To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of 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. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. 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 of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may 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.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: “Standard”, “High Quality”, and “Specific”. The recommended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as “Specific” without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as “Specific” or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is “Standard” unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   “Specific”: Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, 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, please evaluate the safety of the final products or system 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. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

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

RENESAS

User's Manual

V850E/IA3, V850E/IA4

32-bit Single-Chip Microcontrollers

Hardware

V850E/IA3:
- μPD703183
- μPD70F3184

V850E/IA4:
- μPD703185
- μPD703186
- μPD70F3186

Document No. U16543EJ4V0UD00 (4th edition)
Date Published April 2008 N

[MEMO]

NOTES FOR CMOS DEVICES

① VOLTAGE APPLICATION WAVEFORM AT INPUT PIN

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

② HANDLING OF UNUSED INPUT PINS

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

③ PRECAUTION AGAINST ESD

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

④ STATUS BEFORE INITIALIZATION

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

⑤ POWER ON/OFF SEQUENCE

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

The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

⑥ INPUT OF SIGNAL DURING POWER OFF STATE

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements.

Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

These commodities, technology or software, must be exported in accordance with the export administration regulations of the exporting country. Diversion contrary to the law of that country is prohibited.

- **The information in this document is current as of January, 2008. The information is subject to change without notice. For actual design-in, refer to the latest publications of NEC Electronics data sheets or data books, etc., for the most up-to-date specifications of NEC Electronics products. Not all products and/or types are available in every country. Please check with an NEC Electronics sales representative for availability and additional information.**
- No part of this document may be copied or reproduced in any form or by any means without the prior written consent of NEC Electronics. NEC Electronics assumes no responsibility for any errors that may appear in this document.
- NEC Electronics does not assume any liability for infringement of patents, copyrights or other intellectual property rights of third parties by or arising from the use of NEC Electronics products listed in this document or any other liability arising from the use of such products. No license, express, implied or otherwise, is granted under any patents, copyrights or other intellectual property rights of NEC Electronics or others.
- Descriptions of circuits, software and other related information in this document are provided for illustrative purposes in semiconductor product operation and application examples. The incorporation of these circuits, software and information in the design of a customer's equipment shall be done under the full responsibility of the customer. NEC Electronics assumes no responsibility for any losses incurred by customers or third parties arising from the use of these circuits, software and information.
- While NEC Electronics endeavors to enhance the quality, reliability and safety of NEC Electronics products, customers agree and acknowledge that the possibility of defects thereof cannot be eliminated entirely. To minimize risks of damage to property or injury (including death) to persons arising from defects in NEC Electronics products, customers must incorporate sufficient safety measures in their design, such as redundancy, fire-containment and anti-failure features.
- NEC Electronics products are classified into the following three quality grades: "Standard", "Special" and "Specific".
  The "Specific" quality grade applies only to NEC Electronics products developed based on a customer-designated "quality assurance program" for a specific application. The recommended applications of an NEC Electronics product depend on its quality grade, as indicated below. Customers must check the quality grade of each NEC Electronics product before using it in a particular application.
  "Standard": Computers, office equipment, communications equipment, test and measurement equipment, audio and visual equipment, home electronic appliances, machine tools, personal electronic equipment and industrial robots.
  "Special": Transportation equipment (automobiles, trains, ships, etc.), traffic control systems, anti-disaster systems, anti-crime systems, safety equipment and medical equipment (not specifically designed for life support).
  "Specific": Aircraft, aerospace equipment, submersible repeaters, nuclear reactor control systems, life support systems and medical equipment for life support, etc.

The quality grade of NEC Electronics products is "Standard" unless otherwise expressly specified in NEC Electronics data sheets or data books, etc. If customers wish to use NEC Electronics products in applications not intended by NEC Electronics, they must contact an NEC Electronics sales representative in advance to determine NEC Electronics' willingness to support a given application.

(Note)
(1) "NEC Electronics" as used in this statement means NEC Electronics Corporation and also includes its majority-owned subsidiaries.
(2) "NEC Electronics products" means any product developed or manufactured by or for NEC Electronics (as defined above).

M8E 02. 11-1

PREFACE

Readers

This manual is intended for users who wish to understand the functions of the V850E/IA3 (μPD703183, 70F3184) and V850E/IA4 (μPD703185, 703186, 70F3186) to design application systems using the V850E/IA3 and V850E/IA4.

Purpose

This manual is intended to give users an understanding of the hardware functions.

Organization

The **V850E/IA3, V850E/IA4 User's Manual** is divided into two parts: Hardware (this manual) and Architecture (**V850E1 Architecture User's Manual**). The organization of each manual is as follows:

Hardware	Architecture
• Pin functions	• Data type
• CPU function	• Register set
• On-chip peripheral functions	• Instruction format and instruction set
• Flash memory programming	• Interrupts and exceptions
• Electrical specifications	• Pipeline operation

How to Read This Manual

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

- To understand the overall functions of the V850E/IA3 and V850E/IA4
  → Read this manual according to the **CONTENTS**.

- To find the details of a register where the name is known
  → Refer to **APPENDIX B REGISTER INDEX**.

- How to interpret the register format
  → For a bit whose bit number is enclosed in angle brackets < >, its bit name is defined as a reserved word in the device file.

- To understand the details of an instruction function
  → Refer to the **V850E1 Architecture User's Manual**.

- To know the electrical specifications of the V850E/IA3 and V850E/IA4
  → See **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)** and **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**.

The "yyy bit of the xxx register" is described as the "xxx.yyy bit" in this manual. Note with caution that if "xxx.yyy" is described as is in a program, however, the compiler/assembler cannot recognize it correctly.

