and the XT1 oscillator). The supply current of the RL78 microcontrollers is the sum of the values of either I<sub>DD1</sub> or I<sub>DD2</sub>, and I<sub>IT</sub>, when the 12-bit interval timer operates in operation mode or HALT mode. When the low-speed on-chip oscillator is selected, I<sub>FIL</sub> should be added.

- Note 5.** Current flowing only to the watchdog timer (including the operating current of the low-speed on-chip oscillator). The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2 or IDD3 and I<sub>WDT</sub> when the watchdog timer is in operation.
- Note 6.** Current flowing only to the A/D converter. The supply current of the RL78 microcontrollers is the sum of IDD1 or IDD2 and I<sub>AADC</sub> when the A/D converter operates in an operation mode or the HALT mode.
- Note 7.** Current flowing only to the LVD circuit. The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2 or IDD3 and I<sub>LVD</sub> when the LVD circuit is in operation.
- Note 8.** Current flowing during programming of the data flash.
- Note 9.** Current flowing during self-programming.
- Note 10.** For shift time to the SNOOZE mode, see **23.3.3 SNOOZE mode**.
- Note 11.** Current flowing only to the D/A converter. The supply current of the RL78 microcontrollers is the sum of IDD1 or IDD2 and I<sub>DAC</sub> when the D/A converter operates in an operation mode or the HALT mode.
- Note 12.** Current flowing only to the comparator circuit. The supply current of the RL78 microcontrollers is the sum of IDD1, IDD2, or IDD3 and I<sub>CMP</sub> when the comparator circuit is in operation.
- Note 13.** A comparator and D/A converter are provided in products with 96 KB or more code flash memory.

**Remark 1.** f<sub>IL</sub>: Low-speed on-chip oscillator clock frequency

**Remark 2.** f<sub>SUB</sub>: Subsystem clock frequency (XT1 clock oscillation frequency)

**Remark 3.** f<sub>CLK</sub>: CPU/peripheral hardware clock frequency

**Remark 4.** Temperature condition of the TYP. value is TA = 25°C

### 35.4 AC Characteristics

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(1/2)

Items	Symbol	Conditions			MIN.	TYP.	MAX.	Unit	
Instruction cycle (minimum instruction execution time)	T <sub>CY</sub>	Main system clock (f <sub>MAIN</sub> ) operation	HS (high-speed main) mode	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.03125		1	μs	
				2.4 V ≤ V <sub>DD</sub> < 2.7 V	0.0625		1	μs	
		Subsystem clock (f <sub>SUB</sub> ) operation		2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V	28.5	30.5	31.3	μs	
		In the self-programming mode	HS (high-speed main) mode	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	0.03125		1	μs	
				2.4 V ≤ V <sub>DD</sub> < 2.7 V	0.0625		1	μs	
External system clock frequency	f <sub>EX</sub>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			1.0		20.0	MHz	
		2.4 V ≤ V <sub>DD</sub> ≤ 2.7 V			1.0		16.0	MHz	
	f <sub>EXS</sub>				32		35	kHz	
External system clock input high-level width, low-level width	t <sub>EXH</sub> , t <sub>EXL</sub>	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			24			ns	
		2.4 V ≤ V <sub>DD</sub> ≤ 2.7 V			30			ns	
	t <sub>EXHS</sub> , t <sub>EXLS</sub>				13.7			μs	
TI00 to TI03, TI10 to TI13 input high-level width, low-level width	t <sub>TIH</sub> , t <sub>TIL</sub>				1/f <sub>MCK</sub> + 10 Note			ns	
Timer RJ input cycle	fc	TRJIO	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		100			ns	
			2.4 V ≤ EV <sub>D0</sub> < 2.7 V		300			ns	
Timer RJ input high-level width, low-level width	t <sub>TJH</sub> , t <sub>TJL</sub>	TRJIO	2.7 V ≤ EV <sub>D0</sub> ≤ 5.5 V		40			ns	
			2.4 V ≤ EV <sub>D0</sub> < 2.7 V		120			ns	

**Note** The following conditions are required for low voltage interface when EV<sub>D0</sub> < V<sub>DD</sub>  
2.4 V ≤ EV<sub>D0</sub> < 2.7 V: MIN. 125 ns

**Remark** f<sub>MCK</sub>: Timer array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of timer mode register mn (TMRmn). m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3))

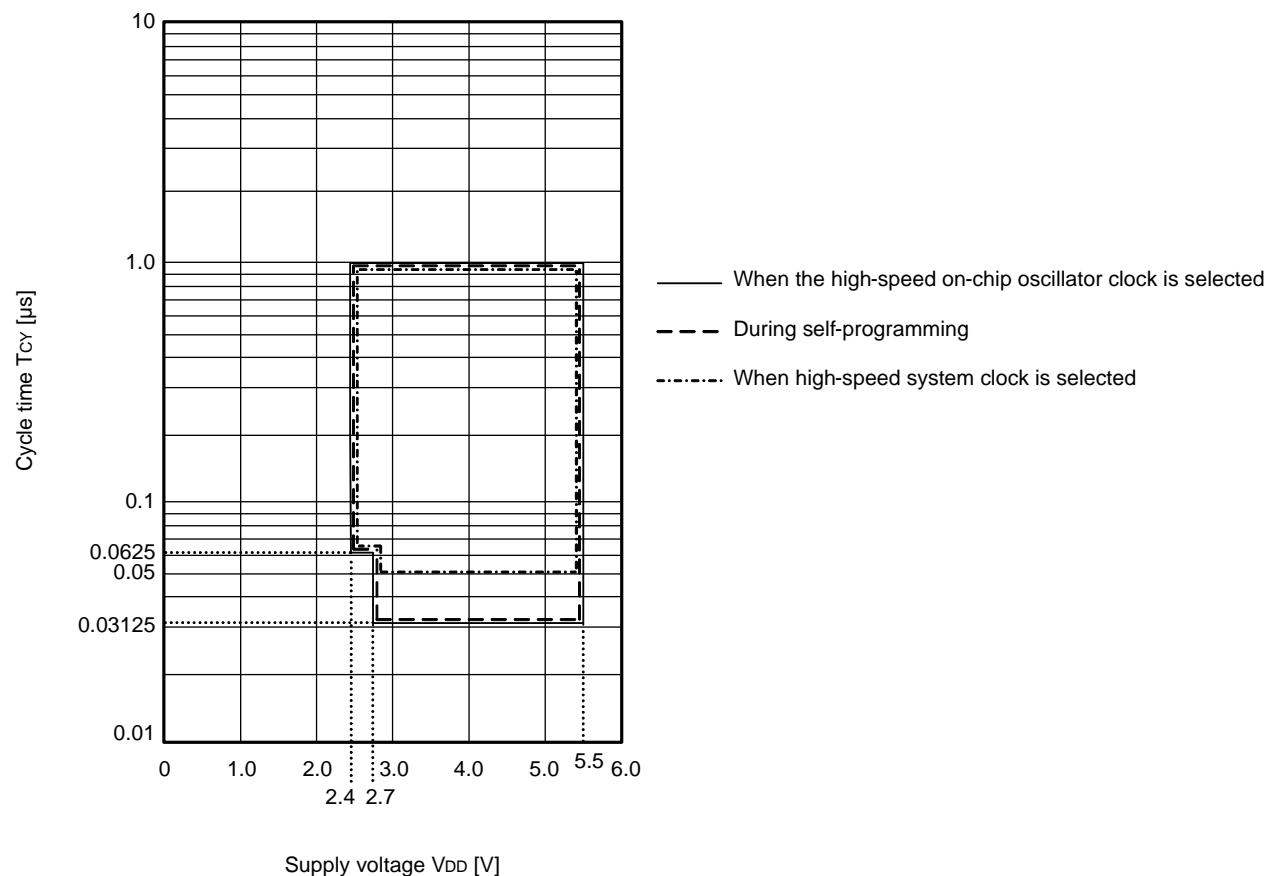
(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, VSS = EVSS0 = EVSS1 = 0 V)

(2/2)

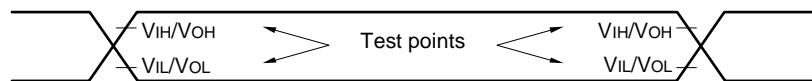
Items	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Timer RD input high-level width, low-level width	tTDIH, tTDIL	TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1, TRDIODC0, TRDIODC1, TRDIOD0, TRDIOD1		3/fCLK			ns
Timer RD forced cutoff signal input low-level width	tTDSIL	P130/INTP0	2MHz < fCLK ≤ 32 MHz	1			μs
			fCLK ≤ 2 MHz	1/fCLK + 1			
Timer RG input high-level width, low-level width	tTGIIH, tTGIL	TRGIOA, TRGIOB		2.5/fCLK			ns
TO00 to TO03, TO10 to TO13, TRJIO0, TRJOO, TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1, TRDIODC0, TRDIODC1, TRDIOD0, TRDIOD1, TRGIOA, TRGIOB output frequency	fro	HS (high-speed main) mode	4.0 V ≤ EVDD0 ≤ 5.5 V			16	MHz
			2.7 V ≤ EVDD0 < 4.0 V			8	MHz
			2.4 V ≤ EVDD0 < 2.7 V			4	MHz
PCLBUZ0, PCLBUZ1 output frequency	fPCL	HS (high-speed main) mode	4.0 V ≤ EVDD0 ≤ 5.5 V			16	MHz
			2.7 V ≤ EVDD0 < 4.0 V			8	MHz
			2.4 V ≤ EVDD0 < 2.7 V			4	MHz
Interrupt input high-level width, low-level width	tINTH, tINTL	INTP0	2.4 V ≤ VDD ≤ 5.5 V	1			μs
		INTP1 to INTP11	2.4 V ≤ EVDD0 ≤ 5.5 V	1			μs
Key interrupt input low-level width	tKR	KR0 to KR7	2.4 V ≤ EVDD0 ≤ 5.5 V	250			ns
RESET low-level width	tRSI			10			μs

## Minimum Instruction Execution Time during Main System Clock Operation

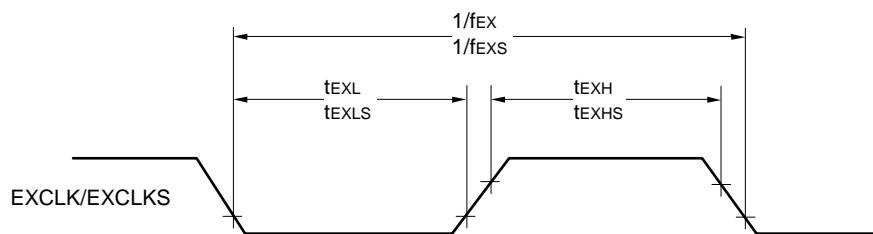
TCY vs VDD (HS (high-speed main) mode)



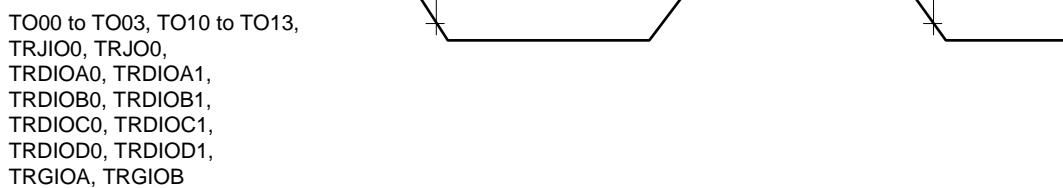
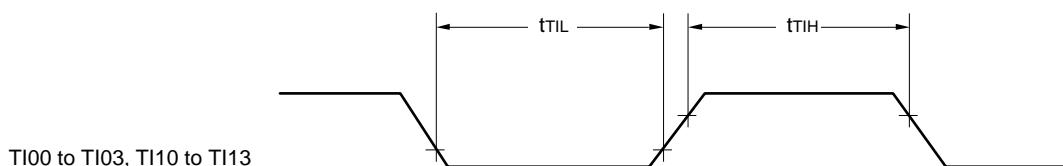
## AC Timing Test Points

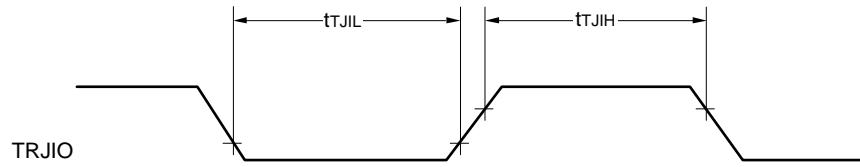


## External System Clock Timing

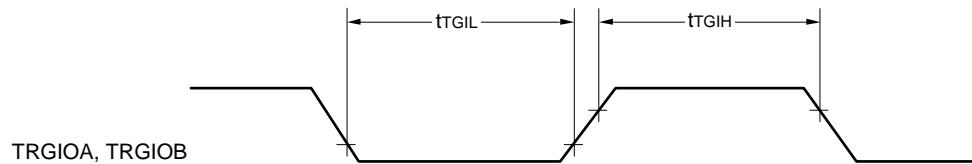
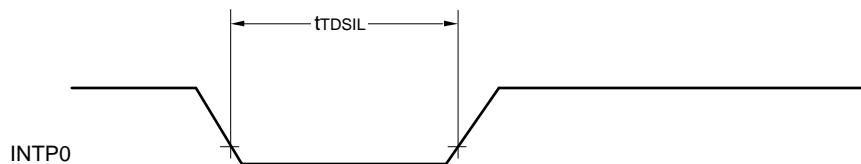


## TI/TO Timing

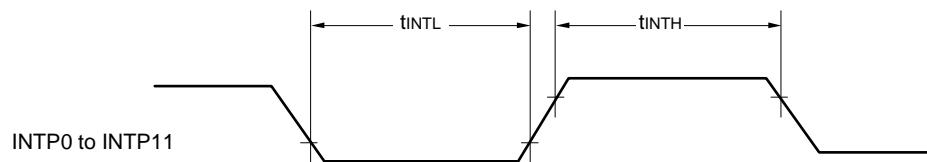




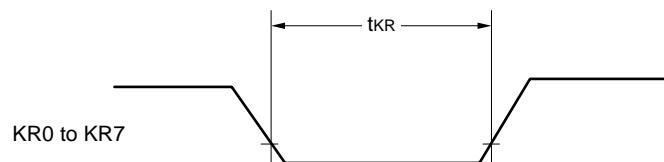
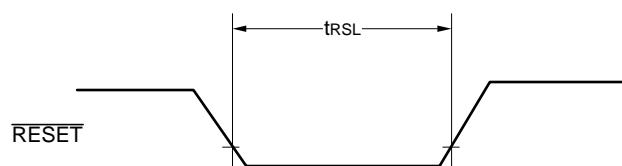
TRDIOA0, TRDIOA1, TRDIOB0, TRDIOB1,  
TRDIOC0, TRDIOC1, TRDIOD0, TRDIOD1



## Interrupt Request Input Timing

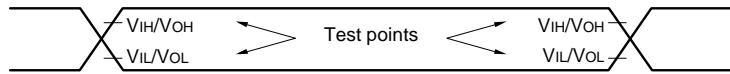


## Key Interrupt Input Timing

RESET Input Timing

## 35.5 Peripheral Functions Characteristics

AC Timing Test Points



### 35.5.1 Serial array unit

#### (1) During communication at same potential (UART mode)

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions	HS (high-speed main) Mode		Unit
			MIN.	MAX.	
Transfer rate Note 1		2.4 V ≤ EVDD0 ≤ 5.5 V Theoretical value of the maximum transfer rate fMCK = fCLK Note 3		fMCK/12 Note 2	bps
				2.6	Mbps

**Note 1.** Transfer rate in the SNOOZE mode is 4800 bps only.

However, the SNOOZE mode cannot be used when FRQSEL4 = 1.

**Note 2.** The following conditions are required for low voltage interface when EVDD0 < VDD.

2.4 V ≤ EVDD0 < 2.7 V: MAX. 1.3 Mbps

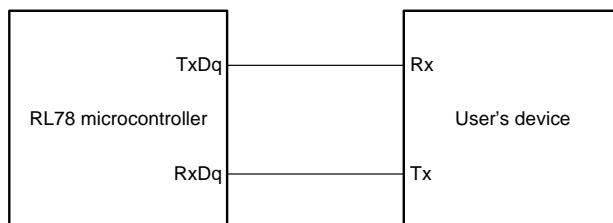
**Note 3.** The maximum operating frequencies of the CPU/peripheral hardware clock (fCLK) are:

HS (high-speed main) mode: 32 MHz (2.7 V ≤ VDD ≤ 5.5 V)

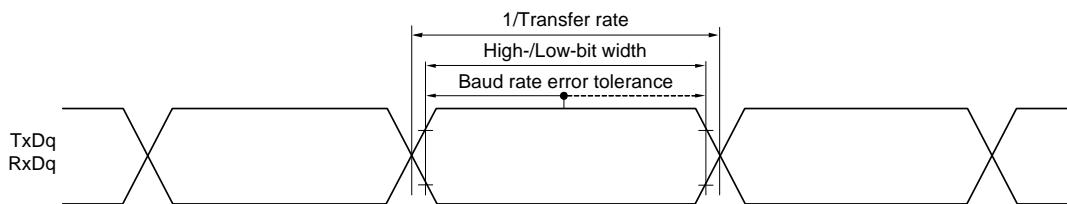
16 MHz (2.4 V ≤ VDD ≤ 5.5 V)

**Caution** Select the normal input buffer for the RxDq pin and the normal output mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg).

UART mode connection diagram (during communication at same potential)



UART mode bit width (during communication at same potential) (reference)



**Remark 1.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)

**Remark 2.** fMCK: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

## (2) During communication at same potential (CSI mode) (master mode, SCKp... internal clock output)

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

Parameter	Symbol	Conditions		HS (high-speed main) mode		Unit
				MIN.	MAX.	
SCKp cycle time	tkCY1	tkCY1 ≥ 4/fCLK	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	250		ns
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	500		ns
SCKp high-/low-level width	tkH1, tkL1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		tkCY1/2 - 24		ns
		2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		tkCY1/2 - 36		ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		tkCY1/2 - 76		ns
Slp setup time (to SCKp↑) Note 1	tSIK1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		66		ns
		2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		66		ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V		113		ns
Slp hold time (from SCKp↓) Note 2	tSIH1			38		ns
Delay time from SCKp↓ to SOp output Note 3	tKS01	C = 30 pF Note 4			50	ns

**Note 1.** When DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 0, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 1. The Slp setup time becomes “to SCKp↓” when DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 1, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 0.

**Note 2.** When DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 0, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 1. The Slp hold time becomes “from SCKp↓” when DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 1, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 0.

**Note 3.** When DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 0, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 1. The delay time to SOp output becomes “from SCKp↑” when DAP<sub>Mn</sub> = 0 and CKP<sub>Mn</sub> = 1, or DAP<sub>Mn</sub> = 1 and CKP<sub>Mn</sub> = 0.

**Note 4.** C is the load capacitance of the SCKp and SOp output lines.

**Caution** Select the normal input buffer for the Slp pin and the normal output mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), g: PIM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** fmCK: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

## (3) During communication at same potential (CSI mode) (slave mode, SCKp... external clock input)

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(1/2)

Parameter	Symbol	Conditions		HS (high-speed main) mode	Unit
		MIN.	MAX.		
SCKp cycle time Note 5	tkCY2	4.0 V ≤ EVDD0 ≤ 5.5 V	20 MHz < fmck fmck ≤ 20 MHz	16/fmck 12/fmck	ns ns
		2.7 V ≤ EVDD0 ≤ 5.5 V	16 MHz < fmck fmck ≤ 16 MHz	16/fmck 12/fmck	ns ns
		2.4 V ≤ EVDD0 ≤ 5.5 V		12/fmck and 1000	ns
SCKp high-/low-level width	tkH2, tkL2	4.0 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 14	ns
		2.7 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 16	ns
		2.4 V ≤ EVDD0 ≤ 5.5 V		tkCY2/2 - 36	ns
Slp setup time (to SCKp↑) Note 1	tsIK2	2.7 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 40	ns
		2.4 V ≤ EVDD0 ≤ 5.5 V		1/fmck + 60	ns
Slp hold time (from SCKp↑) Note 2	tksi2			1/fmck + 62	ns
Delay time from SCKp↓ to SOp output Note 3	tksO2	C = 30 pF Note 4	2.7 V ≤ EVDD0 ≤ 5.5 V		2/fmck + 66 ns
			2.4 V ≤ EVDD0 ≤ 5.5 V		2/fmck + 113 ns

**Note 1.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes "to SCKp↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 2.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes "from SCKp↓" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes "from SCKp↑" when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Note 4.** C is the load capacitance of the SOp output lines.

**Note 5.** The maximum transfer rate when using the SNOOZE mode is 1 Mbps.

**Caution** Select the normal input buffer for the Slp pin and SCKp pin and the normal output mode for the SOp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31), m: Unit number (m = 0, 1),  
n: Channel number (n = 0 to 3), g: PIM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number,  
n: Channel number (mn = 00 to 03, 10 to 13))

## (3) During communication at same potential (CSI mode) (slave mode, SCKp... external clock input)

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

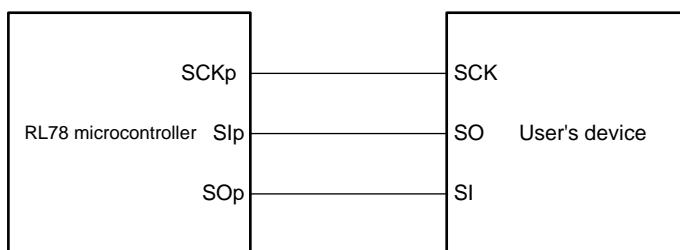
(2/2)

Parameter	Symbol	Conditions		HS (high-speed main) mode	Unit
		MIN.	MAX.		
SSI00 setup time	t <sub>SSI00</sub>	DAPmn = 0	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	240	ns
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	400	ns
		DAPmn = 1	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 240	ns
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 400	ns
SSI00 hold time	t <sub>kSSI00</sub>	DAPmn = 0	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 240	ns
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	1/fMCK + 400	ns
		DAPmn = 1	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	240	ns
			2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V	400	ns

**Caution** Select the normal input buffer for the Slp pin and SCKp pin and the normal output mode for the SOOp pin by using port input mode register g (PIMg) and port output mode register g (POMg).

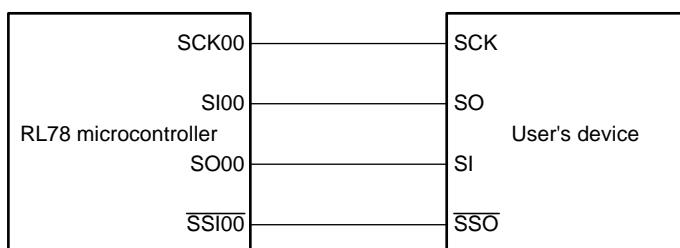
**Remark** p: CSI number (p = 00), m: Unit number (m = 0), n: Channel number (n = 0), g: PIM number (g = 3, 5)

## CSI mode connection diagram (during communication at same potential)



## CSI mode connection diagram (during communication at same potential)

(Slave Transmission of slave select input function (CSI00))

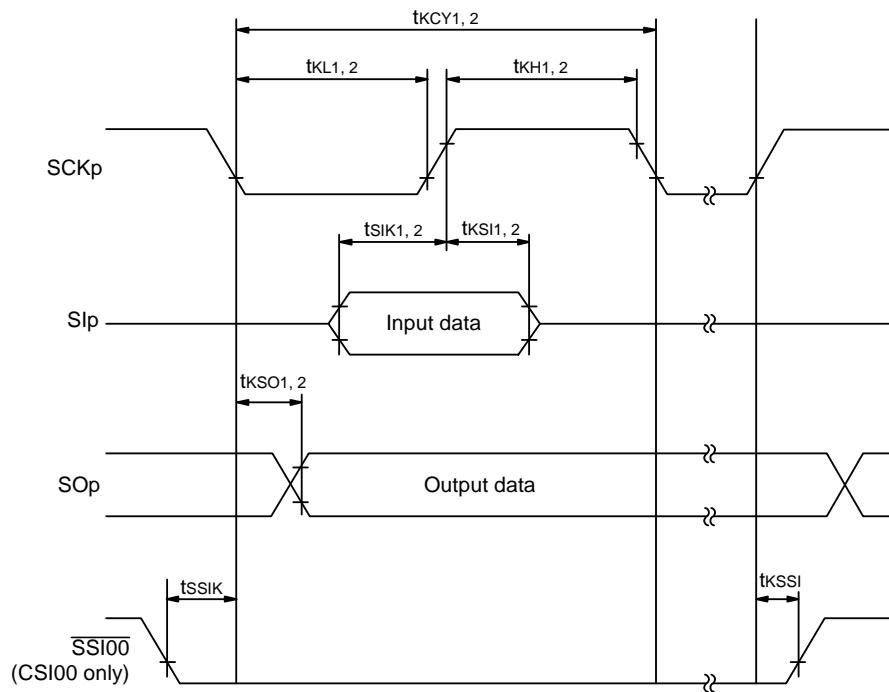


**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)

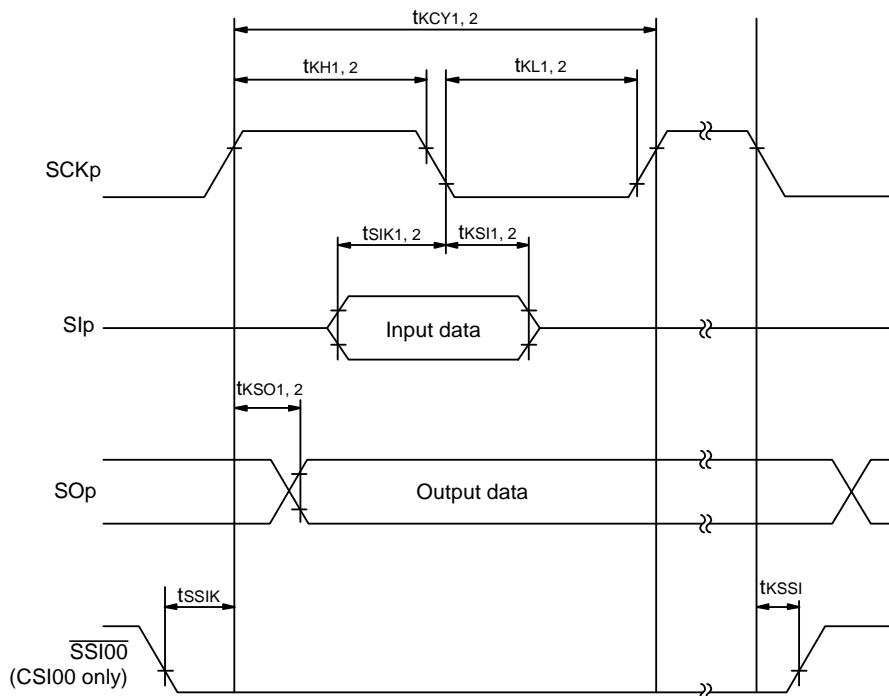
**Remark 2.** m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13)

**CSI mode serial transfer timing (during communication at same potential)**

(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)

**CSI mode serial transfer timing (during communication at same potential)**

(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)

**Remark 1.** p: CSI number (p = 00, 01, 10, 11, 20, 21, 30, 31)**Remark 2.** m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13)

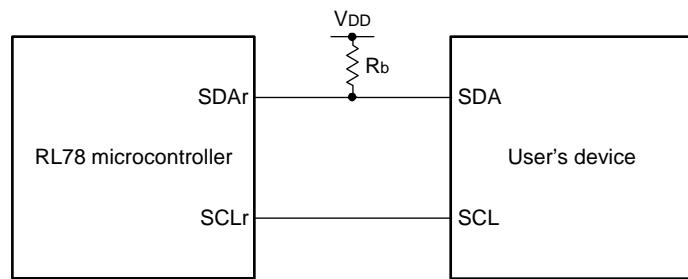
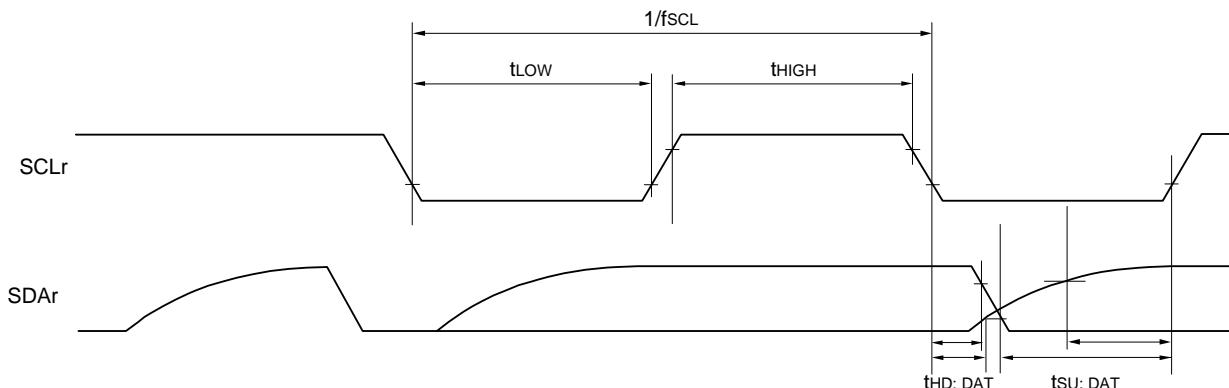
**(4) During communication at same potential (simplified I<sup>2</sup>C mode)**(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
SCL <sub>r</sub> clock frequency	f <sub>SCL</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		400 Note 1	kHz
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ		100 Note 1	kHz
Hold time when SCL <sub>r</sub> = "L"	t <sub>LOW</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1200		ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	4600		ns
Hold time when SCL <sub>r</sub> = "H"	t <sub>HIGH</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1200		ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	4600		ns
Data setup time (reception)	t <sub>SU: DAT</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/f <sub>MCK</sub> + 220 Note 2		ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	1/f <sub>MCK</sub> + 580 Note 2		ns
Data hold time (transmission)	t <sub>HD: DAT</sub>	2.7 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	770	ns
		2.4 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 3 kΩ	0	1420	ns

**Note 1.** The value must also be equal to or less than f<sub>MCK</sub>/4.**Note 2.** Set the f<sub>MCK</sub> value to keep the hold time of SCL<sub>r</sub> = "L" and SCL<sub>r</sub> = "H".

**Caution** Select the normal input buffer and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SDAr pin and the normal output mode for the SCL<sub>r</sub> pin by using port input mode register g (PIMg) and port output mode register h (POMh).

(Remarks are listed on the next page.)

**Simplified I<sup>2</sup>C mode connection diagram (during communication at same potential)****Simplified I<sup>2</sup>C mode serial transfer timing (during communication at same potential)**

**Remark 1.**  $R_b[\Omega]$ : Communication line (SDAr) pull-up resistance,  $C_b[F]$ : Communication line (SDAr, SCLr) load capacitance

**Remark 2.** r: IIC number ( $r = 00, 01, 10, 11, 20, 21, 30, 31$ ), g: PIM number ( $g = 0, 1, 3 \text{ to } 5, 14$ ),

h: POM number ( $h = 0, 1, 3 \text{ to } 5, 7, 14$ )

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number ( $m = 0, 1$ ),

n: Channel number ( $n = 0 \text{ to } 3$ ), mn = 00 to 03, 10 to 13)

## (5) Communication at different potential (1.8 V, 2.5 V, 3 V) (UART mode)

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>D0</sub> = EV<sub>D1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>S0</sub> = EV<sub>S1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
Transfer rate	reception	4.0 V ≤ EV <sub>D0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V		fmck/12 Note 1	bps
		Theoretical value of the maximum transfer rate fmck = fCLK Note 3		2.6	Mbps
		2.7 V ≤ EV <sub>D0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V		fmck/12 Note 1	bps
		Theoretical value of the maximum transfer rate fmck = fCLK Note 3		2.6	Mbps
		2.4 V ≤ EV <sub>D0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V		fmck/12 Notes 1, 2	bps
		Theoretical value of the maximum transfer rate fmck = fCLK Note 3		2.6	Mbps

**Note 1.** Transfer rate in the SNOOZE mode is 4800 bps only.

However, the SNOOZE mode cannot be used when FRQSEL4 = 1.

**Note 2.** The following conditions are required for low voltage interface when EV<sub>D0</sub> < V<sub>DD</sub>.2.4 V ≤ EV<sub>D0</sub> < 2.7 V: MAX. 1.3 Mbps**Note 3.** The maximum operating frequencies of the CPU/peripheral hardware clock (fCLK) are:HS (high-speed main) mode: 32 MHz (2.7 V ≤ V<sub>DD</sub> ≤ 5.5 V)16 MHz (2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V)**Caution** Select the TTL input buffer for the RxDq pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>D0</sub> tolerance (for the 64- to 100-pin products)) mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.**Remark 1.** V<sub>b</sub> [V]: Communication line voltage**Remark 2.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number,

n: Channel number (mn = 00 to 03, 10 to 13)

**Remark 4.** UART2 cannot communicate at different potential when bit 1 (PIOR01) of peripheral I/O redirection register 0 (PIOR0) is 1.

## (5) Communication at different potential (1.8 V, 2.5 V, 3 V) (UART mode)

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(2/2)

Parameter	Symbol	Conditions		HS (high-speed main) mode	Unit
		MIN.	MAX.		
Transfer rate	transmission	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V		Note 1	bps
		Theoretical value of the maximum transfer rate C <sub>b</sub> = 50 pF, R <sub>b</sub> = 1.4 kΩ, V <sub>b</sub> = 2.7 V		2.6 Note 2	Mbps
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V		Note 3	bps
		Theoretical value of the maximum transfer rate C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ, V <sub>b</sub> = 2.3 V		1.2 Note 4	Mbps
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V		Note 5	bps
		Theoretical value of the maximum transfer rate C <sub>b</sub> = 50 pF, R <sub>b</sub> = 5.5 kΩ, V <sub>b</sub> = 1.6 V		0.43 Note 6	Mbps

**Note 1.** The smaller maximum transfer rate derived by using fmck/12 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 4.0 V ≤ EV<sub>DD0</sub> ≤ 5.5 V and 2.7 V ≤ V<sub>b</sub> ≤ 4.0 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.2}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 2.** This value as an example is calculated when the conditions described in the "Conditions" column are met.  
Refer to **Note 1** above to calculate the maximum transfer rate under conditions of the customer.

**Note 3.** The smaller maximum transfer rate derived by using fmck/12 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 2.7 V ≤ EV<sub>DD0</sub> < 4.0 V and 2.3 V ≤ V<sub>b</sub> ≤ 2.7 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\} \times 3} \text{ [bps]}$$

$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{2.0}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 4.** This value as an example is calculated when the conditions described in the "Conditions" column are met.  
Refer to **Note 3** above to calculate the maximum transfer rate under conditions of the customer.

**Note 5.** The smaller maximum transfer rate derived by using fmck/12 or the following expression is the valid maximum transfer rate.

Expression for calculating the transfer rate when 2.4 V ≤ EVDD0 < 3.3 V and 1.6 V ≤ Vb ≤ 2.0 V

$$\text{Maximum transfer rate} = \frac{1}{\{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\} \times 3} \text{ [bps]}$$

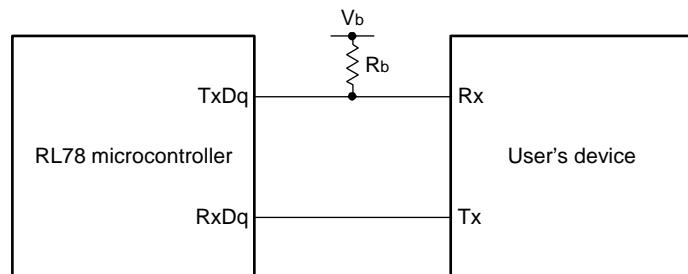
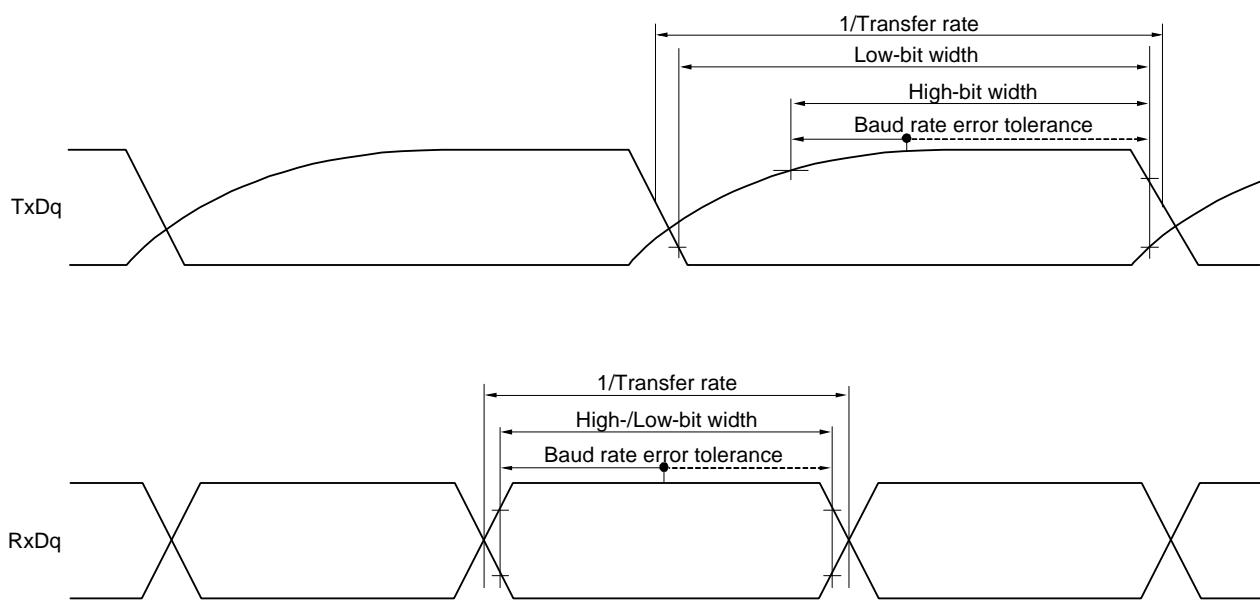
$$\text{Baud rate error (theoretical value)} = \frac{\frac{1}{\text{Transfer rate} \times 2} - \{-C_b \times R_b \times \ln(1 - \frac{1.5}{V_b})\}}{\left(\frac{1}{\text{Transfer rate}}\right) \times \text{Number of transferred bits}} \times 100 [\%]$$

\* This value is the theoretical value of the relative difference between the transmission and reception sides

**Note 6.** This value as an example is calculated when the conditions described in the “Conditions” column are met.  
Refer to **Note 5** above to calculate the maximum transfer rate under conditions of the customer.

**Caution** Select the TTL input buffer for the RxDq pin and the N-ch open drain output (VDD tolerance (for the 30- to 52-pin products)/EVDD tolerance (for the 64- to 100-pin products)) mode for the TxDq pin by using port input mode register g (PIMg) and port output mode register g (POMg). For VIH and Vil, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**UART mode connection diagram (during communication at different potential)****UART mode bit width (during communication at different potential) (reference)**

**Remark 1.**  $R_b[\Omega]$ : Communication line (TxDq) pull-up resistance,

$C_b[F]$ : Communication line (TxDq) load capacitance,  $V_b[V]$ : Communication line voltage

**Remark 2.** q: UART number (q = 0 to 3), g: PIM and POM number (g = 0, 1, 5, 14)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn).

m: Unit number, n: Channel number (mn = 00 to 03, 10 to 13))

**Remark 4.** UART2 cannot communicate at different potential when bit 1 (PIOR01) of peripheral I/O redirection register 0 (PIOR0) is 1.

**(6) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

(1/3)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
SCKp cycle time	t <sub>KCY1</sub>	t <sub>KCY1</sub> ≥ 4/f <sub>CLK</sub>	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	600	ns
			2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	1000	ns
			2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	2300	ns
SCKp high-level width	t <sub>KH1</sub>	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	t <sub>KCY1/2</sub> - 150		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	t <sub>KCY1/2</sub> - 340		ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	t <sub>KCY1/2</sub> - 916		ns
SCKp low-level width	t <sub>KL1</sub>	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	t <sub>KCY1/2</sub> - 24		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	t <sub>KCY1/2</sub> - 36		ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	t <sub>KCY1/2</sub> - 100		ns

**Caution** Select the TTL input buffer for the S<sub>l</sub>p pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SO<sub>p</sub> pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed two pages after the next page.)

**(6) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V) (2/3)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
Slp setup time (to SCKp↑) <sup>Note</sup>	tsIK1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	162		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	354		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	958		ns
Slp hold time (from SCKp↑) <sup>Note</sup>	tKSI1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	38		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	38		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	38		ns
Delay time from SCKp↓ to SOp output <sup>Note</sup>	tKS01	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ		200	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ		390	ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ		966	ns

**Note** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.

**Caution** Select the TTL input buffer for the Slp pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SOp pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the page after the next page.)

**(6) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (master mode, SCKp... internal clock output)**

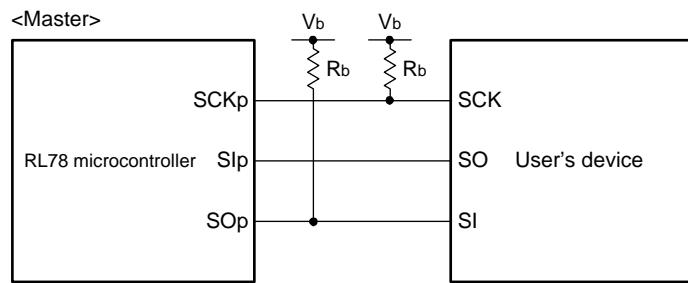
(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V) (3/3)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
Slp setup time (to SCKp <sub>↓</sub> ) <sup>Note</sup>	tsIK1	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	88		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	88		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	220		ns
Slp hold time (from SCKp <sub>↓</sub> ) <sup>Note</sup>	tKS11	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ	38		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ	38		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ	38		ns
Delay time from SCKp <sub>↑</sub> to SO <sub>p</sub> output <sup>Note</sup>	tKS01	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 1.4 kΩ		50	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 2.7 kΩ		50	ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 30 pF, R <sub>b</sub> = 5.5 kΩ		50	ns

**Note** When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.

**Caution** Select the TTL input buffer for the Slp pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SO<sub>p</sub> pin and SCKp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**CSI mode connection diagram (during communication at different potential)**


**Remark 5.**  $R_b[\Omega]$ : Communication line (SCKp, SOp) pull-up resistance,  $C_b[F]$ : Communication line (SCKp, SOp) load capacitance,  $V_b[V]$ : Communication line voltage

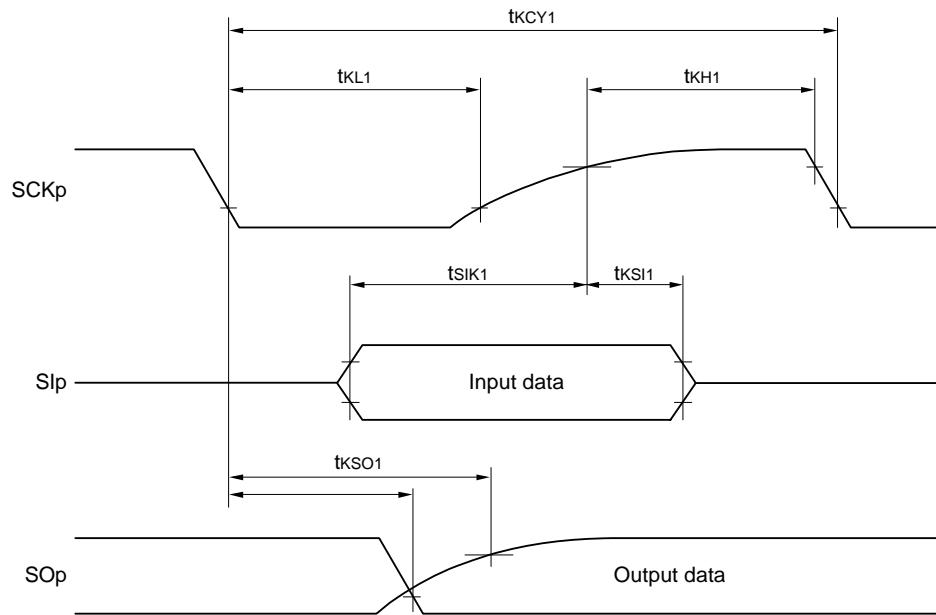
**Remark 6.** p: CSI number ( $p = 00, 01, 10, 20, 30, 31$ ), m: Unit number ( $m = 0, 1$ ), n: Channel number ( $n = 0$  to  $3$ ), g: PIM and POM number ( $g = 0, 1, 3$  to  $5, 14$ )

**Remark 7.** fmck: Serial array unit operation clock frequency

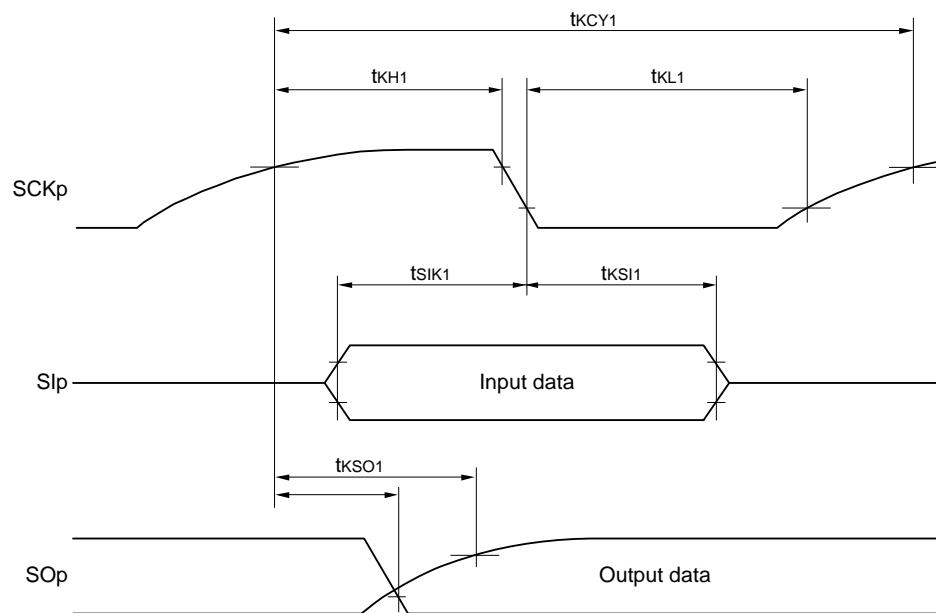
(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number, n: Channel number ( $mn = 00$ ))

**Remark 8.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

**CSI mode serial transfer timing (master mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)**



**CSI mode serial transfer timing (master mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)**



**Remark 1.** p: CSI number (p = 00, 01, 10, 20, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3),  
g: PIM and POM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

**(7) Communication at different potential (1.8 V, 2.5 V, 3 V) (CSI mode) (slave mode, SCKp... external clock input)**

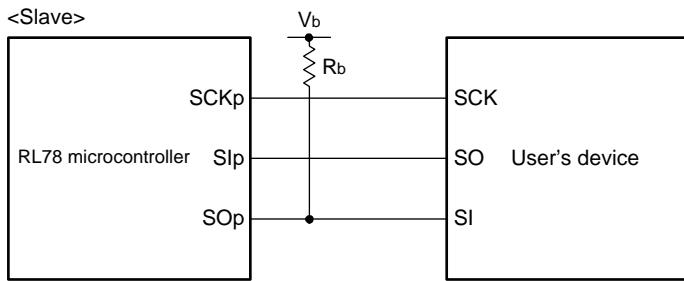
**(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)**

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
SCKp cycle time Note 1	tkCY2	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	24 MHz < fMCK	28/fMCK	ns
			20 MHz < fMCK ≤ 24 MHz	24/fMCK	ns
			8 MHz < fMCK ≤ 20 MHz	20/fMCK	ns
			4 MHz < fMCK ≤ 8 MHz	16/fMCK	ns
			fMCK ≤ 4 MHz	12/fMCK	ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V	24 MHz < fMCK	40/fMCK	ns
			20 MHz < fMCK ≤ 24 MHz	32/fMCK	ns
			16 MHz < fMCK ≤ 20 MHz	28/fMCK	ns
			8 MHz < fMCK ≤ 16 MHz	24/fMCK	ns
			4 MHz < fMCK ≤ 8 MHz	16/fMCK	ns
			fMCK ≤ 4 MHz	12/fMCK	ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V	24 MHz < fMCK	96/fMCK	ns
			20 MHz < fMCK ≤ 24 MHz	72/fMCK	ns
			16 MHz < fMCK ≤ 20 MHz	64/fMCK	ns
			8 MHz < fMCK ≤ 16 MHz	52/fMCK	ns
			4 MHz < fMCK ≤ 8 MHz	32/fMCK	ns
			fMCK ≤ 4 MHz	20/fMCK	ns
SCKp high-/low-level width	tKH2, tKL2	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	tkCY2/2 - 24		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V	tkCY2/2 - 36		ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V	tkCY2/2 - 100		ns
Slp setup time (to SCKp↑) Note 2	tsIK2	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V	1/fMCK + 40		ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V	1/fMCK + 40		ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V	1/fMCK + 60		ns
Slp hold time (from SCKp↑) Note 3	tksI2		1/fMCK + 62		ns
Delay time from SCKp↓ to SOp output Note 4	tksO2	4.0 V ≤ EVDD0 ≤ 5.5 V, 2.7 V ≤ Vb ≤ 4.0 V, Cb = 30 pF, Rb = 1.4 kΩ		2/fMCK + 240	ns
		2.7 V ≤ EVDD0 < 4.0 V, 2.3 V ≤ Vb ≤ 2.7 V, Cb = 30 pF, Rb = 2.7 kΩ		2/fMCK + 428	ns
		2.4 V ≤ EVDD0 < 3.3 V, 1.6 V ≤ Vb ≤ 2.0 V, Cb = 30 pF, Rv = 5.5 kΩ		2/fMCK + 1146	ns

**(Notes, Caution, and Remarks are listed on the next page.)**

- Note 1.** Transfer rate in the SNOOZE mode: MAX. 1 Mbps
- Note 2.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp setup time becomes “to SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.
- Note 3.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The Slp hold time becomes “from SCKp↓” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.
- Note 4.** When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1. The delay time to SOp output becomes “from SCKp↑” when DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.
- Caution** Select the TTL input buffer for the Slp pin and SCKp pin, and the N-ch open drain output (VDD tolerance (for the 30- to 52-pin products)/EVDD tolerance (for the 64- to 100-pin products)) mode for the SOp pin by using port input mode register g (PIMg) and port output mode register g (POMg). For VIH and Vil, see the DC characteristics with TTL input buffer selected.

#### CSI mode connection diagram (during communication at different potential)



**Remark 1.** Rb[Ω]: Communication line (SOp) pull-up resistance, Cb[F]: Communication line (SOp) load capacitance, Vb[V]: Communication line voltage

**Remark 2.** p: CSI number (p = 00, 01, 10, 20, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), g: PIM and POM number (g = 0, 1, 3 to 5, 14)

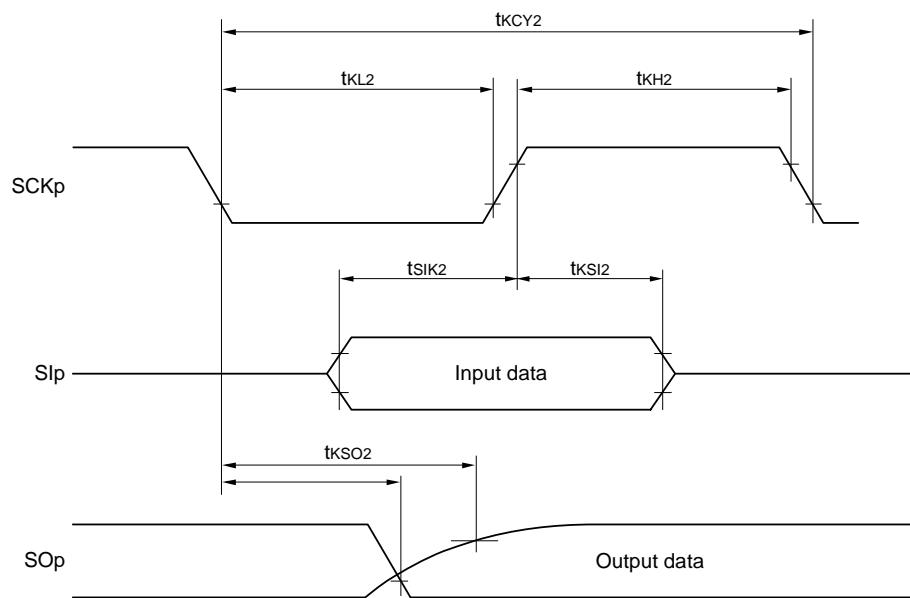
**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn).  
m: Unit number, n: Channel number (mn = 00, 01, 02, 10, 12, 13))

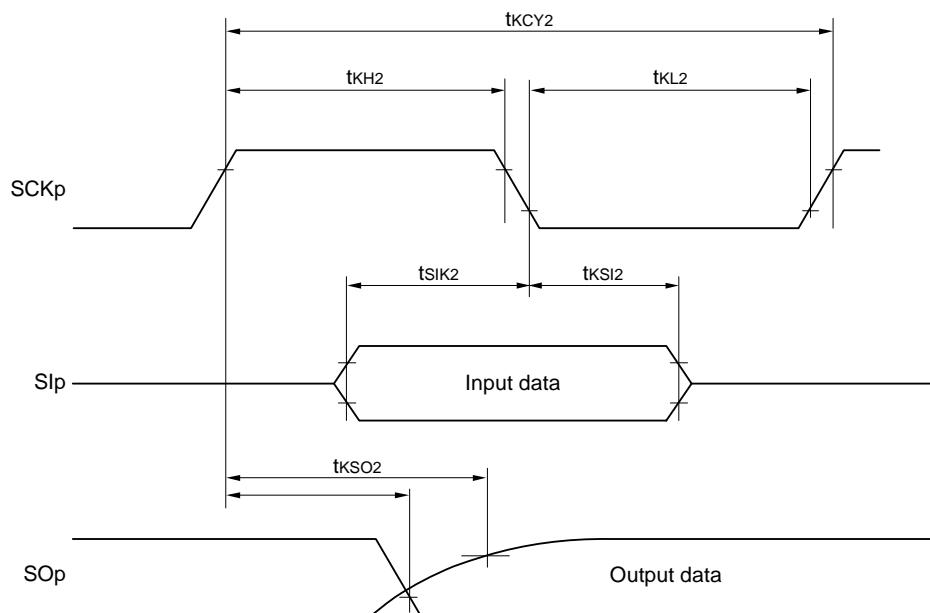
**Remark 4.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

Also, communication at different potential cannot be performed during clock synchronous serial communication with the slave select function.

**CSI mode serial transfer timing (slave mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 0, or DAPmn = 1 and CKPmn = 1.)**



**CSI mode serial transfer timing (slave mode) (during communication at different potential)**  
**(When DAPmn = 0 and CKPmn = 1, or DAPmn = 1 and CKPmn = 0.)**



**Remark 1.** p: CSI number (p = 00, 01, 10, 20, 30, 31), m: Unit number (m = 0, 1), n: Channel number (n = 0 to 3), g: PIM and POM number (g = 0, 1, 3 to 5, 14)

**Remark 2.** CSI01 of 48-, 52-, 64-pin products, and CSI11 and CSI21 cannot communicate at different potential. Use other CSI for communication at different potential.

Also, communication at different potential cannot be performed during clock synchronous serial communication with the slave select function.

(8) Communication at different potential (1.8 V, 2.5 V, 3 V) (simplified I<sup>2</sup>C mode)(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(1/2)

Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
SCL <sub>r</sub> clock frequency	f <sub>SCL</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		400 Note 1	kHz
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ		400 Note 1	kHz
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ		100 Note 1	kHz
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ		100 Note 1	kHz
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ		100 Note 1	kHz
Hold time when SCL <sub>r</sub> = "L"	t <sub>LOW</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1200		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1200		ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	4600		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	4600		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	4650		ns
Hold time when SCL <sub>r</sub> = "H"	t <sub>HIGH</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	620		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	500		ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	2700		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	2400		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	1830		ns

(8) Communication at different potential (1.8 V, 2.5 V, 3 V) (simplified I<sup>2</sup>C mode)(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

(2/2)

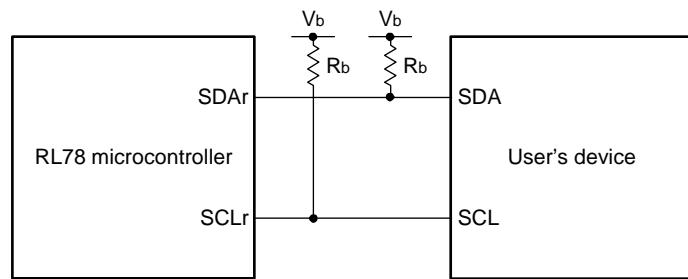
Parameter	Symbol	Conditions	HS (high-speed main) mode		Unit
			MIN.	MAX.	
Data setup time (reception)	t <sub>SU:DAT</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/f <sub>MCK</sub> + 340 Note 2		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	1/f <sub>MCK</sub> + 340 Note 2		ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	1/f <sub>MCK</sub> + 760 Note 2		ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	1/f <sub>MCK</sub> + 760 Note 2		ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	1/f <sub>MCK</sub> + 570 Note 2		ns
Data hold time (transmission)	t <sub>HD:DAT</sub>	4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	770	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 50 pF, R <sub>b</sub> = 2.7 kΩ	0	770	ns
		4.0 V ≤ EV <sub>DD0</sub> ≤ 5.5 V, 2.7 V ≤ V <sub>b</sub> ≤ 4.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.8 kΩ	0	1420	ns
		2.7 V ≤ EV <sub>DD0</sub> < 4.0 V, 2.3 V ≤ V <sub>b</sub> ≤ 2.7 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 2.7 kΩ	0	1420	ns
		2.4 V ≤ EV <sub>DD0</sub> < 3.3 V, 1.6 V ≤ V <sub>b</sub> ≤ 2.0 V, C <sub>b</sub> = 100 pF, R <sub>b</sub> = 5.5 kΩ	0	1215	ns

**Note 1.** The value must also be equal to or less than f<sub>MCK</sub>/4.**Note 2.** Set the f<sub>MCK</sub> value to keep the hold time of SCL<sub>r</sub> = "L" and SCL<sub>r</sub> = "H".

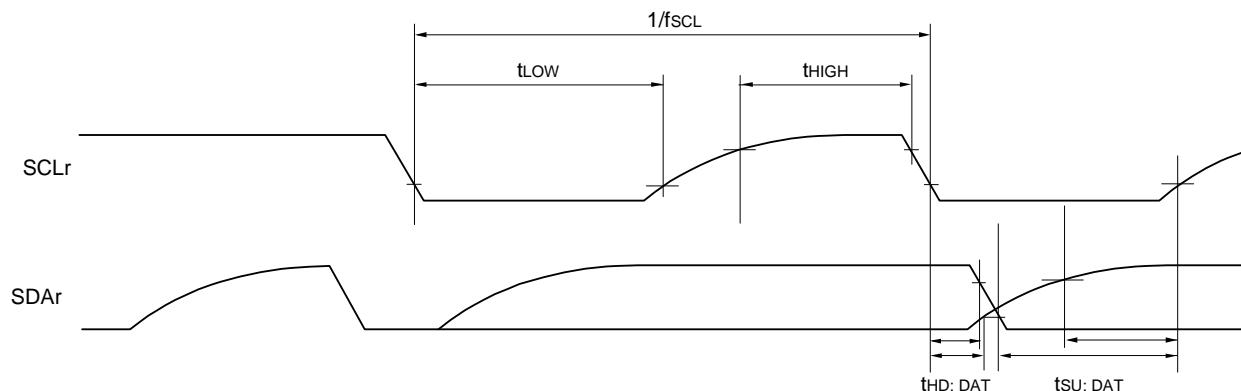
**Caution** Select the TTL input buffer and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SD<sub>Ar</sub> pin and the N-ch open drain output (V<sub>DD</sub> tolerance (for the 30- to 52-pin products)/EV<sub>DD</sub> tolerance (for the 64- to 100-pin products)) mode for the SCL<sub>r</sub> pin by using port input mode register g (PIMg) and port output mode register g (POMg). For V<sub>IH</sub> and V<sub>IL</sub>, see the DC characteristics with TTL input buffer selected.

(Remarks are listed on the next page.)

**Simplified I<sup>2</sup>C mode connection diagram (during communication at different potential)**



**Simplified I<sup>2</sup>C mode serial transfer timing (during communication at different potential)**



**Remark 1.** R<sub>b</sub>[Ω]: Communication line (SDAr, SCLR) pull-up resistance, C<sub>b</sub>[F]: Communication line (SDAr, SCLR) load capacitance, V<sub>b</sub>[V]: Communication line voltage

**Remark 2.** r: IIC number (r = 00, 01, 10, 11, 20, 30, 31), g: PIM, POM number (g = 0, 1, 3 to 5, 14)

**Remark 3.** fmck: Serial array unit operation clock frequency

(Operation clock to be set by the CKSmn bit of serial mode register mn (SMRmn). m: Unit number (m = 0, 1), n: Channel number (n = 0, 2), mn = 00, 01, 02, 10, 12, 13)

### 35.5.2 Serial interface IICA

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions	HS (high-speed main) mode				Unit	
			Standard mode		Fast mode			
			MIN.	MAX.	MIN.	MAX.		
SCLA0 clock frequency	f <sub>SCL</sub>	Fast mode: f <sub>CLK</sub> ≥ 3.5 MHz	—	—	0	400	kHz	
		Standard mode: f <sub>CLK</sub> ≥ 1 MHz	0	100	—	—	kHz	
Setup time of restart condition	t <sub>SU: STA</sub>		4.7		0.6		μs	
Hold time Note 1	t <sub>HD: STA</sub>		4.0		0.6		μs	
Hold time when SCLA0 = "L"	t <sub>LOW</sub>		4.7		1.3		μs	
Hold time when SCLA0 = "H"	t <sub>HIGH</sub>		4.0		0.6		μs	
Data setup time (reception)	t <sub>SU: DAT</sub>		250		100		ns	
Data hold time (transmission) Note 2	t <sub>HD: DAT</sub>		0	3.45	0	0.9	μs	
Setup time of stop condition	t <sub>SU: STO</sub>		4.0		0.6		μs	
Bus-free time	t <sub>BUF</sub>		4.7		1.3		μs	

**Note 1.** The first clock pulse is generated after this period when the start/restart condition is detected.

**Note 2.** The maximum value (MAX.) of t<sub>HD: DAT</sub> is during normal transfer and a wait state is inserted in the ACK (acknowledge) timing.

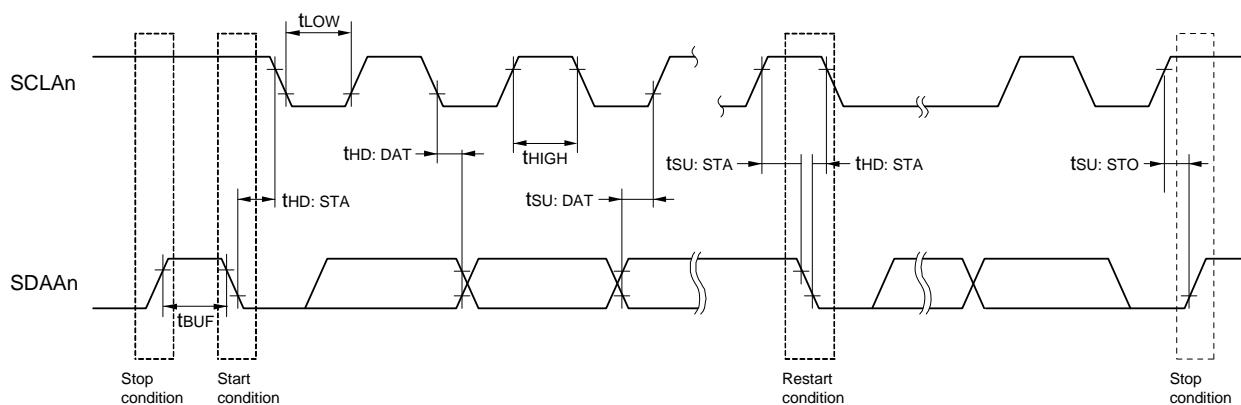
**Caution** The values in the above table are applied even when bit 2 (PIOR02) in the peripheral I/O redirection register 0 (PIOR0) is 1. At this time, the pin characteristics (I<sub>OH1</sub>, I<sub>OL1</sub>, V<sub>OH1</sub>, V<sub>OL1</sub>) must satisfy the values in the redirect destination.

**Remark** The maximum value of C<sub>b</sub> (communication line capacitance) and the value of R<sub>b</sub> (communication line pull-up resistor) at that time in each mode are as follows.

Standard mode: C<sub>b</sub> = 400 pF, R<sub>b</sub> = 2.7 kΩ

Fast mode: C<sub>b</sub> = 320 pF, R<sub>b</sub> = 1.1 kΩ

#### IICA serial transfer timing



**Remark** n = 0, 1

## 35.6 Analog Characteristics

### 35.6.1 A/D converter characteristics

#### Classification of A/D converter characteristics

Input channel	Reference Voltage	Reference voltage (+) = AVREFP Reference voltage (-) = AVREFM	Reference voltage (+) = VDD Reference voltage (-) = Vss	Reference voltage (+) = VBGR Reference voltage (-) = AVREFM
AN10 to ANI14	Refer to 35.6.1 (1).	Refer to 35.6.1 (3).	Refer to 35.6.1 (4).	—
ANI16 to ANI20				
Internal reference voltage Temperature sensor output voltage				

- (1) When reference voltage (+) = AVREFP/ANI0 (ADREFP1 = 0, ADREFP0 = 1), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI2 to ANI14, internal reference voltage, and temperature sensor output voltage

(TA = -40 to +105°C, 2.4 V ≤ AVREFP ≤ VDD ≤ 5.5 V, Vss = 0 V, Reference voltage (+) = AVREFP,

Reference voltage (-) = AVREFM = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		10	bit
Overall error Note 1	AINL	10-bit resolution AVREFP = VDD Note 3	2.4 V ≤ AVREFP ≤ 5.5 V		1.2	±3.5	LSB
Conversion time	tCONV	10-bit resolution Target pin: ANI2 to ANI14	3.6 V ≤ VDD ≤ 5.5 V	2.125		39	μs
			2.7 V ≤ VDD ≤ 5.5 V	3.1875		39	μs
			2.4 V ≤ VDD ≤ 5.5 V	17		39	μs
	tCONV	10-bit resolution Target pin: Internal reference voltage, and temperature sensor output voltage (HS (high-speed main) mode)	3.6 V ≤ VDD ≤ 5.5 V	2.375		39	μs
			2.7 V ≤ VDD ≤ 5.5 V	3.5625		39	μs
			2.4 V ≤ VDD ≤ 5.5 V	17		39	μs
Zero-scale error Notes 1, 2	Ezs	10-bit resolution AVREFP = VDD Note 3	2.4 V ≤ AVREFP ≤ 5.5 V			±0.25	%FSR
Full-scale error Notes 1, 2	EFS	10-bit resolution AVREFP = VDD Note 3	2.4 V ≤ AVREFP ≤ 5.5 V			±0.25	%FSR
Integral linearity error Note 1	ILE	10-bit resolution AVREFP = VDD Note 3	2.4 V ≤ AVREFP ≤ 5.5 V			±2.5	LSB
Differential linearity error Note 1	DLE	10-bit resolution AVREFP = VDD Note 3	2.4 V ≤ AVREFP ≤ 5.5 V			±1.5	LSB
Analog input voltage	VAIN	ANI2 to ANI14		0		AVREFP	V
		Internal reference voltage output (2.4 V ≤ VDD ≤ 5.5 V, HS (high-speed main) mode)		VBGR Note 4			V
		Temperature sensor output voltage (2.4 V ≤ VDD ≤ 5.5 V, HS (high-speed main) mode)		VTMPS25 Note 4			V

Note 1. Excludes quantization error (±1/2 LSB).

Note 2. This value is indicated as a ratio (%FSR) to the full-scale value.

Note 3. When AVREFP < VDD, the MAX. values are as follows.

Overall error: Add ±1.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.05%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±0.5 LSB to the MAX. value when AVREFP = VDD.

Note 4. Refer to 35.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

- (2) When reference voltage (+) = AVREFP/ANI0 (ADREFP1 = 0, ADREFP0 = 1), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI16 to ANI20

(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, 2.4 V ≤ AVREFP ≤ VDD ≤ 5.5 V,

Vss = EVSS0 = EVSS1 = 0 V, Reference voltage (+) = AVREFP, Reference voltage (-) = AVREFM = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		10	bit
Overall error Note 1	AINL	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	2.4 V ≤ AVREFP ≤ 5.5 V		1.2	±5.0	LSB
Conversion time	tCONV	10-bit resolution Target ANI pin: ANI16 to ANI20	3.6 V ≤ VDD ≤ 5.5 V	2.125		39	μs
			2.7 V ≤ VDD ≤ 5.5 V	3.1875		39	μs
			2.4 V ≤ VDD ≤ 5.5 V	17		39	μs
Zero-scale error Notes 1, 2	Ezs	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	2.4 V ≤ AVREFP ≤ 5.5 V			±0.35	%FSR
Full-scale error Notes 1, 2	EFS	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	2.4 V ≤ AVREFP ≤ 5.5 V			±0.35	%FSR
Integral linearity error Note 1	ILE	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	2.4 V ≤ AVREFP ≤ 5.5 V			±3.5	LSB
Differential linearity error Note 1	DLE	10-bit resolution EVDD0 ≤ AVREFP = VDD Notes 3, 4	2.4 V ≤ AVREFP ≤ 5.5 V			±2.0	LSB
Analog input voltage	VAIN	ANI16 to ANI20		0		AVREFP and EVDD0	V

**Note 1.** Excludes quantization error (±1/2 LSB).

**Note 2.** This value is indicated as a ratio (%FSR) to the full-scale value.

**Note 3.** When EVDD0 ≤ AVREFP ≤ VDD, the MAX. values are as follows.

Overall error: Add ±1.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.05%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±0.5 LSB to the MAX. value when AVREFP = VDD.

**Note 4.** When AVREFP < EVDD0 ≤ VDD, the MAX. values are as follows.

Overall error: Add ±4.0 LSB to the MAX. value when AVREFP = VDD.

Zero-scale error/Full-scale error: Add ±0.20%FSR to the MAX. value when AVREFP = VDD.

Integral linearity error/ Differential linearity error: Add ±2.0 LSB to the MAX. value when AVREFP = VDD.

- (3) When reference voltage (+) = V<sub>DD</sub> (ADREFP1 = 0, ADREFP0 = 0), reference voltage (-) = V<sub>SS</sub> (ADREFM = 0), target pin: ANI0 to ANI14, ANI16 to ANI20, internal reference voltage, and temperature sensor output voltage

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V, Reference voltage (+) = V<sub>DD</sub>, Reference voltage (-) = V<sub>SS</sub>)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution	RES		8		10	bit
Overall error Note 1	AINL	10-bit resolution	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V		1.2	±7.0 LSB
Conversion time	t <sub>CONV</sub>	10-bit resolution Target pin: ANI0 to ANI14, ANI16 to ANI20	3.6 V ≤ V <sub>DD</sub> ≤ 5.5 V	2.125		39 μs
			2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	3.1875		39 μs
			2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V	17		39 μs
		10-bit resolution Target pin: internal reference voltage, and temperature sensor output voltage (HS (high-speed main) mode)	3.6 V ≤ V <sub>DD</sub> ≤ 5.5 V	2.375		39 μs
			2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V	3.5625		39 μs
			2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V	17		39 μs
Zero-scale error Notes 1, 2	E <sub>ZS</sub>	10-bit resolution	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±0.60 %FSR
Full-scale error Notes 1, 2	E <sub>FS</sub>	10-bit resolution	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±0.60 %FSR
Integral linearity error Note 1	I <sub>LE</sub>	10-bit resolution	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±4.0 LSB
Differential linearity error Note 1	D <sub>LE</sub>	10-bit resolution	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±2.0 LSB
Analog input voltage	V <sub>AIN</sub>	ANI0 to ANI14		0		V <sub>DD</sub> V
		ANI16 to ANI20		0		EV <sub>DD0</sub> V
		Internal reference voltage (2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode)		V <sub>BGR</sub> Note 3		V
		Temperature sensor output voltage (2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode)		V <sub>TMP25</sub> Note 3		V

Note 1. Excludes quantization error (±1/2 LSB).

Note 2. This value is indicated as a ratio (% FSR) to the full-scale value.

Note 3. Refer to 35.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

- (4) When reference voltage (+) = Internal reference voltage (ADREFP1 = 1, ADREFP0 = 0), reference voltage (-) = AVREFM/ANI1 (ADREFM = 1), target pin: ANI0, ANI2 to ANI14, ANI16 to ANI20

(TA = -40 to +105°C, 2.4 V ≤ VDD ≤ 5.5 V, 1.6 V ≤ EVDD = EVDD1 ≤ VDD, VSS = EVSS0 = EVSS1 = 0 V,

Reference voltage (+) = VBGR Note 3, Reference voltage (-) = AVREFM = 0 V Note 4, HS (high-speed main) mode)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES			8		bit	
Conversion time	tCONV	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V	17		39	μs
Zero-scale error Notes 1, 2	Ezs	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±0.60	% FSR
Integral linearity error Note 1	ILE	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±2.0	LSB
Differential linearity error Note 1	DLE	8-bit resolution	2.4 V ≤ VDD ≤ 5.5 V			±1.0	LSB
Analog input voltage	VAIN			0		VBGR Note 3	V

**Note 1.** Excludes quantization error (±1/2 LSB).

**Note 2.** This value is indicated as a ratio (% FSR) to the full-scale value.

**Note 3.** Refer to 35.6.2 Temperature sensor characteristics/internal reference voltage characteristic.

**Note 4.** When reference voltage (-) = Vss, the MAX. values are as follows.

Zero-scale error: Add ±0.35%FSR to the MAX. value when reference voltage (-) = AVREFM.

Integral linearity error: Add ±0.5 LSB to the MAX. value when reference voltage (-) = AVREFM.

Differential linearity error: Add ±0.2 LSB to the MAX. value when reference voltage (-) = AVREFM.

### 35.6.2 Temperature sensor characteristics/internal reference voltage characteristic

(TA = -40 to +105°C, 2.4 V ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V, HS (high-speed main) mode)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Temperature sensor output voltage	V <sub>TMPS25</sub>	Setting ADS register = 80H, TA = +25°C		1.05		V
Internal reference voltage	V <sub>BGR</sub>	Setting ADS register = 81H	1.38	1.45	1.5	V
Temperature coefficient	F <sub>VTMPS</sub>	Temperature sensor that depends on the temperature		-3.6		mV/°C
Operation stabilization wait time	t <sub>AMP</sub>		5			μs

### 35.6.3 D/A converter characteristics

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>SS0</sub> = EV<sub>SS1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>SS0</sub> = EV<sub>SS1</sub> = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Resolution	RES					8	bit
Overall error	AINL	R <sub>load</sub> = 4 MΩ	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±2.5	LSB
		R <sub>load</sub> = 8 MΩ	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V			±2.5	LSB
Settling time	t <sub>SET</sub>	C <sub>load</sub> = 20 pF	2.7 V ≤ V <sub>DD</sub> ≤ 5.5 V			3	μs
			2.4 V ≤ V <sub>DD</sub> < 2.7 V			6	μs

### 35.6.4 Comparator

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>VDD0</sub> = EV<sub>VDD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage range	I <sub>Vref</sub>			0		EV <sub>VDD0</sub> - 1.4	V
	I <sub>Vcmp</sub>			-0.3		EV <sub>VDD0</sub> + 0.3	V
Output delay	t <sub>d</sub>	V <sub>DD</sub> = 3.0 V Input slew rate > 50 mV/μs	Comparator high-speed mode, standard mode			1.2	μs
			Comparator high-speed mode, window mode			2.0	μs
			Comparator low-speed mode, standard mode		3.0	5.0	μs
High-electric-potential reference voltage	V <sub>TW+</sub>	Comparator high-speed mode, window mode			0.76 V <sub>DD</sub>		V
Low-electric-potential reference voltage	V <sub>TW-</sub>	Comparator high-speed mode, window mode			0.24 V <sub>DD</sub>		V
Operation stabilization wait time	t <sub>CMP</sub>			100			μs
Internal reference voltage Note	V <sub>BGR</sub>	2.4 V ≤ V <sub>DD</sub> ≤ 5.5 V, HS (high-speed main) mode		1.38	1.45	1.50	V

**Note** Not usable in sub-clock operation or STOP mode.

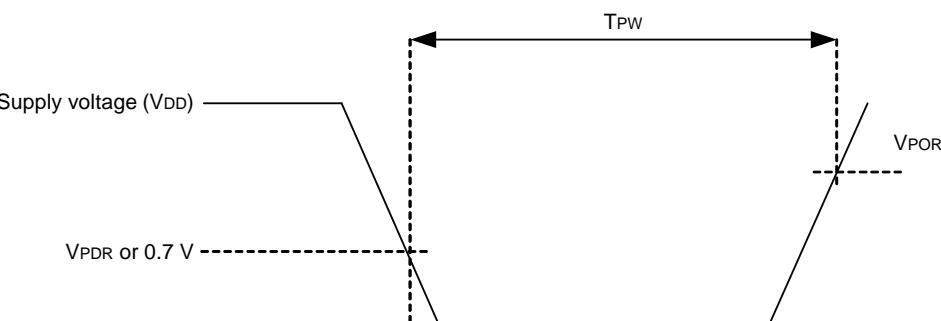
### 35.6.5 POR circuit characteristics

(TA = -40 to +105°C, V<sub>SS</sub> = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Power on/down reset threshold	V <sub>POR</sub>	Voltage threshold on V <sub>DD</sub> rising	1.45	1.51	1.57	V
	V <sub>PDR</sub>	Voltage threshold on V <sub>DD</sub> falling Note 1	1.44	1.50	1.56	V
Minimum pulse width Note 2	T <sub>PW</sub>		300			μs

**Note 1.** However, when the operating voltage falls while the LVD is off, enter STOP mode, or enable the reset status using the external reset pin before the voltage falls below the operating voltage range shown in 35.4 AC Characteristics.

**Note 2.** Minimum time required for a POR reset when V<sub>DD</sub> exceeds below V<sub>PDR</sub>. This is also the minimum time required for a POR reset from when V<sub>DD</sub> exceeds below 0.7 V to when V<sub>DD</sub> exceeds V<sub>POR</sub> while STOP mode is entered or the main system clock is stopped through setting bit 0 (HIOSTOP) and bit 7 (MSTOP) in the clock operation status control register (CSC).



### 35.6.6 LVD circuit characteristics

#### (1) Reset Mode and Interrupt Mode

(TA = -40 to +105°C, VPDR ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Voltage detection threshold	VLVD0	Rising edge	3.90	4.06	4.22	V
		Falling edge	3.83	3.98	4.13	V
	VLVD1	Rising edge	3.60	3.75	3.90	V
		Falling edge	3.53	3.67	3.81	V
	VLVD2	Rising edge	3.01	3.13	3.25	V
		Falling edge	2.94	3.06	3.18	V
	VLVD3	Rising edge	2.90	3.02	3.14	V
		Falling edge	2.85	2.96	3.07	V
	VLVD4	Rising edge	2.81	2.92	3.03	V
		Falling edge	2.75	2.86	2.97	V
	VLVD5	Rising edge	2.70	2.81	2.92	V
		Falling edge	2.64	2.75	2.86	V
	VLVD6	Rising edge	2.61	2.71	2.81	V
		Falling edge	2.55	2.65	2.75	V
	VLVD7	Rising edge	2.51	2.61	2.71	V
		Falling edge	2.45	2.55	2.65	V
Minimum pulse width	tLW		300			μs
Detection delay time					300	μs

**(2) Interrupt & Reset Mode**

(TA = -40 to +105°C, VPDR ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Voltage detection threshold	V <sub>LVDD0</sub>	V <sub>POC2</sub> , V <sub>POC1</sub> , V <sub>Poco</sub> = 0, 1, 1, falling reset voltage		2.64	2.75	2.86	V
	V <sub>LVDD1</sub>	LVIS1, LVIS0 = 1, 0	Rising release reset voltage	2.81	2.92	3.03	V
			Falling interrupt voltage	2.75	2.86	2.97	V
	V <sub>LVDD2</sub>	LVIS1, LVIS0 = 0, 1	Rising release reset voltage	2.90	3.02	3.14	V
			Falling interrupt voltage	2.85	2.96	3.07	V
	V <sub>LVDD3</sub>	LVIS1, LVIS0 = 0, 0	Rising release reset voltage	3.90	4.06	4.22	V
			Falling interrupt voltage	3.83	3.98	4.13	V

**35.6.7 Power supply voltage rising slope characteristics**

(TA = -40 to +105°C, Vss = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Power supply voltage rising slope	S <sub>VDD</sub>				54	V/ms

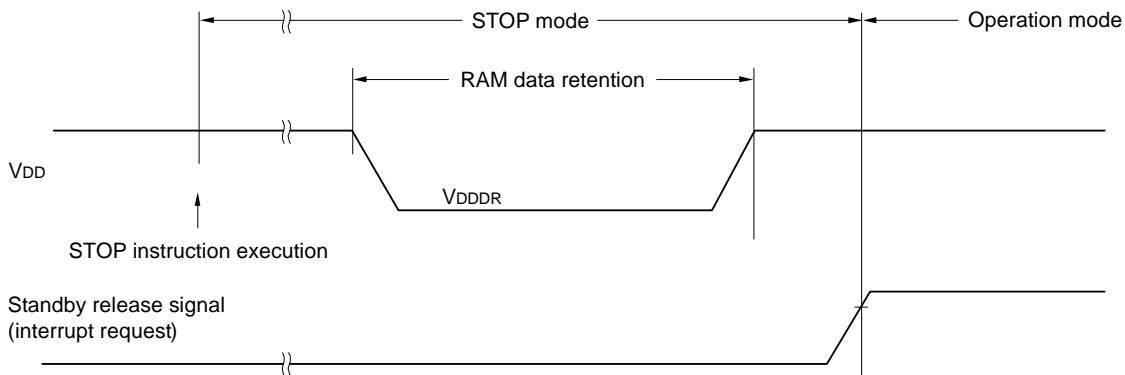
**Caution** Make sure to keep the internal reset state by the LVD circuit or an external reset until Vdd reaches the operating voltage range shown in 35.4 AC Characteristics.

### 35.7 RAM Data Retention Characteristics

(TA = -40 to +105°C, Vss = 0V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Data retention supply voltage	VDDDR		1.44 Note		5.5	V

**Note** The value depends on the POR detection voltage. When the voltage drops, the RAM data is retained before a POR reset is effected, but RAM data is not retained when a POR reset is effected.



### 35.8 Flash Memory Programming Characteristics

(TA = -40 to +105°C, 2.4 V ≤ VDD ≤ 5.5 V, Vss = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
System clock frequency	fCLK	2.4 V ≤ VDD ≤ 5.5 V	1		32	MHz
Number of code flash rewrites Notes 1, 2, 3	Cerwr	Retained for 20 years TA = 85°C Note 4	1,000			Times
Number of data flash rewrites Notes 1, 2, 3		Retained for 1 year TA = 25°C		1,000,000		
		Retained for 5 years TA = 85°C Note 4	100,000			
		Retained for 20 years TA = 85°C Note 4	10,000			

**Note 1.** 1 erase + 1 write after the erase is regarded as 1 rewrite. The retaining years are until next rewrite after the rewrite.

**Note 2.** When using flash memory programmer and Renesas Electronics self-programming library

**Note 3.** These are the characteristics of the flash memory and the results obtained from reliability testing by Renesas Electronics Corporation.

**Note 4.** This temperature is the average value at which data are retained.

### 35.9 Dedicated Flash Memory Programmer Communication (UART)

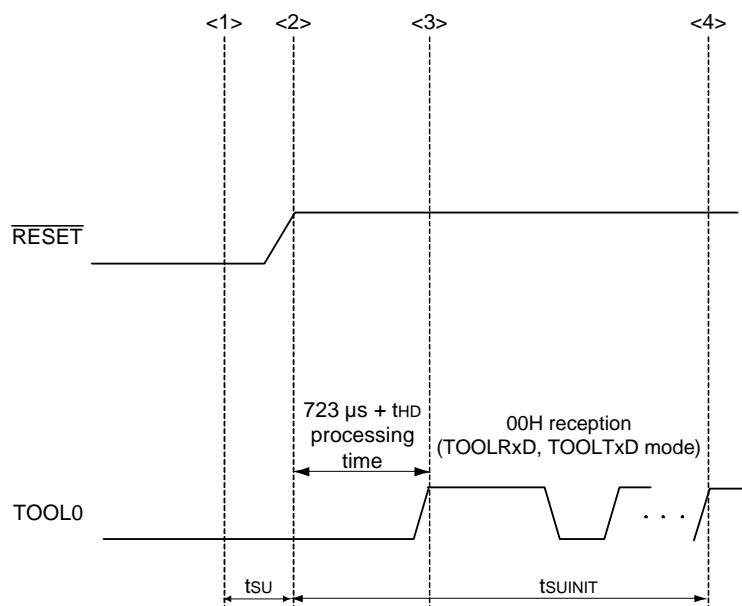
(TA = -40 to +105°C, 2.4 V ≤ EVDD0 = EVDD1 ≤ VDD ≤ 5.5 V, Vss = EVSS0 = EVSS1 = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Transfer rate		During serial programming	115,200		1,000,000	bps

### 35.10 Timing of Entry to Flash Memory Programming Modes

(TA = -40 to +105°C, 2.4 V ≤ EV<sub>DD0</sub> = EV<sub>DD1</sub> ≤ V<sub>DD</sub> ≤ 5.5 V, V<sub>SS</sub> = EV<sub>VSS0</sub> = EV<sub>VSS1</sub> = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
How long from when an external reset ends until the initial communication settings are specified	tsUINIT	POR and LVD reset must end before the external reset ends.			100	ms
How long from when the TOOL0 pin is placed at the low level until an external reset ends	tsU	POR and LVD reset must end before the external reset ends.	10			μs
How long the TOOL0 pin must be kept at the low level after an external reset ends (excluding the processing time of the firmware to control the flash memory)	tHD	POR and LVD reset must end before the external reset ends.	1			ms



<1> The low level is input to the TOOL0 pin.

<2> The external reset ends (POR and LVD reset must end before the external reset ends).

<3> The TOOL0 pin is set to the high level.

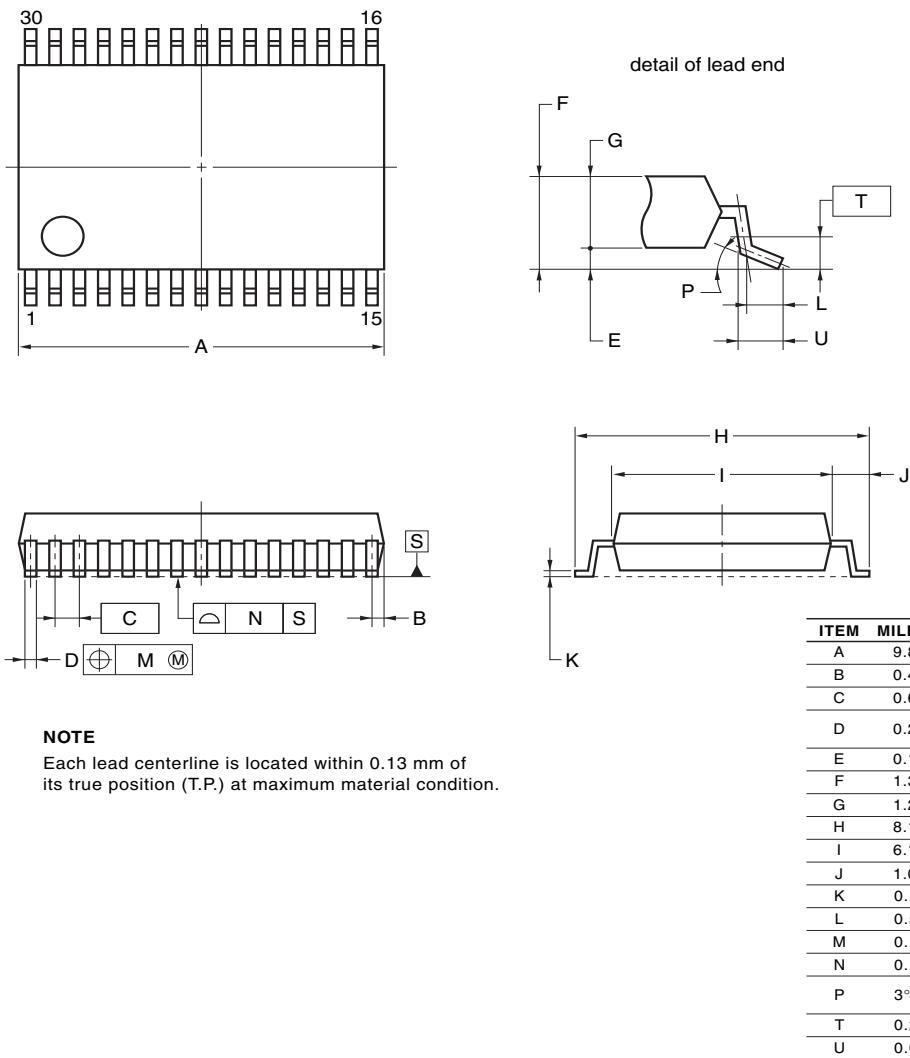
<4> Setting of the flash memory programming mode by UART reception and complete the baud rate setting.

**Remark** tsUINIT: The segment shows that it is necessary to finish specifying the initial communication settings within 100 ms from when the external resets end.  
 tsU: How long from when the TOOL0 pin is placed at the low level until a pin reset ends  
 tHD: How long to keep the TOOL0 pin at the low level from when the external resets end  
 (excluding the processing time of the firmware to control the flash memory)

## CHAPTER 36 PACKAGE DRAWINGS

### <R> 36.1 30-pin Package

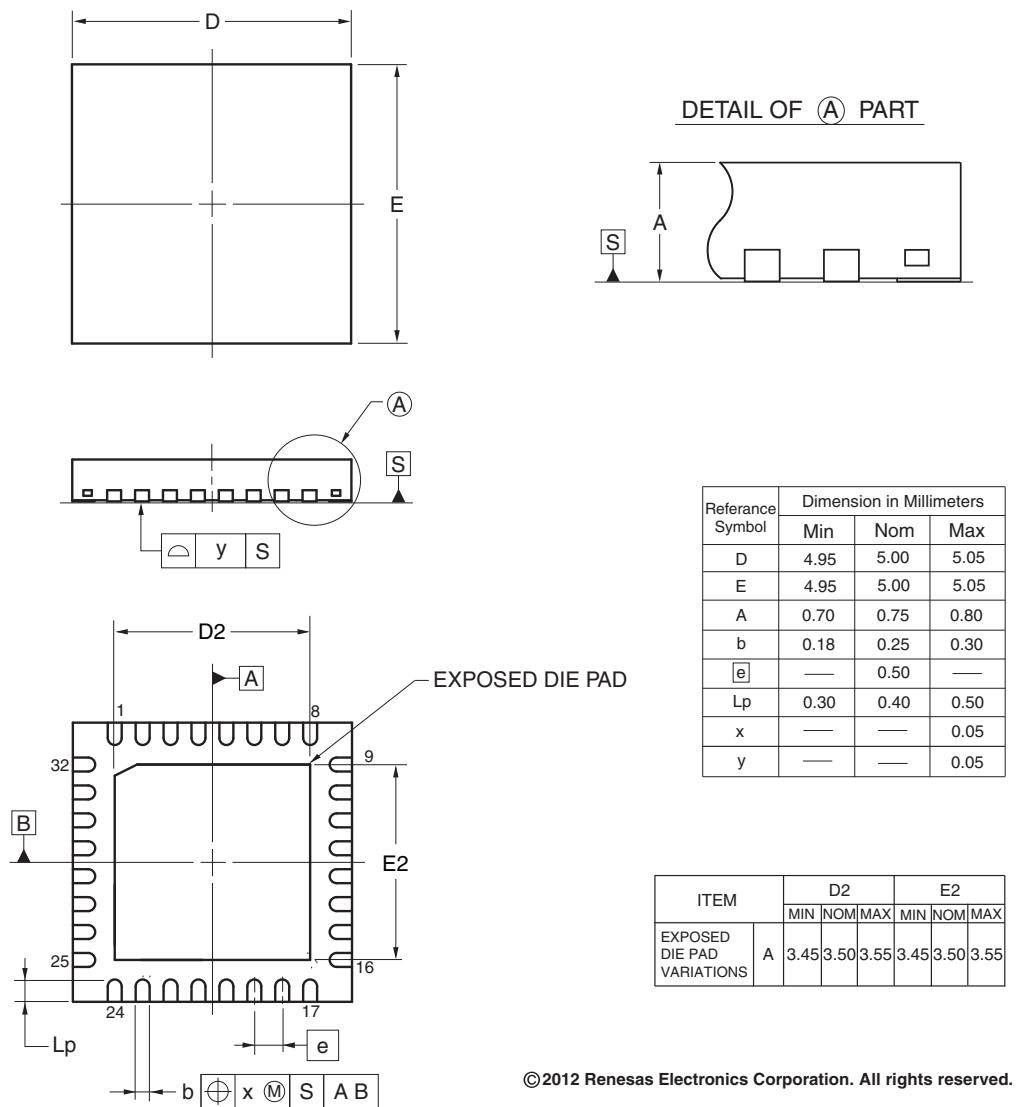
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LSSOP30-0300-0.65	PLSP0030JB-B	S30MC-65-5A4-3	0.18



©2012 Renesas Electronics Corporation. All rights reserved.

## &lt;R&gt; 36.2 32-pin Package

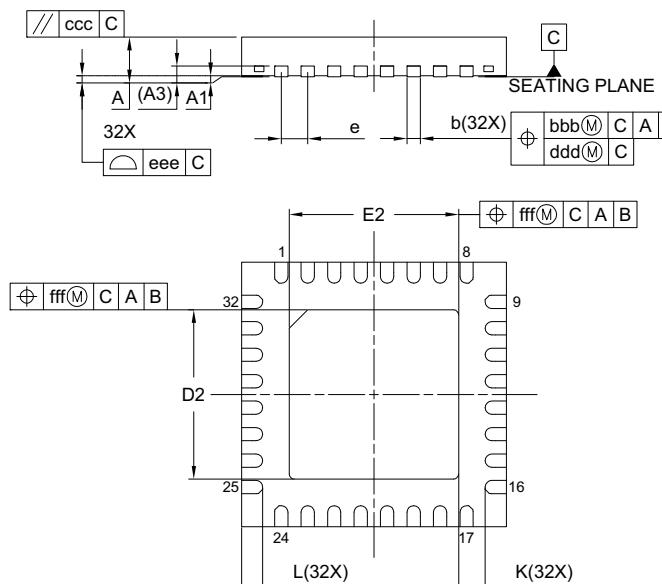
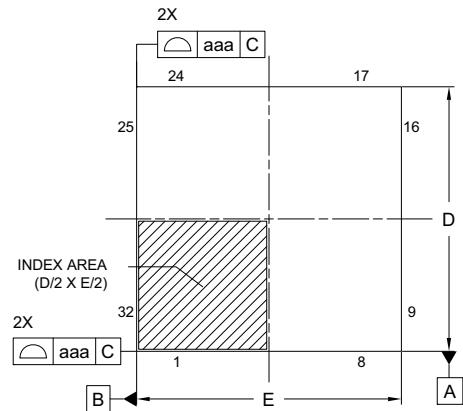
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-HWQFN32-5x5-0.50	PWQN0032KB-A	P32K8-50-3B4-4	0.06



©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

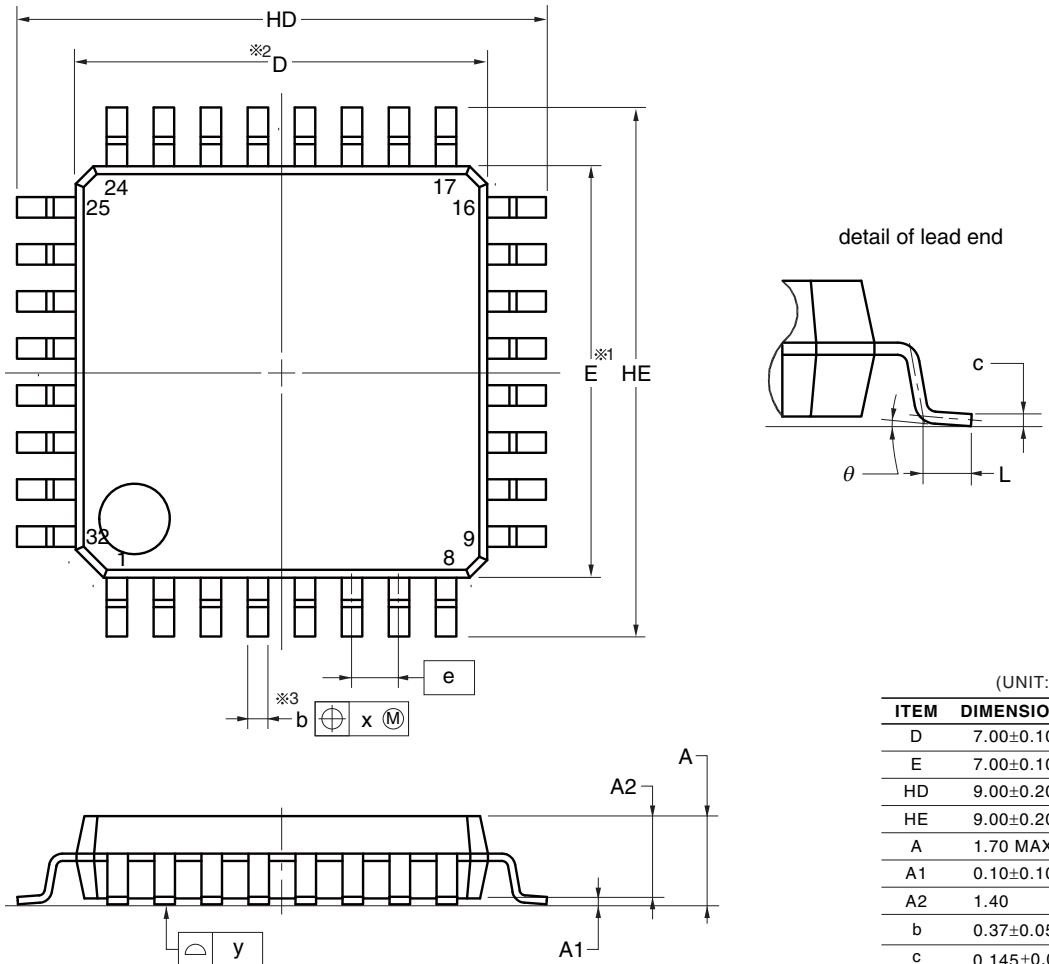
JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN032-5x5-0.50	PWQN0032KE-A	0.06



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	—	—	0.80
A <sub>1</sub>	0.00	0.02	0.05
A <sub>3</sub>	0.203 REF.		
b	0.18	0.25	0.30
D	5.00 BSC		
E	5.00 BSC		
e	0.50 BSC		
L	0.35	0.40	0.45
K	0.20	—	—
D <sub>2</sub>	3.15	3.20	3.25
E <sub>2</sub>	3.15	3.20	3.25
aaa	0.15		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP) [g]
P-LQFP32-7x7-0.80	PLQP0032GB-A	P32GA-80-GBT-1	0.2

**NOTE**

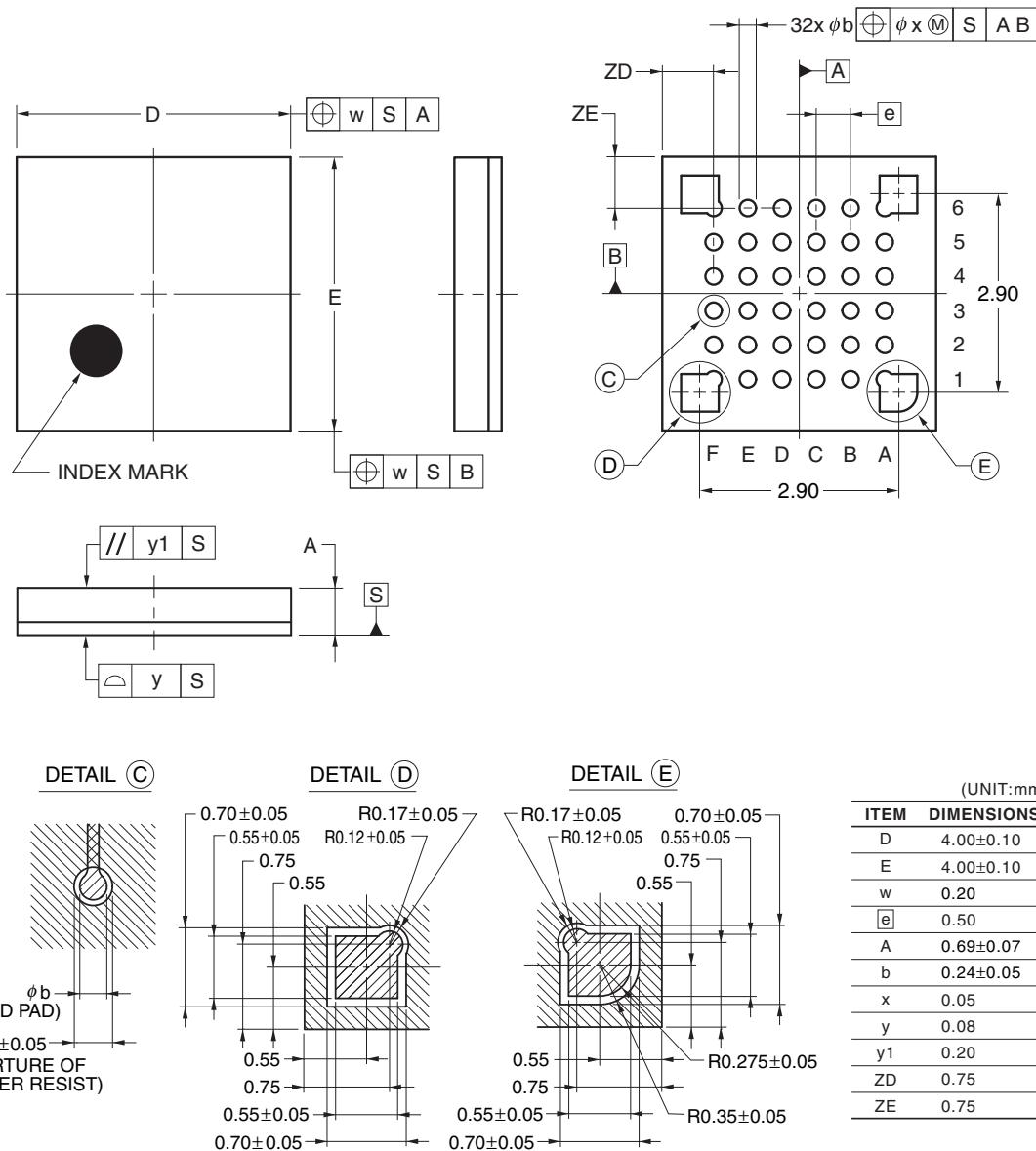
1. Dimensions “\*1” and “\*2” do not include mold flash.
2. Dimension “\*3” does not include trim offset.

(UNIT:mm)	
ITEM	DIMENSIONS
D	7.00±0.10
E	7.00±0.10
HD	9.00±0.20
HE	9.00±0.20
A	1.70 MAX.
A1	0.10±0.10
A2	1.40
b	0.37±0.05
c	0.145±0.055
L	0.50±0.20
θ	0° to 8°
e	0.80
x	0.20
y	0.10

© 2012 Renesas Electronics Corporation. All rights reserved.

## &lt;R&gt; 36.3 36-pin Package

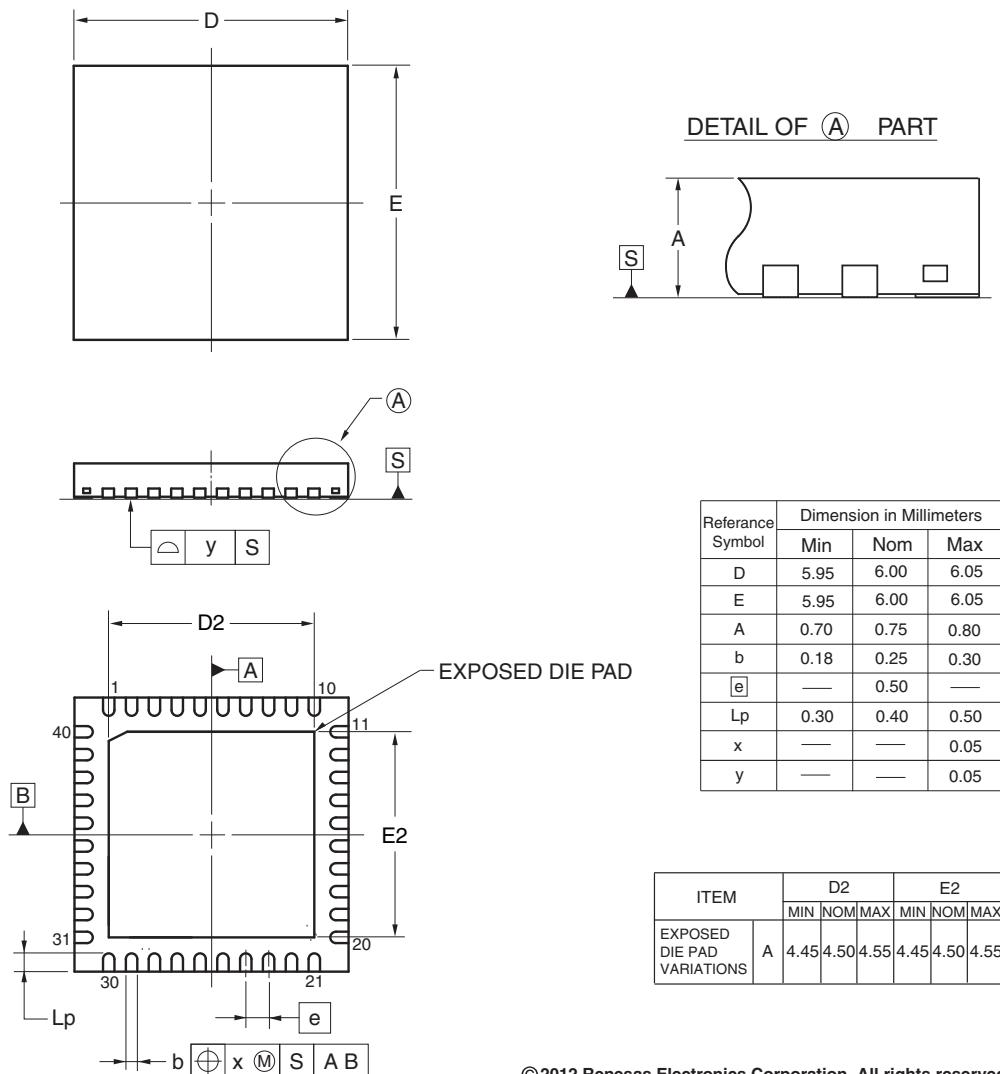
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-WFLGA36-4x4-0.50	PWLG0036KA-A	P36FC-50-AA4-2	0.023



©2012 Renesas Electronics Corporation. All rights reserved.

## &lt;R&gt; 36.4 40-pin Package

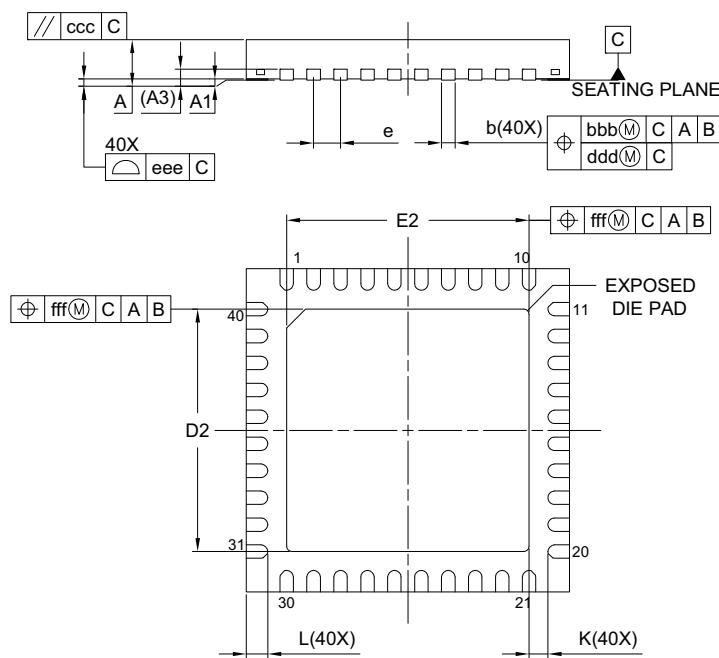
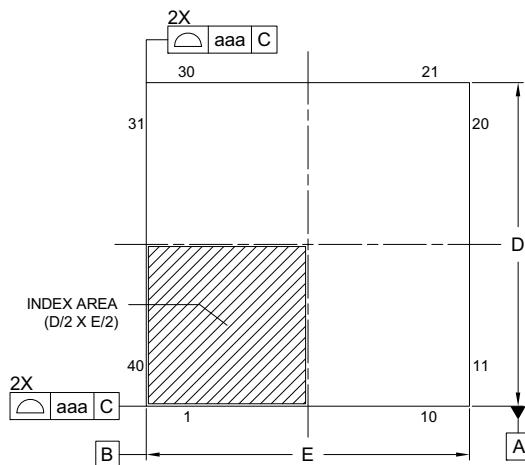
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-HWQFN40-6x6-0.50	PWQN0040KC-A	P40K8-50-4B4-4	0.09



©2012 Renesas Electronics Corporation. All rights reserved.

<R>

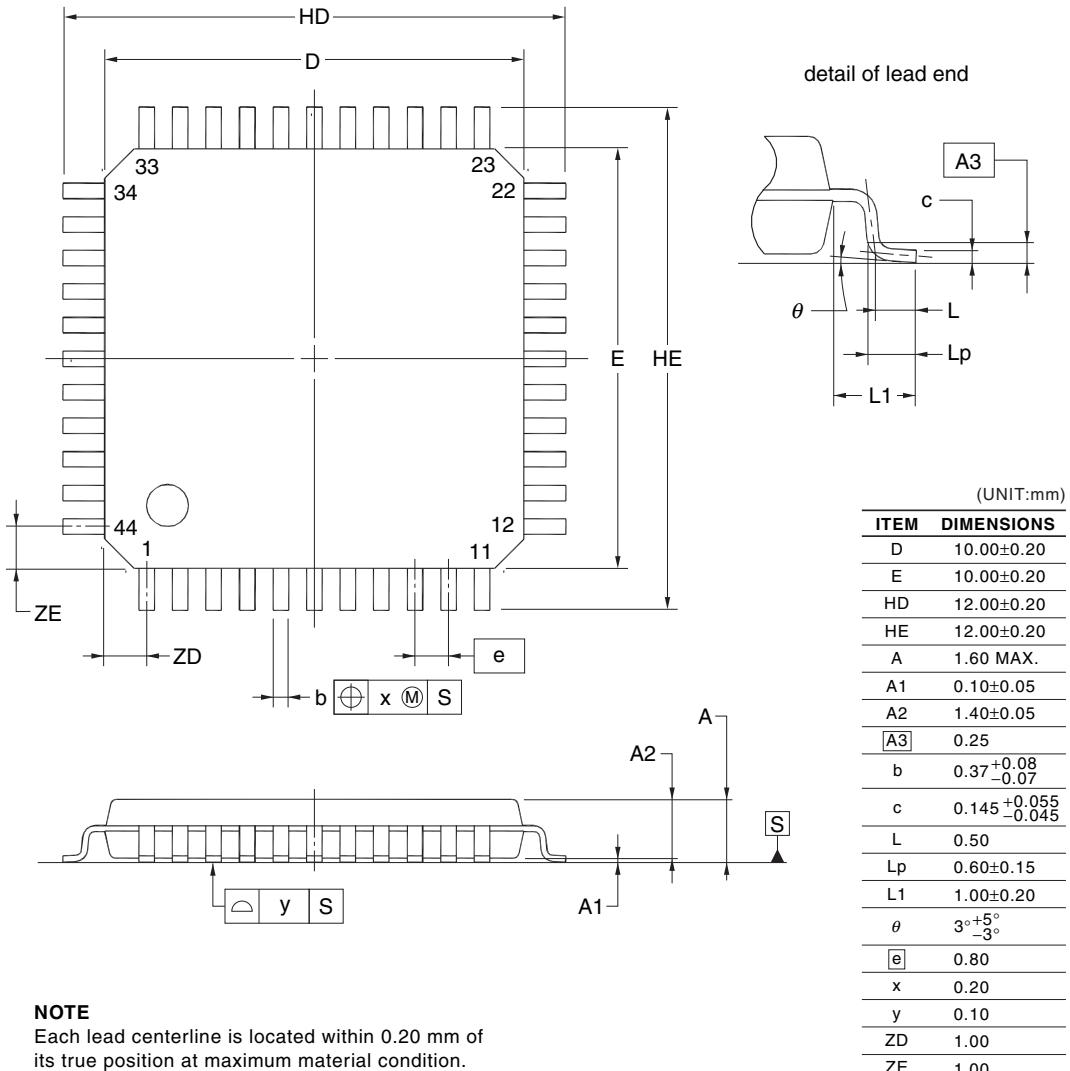
JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN040-6x6-0.50	PWQN0040KD-A	0.08



Reference Symbol	Dimension in Millimeters		
	Min.	Nom.	Max.
A	—	—	0.80
A <sub>1</sub>	0.00	0.02	0.05
A <sub>3</sub>	0.203 REF.		
b	0.18	0.25	0.30
D	6.00 BSC		
E	6.00 BSC		
e	0.50 BSC		
L	0.30	0.40	0.50
K	0.20	—	—
D <sub>2</sub>	4.45	4.50	4.55
E <sub>2</sub>	4.45	4.50	4.55
aaa	0.15		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

## &lt;R&gt; 36.5 44-pin Package

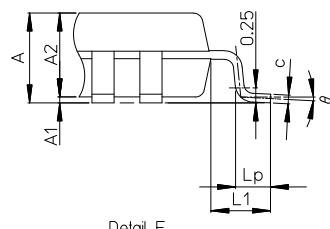
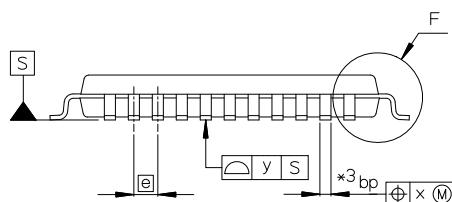
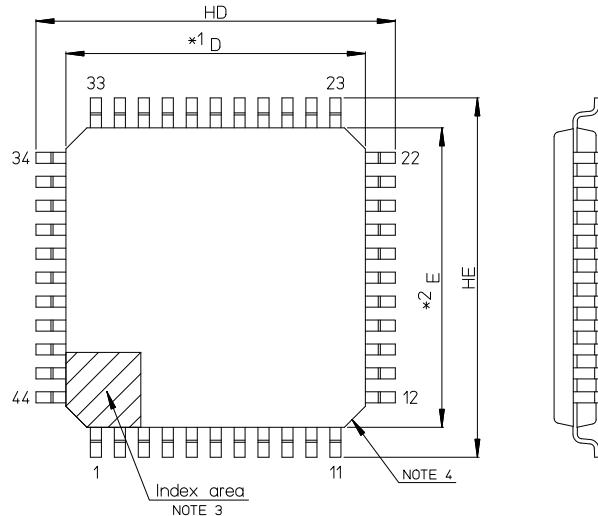
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP44-10x10-0.80	PLQP0044GC-A	P44GB-80-UES-2	0.36



©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP44-10x10-0.80	PLQP00440C-D	—	0.36g

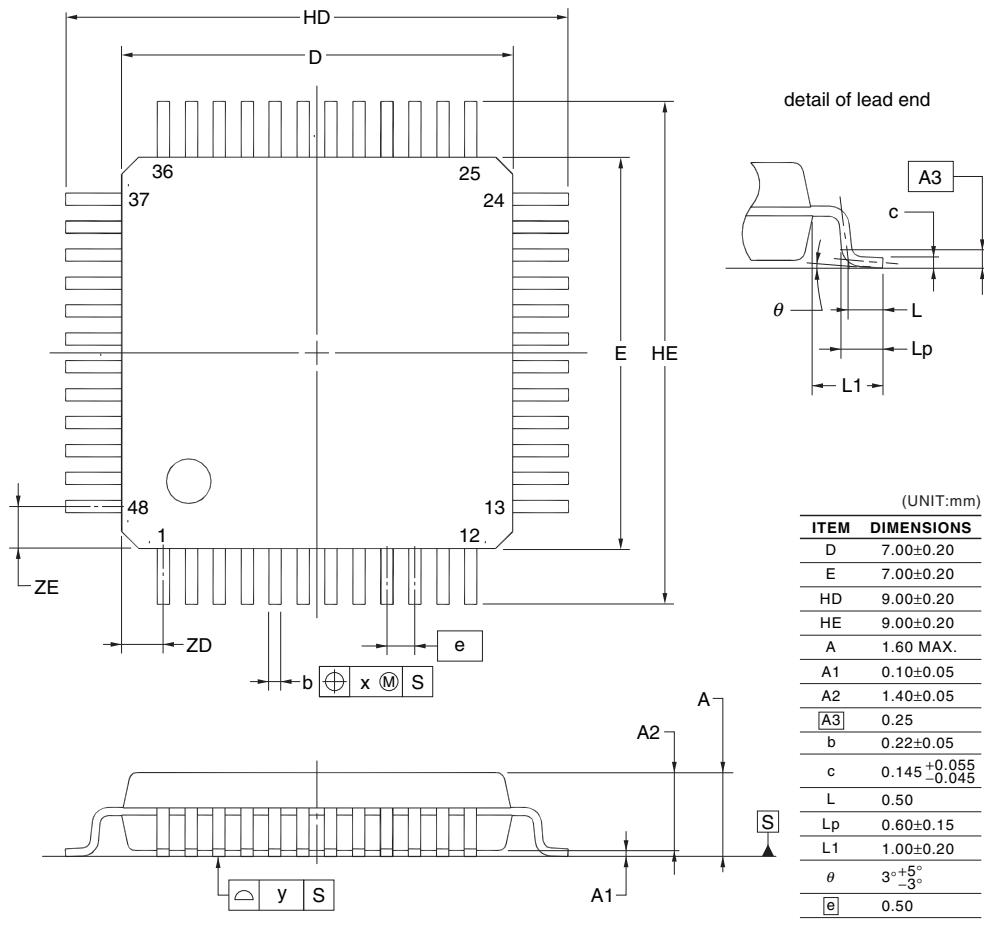


- NOTE)
1. DIMENSIONS '\*1' AND '\*2' DO NOT INCLUDE MOLD FLASH.
  2. DIMENSION '\*3' DOES NOT INCLUDE TRIM OFFSET.
  3. PIN 1 VISUAL INDEX FEATURE MAY VARY, BUT MUST BE LOCATED WITHIN THE HATCHED AREA.
  4. CHAMFERS AT CORNERS ARE OPTIONAL, SIZE MAY VARY.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	9.8	10.0	10.2
E	9.8	10.0	10.2
A2	—	1.4	—
HD	11.8	12.0	12.2
HE	11.8	12.0	12.2
A	—	—	1.6
A1	0.05	—	0.15
bp	0.22	0.37	0.45
c	0.09	—	0.20
θ	0 °	3.5 °	8 °
⊖	—	0.80	—
x	—	—	0.20
y	—	—	0.10
Lp	0.45	0.6	0.75
L1	—	1.0	—

## &lt;R&gt; 36.6 48-pin Package

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LFQFP48-7x7-0.50	PLQP0048KF-A	P48GA-50-8EU-1	0.16

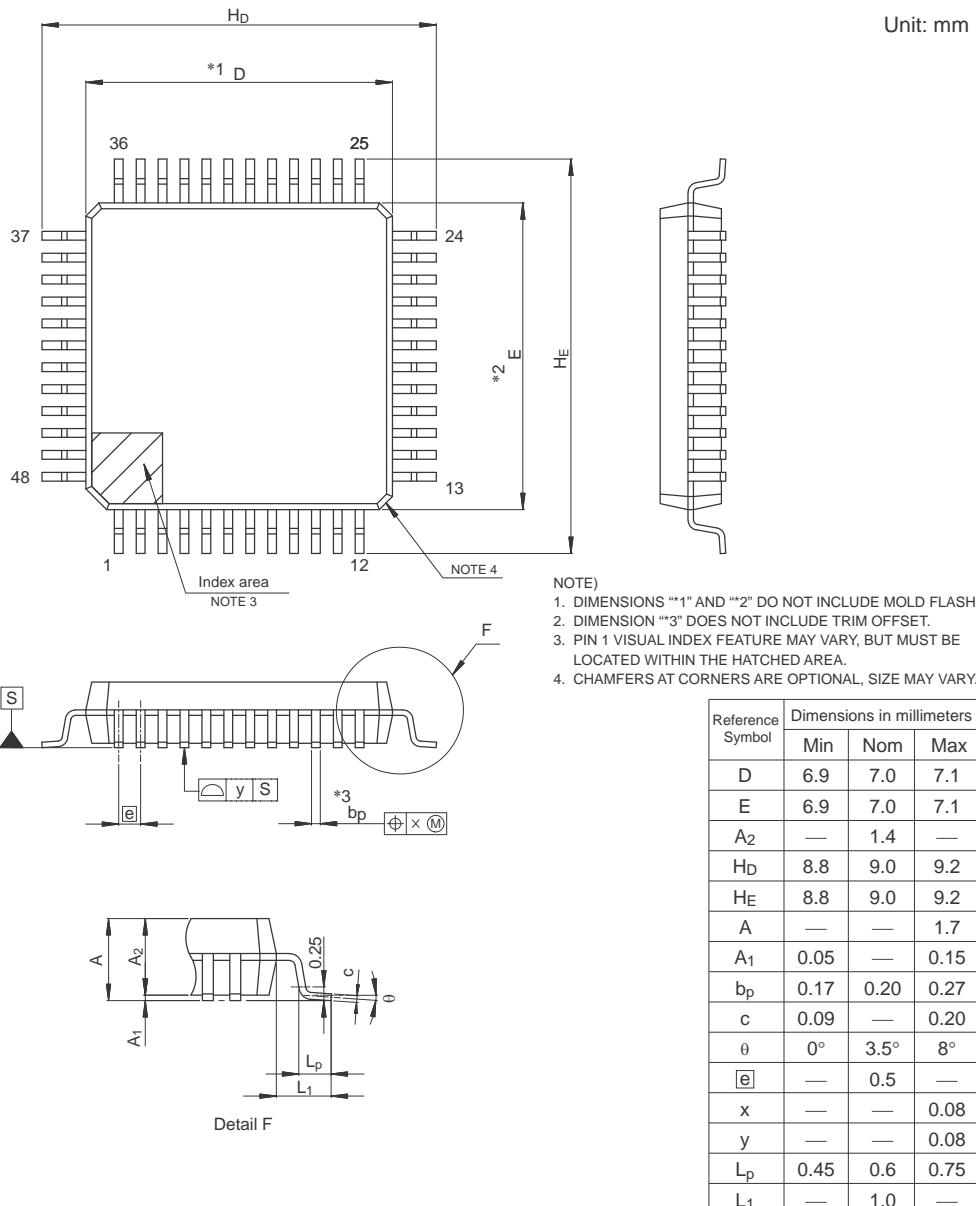
**NOTE**

Each lead centerline is located within 0.08 mm of its true position at maximum material condition.

©2012 Renesas Electronics Corporation. All rights reserved.

<R>

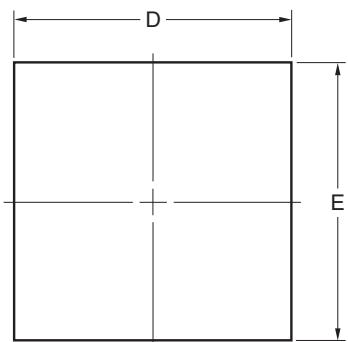
JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-LFQFP48-7x7-0.50	PLQP0048KB-B	—	0.2



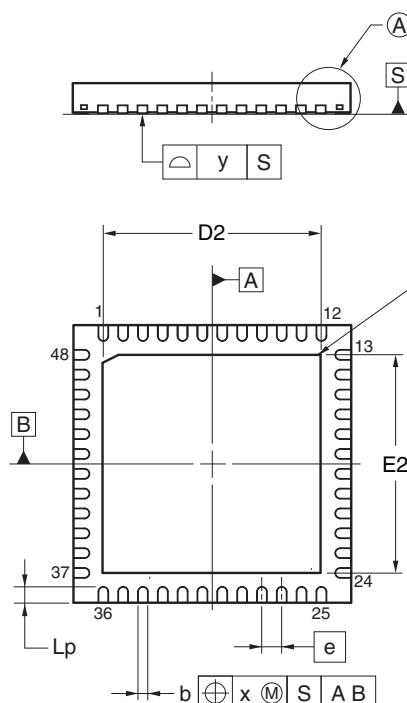
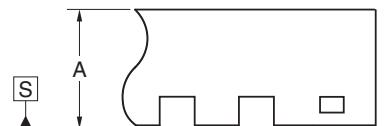
© 2015 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-HWQFN48-7x7-0.50	PWQN0048KB-A	48PQN-A P48K8-50-5B4-5	0.13



DETAIL OF (A) PART



EXPOSED DIE PAD

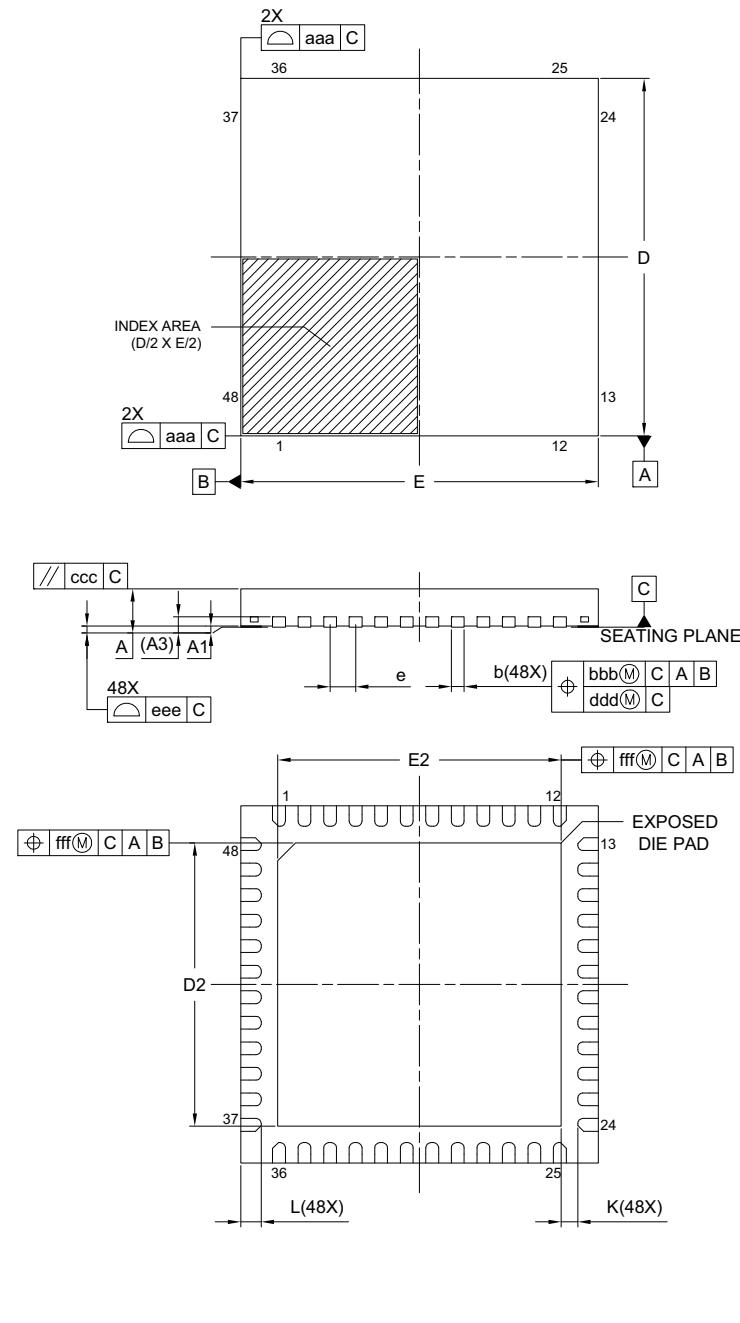
Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	6.95	7.00	7.05
E	6.95	7.00	7.05
A	0.70	0.75	0.80
b	0.18	0.25	0.30
[e]	—	0.50	—
Lp	0.30	0.40	0.50
x	—	—	0.05
y	—	—	0.05

ITEM	D2			E2			
	MIN	NOM	MAX	MIN	NOM	MAX	
EXPOSED DIE PAD VARIATIONS	A	5.45	5.50	5.55	5.45	5.50	5.55

©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

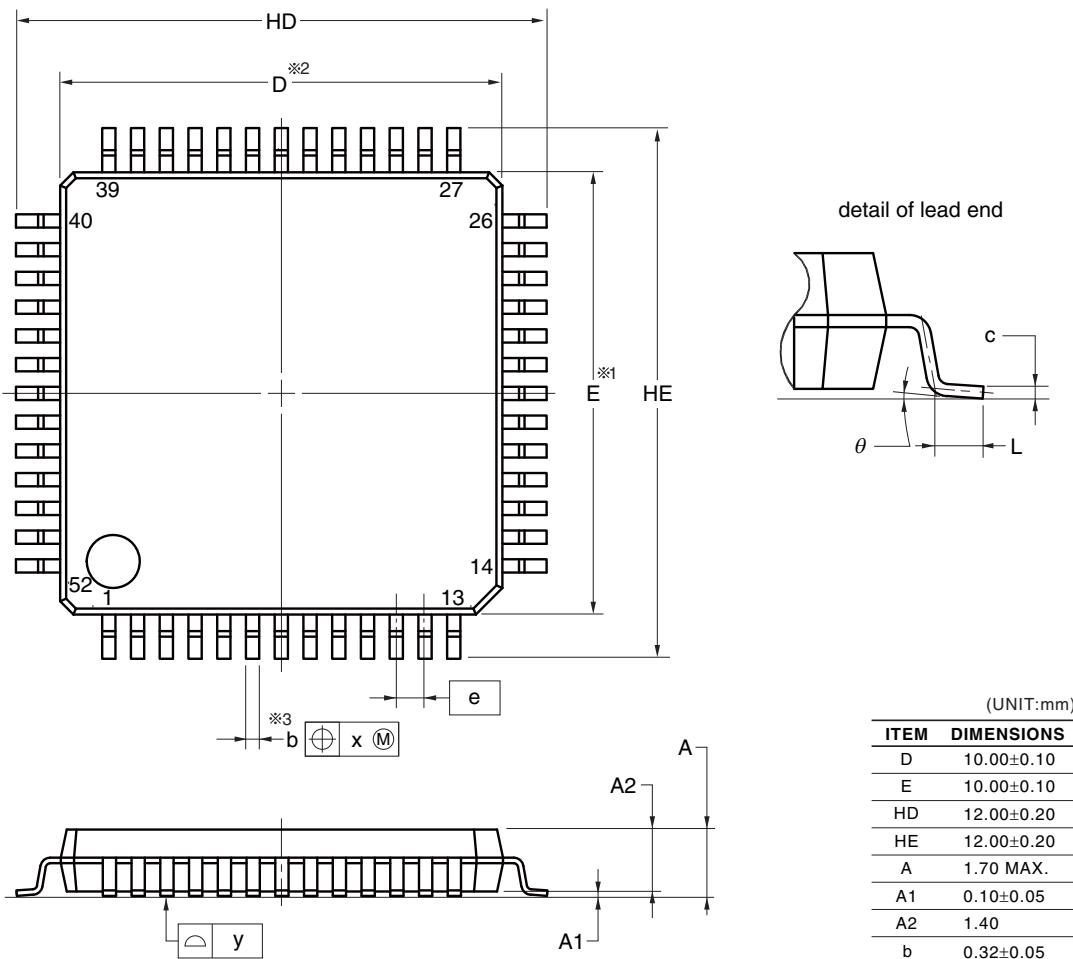
JEITA Package code	RENESAS code	MASS(TYP.)[g]
P-HWQFN048-7x7-0.50	PWQN0048KE-A	0.13



&lt;R&gt;

## 36.7 52-pin Package

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP52-10x10-0.65	PLQP0052JA-A	P52GB-65-GBS-1	0.3



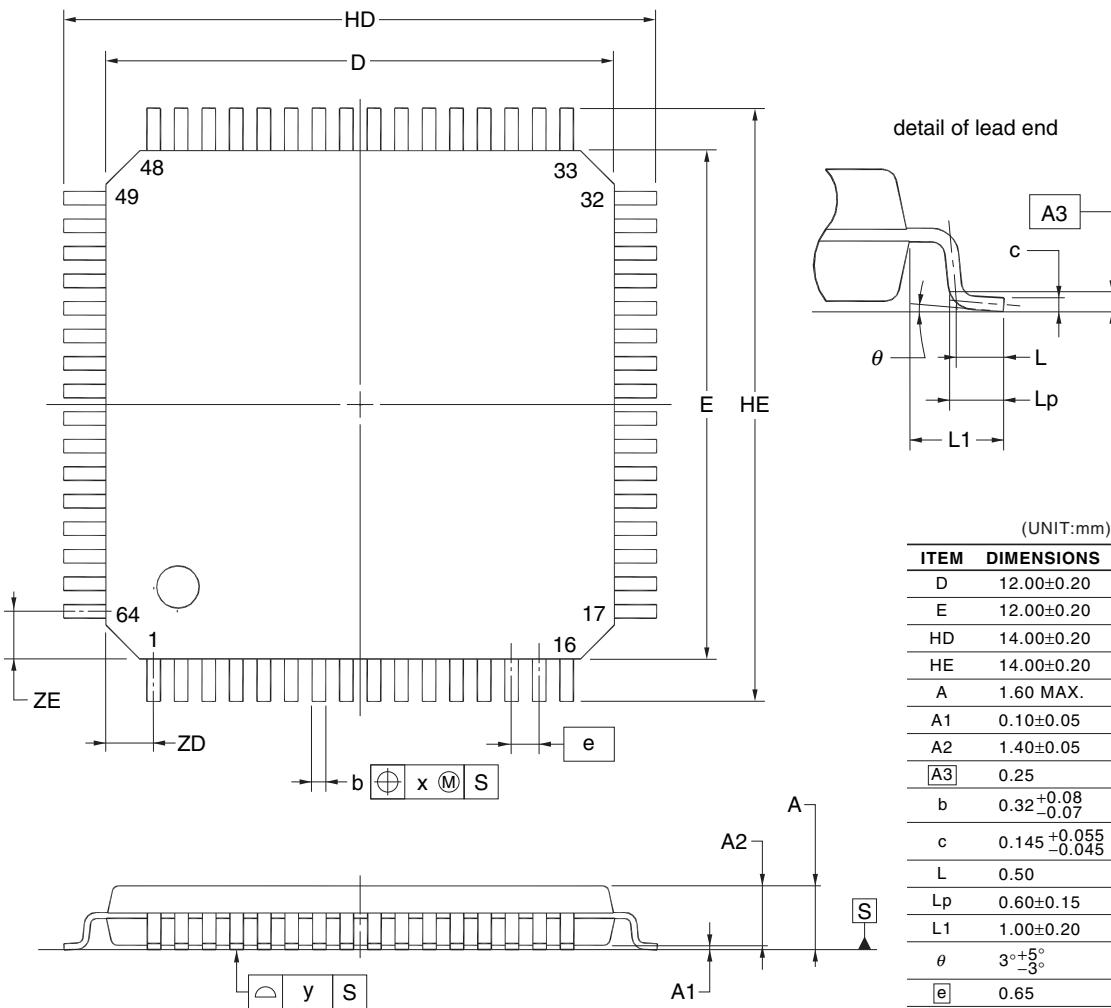
### NOTE

1. Dimensions “\*1” and “\*2” do not include mold flash.
2. Dimension “\*3” does not include trim offset.

© 2012 Renesas Electronics Corporation. All rights reserved.

## &lt;R&gt; 36.8 64-pin Package

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP64-12x12-0.65	PLQP0064JA-A	P64GK-65-UET-2	0.51

**NOTE**

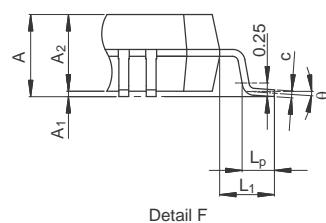
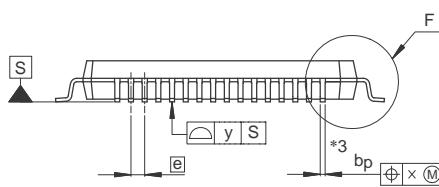
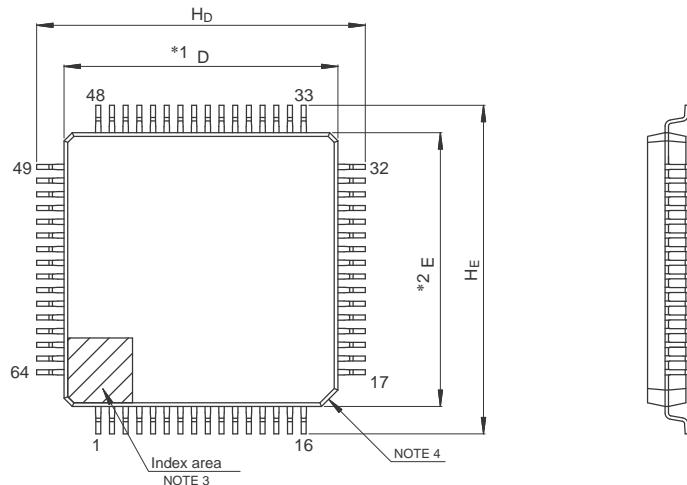
Each lead centerline is located within 0.13 mm of its true position at maximum material condition.

©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-LFQFP64-10x10-0.50	PLQP0064KB-C	—	0.3

Unit: mm



## NOTE)

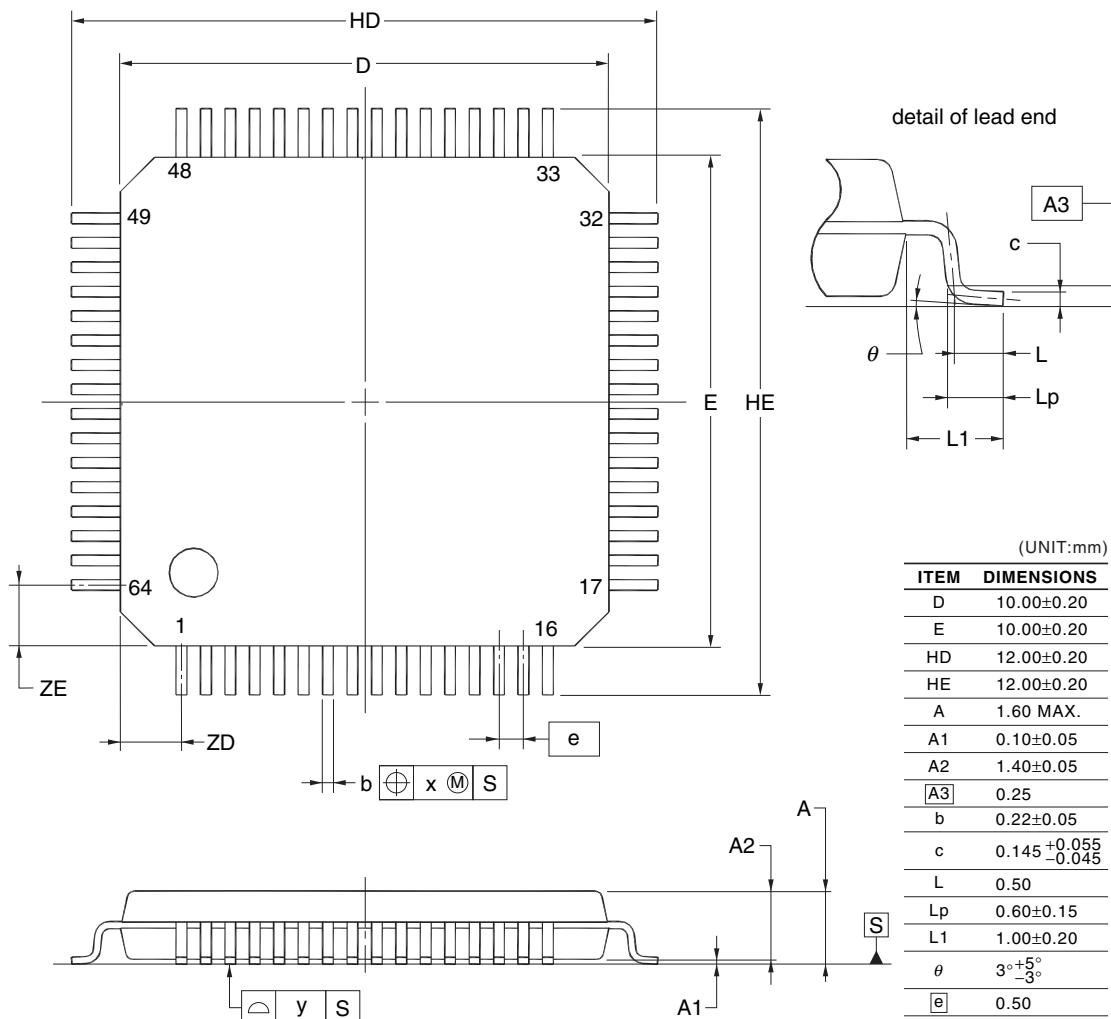
1. DIMENSIONS \*\*1 AND \*\*2 DO NOT INCLUDE MOLD FLASH.
2. DIMENSION \*\*3 DOES NOT INCLUDE TRIM OFFSET.
3. PIN 1 VISUAL INDEX FEATURE MAY VARY, BUT MUST BE LOCATED WITHIN THE HATCHED AREA.
4. CHAMFERS AT CORNERS ARE OPTIONAL, SIZE MAY VARY.

Reference Symbol	Dimensions in millimeters		
	Min	Nom	Max
D	9.9	10.0	10.1
E	9.9	10.0	10.1
A <sub>2</sub>	—	1.4	—
H <sub>D</sub>	11.8	12.0	12.2
H <sub>E</sub>	11.8	12.0	12.2
A	—	—	1.7
A <sub>1</sub>	0.05	—	0.15
b <sub>p</sub>	0.15	0.20	0.27
c	0.09	—	0.20
θ	0°	3.5°	8°
e	—	0.5	—
x	—	—	0.08
y	—	—	0.08
L <sub>p</sub>	0.45	0.6	0.75
L <sub>1</sub>	—	1.0	—

© 2015 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LFQFP64-10x10-0.50	PLQP0064KF-A	P64GB-50-UEU-2	0.35

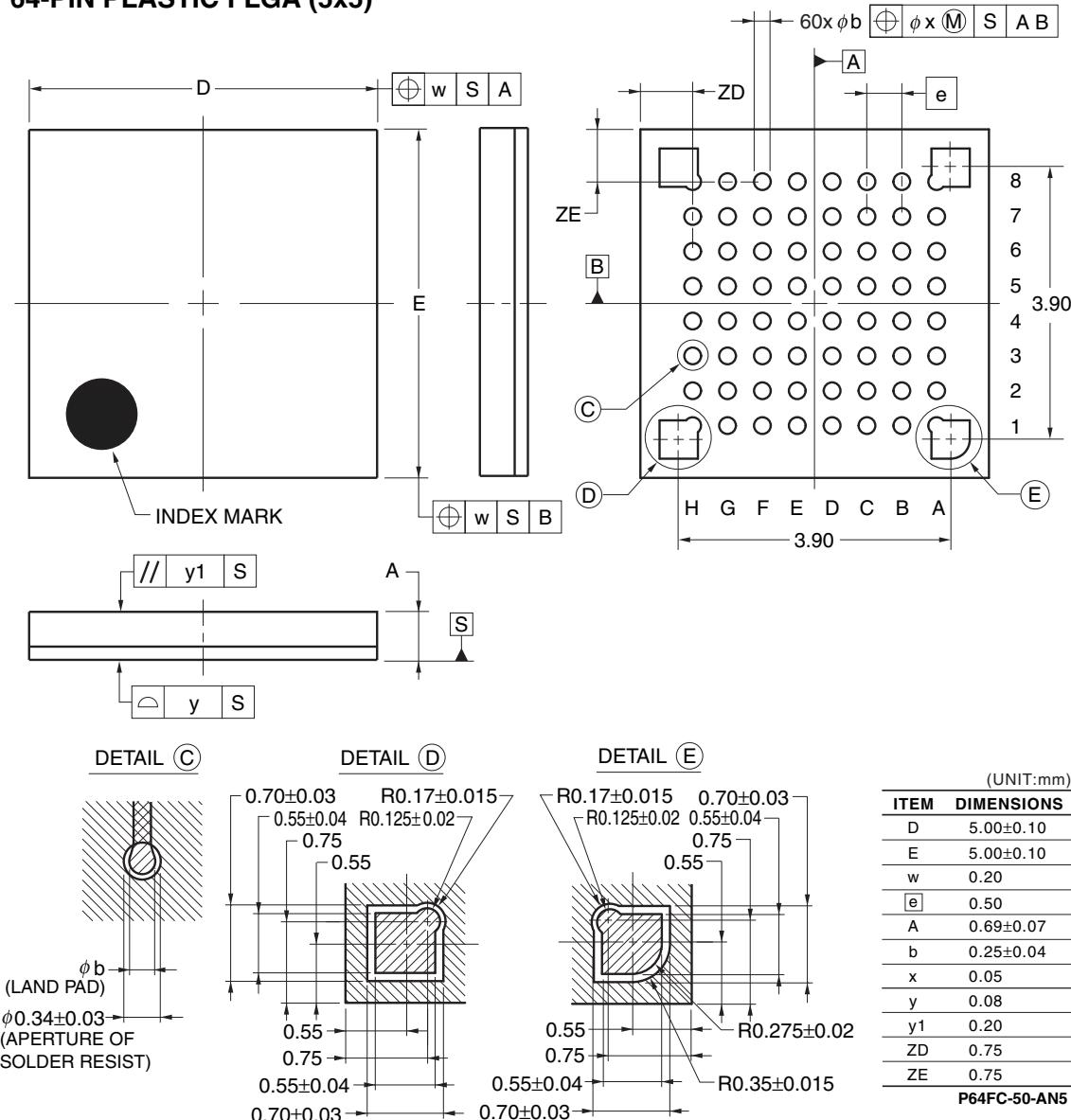
**NOTE**

Each lead centerline is located within 0.08 mm of its true position at maximum material condition.

©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

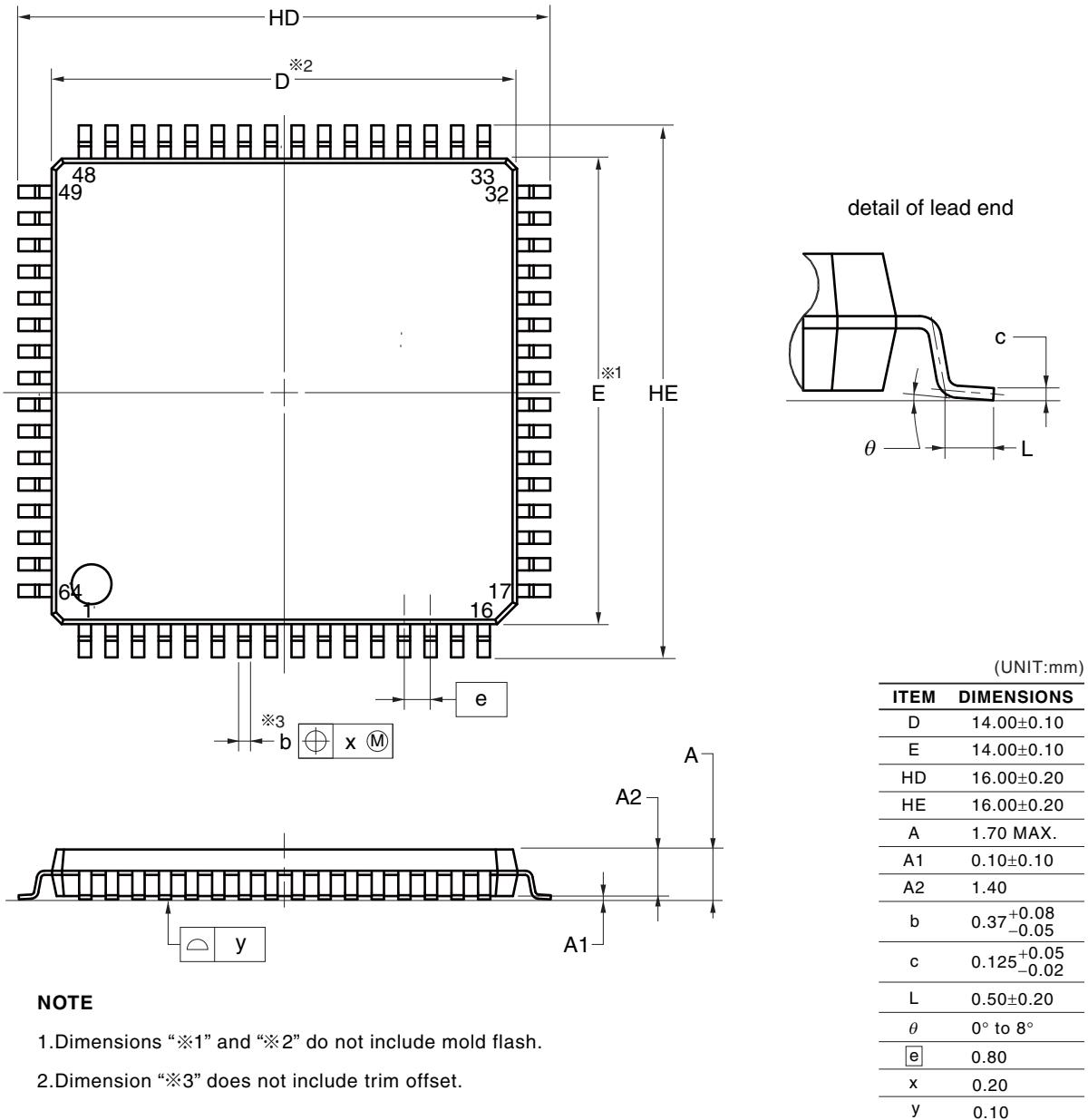
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-WFLGA64-5x5-0.50	PWLG0064KA-A	P64FC-50-AN5	0.037

**64-PIN PLASTIC FLGA (5x5)**

© 2011 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

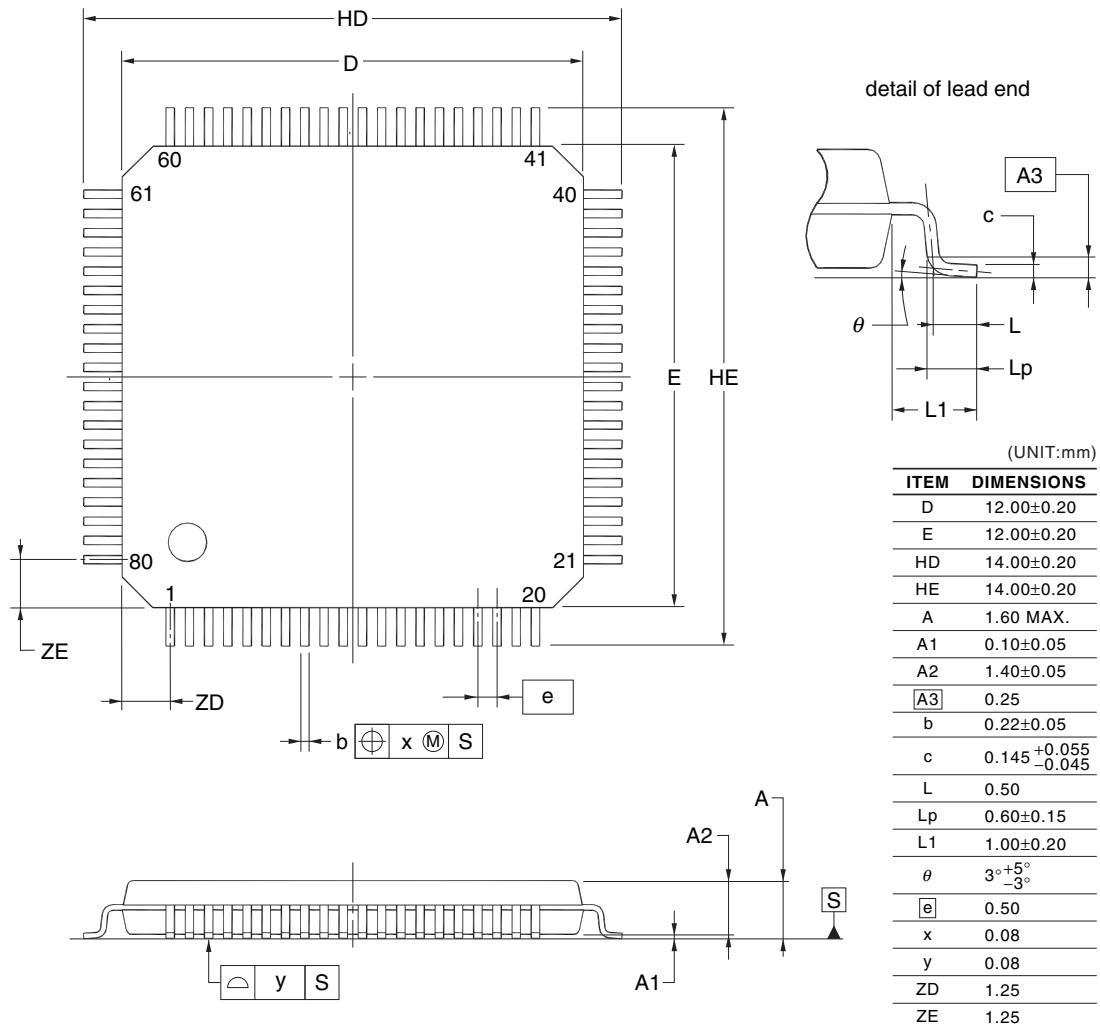
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP64-14x14-0.80	PLQP0064GA-A	P64GC-80-GBW-1	0.7



© 2012 Renesas Electronics Corporation. All rights reserved.

## &lt;R&gt; 36.9 80-pin Package

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LFQFP80-12x12-0.50	PLQP0080KE-A	P80GK-50-8EU-2	0.53

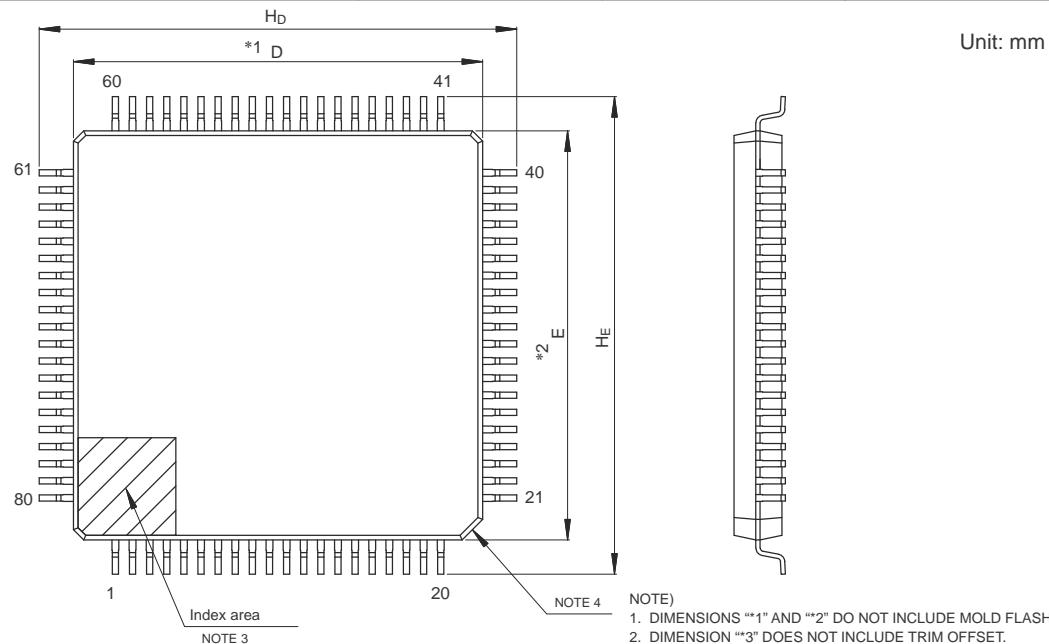
**NOTE**

Each lead centerline is located within 0.08 mm of its true position at maximum material condition.

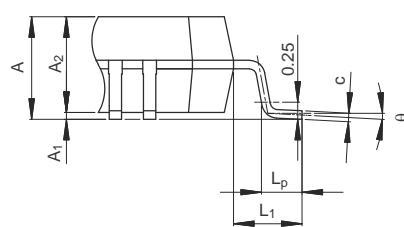
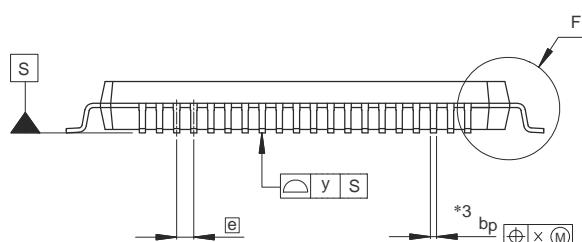
©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-LFQFP80-12x12-0.50	PLQP0080KB-B	—	0.5



- NOTE)
1. DIMENSIONS “\*1” AND “\*2” DO NOT INCLUDE MOLD FLASH.
  2. DIMENSION “\*3” DOES NOT INCLUDE TRIM OFFSET.
  3. PIN 1 VISUAL INDEX FEATURE MAY VARY, BUT MUST BE LOCATED WITHIN THE HATCHED AREA.
  4. CHAMFERS AT CORNERS ARE OPTIONAL, SIZE MAY VARY.

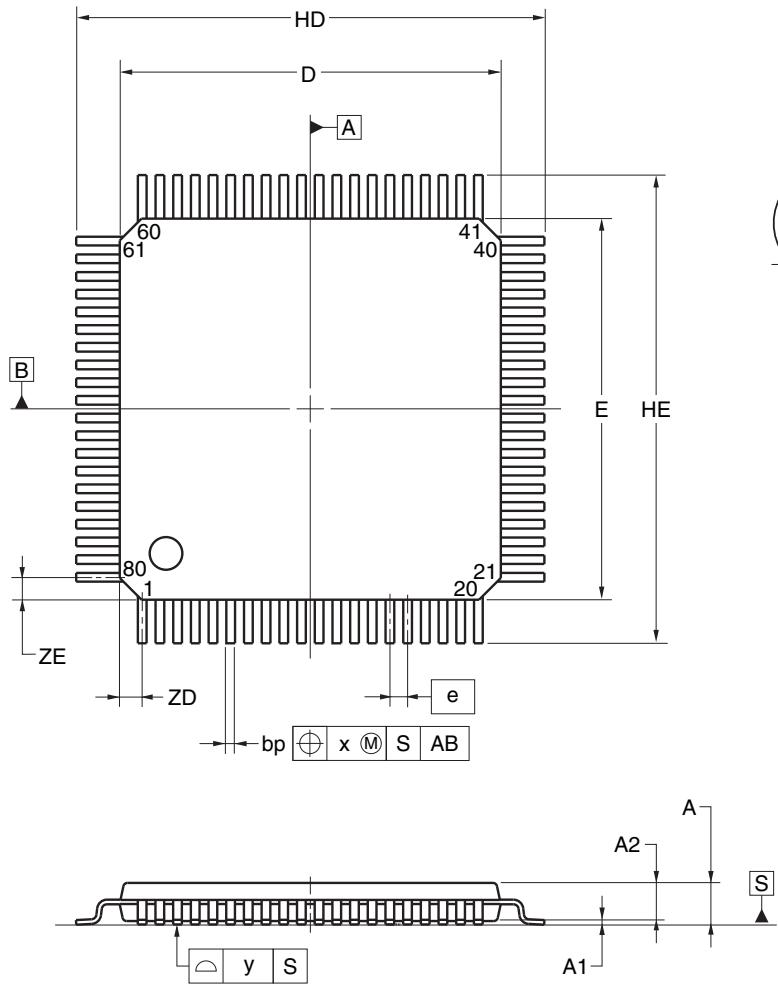


Reference Symbol	Dimensions in millimeters		
	Min	Nom	Max
D	11.9	12.0	12.1
E	11.9	12.0	12.1
A <sub>2</sub>	—	1.4	—
H <sub>D</sub>	13.8	14.0	14.2
H <sub>E</sub>	13.8	14.0	14.2
A	—	—	1.7
A <sub>1</sub>	0.05	—	0.15
b <sub>p</sub>	0.15	0.20	0.27
c	0.09	—	0.20
θ	0°	3.5°	8°
[e]	—	0.5	—
x	—	—	0.08
y	—	—	0.08
L <sub>p</sub>	0.45	0.6	0.75
L <sub>1</sub>	—	1.0	—

© 2017 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP80-14x14-0.65	PLQP0080JB-E	P80GC-65-UBT-2	0.69



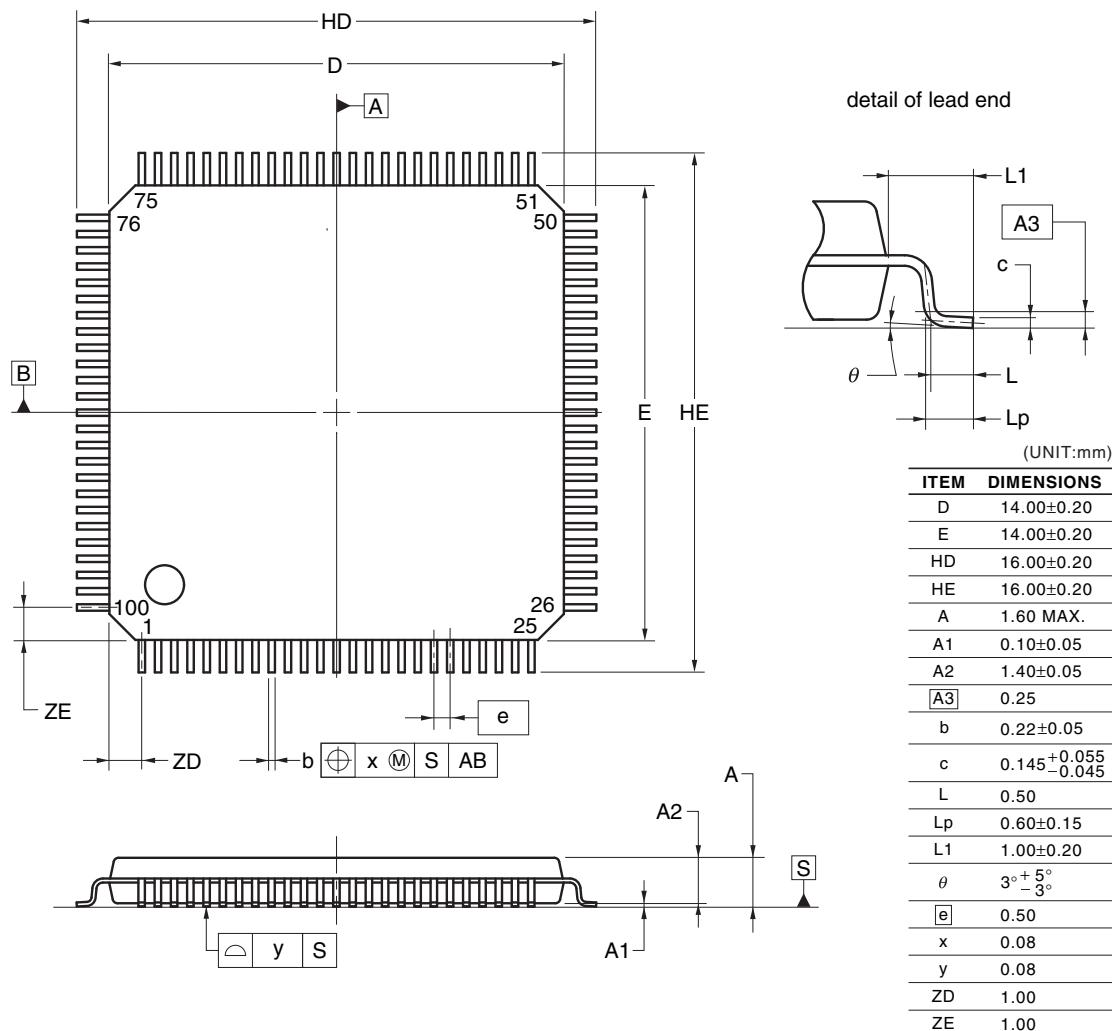
Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	13.80	14.00	14.20
E	13.80	14.00	14.20
HD	17.00	17.20	17.40
HE	17.00	17.20	17.40
A	—	—	1.70
A1	0.05	0.125	0.20
A2	1.35	1.40	1.45
<b>A3</b>	—	0.25	—
bp	0.26	0.32	0.38
c	0.10	0.145	0.20
L	—	0.80	—
Lp	0.736	0.886	1.036
L1	1.40	1.60	1.80
$\theta$	0°	3°	8°
<b>e</b>	—	0.65	—
x	—	—	0.13
y	—	—	0.10
ZD	—	0.825	—
ZE	—	0.825	—

© 2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

## 36.10 100-pin Package

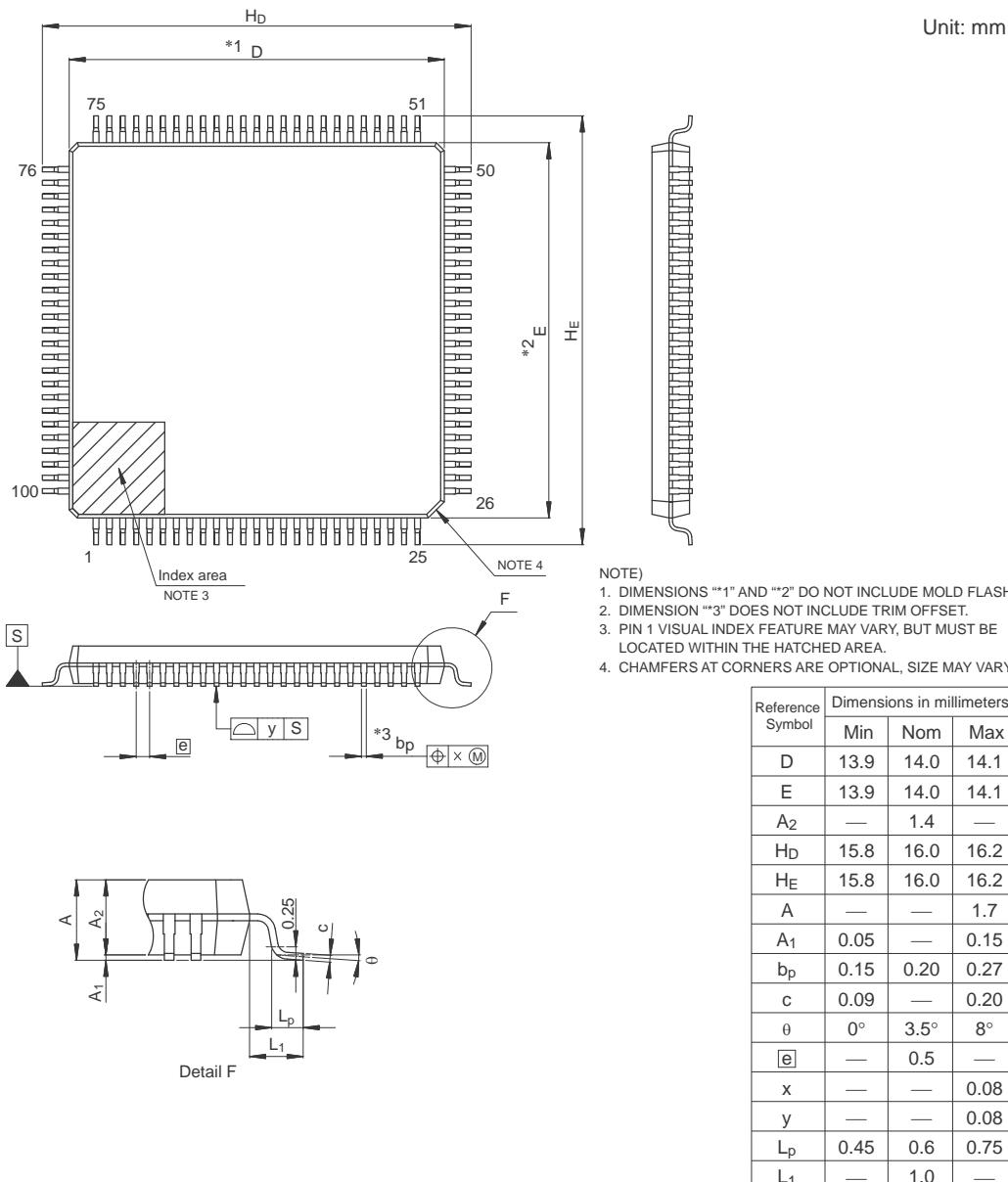
JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP) [g]
P-LFQFP100-14x14-0.50	PLQP0100KE-A	P100GC-50-GBR-1	0.69



©2012 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

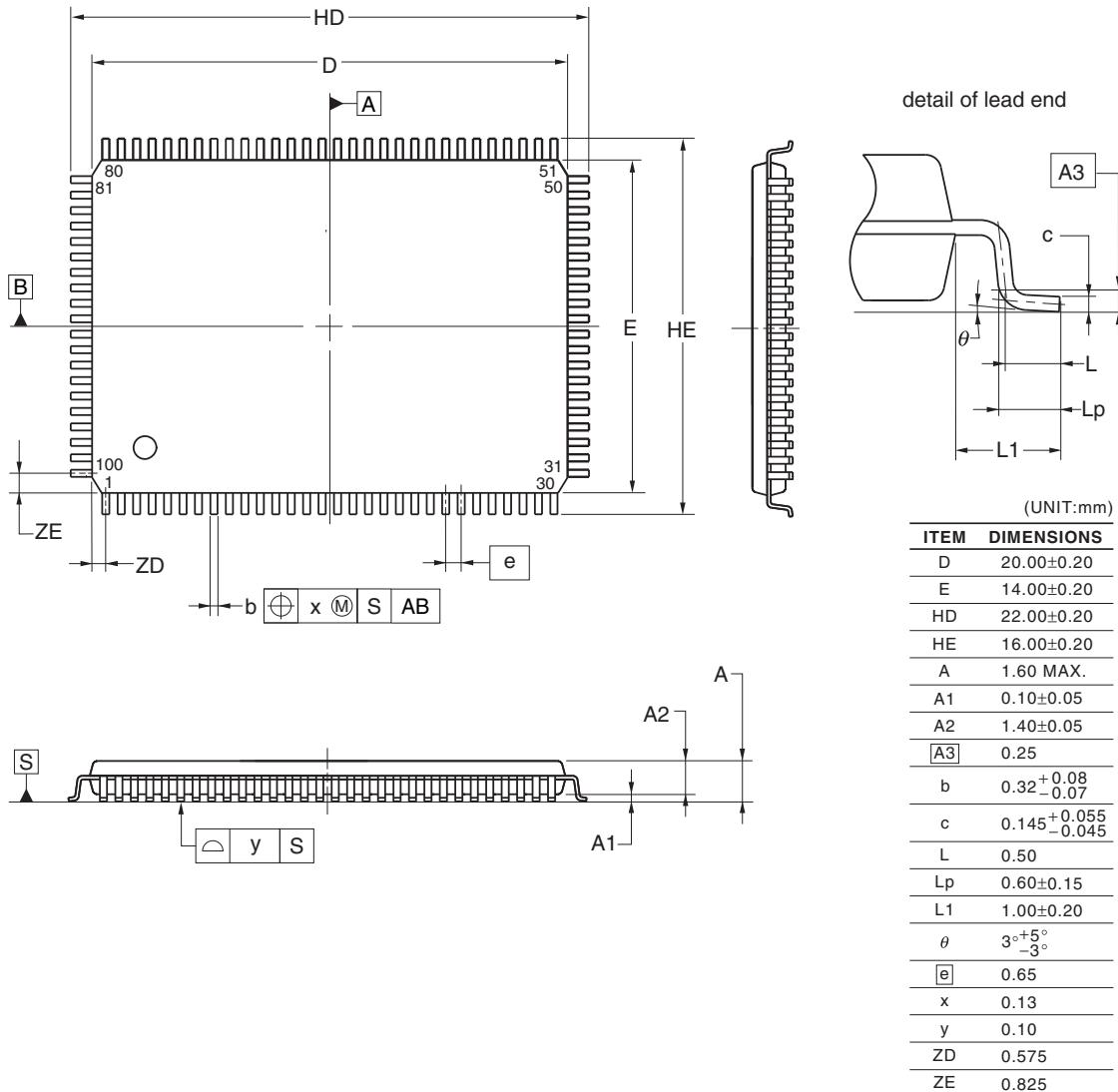
JEITA Package Code	RENESAS Code	Previous Code	MASS (Typ) [g]
P-LFQFP100-14x14-0.50	PLQP0100KB-B	—	0.6



© 2015 Renesas Electronics Corporation. All rights reserved.

&lt;R&gt;

JEITA Package Code	RENESAS Code	Previous Code	MASS (TYP.) [g]
P-LQFP100-14x20-0.65	PLQP0100JC-A	P100GF-65-GBN-1	0.92



© 2012 Renesas Electronics Corporation. All rights reserved.

## APPENDIX A REVISION HISTORY

### A.1 Major Revisions in This Edition

Page	Description	Classification
<b>CHAPTER 1 OUTLINE</b>		
p.5	Modification of Figure 1-1 Part Number, Memory Size, and Package of RL78/G14	(d)
p.6 to 9	Addition of title and modification of description in Table 1-1 List of Ordering Part Numbers	(d)
<b>CHAPTER 36 PACKAGE DRAWINGS</b>		
p.1296 to 1320	Addition and modification of all in CHAPTER 36 PACKAGE DRAWINGS	(d)

**Remark** "Classification" in the above table classifies revisions as follows.

- (a): Error correction, (b): Addition/change of specifications, (c): Addition/change of description or note,
- (d): Addition/change of package, part number, or management division, (e): Addition/change of related documents

## A.2 Revision History of Preceding Editions

Here is the revision history of the preceding editions. Chapter indicates the chapter of each edition.

(1/26)

Edition	Description	Chapter
Rev. 3.30	Modification of pin type of P60 and P61 in 2.1.1 30-pin (Code Flash Memory 16 KB to 64 KB)	CHAPTER 2 PIN FUNCTIONS
	Modification of pin type of P60 to P62 in 2.1.2 32-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P62 in 2.1.3 36-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P62 in 2.1.4 40-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P63 in 2.1.5 44-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P63 in 2.1.6 48-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P63 in 2.1.7 52-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 to P63 in 2.1.8 64-pin (Code Flash Memory 16 KB to 64 KB)	
	Modification of pin type of P60 and P61 in 2.1.9 30-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 to P62 in 2.1.10 32-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 to P62 in 2.1.11 36-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 and P61 in 2.1.12 40-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 to P63 in 2.1.13 44-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 to P63 in 2.1.14 48-pin (Code Flash Memory 96 KB to 512 KB)	
	Modification of pin type of P60 to P63 in 2.1.15 52-pin (Code Flash Memory 96 KB to 256 KB)	
	Modification of pin type of P60 to P63 in 2.1.16 64-pin (Code Flash Memory 96 KB to 512 KB)	
	Modification of pin type of P60 to P63 in 2.1.17 80-pin	
	Modification of pin type of P60 to P63 in 2.1.18 100-pin	
	Addition of caution in Figure 2 - 8 Pin Block Diagram of Pin Type 7-1-4	
	Addition of caution in Figure 2 - 10 Pin Block Diagram of Pin Type 7-1-8	
	Addition of caution in Figure 2 - 12 Pin Block Diagram of Pin Type 7-3-4	
	Addition of caution in Figure 2 - 14 Pin Block Diagram of Pin Type 7-6-8	
	Addition of caution in Figure 2 - 15 Pin Block Diagram of Pin Type 8-1-3	
	Addition of cautions 1 and 2 in Figure 2 - 16 Pin Block Diagram of Pin Type 8-1-4	
	Addition of caution in Figure 2 - 17 Pin Block Diagram of Pin Type 8-1-7	
	Addition of cautions 1 and 2 in Figure 2 - 18 Pin Block Diagram of Pin Type 8-1-8	
	Addition of cautions 1 and 2 in Figure 2 - 19 Pin Block Diagram of Pin Type 8-3-4	
	Addition of caution in Figure 2 - 20 Pin Block Diagram of Pin Type 8-6-6	

(2/26)

Edition	Description	Chapter
Rev. 3.30	Addition of cautions 1 and 2 in Figure 2 - 21 Pin Block Diagram of Pin Type 8-6-8 Modification of the pin type (12-1-2 → 12-1-5) and addition of caution in Figure 2 - 22 Pin Block Diagram of Pin Type 12-1-5	CHAPTER 2 PIN FUNCTIONS
	Modification of vector table addresses in Tables 3 - 6 and 3 - 7 Vector Table Modification of addresses (00000H and 00001H) in (1) Program counter (PC) of 3.2.1 Control registers	CHAPTER 3 CPU ARCHITECTURE
	Addition of F0139H in Table 3 - 16 Extended Special Function Register (2nd SFR) List (3/9)	
	Modification of note 1 in 4.2.8 Port 7 Modification of the setting of POMx register in Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (1/6)	CHAPTER 4 PORT FUNCTIONS
	Modification of Table 4 - 19 Setting Examples of Registers When Using P130 to P147 Pin Function	
	Modification of caution 6 in Figure 5 - 4 Format of Clock operation status control register (CSC)	CHAPTER 5 CLOCK GENERATOR
	Modification of Table 5 - 8 Changing CPU Clock (1/2) Modification of description in 5.6.7 Conditions before clock oscillation is stopped	
	Addition of remark 2 in (1) X1 oscillation of 5.7 Resonator and Oscillator Constants	
	Addition of remark in (2) XT1 oscillation (crystal resonator) of 5.7 Resonator and Oscillator Constants	
	Modification of Figure 6 - 66 Operation Procedure When Input Signal High-/Low-Level Width Measurement Function Is Used	CHAPTER 6 TIMER ARRAY UNIT
	Addition of note 2 to TOA1 and TOA0 bits in Figure 8 - 10 Format of Timer RD output control register (TRDOCR) [Output Compare Function] Addition of note 2 to TOA1 and TOA0 bits and modification of description for TOC0 bit in Figure 8 - 11 Format of Timer RD output control register (TRDOCR) [PWM Function]	CHAPTER 8 TIMER RD
	Addition of Figure 8 - 12 Format of Timer RD output control register (TRDOCR) [Reset Synchronous PWM Mode, Complementary PWM Mode]	
	Addition of note 2 to TOA0 bit in Figure 8 - 13 Format of Timer RD output control register (TRDOCR) [PWM3 Mode]	
	Addition of description of TRDCR1 in Figure 8 - 19 Format of Timer RD control register i (TRDCR <i>i</i> )( <i>i</i> = 0, 1)[Complementary PWM Mode]	
	Deletion of description of TRD1 in Figure 8 - 32 Format of Timer RD counter 0(TRD0) [Reset Synchronous PWM Mode and PWM3 Mode]	
	Deletion of description of TRD1 in Figure 8 - 33 Format of Timer RD counter 0 (TRD0) [Complementary PWM Mode (TRD0)]	
	Deletion of description of TRD0 in Figure 8 - 34 Format of Timer RD counter 1 (TRD1) [Complementary PWM Mode (TRD1)]	
	Modification of Figure 8 - 57 Block Diagram of Reset Synchronous PWM Mode	
	Modification of Figure 8 - 59 Block Diagram of Complementary PWM Mode	
	Deletion of description in Figure 14 - 4 Timing Chart When A/D Voltage Comparator Is Used	CHAPTER 14 A/D CONVERTER
	Modification of Figure 14 - 19 Example of Software Trigger Mode (Select Mode, Sequential Conversion Mode) Operation Timing	
	Modification of Figure 14 - 20 Example of Software Trigger Mode (Select Mode, One-Shot Conversion Mode) Operation Timing	

(3/26)

Edition	Description	Chapter
Rev. 3.30	Modification of Figure 14 - 21 Example of Software Trigger Mode (Scan Mode, Sequential Conversion Mode) Operation Timing	CHAPTER 14 A/D CONVERTER
	Modification of Figure 14 - 22 Example of Software Trigger Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of Figure 14 - 23 Example of Hardware Trigger No-Wait Mode (Select Mode, Sequential Conversion Mode) Operation	
	Modification of Figure 14 - 24 Example of Hardware Trigger No-Wait Mode (Select Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of Figure 14 - 25 Example of Hardware Trigger No-Wait Mode (Scan Mode, Sequential Conversion Mode) Operation Timing	
	Modification of Figure 14 - 26 Example of Hardware Trigger No-Wait Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of Figure 14 - 27 Example of Hardware Trigger Wait Mode (Select Mode, Sequential Conversion Mode) Operation Timing	
	Modification of Figure 14 - 28 Example of Hardware Trigger Wait Mode (Select Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of Figure 14 - 29 Example of Hardware Trigger Wait Mode (Scan Mode, Sequential Conversion Mode) Operation Timing	
	Modification of Figure 14 - 30 Example of Hardware Trigger Wait Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of description in Figure 14 - 31 Setting up Software Trigger Mode	
	Modification of description in Figure 14 - 32 Setting up Hardware Trigger No-Wait Mode	
	Modification of description in Figure 14 - 34 Setup when temperature sensor output voltage/internal reference voltage is selected	
	Modification of description in Figure 14 - 35 Setting up Test Mode	
	Modification of description in Figure 14 - 39 Flowchart for Setting up SNOOZE Mode	
	Modification of description for PRSn bit in Figure 18 - 15 Format of IICA control register n1 (IICCTLn1) (2/2)	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of vector table addresses in Tables 21 – 1 to 21 - 4 Interrupt Source List	CHAPTER 21 INTERRUPT FUNCTIONS
	Modification of addresses (00000H and 00001H)	CHAPTER 24 RESET FUNCTION
	Modification of addresses in (00000H, 00001H) in Table 24 - 2 Hardware Statuses After Reset Acknowledgment	
	Modification of notes 3 and 4 in Figure 25 – 2 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (1/3)	CHAPTER 25 POWER-ON-RESET CIRCUIT
	Modification of note 3 in Figure 25 - 3 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (2/3)	
	Modification of description in the table in 26.1 Functions of Voltage Detector	CHAPTER 26 VOLTAGE DETECTOR
	Title of Figure 26 - 4 Format of User Option Byte (000C1H/010C1H) corrected from Table 26 - 1 Format of User Option Byte (000C1H/010C1H)	
	Addition of note in Table 31 - 1 On-Chip Debug Security ID	CHAPTER 31 ON-CHIP DEBUG FUNCTION
	Addition of maximum values in (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products of 35.3.2 Supply current characteristics	CHAPTER 35 ELECTRICAL SPECIFICATIONS (G: INDUSTRIAL APPLICATIONS TA = -40 to +105°C)

(4/26)

Edition	Description	Chapter
Rev. 3.20	Deletion of R5F104JK and R5F104JL from the list of ROM and RAM capacities and modification of <b>note</b>	CHAPTER 1 OUTLINE
	Deletion of ordering part numbers of R5F104JK and R5F104JL from 52-pin plastic LQFP package in <b>1.2 Ordering Information</b>	
	Deletion of <b>note 2</b> in <b>1.2 Ordering Information</b>	
	Deletion of <b>note 2</b> in <b>1.3.7 52-pin products</b>	
	Modification of description in <b>1.6 Outline of Functions</b>	
	Deletion of description of 52-pin in <b>1.6 Outline of Functions</b>	
	Modification of note of <b>1.6 Outline of Functions</b>	
	Renamed to <b>2.1.15 52-pin (Code Flash Memory 96 KB to 256 KB)</b> and deletion of <b>note 3</b>	CHAPTER 2 PIN FUNCTIONS
	Modification of title of figures 3 – 9 and 3 – 10 in <b>3.1 Memory Space</b>	CHAPTER 3 CPU ARCHITECTURE
	Modification of <b>remark</b> of tables 3 - 3 and 3 - 4	
	Modification of description in table 3 – 5	
	Modification of description in table 3 – 8	
	Modification of <b>cautions 4 and 5</b> in (3) Stack pointer (SP) in <b>3.2.1 Control registers</b>	
	Modification of <b>note 2</b> of table 4 - 7	CHAPTER 4 PORT FUNCTIONS
	Addition of description in figure 6 - 12	CHAPTER 6 TIMER ARRAY UNIT
	Modification of <b>caution</b> in <b>6.9.1 Operation as one-shot pulse output function</b>	
	Modification of error in figure 8 – 58	CHAPTER 8 TIMER RD
	Modification of <b>note</b> of figure 17 - 19	CHAPTER 17 SERIAL ARRAY UNIT
	Addition of description in <b>18.3.6 IICA low-level width setting register n (IICWL<sub>n</sub>)</b>	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of <b>cautions 3 and 4</b> of figure 19 – 2	CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)
	Modification of description in <b>19.5.2 Allocation of DTC Control Data Area and DTC Vector Table Area</b>	
	Addition of description in <b>19.5.3 DTC Pending Instruction</b>	
	Addition of <b>caution</b> in <b>21.4.4 Interrupt servicing during division instruction</b>	CHAPTER 21 INTERRUPT FUNCTIONS
	Addition of description in <b>21.4.5 Interrupt request hold</b>	
	Modification of <b>note</b> of figure 27 – 11	CHAPTER 27 SAFETY FUNCTIONS
	Modification of <b>note 1</b> of figure 31 – 2	CHAPTER 31 ON-CHIP DEBUG FUNCTION
	Addition of <b>caution</b> to table 33 - 16	CHAPTER 33 INSTRUCTION SET
	Modification of <b>34.3.2 Supply current characteristics</b>	CHAPTER 34 ELECTRICAL SPECIFICATIONS (TA = -40 to +85°C)
	Modification of <b>35.3.2 Supply current characteristics</b>	CHAPTER 35 ELECTRICAL SPECIFICATIONS (G: INDUSTRIAL APPLICATIONS TA = -40 to +105°C)
	Modification of part number in <b>36.7 52-pin products</b>	CHAPTER 36 PACKAGE DRAWINGS

(5/26)

Edition	Description	Chapter
Rev. 3.10	Modification of note of ROM, RAM capacities	CHAPTER 1 OUTLINE
	Modification of Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14	
	Addition and modification of ordering part number	
	Addition of description to 1.3.1 30-pin products to 1.3.10 100-pin products	
	Modification of description and note in 1.6 Outline of Functions	
	Modification of note 1 of Figures 3 - 1 to 3 - 10 Memory Map	CHAPTER 3 CPU ARCHITECTURE
	Modification of cautions 2 and 3 in 3.1.3 Internal data memory space	
	Modification of description in Figure 3 - 43 Example of CALL, CALLT	
	Addition of description in 5.1 Functions of Clock Generator	CHAPTER 5 CLOCK GENERATOR
	Addition of caution to <5> in 5.6.2 Example of setting X1 oscillation clock	
	Modification of description in Figure 6 - 19	CHAPTER 6 TIMER ARRAY UNIT
	Addition of note in Table 13 - 3 Setting of Overflow Time of Watchdog Timer	CHAPTER 13 WATCHDOG TIMER
	Addition of 13.5 Restrictions on the Watchdog Timer	
	Addition of description in Table 15 - 1 Output Pin of D/A Converter	CHAPTER 15 D/A CONVERTER
	Modification of Figure 17 - 74 Timing Chart of SNOOZE Mode Operation (once startup) (Type 1: DAPmn = 0, CKPmn = 0)	CHAPTER 17 SERIAL ARRAY UNIT
	Modification of Figure 17 - 76 Timing Chart of SNOOZE Mode Operation (continuous startup) (Type 1: DAPmn = 0, CKPmn = 0)	
	Addition of caution 5 in 17.7.3 SNOOZE mode function	
	Modification of Figure 17 - 123 Timing Chart of SNOOZE Mode Operation (EOCm1 = 0, SSECm = 0/1)	
	Modification of Figure 17 - 124 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 0)	
	Modification of Figure 17 - 126 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 1)	
	Modification of description in 18.5.14 Communication reservation	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of note 1 in Figure 18 - 34 Communication Reservation Protocol	
	Modification of note in Figure 18 - 37 Master Operation in Multi-Master System (2/3)	
	Addition of description to the beginning	CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)
	Modification of note in Table 19 - 1 DTC Specifications (2/2)	
	Modification of Figure 19 - 3 Control Data Allocation	
	Addition of Figure 19 - 4 Start Address of Control Data and Vector Table	
	Modification of Figure 19 - 16 Example 1 of using normal mode: Consecutively capturing A/D conversion results	
	Addition of target products to the beginning	CHAPTER 34 ELECTRICAL SPECIFICATIONS (TA = -40 to +85°C)
	Modification of conditions of Slp setup time	
	Renamed to 34.7 RAM Data Retention Characteristics, and modification of note	
	Addition of target products to the beginning	CHAPTER 35 ELECTRICAL SPECIFICATIONS (G: INDUSTRIAL APPLICATIONS TA = -40 to +105°C)
	Modification of conditions of Slp setup time	
	Renamed to 35.7 RAM Data Retention Characteristics, and modification of note	
	Addition of part number in 36.3 36-pin products	CHAPTER 36 PACKAGE DRAWINGS
	Addition of part number in 36.8 64-pin products	

(6/26)

Edition	Description	Chapter
Rev. 3.00	Addition of products with maximum 512 KB flash ROM and 48 KB RAM	ALL
	Modification of 1.1 Features	CHAPTER 1 OUTLINE
	Modification of ROM, RAM capacities and addition of note 3	
	Modification of Figure 1 - 1 Part Number, Memory Size, and Package of RL78/G14	
	Addition of part number	
	Modification of 1.3.6 48-pin products	
	Modification of 1.3.7 52-pin products	
	Modification of 1.3.8 64-pin products	
	Modification of 1.3.9 80-pin products	
	Modification of 1.3.10 100-pin products	
	Modification of operating ambient temperature in 1.6 Outline of Functions	
	Addition of table of 48-pin, 52-pin, 64-pin products (code flash memory 384 KB to 512 KB)	
	Addition of table of 80-pin, 100-pin products (code flash memory 384 KB to 512 KB)	
	Modification from after reset to after reset release	CHAPTER 2 PIN FUNCTIONS
	Modification of 2.1.14 48-pin (Code Flash Memory 96 KB to 512 KB)	
	Modification of 2.1.15 52-pin (Code Flash Memory 96 KB to 512 KB)	
	Modification of 2.1.16 64-pin (Code Flash Memory 96 KB to 512 KB)	
	Modification of 2.1.17 80-pin	
	Modification of 2.1.18 100-pin	
	Deletion of old Figure 2 - 5 Pin Block Diagram of Pin Type 4-3-1	
	Deletion of old Figure 2 - 7 Pin Block Diagram of Pin Type 4-6-1	
	Addition of Figure 2 - 9 Pin Block Diagram of Pin Type 7-1-7	
	Addition of Figure 2 - 10 Pin Block Diagram of Pin Type 7-1-8	
	Addition of Figure 2 - 17 Pin Block Diagram of Pin Type 8-1-7	
	Addition of Figure 2 - 18 Pin Block Diagram of Pin Type 8-1-8	
	Addition of Figure 3 - 9 Memory Map (R5F104xK (x = G, J, L, M, P))	CHAPTER 3 CPU ARCHITECTURE
	Addition of Figure 3 - 10 Memory Map (R5F104xL (x = G, J, L, M, P))	
	Addition of Table 3 - 3 Correspondence Between Address Values and Block Numbers in Flash Memory (3/4)	
	Addition of Table 3 - 4 Correspondence Between Address Values and Block Numbers in Flash Memory (4/4)	
	Modification of Table 3 - 5 Internal ROM Capacity	
	Modification of Table 3 - 8 Internal RAM Capacity, cautions 3, and 4	
	Modification of Figure 3 - 12 Correspondence Between Data Memory and Addressing	
	Modification of 3.2.1 Control registers	
	Modification of cautions 4 and 5	
	Modification of 3.2.2 General-purpose registers	

(7/26)

Edition	Description	Chapter
Rev. 3.00	Modification of note 2 in Table 4 - 7 PMxx, Pxx, PUxx, PIMxx, POMxx, PMCxx registers and the bits mounted on each product (30-pin products to 64-pin products) (1/3)	CHAPTER 4 PORT FUNCTIONS
	Modification of Figure 4 - 8 Format of Peripheral I/O redirection register 0 (PIOR0)	
	Modification of 4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers	
	Modification of Table 4 - 15 Setting Examples of Registers When Using P00 to P17 Pin Function (2/3)	
	Modification of Table 4 - 15 Setting Examples of Registers When Using P00 to P17 Pin Function (3/3)	
	Modification of note in Table 4 - 16 Setting Examples of Registers When Using P20 to P27 Pin Function	
	Modification of Table 4 - 17 Setting Examples of Registers When Using P30 to P120 Pin Function (2/6)	
	Modification of note in Table 4 - 19 Setting Examples of Registers When Using P130 to P147 Pin Function	
	Modification of Figure 5 - 1 Block Diagram of Clock Generator	CHAPTER 5 CLOCK GENERATOR
	Modification of (1) X1 oscillation	
	Modification of (2) XT1 oscillation (crystal resonator)	
	Modification of Figure 6 - 12 Format of Timer mode register mn (TMRmn) (1/4)	CHAPTER 6 TIMER ARRAY UNIT
	Modification of caution 1 in Figure 6 - 20 Format of Timer input select register 0 (TIS0)	
	Deletion of caution in Figure 6 - 42 TOOn Pin Statuses by Collective Manipulation of TOOn Bit	
	Modification of Figure 6 - 69 Example of Set Contents of Registers to Delay Counter	
	Modification of Figure 7 - 11 Operation Example in Timer Mode	CHAPTER 7 TIMER RJ
	Modification of Figure 7 - 12 Operation Example in Pulse Output Mode	
	Modification of Figure 7 - 13 Operation Example 1 in Event Counter Mode	
	Modification of Figure 7 - 15 Operation Example in Pulse Width Measurement Mode	
	Modification of Figure 7 - 16 Operation Example in Pulse Period Measurement Mode	
	Modification of Table 8 - 7 TRDGRji Register Functions in Complementary PWM Mode	CHAPTER 8 TIMER RD
	Modification of Table 8 - 11 Timer RD Modes and Event Output to ELC/DTC	
	Modification of Figure 8 - 53 Block Diagram of PWM Function	
	Modification of Figure 8 - 56 Block Diagram of Reset Synchronous PWM Mode	
	Modification of Figure 8 - 58 Block Diagram of Complementary PWM Mode	
	Modification of Figure 8 - 61 Block Diagram of PWM3 Mode	
	Modification of Table 14 - 6 A/D Conversion Time Selection (4/4)	CHAPTER 14 A/D CONVERTER
	Modification of Figure 14 - 22 Example of Software Trigger Mode (Scan Mode, One-Shot Conversion Mode) Operation Timing	
	Modification of (2) Input range of ANI0 to ANI14 and ANI16 to ANI20 pins in 14.10 Cautions for A/D Converter	

(8/26)

Edition	Description	Chapter
Rev. 3.00	Modification of note 2 in 17.2.2 Lower 8/9 bits of the serial data register mn (SDRmn)	CHAPTER 17 SERIAL ARRAY UNIT
	Modification of caution 4 in Figure 17 - 11 Format of Serial data register mn (SDRmn)	
	Modification of Figure 17 - 113 Flowchart of UART Transmission (in Single-Transmission Mode)	
	Modification of Figure 17 - 123 Timing Chart of SNOOZE Mode Operation (EOCm1 = 0, SSECm = 0/1)	
	Modification of Figure 17 - 124 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 0)	
	Modification of Figure 17 - 126 Timing Chart of SNOOZE Mode Operation (EOCm1 = 1, SSECm = 1)	
	Modification of Figure 17 - 134 Flowchart for LIN Reception	
	Modification of Figure 18 - 35 Master Operation in Single-Master System	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of Figure 18 - 36 Master Operation in Multi-Master System (1/3)	
	Modification of Figure 18 - 39 Slave Operation Flowchart (1)	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of cautions 3 and 4 in 19.3.1 Allocation of DTC Control Data Area and DTC Vector Table Area	CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)
	Modification of Figure 19 - 13 DTC Internal Operation Flowchart	
	Modification of 19.5.2 Allocation of DTC Control Data Area and DTC Vector Table Area	
	Modification of Figure 23 - 8 When the Interrupt Request Signal is not Generated in the SNOOZE Mode	CHAPTER 23 STANDBY FUNCTION
	Modification of note 3 in Figure 25 - 3 Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector (2/3)	CHAPTER 25 POWER-ON-RESET CIRCUIT
	Modification of table in 26.1 Functions of Voltage Detector	CHAPTER 26 VOLTAGE DETECTOR
	Modification of Operation in LVD interrupt mode in 26.4.2 When used as interrupt mode	
	Modification of Figure 27 - 1 Format of Flash memory CRC control register (CRC0CTL)	CHAPTER 27 SAFETY FUNCTIONS
	Modification of note in Figure 27 - 11 Invalid access detection area	
	Modification of Figure 29 - 6 Format of Option Byte (000C2H/010C2H)	
	Modification of Table 30 - 10 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value)	CHAPTER 30 FLASH MEMORY
	Modification of note 1 in Figure 31 - 2 Memory Spaces Where Debug Monitor Programs Are Allocated	CHAPTER 31 ON-CHIP DEBUG FUNCTION
	Addition of (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products	CHAPTER 34 ELECTRICAL SPECIFICATIONS (A, D: TA = -40 to +85 °C)
	Modification of 34.7 Data Memory Retention Characteristics	
	Addition of (3) Flash ROM: 384 to 512 KB of 48- to 100-pin products	CHAPTER 35 ELECTRICAL SPECIFICATIONS (G: TA = -40 to +105 °C)
	Modification of 35.7 Data Memory Retention Characteristics	
	Addition and modification of 36.6 48-pin products	CHAPTER 36 PACKAGE DRAWINGS
	Modification of 36.7 52-pin products	
	Addition and modification of 36.8 64-pin products	
	Addition and modification of 36.9 80-pin products	
	Addition and modification of 36.10 100-pin products	
Rev. 2.00	Deletion of overline in SCK and SCKxx	All
	Modification of OSMC register name (from operation speed mode control register to subsystem clock supply mode control register)	
	Modification of Pin identification (from TRDCLK0 to TRDCLK)	

(9/26)

Edition	Description	Chapter
Rev. 2.00	Modification of 1.1 Features	CHAPTER 1 OUTLINE
	Modification of 1.2 Ordering Information	
	Modification of package type in 1.3 Pin Configuration (Top View)	
	Modification of description of subsystem clock in 1.6 Outline of Functions	
	Modification of description of timer output in 1.6 Outline of Functions	
	Modification of error of data transfer controller in 1.6 Outline of Functions	
	Modification of error of event link controller in 1.6 Outline of Functions	
	Addition of caution in 2.1 Port Function	CHAPTER 2 PIN FUNCTIONS
	Modification of tables and remarks in 2.1.1 to 2.1.18	
	Modification of Table in 2.3 Connection of Unused Pins	
	Modification of error of unused pins in P60 to P63	
	Modification of 2.4 Pin Block Diagrams	
	Modification of notes and cautions in Figures 3 - 1 to 3 - 8	CHAPTER 3 CPU ARCHITECTURE
	Modification of description and extension of specification in 3.1.3 Internal data memory space	
	Modification of Figure in 3.1.6 Data memory addressing	
	Modification of error in (3) Stack pointer (SP)	
	Modification of description in 3.2.2 General-purpose registers	
	Modification of description in 3.2.3 ES and CS registers	
	Modification of description in 3.2.4 Special function registers (SFRs)	
	Modification of description in 3.2.5 Extended special function registers (2nd SFRs: 2nd Special Function Registers)	
	Modification of error in Manipulable Bit Range in Event output destination select registers 00 to 25	
	Modification of description in 3.4.3 Direct addressing	
	Modification of description in 3.4.6 Register indirect addressing	
	Modification of description in 3.4.7 Based addressing	
	Modification of description in 3.4.8 Based indexed addressing	
	Modification of description in 3.4.9 Stack addressing	
	Modification of descriptions in 4.2.1 Port 0 to 4.2.15 Port 15	CHAPTER 4 PORT FUNCTIONS
	Addition of caution in 4.3 Registers Controlling Port Function	
	Addition of description and caution in 4.3.5 Port output mode registers (POMxx)	
	Addition of caution in 4.3.7 A/D port configuration register (ADPC)	
	Addition of description and caution in 4.3.10 Global digital input disable register (GDIDIS)	
	Addition of 4.4.4 Handling different potential (1.8 V, 2.5 V, 3 V) by using EVDD ≤ VDD	
	Modification of description in 4.4.5 Handling different potential (1.8 V, 2.5 V, 3 V) by using I/O buffers	
	Modification of description in 4.5 Register Settings When Using Alternate Function	CHAPTER 4 PORT FUNCTIONS

(10/26)

Edition	Description	Chapter
Rev. 2.00	Extension of selected clock in 5.1 Functions of Clock Generator	CHAPTER 5 CLOCK GENERATOR
	Modification of Figure 5 - 1 Block Diagram of Clock Generator	
	Modification of descriptions in 5.3.1 Clock operation mode control register (CMC)	
	Modification of descriptions in 5.3.6 Peripheral enable registers 0, 1 (PER0, PER1)	
	Modification of descriptions in 5.3.7 Subsystem clock supply mode control register (OSMC)	
	Modification of descriptions in 5.3.8 High-speed on-chip oscillator frequency select register (HOCODIV)	
	Modification of descriptions in Figure 5 - 19	
	Extension of selected clock in 5.6.1 Example of setting high-speed on-chip oscillator	
	Modification of descriptions in 5.6.2 Example of setting X1 oscillation clock	
	Modification of descriptions in 5.6.4 CPU clock status transition diagram	
	Modification of descriptions in 5.6.5 Condition before changing CPU clock and processing after changing CPU clock	
	Addition of 5.7 Resonator and Oscillator Constants	
	Modification of error of figure in (7) Delay counter in 6.1.1 Independent channel operation function	CHAPTER 6 TIMER ARRAY UNIT
	Modification of description in Figures 6 - 1 to 6 - 5	
	Modification of description in 6.2.2 Timer data register mn (TDRmn)	
	Modification of description in 6.3.1 Peripheral enable register 0 (PER0)	
	Modification of description in 6.3.2 Timer clock select register m (TPSm)	
	Modification of description in Figure 6 - 15 Format of Timer mode register mn (TMRmn) (4/4)	
	Modification of description in 6.3.7 Timer channel stop register m (TTm)	
	Modification of description in 6.3.8 Timer input select register 0 (TIS0)	
	Modification of description in Figure 6 - 21 Format of Timer output enable register m (TOEm)	
	Addition of description in 6.3.14 Noise filter enable registers 1, 2 (NFEN1, NFEN2)	
	Modification of description in 6.3.15 Registers controlling port functions of pins to be used for timer I/O	
	Modification of description in 6.5.2 Start timing of counter	
	Addition of description in (2) Operation of event counter mode	
	Modification of error of (3) Operation of capture mode (input pulse interval measurement)	
	Modification of error of 6.6.3 Cautions on Channel Output Operation	
	Addition of 6.7 Timer Input (TImn) Control	
	Modification of error in Figures 6 - 48 to 6 - 50	
	Addition of description in Figure 6 - 51	
	Modification of error in Figures 6 - 53 and 6 - 54	
	Addition of description in Figure 6 - 55	
	Addition of description in Figure 6 - 58	
	Addition of description in Figure 6 - 59	
	Modification of error and addition of description in Figures 6 - 61 and 6 - 62	
	Addition of description in Figure 6 - 63	

(11/26)

Edition	Description	Chapter
Rev. 2.00	Modification of error and addition of description in Figures 6 - 64 and 6 - 65	CHAPTER 6 TIMER ARRAY UNIT
	Modification of error in Figure 6 - 67	
	Modification of error and addition of description in Figures 6 - 69 and 6 - 70	
	Modification of error in Figure 6 - 71	
	Addition of description in Figures 6 - 75 and 6 - 76	
	Modification of error of count value in Figure 7 - 16 Operation Example in Pulse Period Measurement Mode	CHAPTER 7 TIMER RJ
	Addition of description in 8.3.10 Timer RD digital filter function select register i (TRDDFi) (i = 0 or 1)	CHAPTER 8 TIMER RD
	Modification of description in 8.3.14 Timer RD status register 0 (TRDSR0)	
	Modification of description in 8.3.15 Timer RD status register 1 (TRDSR1)	
	Modification of error in Figure 8 - 45 Pulse Output Forced Cutoff	
	Modification of error in Table 8 - 11 Timer RD Modes and Event Output to ELC/DTC	
	Modification of Figure 8 - 46 Block Diagram of Input Capture Function	
	Modification of error in Figure 8 - 48 Block Diagram of Digital Filter	
	Modification of Figure 8 - 49 Block Diagram of Output Compare Function	
	Modification of Figure 8 - 53 Block Diagram of PWM Function	
	Modification of Figure 8 - 56 Block Diagram of Reset Synchronous PWM Mode	
	Modification of Figure 8 - 58 Block Diagram of Complementary PWM Mode	
	Modification of Figure 8 - 61 Block Diagram of PWM3 Mode	
	Modification of description in 8.6 Timer RD Interrupt	
	Addition of description in (3) TRDi Register (i = 0 or 1) in 8.7.1 SFR Read/Write Access	
	Addition of description in 8.7.5 Procedure for Setting Pins TRDIOAi, TRDIOBi, TRDIOCi, and TRDIODi (i = 0 or 1)	
	Addition of description in 8.7.8 Complementary PWM Mode	
	Modification of error in 9.3.2 Timer RG mode resister (TRGMR)	CHAPTER 9 TIMER RG
	Modification of description in 9.3.6 Timer RG status resister (TRGSR)	
	Modification of error in Figure 9 - 15 Block Diagram of Digital Filter	
	Addition of 9.5 Timer RG Interrupt	
	Modification of error in 9.6.4 Procedure for Setting Pins TRGIOA and TRGIOB	
	Addition of description in 10.1 Functions of Real-time Clock	CHAPTER 10 REAL-TIME CLOCK
	Modification of error in 10.2 Configuration of Real-time Clock	
	Addition of description in 10.3.1 Peripheral enable register 0 (PER0)	
	Addition of caution in 10.3.3 Real-time clock control register 0 (RTCC0)	
	Addition of 10.4.6 Example of watch error correction of real-time clock	
	Modification of error in 11.2 Configuration of 12-bit Interval Timer	CHAPTER 11 12-BIT INTERVAL TIMER
	Modification of description in 11.3.1 Peripheral enable register 0 (PER0)	
	Modification of error in 11.3.3 12-bit interval timer control register (ITMC)	
	Addition of 11.4.2 Start of count operation and re-enter to HALT/STOP mode after returned from HALT/STOP mode	
	Modification of description in 12.1 Functions of Clock Output/Buzzer Output Controller	CHAPTER 12 CLOCK OUTPUT/BUZZER OUTPUT CONTROLLER
	Modification of note in 12.3.1 Clock output select registers n (CKSn)	

(12/26)

Edition	Description	Chapter
Rev. 2.00	Modification of description in 12.3.2 Registers controlling port functions of pins to be used for clock or buzzer output	CHAPTER 12 CLOCK OUTPUT/BUZZER OUTPUT CONTROLLER
	Addition of description in 12.4.1 Operation as output pin	
	Addition of 12.5 Cautions of clock output/buzzer output controller	
	Addition of description in 13.1 Functions of Watchdog Timer	CHAPTER 13 WATCHDOG TIMER
	Addition of description in 13.2 Configuration of Watchdog Timer	
	Modification of description in 13.3 Register Controlling Watchdog Timer	
	Modification of cautions in 13.4 Operation of Watchdog Timer	
	Modification of description in 14.1 Function of A/D Converter	CHAPTER 14 A/D CONVERTER
	Modification of error in Figure 14 - 1 Block Diagram of A/D Converter	
	Modification of caution in 14.3.1 Peripheral enable register 0 (PER0)	
	Modification of description in 14.3.2 A/D converter mode register 0 (ADM0)	
	Modification of caution in 14.3.3 A/D converter mode register 1 (ADM1)	
	Addition of description in 14.3.4 A/D converter mode register 2 (ADM2)	
	Modification of description in 14.3.5 10-bit A/D conversion result register (ADCR)	
	Modification of description in 14.3.6 8-bit A/D conversion result register (ADCRH)	
	Modification of caution in 14.3.7 Analog input channel specification register (ADS)	
	Addition of caution in 14.3.8 Conversion result comparison upper limit setting register (ADUL)	
	Addition of caution in 14.3.9 Conversion result comparison lower limit setting register (ADLL)	
	Modification of description in 14.3.10 A/D test register (ADTES)	
	Modification of description in 14.3.11 Registers controlling port function of analog input pins	
	Addition of note in 14.4 A/D Converter Conversion Operations	
	Modification of Figures 14 - 19 to 14 - 30	
	Modification of error in 14.7 A/D Converter Setup Flowchart	
	Addition of description in 14.8 SNOOZE Mode Function	
	Modification of error in Figures 14 - 42 and 14 - 43	
	Modification of description in (2) Input range of ANI0 to ANI14 and ANI16 to ANI20 pins in 14.10 Cautions for A/D Converter	
	Modification of description in (4) Noise countermeasures in 14.10 Cautions for A/D Converter	
	Modification of error in Figure 14 - 46	
	Modification of description in (5) Analog input (ANIn) pins in 14.10 Cautions for A/D Converter	
	Modification of description in (6) Input impedance of analog input (ANIn) pins in 14.10 Cautions for A/D Converter	
	Addition of 15.3.4 Event output destination select register n (ELSELRn), n = 00 to 25	CHAPTER 15 D/A CONVERTER
	Modification of description in 15.3.5 Registers controlling port functions of analog input pins	
	Modification of caution in 15.4 Operations of D/A Converter	
	Modification of description in 15.4.2 Operation in Real-Time Output Mode	
	Addition of 15.4.3 Timing for Outputting D/A Conversion Value	

(13/26)

Edition	Description	Chapter
Rev. 2.00	Modification of Figure 16 - 1 Comparator Block Diagram	CHAPTER 16 COMPARATOR
	Modification of description in 16.3.2 Comparator mode setting register (COMPMDR)	
	Modification of description in 16.3.5 Registers controlling port functions of analog input pins	
	Modification of description in Table 16 - 2 Procedure for Setting Comparator Associated Registers	
	Modification of description in 16.4.3 Event signal output to event link controller (ELC)	
	Modification of description in 16.4.4 Comparator i Output (i = 0 or 1)	
	Modification of description in 17.1 Functions of Serial Array Unit	CHAPTER 17 SERIAL ARRAY UNIT
	Modification of Figure 17 - 1 Block Diagram of Serial Array Unit 0	
	Modification of Figure 17 - 2 Block Diagram of Serial Array Unit 1	
	Modification of description in 17.2.2 Lower 8/9 bits of the serial data register mn (SDRmn)	
	Modification of caution in 17.3.1 Peripheral enable register 0 (PER0)	
	Modification of note in 17.3.2 Serial clock select register m (SPSm)	
	Addition of note in 17.3.3 Serial mode register mn (SMRmn)	
	Modification of description in 17.3.4 Serial communication operation setting register mn (SCRmn)	
	Modification of description in 17.3.5 Serial data register mn (SDRmn)	
	Addition of note in 17.3.6 Serial flag clear trigger register mn (SIRmn)	
	Addition of note in 17.3.7 Serial status register mn (SSRmn)	
	Modification of note in 17.3.8 Serial channel start register m (SSm)	
	Modification of note in 17.3.9 Serial channel stop register m (STm)	
	Modification of note in 17.3.11 Serial output enable register m (SOEm)	
	Modification of note in 17.3.12 Serial output register m (SOm)	
	Addition of description in 17.3.13 Serial output level register m (SOLm)	
	Modification of description in 17.3.14 Serial standby control register m (SSCm)	
	Modification of description in 17.3.16 Noise filter enable register 0 (NFENO)	
	Modification of description in 17.3.17 Registers controlling port functions of serial input/output pins	
	Modification of description in 17.5 Operation of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21, CSI30, CSI31) Communication	
	Modification of description in 17.5.1 Master transmission	
	Modification of description in 17.5.2 Master reception	
	Modification of description in 17.5.3 Master transmission/reception	
	Modification of description in 17.5.4 Slave transmission	
	Modification of description in 17.5.5 Slave reception	
	Modification of description in 17.5.6 Slave transmission/reception	
	Modification of description in 17.5.7 SNOOZE mode function	
	Modification of note in 17.6 Clock Synchronous Serial Communication with Slave Select Input Function	
	Modification of Figure 17 - 80	
	Modification of description in 17.6.1 Slave transmission	
	Modification of description in 17.6.2 Slave reception	

(14/26)

Edition	Description	Chapter
Rev. 2.00	Modification of description in 17.6.3 Slave transmission/reception	CHAPTER 17 SERIAL ARRAY UNIT
	Modification of description in 17.6.4 Calculating transfer clock frequency	
	Modification of description in 17.7 Operation of UART (UART0 to UART3) Communication	
	Modification of description in 17.7.1 UART transmission	
	Modification of description in 17.7.2 UART reception	
	Modification of description in 17.7.3 SNOOZE mode function	
	Modification of description in 17.8 LIN Communication Operation	
	Modification of description in 17.9 Operation of Simplified I <sup>2</sup> C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) Communication	
	Modification of description in 17.9.1 Address field transmission	
	Modification of description in 17.9.2 Data transmission	
	Modification of description in 17.9.3 Data reception	
	Modification of description in 17.9.4 Stop condition generation	
	Modification of caution in 17.9.5 Calculating transfer rate	
	Modification of description in 17.9.6 Procedure for processing errors that occurred during simplified I <sup>2</sup> C (IIC00, IIC01, IIC10, IIC11, IIC20, IIC21, IIC30, IIC31) communication	
	Modification of descriptions (from 64-pin to 100-pin)	CHAPTER 18 SERIAL INTERFACE IICA
	Modification of error in Figure 18 - 1 Block Diagram of Serial Interface IICA	
	Modification of caution in 18.3.1 Peripheral enable register 0 (PER0)	
	Modification of description in 18.3.5 IICA control register n1 (IICCTLn1)	
	Modification of description in 18.3.6 IICA low-level width setting register n (IICWLn)	
	Modification of description in 18.3.7 IICA high-level width setting register n (IICWHn)	
	Modification of error in 18.4.2 Setting transfer clock by using IICWLn and IICWHn registers	
	Addition of description in Figure 18 - 29 Flow When Setting WUPn = 1	
	Addition of description in Figure 18 - 30 Flow When Setting WUPn = 0 upon Address Match (Including Extension Code Reception)	
	Modification of description in Figure 18 - 31 When Operating as Master Device after Releasing STOP Mode other than by INTIICAn	
	Modification of description in Figure 18 - 34 Communication Reservation Protocol	
	Modification of error in (3) If other I <sup>2</sup> C communications are already in progress in 18.5.15 Cautions	
	Modification of error in Figure 18 - 35 Master Operation in Single-Master System	
	Modification of error in Figure 18 - 37 Master Operation in Multi-Master System (2/3)	
	Modification of error in Figure 18 - 38 Master Operation in Multi-Master System (3/3)	
	Modification of error in Figure 18 - 39 Slave Operation Flowchart (1)	
	Modification of Figure 19 - 1 DTC Block Diagram	CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)
	Modification of caution in 19.3.11 DTC activation enable register i (DTCENi) (i = 0 to 4)	
	Modification of error in Table 19 - 6 Register Functions in Normal Mode	
	Modification of error in (1) Example 1 of using normal mode: Consecutively capturing A/D conversion results	

(15/26)

Edition	Description	Chapter
Rev. 2.00	Modification of error in (2) Example 2 of using normal mode: UART0 consecutive transmission	CHAPTER 19 DATA TRANSFER CONTROLLER (DTC)
	Modification of error in (1) Example 1 of using repeat mode: Outputting a stepping motor control pulse using ports	
	Modification of error in (2) Example 2 of using repeat mode: Outputting a sine wave using the 8-bit D/A converter	
	Addition of description in 19.4.4 Chain Transfers	
	Modification of error in (1) Example of using chain transfers: Consecutively capturing A/D conversion results and UART0 transmission	
	Addition of description in 19.5.1 Setting DTC Control Data and Vector Table	
	Modification of description in 19.5.3 DTC Pending Instruction	
	Modification of description in 19.5.4 Operation when Accessing Data Flash Memory Space	
	Addition of description in 20.3.1 Event output destination select register n (ELSELRn) (n = 00 to 25)	CHAPTER 20 EVENT LINK CONTROLLER (ELC)
	Modification of description in 20.4 ELC Operation	
	Addition of description in 21.2 Interrupt Sources and Configuration	CHAPTER 21 INTERRUPT FUNCTIONS
	Modification of description in 21.3 Registers Controlling Interrupt Functions	
	Modification of description in 21.4.1 Maskable interrupt request acknowledgment	
	Addition of caution in 23.3 Standby Function Operation	CHAPTER 23 STANDBY FUNCTION
	Modification of description in reset function	CHAPTER 24 RESET FUNCTION
	Modification of description in 24.1 Timing of Reset Operation	
	Modification of caution in 24.2 Register for Confirming Reset Source	
	Modification of description in 25.1 Functions of Power-on-reset Circuit	CHAPTER 25 POWER-ON-RESET CIRCUIT
	Modification of description in 25.3 Operation of Power-on-reset Circuit	
	Modification of description in 26.1 Functions of Voltage Detector	CHAPTER 26 VOLTAGE DETECTOR
	Modification of description in 26.2 Configuration of Voltage Detector	
	Modification of description in 26.3 Registers Controlling Voltage Detector	
	Modification of description in 26.4 Operation of Voltage Detector	
	Modification of description in 26.5 Cautions for Voltage Detector	
	Addition of description in 27.3.1 Flash memory CRC operation function (high-speed CRC)	CHAPTER 27 SAFETY FUNCTIONS
	Addition of description in 27.3.2 CRC operation function (general-purpose CRC)	
	Modification of error in Figure 27 - 6 CRC Operation Function (General-Purpose CRC)	
	Modification of caution in 27.3.3.1 RAM parity error control register (RPECTL)	
	Modification of error in 27.3.6 Invalid memory access detection function	
	Modification of description in 27.3.7 Frequency detection function	
	Modification of description in 27.3.8 A/D test function	
	Modification of description in 27.3.9 Digital output signal level detection function for I/O pins	
	Addition of description in 29.1 Functions of Option Bytes	CHAPTER 29 OPTION BYTE
	Modification of error and addition of description in 29.2 Format of User Option Byte	

(16/26)

Edition	Description	Chapter
Rev. 2.00	Addition of description in flash memory	CHAPTER 30 FLASH MEMORY
	Addition of description in 30.1 Serial Programming Using Flash Memory Programmer	
	Modification of description in 30.2 Serial Programming Using External Device (that Incorporates UART)	
	Addition of description in 30.3 Connection of Pins on Board	
	Addition of description in 30.4.2 Flash memory programming mode	
	Modification of description in 30.4.4 Communication commands	
	Addition of 30.5 Processing Time for Each Command When PG-FP5 Is in Use (Reference Value)	
	Modification of description in 30.6 Self-Programming	
	Modification of description in 30.7 Security Settings	
	Addition of 30.8.1 Data flash overview	
	Addition of 30.8.3 Procedure for accessing data flash memory	
	Modification of error in 31.1 Connecting E1 On-chip Debugging Emulator	CHAPTER 31 ON-CHIP DEBUG FUNCTION
	Addition of description in Table 33 - 2 Symbols in "Operation" Column	
	Modification of Note 2 in 33.2 Operation List	
	Modification of error of flags when CMP0 instruction is executed in 33.2 Operation List	CHAPTER 33 INSTRUCTION SET
	Modification of description of Tables in 34.1 Absolute Maximum Ratings	
	Modification of Tables, notes, cautions, and remarks in 34.2 Oscillator Characteristics	
	Modification of error of conditions of low level input voltage in 34.3.1 Pin characteristics	
	Modification of error of conditions of low level output voltage in 34.3.1 Pin characteristics	
	Modification of Notes and Remarks in 34.3.2 Supply current characteristics	
	Addition of Minimum Instruction Execution Time during Main System Clock Operation	
	Addition of AC Timing Test Points	
	Addition of LS mode and LV mode characteristics in 34.5.1 Serial array unit	
	Addition of LS mode and LV mode characteristics in 34.5.2 Serial interface IICA	
	Addition of characteristics about conversion of internal reference voltage and temperature sensor in 34.6.1 A/D converter characteristics	
	Addition of characteristic in 34.6.4 Comparator	
	Deletion of detection delay in 34.6.5 POR circuit characteristics	
	Modification of 34.6.7 Power supply voltage rising slope characteristics	
New	Modification of 34.7 Data Memory STOP Mode Low Supply Voltage Data Retention Characteristics	CHAPTER 34 ELECTRICAL SPECIFICATIONS (A, D: TA = -40 to +85 °C)
	Addition of characteristic in 34.8 Flash Memory Programming Characteristics	
	Addition of description in 34.10 Timing for Switching Flash Memory Programming Modes	
	New	
	Modification of 36.1 30-pin products to 36.10 100-pin products	

(17/26)

Edition	Description	Chapter
Rev. 1.00	Change of 1.2 Ordering Information	CHAPTER 1 PIN FUNCTIONS
	Addition of Figure 1-1. Part Number, Memory Size, and Package of RL78/G14	
	Change of 1.3 Pin Configuration (Top View)	
	Addition of caution to 1.6 Outline of Functions	
	Change of 2.1.1 30-pin products to 2.1.10 100-pin products	CHAPTER 2 PIN FUNCTIONS
	Addition of remark to 2.2.16 VDD, EVDD0, EVDD1, Vss, EVSS0, EVSS1	
	Change of Table 2-3. Connection of Unused Pins (100-pin products)	
	Figure 3-1 to Figure 3-8 • Change of and addition of note • Addition of caution	CHAPTER 3 CPU ARCHITECTURE
	Change of caution 2 in and addition of caution 3 to 3.1.3 Internal data memory space	
	Figure 3-10 to Figure 3-17 • Change of and addition of note • Addition of caution	
	Change of caution 3 in and addition of caution 4 to 3.2.1 (3) Stack pointer (SP)	
	Change of caution 2 in and addition of caution 3 to 3.2.2 General-purpose registers	
	Change of Table 3-6. Extended SFR (2nd SFR) List	
	Change of Figure 3-27 Outline of Table Indirect Addressing	
	Change of the block diagram of port. Addition of table of settings of register.	CHAPTER 4 PORT FUNCTIONS
	Change of description in 4.2.1 Port 0	
	Change of Table 4-5 to Table 4-8	
	Change of Figure 4-62. Format of Port Input Mode Register (100-pin products)	
	Change of Figure 4-64. Format of Port Mode Control Register (100-pin products)	
	Change of description in 4.3 (10) Global digital input disable register (GDIDIS)	
	Change of 4.4.4 Connecting to external device with different potential (1.8 V, 2.5 V, 3 V)	
	Change of 4.5 Settings of Port Mode Register, and Output Latch When Using Alternate Function	
	Addition of 4.6.2 Cautions on the pin settings on the products other than 100-pin	
	Change of note in 5.1 Functions of Clock Generator	CHAPTER 5 CLOCK GENERATOR
	Change of block diagram and note in Figure 5-1. Block Diagram of Clock Generator	
	Addition of note to Figure 5-2. Format of Clock Operation Mode Control Register (CMC)	
	Change of note in Figure 5-3. Format of System Clock Control Register (CKC)	
	Change of Figure 5-6. Format of Oscillation Stabilization Time Select Register (OSTS)	
	Change of note in Figure 5-8. Format of Peripheral Enable Register 1 (PER1)	
	Change of Figure 5-9. Format of Operation Speed Mode Control Register (OSMC)	
	Change of 5.3 (8) High-speed on-chip oscillator frequency select register (HOCODIV)	
	Change of 5.3 (9) High-speed on-chip oscillator trimming register (HIOTRM)	
	Addition of note 3 to Figure 5-15. Clock Generator Operation When Power Supply Voltage is Turned On	
	Change of 5.6.2 Example of setting X1 oscillation clock	

(18/26)

Edition	Description	Chapter
Rev. 1.00	Change of Table 5-3. (6) CPU clock changing from high-speed system clock (C) to high speed on-chip oscillator clock (B)	CHAPTER 5 CLOCK GENERATOR
	Change of Table 5-3. (9) CPU clock changing from subsystem clock (D) to high-speed system clock (C)	
	Addition of 5.7 Operation-Verified Resonators and Reference Oscillator Constants As of December 2011	
	Addition of description to 6.2 (1) Timer count register mn (TCRmn)	CHAPTER 6 TIMER ARRAY UNIT
	Addition of note to Table 6-4. Interval Times Available for Operation Clock CKSm2 or CKSm3	
	Addition of caution to Figure 6-13. Format of Timer Input Select register 0 (TIS0)	
	Change of Figure 6-14. Format of Timer Output Enable register m (TOEm)	
	Addition of remark to 6.3 (15) Port mode registers 0, 1, 3, 6 (PM0, PM1, PM3, PM6)	
	Change of 6.4.1 Basic rules of simultaneous channel operation function	
	Change of Figure 6-23. Start Timing (In Interval Timer Mode)	
	Change of Figure 6-25. Start Timing (In Capture Mode)	
	Change of Figure 6-26. Start Timing (In One-count Mode)	
	Change of Figure 6-27. Start Timing (In Capture & One-count Mode)	
	Change of note in Figure 6-37. Block Diagram of Operation as Interval Timer/Square Wave Output	
	Addition of note to Figure 6-49. Block Diagram of Operation as Input Pulse Interval Measurement	
	Addition of note to Figure 6-53. Block Diagram of Operation as Input Signal High-/Low-Level Width Measurement	
	Change of description in 6.7.6 Operation as delay counter	
	Change of note in Figure 6-57. Block Diagram of Operation as Delay Counter	
	Addition of 6.9 Cautions When Using Timer Array Unit	
	Addition of note to Figure 7-1. Timer RJ Block Diagram	CHAPTER 7 TIMER RJ
	Addition of note to Table 7-3. Timer RJ Register Configuration	
	Addition of 7. 3. 2 Operation speed mode control register (OSMC)	
	Change of note in and addition of caution to 7. 3. 3 Timer RJ Counter Register 0 (TRJ0), Timer RJ Reload Register	
	Change of Figure 7-6. Format of Timer RJ I/O Control Register 0 (TRJIOCO)	
	Addition of note to Figure 7-7. Format of Timer RJ Mode Register 0 (TRJMR0)	
	Addition of note to Figure 7-8. Format of Timer RJ Event Pin Select Register 0 (TRJISR0)	
	Change of Figure 7-12. Operation Example in Pulse Output Mode	
	Addition of Figure 7-14. Operation Example 2 in Event Counter Mode	
	Addition of description to 7. 4. 7 Coordination with Event Link Controller (ELC)	
	Change of Table 7-7. TRJIO0 Pin Setting	
	Addition of description to 7. 5. 1 Count Operation Start and Stop Control	
	Change of 7. 5. 4 When Changing Mode	
	Addition of 7. 5. 11 Digital Filter	
	Addition of 7. 5. 12 When Selecting fil as Count Source	

(19/26)

Edition	Description	Chapter
Rev. 1.00	Change of note in Table 8-2. Timer RD Register Configuration	CHAPTER 8 TIMER RD
	Addition of caution to Figure 8-2. Format of Peripheral Enable Register 1 (PER1)	
	Change of note in Figure 8-3. to Figure 8-37.	
	Change of Figure 8-5. Format of Timer RD Mode Register (TRDMDR)	
	Change of Figure 8-6. Format of Timer RD PWM Function Select Register (TRDPMR) [Timer Mode]	
	Change of Figure 8-9. Format of Timer RD Output Master Enable Register 2 (TRDOER2) [PWM Function, Reset Synchronous PWM Mode, Complementary PWM Mode, and PWM3 Mode]	
	Change of note in Table 8-9. Count Source Selection	
	Change of note in Table 8-12 to Table 8-17	
	Addition of 8. 4 Timer RD Interrupt	
	Change of 8. 5. 1 SFR Read/Write Access	
	Change of 8. 5. 3 Count Source	
	Change of note in Figure 9-8. Format of Timer RG I/O Control Register (TRGIOR)	CHAPTER 9 TIMER RG
	Addition of note to 9.4.3 Count Source Switching	
	Addition of 9.4.7 Input Capture Operation when Count is Stopped	
	Change of caution in 10.1 Functions of Real-time Clock	CHAPTER 10 REAL-TIME CLOCK
	Change of block diagram in and addition of caution to Figure 10-1. Block Diagram of Real-time Clock	
	Change of Figure 10-3. Format of Operation Speed Mode Control Register (OSMC)	
	Addition of caution 2 to Figure 10-4. Format of Real-time Clock Control Register 0 (RTCC0)	
	Change of Figure 10-5. Format of Real-time Clock Control Register 1 (RTCC1)	
	Change of Figure 10-17. Procedure for Starting Operation of Real-time Clock	
	Addition of description about HALT mode to 10.4.2 Shifting to HALT/STOP mode after starting operation	
	Addition of description to 10.4.3 Reading/writing real-time clock	
	Addition of caution 2 to Figure 10-20. Procedure for Writing Real-time Clock	
	Change of Figure 10-22. 1 Hz Output Setting Procedure	
	Change of Figure 11-3. Format of Operation Speed Mode Control Register (OSMC)	CHAPTER 11 12-BIT INTERVAL TIMER
	Change of caution 2 in and addition of caution 3 to Figure 11-4. Format of 12-bit Interval Timer Control Register (ITMC)	
	Change of note in and addition of remark to Figure 12-1. Block Diagram of Clock Output/Buzzer Output Controller	CHAPTER 12 CLOCK OUTPUT/BUZZER OUTPUT CONTROLLER
	Change of Figure 12-2. Format of Clock Output Select Register n (CKSn)	
	Change of 13.3 (1) Watchdog timer enable register (WDTE)	CHAPTER 13 WATCHDOG TIMER
	Change of caution in Figure 13-2. Format of Watchdog Timer Enable Register (WDTE)	
	Change of caution in 13.4.1 Controlling operation of watchdog timer	
	Change of Figure 14-1. Block Diagram of A/D Converter	CHAPTER 14 A/D CONVERTER
	Addition of note to Table 14-1. Settings of ADCS and ADCE Bits	
	Change of Table 14-2. Setting and Clearing Conditions for ADCS Bit	

(20/26)

Edition	Description	Chapter
Rev. 1.00	Change of Figure 14-4. Timing Chart When A/D Voltage Comparator Is Used and addition of note	CHAPTER 14 A/D CONVERTER
	Change of Table 14-3. A/D Conversion Time Selection	
	Change of Figure 14-7. Format of A/D Converter Mode Register 2 (ADM2) and addition of note and caution	
	Change of note in and addition of caution to 14.3 (7) Analog input channel specification register (ADS)	
	Change of Figure 14-14. Format of A/D Test Register (ADTES)	
	Change of Figure 14-16. Formats of Port Mode Control Registers 0, 10, 12, and 14 (PMC0, PMC10, PMC12, PMC14)	
	Change of 14.7.4 Setup when using temperature sensor (example for software trigger mode and one-shot conversion mode)	
	Addition of description to 14.10 (2) Input range of ANI0 to ANI14 and ANI16 to ANI20 pins	
	Change of Table 14-6. Resistance and Capacitance Values of Equivalent Circuit (Reference Values)	
	Addition of 14.10 (12) Temperature sensor	
	Change of Figure 15-1 Block Diagram of D/A Converter	CHAPTER 15 D/A CONVERTER
	Change of Figure 16-3 Format of Comparator Mode Setting Register (COMPMDR)	CHAPTER 16 COMPARATOR
	Change of Figure 16-5 Format of Comparator Output Control Register (COMPOCR)	
	Addition of note to Table 16-5 Procedure for Setting Comparator Associated Registers	
	Addition of Figure 16-13 Digital Filter and Interrupt Request/ELC Event Output Operation	
	Addition of caution to 16.4.5 Stopping or Supplying Comparator Clock	
	Change the entire chapter	CHAPTER 17 SERIAL ARRAY UNIT
	Change of Figure 18-7. Format of IICA Status Register 0 (IICSO)	CHAPTER 18 SERIAL INTERFACE IICA
	Change of 18.4.2 Setting transfer clock by using IICWL0 and IICWH0 registers	
	Change of procedure <2> in Figure 18-33. (1) Start condition ~ address ~ data	
	Addition of caution to 19.2.1 Allocation of DTC Control Data Area and DTC Vector Table Area	CHAPTER 19 DTC
	Change of note in 19.2.2 DTC Control Data Allocation	
	Addition of caution to Figure 19-5. to Figure 19-10.	
	Addition and deletion of caution to 19.2.12 DTC Base Address Register (DTCBAR)	
	Addition of 19.3.2 (1) Example 1 of using normal mode	
	Addition of 19.3.2 (2) Example 2 of using normal mode	
	Addition of 19.3.3 (1) Example 1 of using repeat mode	
	Addition of 19.3.3 (2) Example 2 of using repeat mode	
	Addition of 19.3.4 (1) Example of using chain transfers	
	Addition of description to 19.4.2 Allocation of DTC Control Data Area and DTC Vector Table Area	
	Addition of description to 19.4.3 DTC Pending Instruction	
	Addition of 19.4.4 Operation when Accessing Data Flash Memory Space	
	Addition of description to 19.4.6 DTC Response Time	
	Addition of description to 19.4.7 DTC Activation Sources	

(21/26)

Edition	Description	Chapter
Rev. 1.00	Change of 20.2.1 Event Output Destination Select Register n (ELSELRn) (n = 00 to 25)	CHAPTER 20 EVENT LINK CONTROLLER (ELC)
	Addition of note to Table 20-3. Correspondence Between Values Set to ELSELRn (n = 00 to 25) Registers and Operation of Link Destination Peripheral Functions at Reception	
	Addition of note to Table 21-1. Interrupt Source List (3/4)	CHAPTER 21 INTERRUPT FUNCTIONS
	Change of Table 21-4. Time from Generation of Maskable Interrupt Until Servicing	
	Change of Figure 21-9. Interrupt Request Acknowledgment Timing (Maximum Time)	
	Addition of 21.4.4 Interrupt servicing during division instruction	
	Change of Figure 23-2. Format of Oscillation Stabilization Time Select Register (OSTS)	CHAPTER 23 STANDBY FUNCTION
	Addition of note to Figure 23-3. HALT Mode Release by Interrupt Request Generation	
	Addition of note to Figure 23-4. HALT Mode Release by Reset	
	Change of caution 2 in Table 23-2. Operating Statuses in STOP Mode	
	Addition of note to Figure 23-5. STOP Mode Release by Interrupt Request Generation	
	Addition of note to Figure 23-6. STOP Mode Release by Reset	
	Change of caution 1 in CHAPTER 24 RESET FUNCTION	CHAPTER 24 RESET FUNCTION
	Change of Figure 24-2. Timing of Reset by <u>RESET</u> Input	
	Change of Figure 24-3. Timing of Reset Due to Execution of Illegal Instruction or Watchdog Timer Overflow	
	Change of Figure 24-4. Timing of Reset in STOP Mode by <u>RESET</u> Input	
	Change of Table 24-1. Operation Statuses During Reset Period	
	Change of Table 24-2. Hardware Statuses After Reset Acknowledgment	
	Change of caution 3 in Figure 24-5. Format of Reset Control Flag Register (RESF)	
	Change of Figure 25-2. Timing of Generation of Internal Reset Signal by Power-on-reset Circuit and Voltage Detector	CHAPTER 25 POWER-ON-RESET CIRCUIT
	Change of 26.1 Functions of Voltage Detector	CHAPTER 26 VOLTAGE DETECTOR
	Addition of note to and change of note in Figure 26-2. Format of Voltage Detection Register (LVIM)	
	Change of Figure 26-3. Format of Voltage Detection Level Register (LVIS)	
	Change of Table 26-1. LVD Operation Mode and Detection Voltage Settings for User Option Byte (000C1H/010C1H)	
	Change of 26.4.1 When used as reset mode	
	Change of Figure 26-4. Timing of Voltage Detector Internal Reset Signal Generation (Option Byte LVIMDS1, LVIMDS0 = 1, 1)	
	Change of 26.4.2 When used as interrupt mode	
	Change of Figure 26-5. Timing of Voltage Detector Internal Interrupt Signal Generation (Option Byte LVIMDS1, LVIMDS0 = 0, 1)	
	Change of 26.4.3 When used as interrupt and reset mode	
	Change of Figure 26-6. Timing of Voltage Detector Reset Signal and Interrupt Signal Generation (Option Byte LVIMDS1, LVIMDS0 = 1, 0)	
	Addition of Figure 26-7. Processing Procedure After an Interrupt Is Generated	
	Addition of Figure 26-8. Initial Setting of Interrupt and Reset Mode	
	Change of Figure 26-10. Delay from the time LVD reset source is generated until the time LVD reset has been generated or released	CHAPTER 26 VOLTAGE DETECTOR
	Change the entire chapter	CHAPTER 27 SAFETY FUNCTIONS
	Change of Table 28-1. Regulator Output Voltage Conditions	CHAPTER 28 REGULATOR

(22/26)

Edition	Description	Chapter
Rev. 1.00	Change of 29.1.1 (2) 000C1H/010C1H	CHAPTER 29 OPTION BYTE
	Change of Figure 29-2. Format of User Option Byte (000C1H/010C1H)	
	Change of Figure 29-3. Format of User Option Byte (000C2H/010C2H)	
	Change of 30.1.2 Communication Mode	CHAPTER 30 FLASH MEMORY
	Change of 30.2.2 Communication Mode	
	Addition of description to 30.4.1 Data flash overview	
	Change of 30.5.2 Flash memory programming mode	
	Addition of 30.5.5 Description of signature data	
	Change of caution in 30.7 Flash Memory Programming by Self-Programming	
	Change of Figure 31-2. Memory Spaces Where Debug Monitor Programs Are Allocated	CHAPTER 31 ON-CHIP DEBUG FUNCTION
	Change of 33.2 Operation List	CHAPTER 33 INSTRUCTION SET
	Change of 34.2 Absolute Maximum Ratings	CHAPTER 34 ELECTRICAL SPECIFICATIONS (TARGET)
	Change of 34.3.2 On-chip oscillator characteristics	
	Change of 34.4.1 Pin characteristics	
	Change of 34.4.2 Supply current characteristics	
	Change of 34.6.1 Serial array unit	
	Change of 34.7.1 A/D converter characteristics	
	Change of 34.7.2 Temperature sensor characteristics	
	Change of 34.7.3 D/A converter characteristics	
	Change of 34.7.4 Comparator	
	Change of 34.7.5 POR circuit characteristics	
	Change of 34.8 Power Supply Rise Time	
	Change of 34.10 Flash Memory Programming Characteristics	
	Change of 34.11 Timing Specs for Switching Modes	
	Change of 35.10 100-pin products	CHAPTER 35 PACKAGE DRAWINGS
Rev. 0.02	Addition of 80-pin products and 100-pin products	Throughout
	Change of 1.1 Features	CHAPTER 1 PIN FUNCTIONS
	Change of 1.3.1 30-pin products to 1.3.8 64-pin products and addition of note	
	Change of 1.5.1 30-pin products to 1.5.8 64-pin products and addition of note	
	Change of 1.6 Outline of Functions	
	Change the entire chapter	CHAPTER 2 PIN FUNCTIONS

(23/26)

Edition	Description	Chapter
Rev. 0.02	Addition of caution to Figure 3-1 to Figure 3-3	CHAPTER 3 CPU ARCHITECTURE
	Change of note and addition of caution to Figure 3-4	
	Addition of Figure 3-5 to Figure 3-8	
	Change of description of 3.1.2 Mirror area	
	Change of Figure 3-9. Format of Configuration of Processor Mode Control Register (PMC)	
	Change of caution in 3.1.3 Internal data memory space	
	Addition of caution to Figure 3-10 to Figure 3-12	
	Change of note and addition of caution to Figure 3-13	
	Addition of Figure 3-14 to Figure 3-17	
	Change of caution in 3.2.1 (3) Stack pointer (SP)	
	Change of caution in 3.2.2 General-purpose registers	
	Change of Table 3-5. SFR List	
	Change of Table 3-6. Extended SFR (2nd SFR) List	
	Change the entire chapter	CHAPTER 4 PORT FUNCTIONS
	Change of description in 5. 1 (1) <2>Internal high-speed oscillator	CHAPTER 5 CLOCK GENERATOR
	Change of 5.1 (2) Subsystem clock	
	Change of remark in 5.1 Functions of Clock Generator	
	Addition of high-speed on-chip oscillator frequency select register (HOCODIV)	
	Change of Figure 5-1. Block Diagram of Clock Generator	
	Change of remark in Figure 5-3. Format of System Clock Control Register (CKC)	
	Change of 5.3 (6) Peripheral enable registers 0, 1 (PER0, PERO)	
	Change of description in 5.6.1 Example of setting internal high-speed oscillator	
	Change of Figure 5-16. CPU Clock Status Transition Diagram	
	Addition of 6.1.3 8-bit timer operation function (channels 1 and 3 only)	CHAPTER 6 TIMER ARRAY UNIT
	Addition of remark to Figure 6-7. Format of Timer Clock Select register m (TPSm) (1/2)	
	Addition of remark to Table 6-4. Interval Times Available for Operation Clock CKSm2 or CKSm3	
	Change of Figure 6-8. Format of Timer Mode Register mn (TMRmn)	
	Change of 6. 3 (6) Timer channel start register m (TSm)	
	Addition of 6.5 Operation Timing of Counter	
	Change of Figure 6-34. Set/Reset Timing Operating Statuses	
	Change of Figure 6-41. Operation Procedure of Interval Timer/Square Wave Output Function (2/2)	
	Change of Figure 6-45. Operation Procedure When External Event Counter Function Is Used	
	Change of Figure 6-49. Operation Procedure When Frequency Divider Function Is Used	
	Change of Figure 6-61. Operation Procedure When Delay Counter Function Is Used	
	Change of Figure 6-66. Operation Procedure of One-Shot Pulse Output Function (2/2)	
	Change of Figure 6-71. Operation Procedure When PWM Function Is Used (2/2)	
	Change of Figure 6-76. Operation Procedure When Multiple PWM Output Function Is Used (2/2)	

(24/26)

Edition	Description	Chapter
Rev. 0.02	Change of Figure 7-1. Timer RJ Block Diagram	CHAPTER 7 TIMER RJ
	Addition of caution to Figure 7-2. Format of Peripheral Enable Register 1 (PER1)	
	Change of 7. 5. 7 When Timer RJ Operating Clock is Stopped	
	Change from INT0 of pulse output forced cutoff signal input to INTP0	
	Change of description of operation clock	
	Addition of note to Table 8-2. Timer RD Register Configuration	
	Addition of caution to Figure 8-2. Format of Peripheral Enable Register 1 (PER1)	
	Addition of note to Figure 8-3. to Figure 8-37.	
	Change of 8.3.1 (4) Pulse Output Forced Cutoff	
	Change of 8. 4. 3 Count Source	
	Change of 8. 4. 4 Input Capture Function	CHAPTER 8 TIMER RD
	Addition of note to Table 9-2. Timer RD Register Configuration	
	Change of 9.2.2 Timer RG Mode Register (TRGMR)	
	Change of 9.2.4 Timer RG Control Register (TRGCR)	
	Change of 9.2.5 Timer RG Interrupt Enable Register (TRGIER)	
	Change of 9.2.6 Timer RG Status Register (TRGSR)	
	Change of 9.2.7 Timer RG I/O Control Register (TRGIOR)	
	Change of 9.3.1 (5) Event Output to Event Link Controller (ELC)	
	Change of 9.3.4 (2) Operation Example	
	Change of 9.3.5 Phase Counting Mode	
	Change of 9.4.4 Procedure for Setting Pins TRGIOA and TRGIOB	CHAPTER 9 TIMER RG
	Change of Table 14-3. A/D Conversion Time Selection	
	Change of 14.3 (7) Analog input channel specification register (ADS)	
	Change of 14.3 (11) A/D port configuration register (ADPC)	
	Change of note 2 and remark in 14.8 SNOOZE Mode Function	CHAPTER 14 A/D CONVERTER
	Addition of chapter	
	Addition of chapter	
	Change of maximum transfer rate and change of description of SNOOZE mode in 17.1.1 3-wire serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21)	CHAPTER 15 D/A CONVERTER
	Change of description of SNOOZE mode in 17.1.2 UART (UART0 to UART2)	
	Change of Figure 17-2. Block Diagram of Serial Array Unit 1	
	Change of transfer clock setting and caution 2 in Figure 17-9. Format of Serial Data Register mn (SDRmn)	
	Change of 17.3 (14) Serial standby control register m (SSCm)	
	Change of 17.3 (16) Noise filter enable register 0 (NFEN0)	
	Change of maximum transfer rate and change of description of SNOOZE mode in 17.5 Operation of 3-Wire Serial I/O (CSI00, CSI01, CSI10, CSI11, CSI20, CSI21) Communication	
		CHAPTER 16 COMPARATOR
		CHAPTER 17 SERIAL ARRAY UNIT

(25/26)

Edition	Description	Chapter
Rev. 0.02	Change of transfer rate in 17.5.1 Master transmission	CHAPTER 17 SERIAL ARRAY UNIT
	Change of transfer rate in 17.5.2 Master reception	
	Change of transfer rate in 17.5.3 Master transmission/reception	
	Change of description in 17.5.7 SNOOZE mode function	
	Change of description in 17.6 Clock Synchronous Serial Communication with Slave Select Input Function	
	Change of description of SNOOZE mode in 17.7 Operation of UART (UART0 to UART2) Communication	
	Addition of note to 17.7.1 UART transmission	
	Change of note 2 in Figure 17-102. Example of Contents of Registers for UART Transmission of UART (UART0 to UART2) (1/2)	
	Addition of note to 17.7.2 UART reception	
	Change of note 2 in Figure 17-110. Example of Contents of Registers for UART Reception of UART (UART0 to UART2) (1/2)	
	Change of description of 17.7.3 SNOOZE mode function	
	Change of caution and remark 1 in 17.7.4 Calculating baud rate	
	Change of transfer rate setting example of 17.9.5 Calculating transfer rate	
	Addition of Caution to Figure 18-9. Format of IICA Control Register 01 (IICCTL01)	CHAPTER 18 SERIAL INTERFACE IICA
	Change of 18.3 (6) IICA low-level width setting register 0 (IICWL0)	
	Change of caution in 18.4.2 Setting transfer clock by using IICWL0 and IICWHO registers	
	Change of 18.6 Timing Charts	
	Change of 19.2.1 Allocation of DTC Control Data Area and DTC Vector Table Area	CHAPTER 19 DTC
	Change of Table 19-5. DTC Activation Sources and DTC Vector Addresses	
	Change of Table 19-6. Correspondences between Interrupt Sources and Bits DTCENi0 to DTCENi7	
	Addition of caution to 19.2.12 DTC Base Address Register (DTCBAR)	
	Addition of 19.4 Notes on DTC	
	Change of Table 20-1. DTC Register Configuration	CHAPTER 20 EVENT LINK CONTROLLER (ELC)
	Change of 20.2.1 Event Output Destination Select Register n (ELSELRn) (n = 00 to 25)	
	Change of Table 21-1. Interrupt Source List	CHAPTER 21 INTERRUPT FUNCTIONS
	Change of Table 21-2. Flags Corresponding to Interrupt Request Sources	
	Change of Figure 21-2. Format of Interrupt Request Flag Registers (IF0L, IF0H, IF1L, IF1H, IF2L, IF2H) (100-pin)	
	Change of Figure 21-3. Format of Interrupt Mask Flag Registers (MK0L, MK0H, MK1L, MK1H, MK2L, MK2H) (100-pin)	
	Change of Figure 21-4. Format of Priority Specification Flag Registers (PR00L, PR00H, PR01L, PR01H, PR02L, PR02H, PR10L, PR10H, PR11L, PR11H, PR12L, PR12H) (100-pin)	
	Change of Figure 21-5. Format of External Interrupt Rising Edge Enable Registers (EGP0, EGP1) and External Interrupt Falling Edge Enable Registers (EGN0, EGN1) (100-pin)	
	Change of Table 21-3. Ports Corresponding to EGPN and EGNn bits	
	Change of Table 23-1. Operating Statuses in HALT Mode	CHAPTER 23 STANDBY FUNCTION
	Change of Table 23-2. Operating Statuses in STOP Mode	
	Change of Table 23-3. Operating Statuses in SNOOZE Mode	

(26/26)

Edition	Description	Chapter
Rev. 0.02	Change of Table 24-1. Operation Statuses During Reset Period	CHAPTER 24 RESET FUNCTION
	Change of caution in Figure 24-5. Format of Reset Control Flag Register (RESF)	
	Change of Table 26-1. LVD Operation Mode and Detection Voltage Settings for User Option Byte (000C1H/010C1H)	CHAPTER 26 VOLTAGE DETECTOR
	Change of 26.4.2 When used as interrupt mode	
	Addition of caution to Figure 27-6. Format of RAM Parity Error Control Register (RPECTL)	CHAPTER 27 SAFETY FUNCTIONS
	Change of Figure 29-2. Format of User Option Byte (000C1H/010C1H)	CHAPTER 29 OPTION BYTE
	Change of Figure 29-3. Format of User Option Byte (000C2H/010C2H)	
	Change of Table 30-1. Wiring Between RL78/G14 and Dedicated Flash Memory Programmer	CHAPTER 30 FLASH MEMORY
	Change of Figure 30-2. Communication with Dedicated Flash Memory Programmer	
	Change of Table 30-2. Pin Connection	
	Change of 30.3.1 P40/TOOL0 pin	
	Change of description of 30.4.1 Data flash overview	
	Change of description of 30.5.2 Flash memory programming mode	
	Change of Figure 30-8. Setting of Flash Memory Programming Mode	
	Change of Table 30-7. Flash Memory Control Commands	
	Addition of caution to 30.7 Flash Memory Programming by Self-Programming	
	Addition of Chapter	CHAPTER 34 ELECTRICAL SPECIFICATIONS (TARGET)
	Addition of Chapter	CHAPTER 35 PACKAGE DRAWINGS

---

**RL78/G14 User's Manual: Hardware**

Publication Date:    Rev. 0.01    Feb 28, 2011  
                            Rev. 3.31    Nov 2, 2020

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 Corporation**

TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

**Renesas Electronics America Inc. Milpitas Campus**

1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.

Tel: +1-408-432-8888, Fax: +1-408-434-5351

**Renesas Electronics America Inc. San Jose Campus**

6024 Silver Creek Valley Road, San Jose, CA 95138, USA

Tel: +1-408-284-8200, Fax: +1-408-284-2775

**Renesas Electronics Canada Limited**

9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3

Tel: +1-905-237-2004

**Renesas Electronics Europe GmbH**

Arcadiastrasse 10, 40472 Dusseldorf, Germany

Tel: +49-211-6503-0, Fax: +49-211-6503-1327

**Renesas Electronics (China) Co., Ltd.**

Room 101-T01, Floor 1, Building 7, Yard No. 7, 8th Street, Shangdi, Haidian District, Beijing 100085, 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, 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, #06-02 Singapore 339949

Tel: +65-6213-0200, Fax: +65-6213-0300

**Renesas Electronics Malaysia Sdn.Bhd.**

Unit No.3A-1 Level 3A Tower 8 UOA Business Park, No 1 Jalan Pengaturcara U1/51A, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia

Tel: +60-3-5022-1288, Fax: +60-3-5022-1290

**Renesas Electronics India Pvt. Ltd.**

No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India

Tel: +91-80-67208700

**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

RL78/G14



Renesas Electronics Corporation

R01UH0186EJ0331