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.

Conventions

Data significance:	Higher digits on the left and lower digits on the right
Active low representation:	$\overline{xxx}$ (overscore over pin or signal name)
Memory map address:	Higher addresses on the top and lower addresses on the bottom
Note:	Footnote for item marked with **Note** in the text
Caution:	Information requiring particular attention
Remark:	Supplementary information
Numeric representation:	Binary … xxxx or xxxxB
	Decimal … xxxx
	Hexadecimal … xxxxH
Prefix indicating power of 2 (address space, memory capacity):	
	K (kilo): $2^{10} = 1,024$
	M (mega): $2^{20} = 1,024^2$
	G (giga): $2^{30} = 1,024^3$
Data type:	Word … 32 bits
	Halfword … 16 bits
	Byte … 8 bits

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

Documents related to V850E/IA3 and V850E/IA4

Document Name	Document No.
V850E1 Architecture User's Manual	U14559E
V850E/IA3, V850E/IA4 Hardware User's Manual	This manual
Inverter Control by V850 Series Vector Control by Hole Sensor Application Note	U17338E
Inverter Control by V850 Series Vector Control by Encoder Application Note	U17324E
Inverter Control by V850 Series 120° Excitation Method Control by Zero-Cross Detection Application Note	U17209E

Documents related to development tools (user's manuals)

Document Name		Document No.
QB-V850EIA4 (In-circuit emulator)		U17167E
QB-V850MINI (On-chip debug emulator)		U17638E
QB-MINI2 (On-chip debug emulator with programming function)		U18371E
CA850 (Ver. 3.20) (C compiler package)	Operation	U18512E
	C Language	U18513E
	Assembly Language	U18514E
	Link Directive	U18515E
PM+ (Ver. 6.30) (Project manager)		U18416E
ID850QB (Ver. 3.40) (Integrated debugger)	Operation	U18604E
TW850 (Ver. 2.00) (Performance analysis tuning tool)		U17241E
RX850 (Ver. 3.20) (Real-time OS)	Basics	U13430E
	Installation	U17419E
	Technical	U13431E
	Task Debugger	U17420E
RX850 Pro (Ver. 3.21) (Real-time OS)	Basics	U18165E
	Installation	U17421E
	Technical	U13772E
	Task Debugger	U17422E
AZ850 (Ver. 3.30) (System performance analyzer)		U17423E
PG-FP4 Flash Memory Programmer		U15260E
PG-FP5 Flash Memory Programmer		U18865E

CONTENTS

CHAPTER 1 INTRODUCTION

The V850E/IA3 and V850E/IA4 are products of the NEC Electronics V850 single-chip microcontrollers. This chapter gives a simple outline of the V850E/IA3 and V850E/IA4.

1.1 Overview

The V850E/IA3 and V850E/IA4 are 32-bit single-chip microcontrollers that integrate the V850E1 CPU, which is a 32-bit RISC-type CPU core for ASIC, newly developed as the CPU core central to system LSI for the current age of system-on-chip. This device incorporates ROM, RAM, and various peripheral functions such as DMA controller, timer counter, watchdog timer, serial interfaces, an A/D converter, an A/D converter of first-order $\Delta\Sigma$ conversion method, ROM correction, and on-chip debugging for realizing high-capacity data processing and sophisticated real-time control.

(1) V850E1 CPU

The V850E1 CPU is a CPU core of the V850 CPU, which is the CPU core integrated in the V850 microcontrollers, and has added instructions supporting high-level languages, such as C-language switch statement processing, table lookup branching, stack frame creation/deletion, and data conversion. This enhances the performance of both data processing and control.

It is possible to use the software resources of the V850 CPU integrated system since the instruction codes of the V850E1 are upwardly compatible at the object code level with those of the V850 CPU.

(2) On-chip flash memory (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4))

The on-chip flash memory version (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4)) has on-chip flash memory, which is capable of high-speed access, and since it is possible to rewrite a program with the V850E/IA3 and V850E/IA4 mounted as is in the application system, system development time can be reduced and system maintainability after shipment can be markedly improved.

(3) A full range of middleware and development environment products

The V850E/IA3 and V850E/IA4 can execute middleware such as JPEG, JBIG, and MH/MR/MMR at high speed. Also, middleware that enables speech recognition, voice synthesis, and other such processing is available, and by including these middleware programs, a multimedia system can be easily realized.

A development environment system that includes an optimized C compiler, debugger, on-chip debug emulator (μPD70F3186 (V850E/IA4) only), system performance analyzer, and other elements is also available.

Table 1-1 shows the differences in functions between the V850E/IA3 and V850E/IA4.

Table 1-1. Differences in Functions Between V850E/IA3 and V850E/IA4

Item		V850E/IA3	V850E/IA4
Internal ROM/RAM (mask ROM version)		128 KB/6 KB	128 KB/6 KB 256 KB/12 KB
Port function	I/O	44	56
	Input	6	8
	On-chip pull-up resistor	Provided (ports 0, 1, 3, 4, DL only)	Provided (ports 0 to 5, DL only)
Interrupt source		External interrupt: 7 (without NMI) Internal interrupt: 48	External interrupt: 8 (without NMI) Internal interrupt: 52
Timers Q0, Q1		Timer Q0 Timer Q1 (without output pin)	Timer Q0 Timer Q1
Timers P0 to P3		Timer P0 Timer P1 (without output pin) Timer P2 Timer P3 (without output pin)	Timer P0 Timer P1 (without output pin) Timer P2 Timer P3
Timers ENC10, ENC11		Timer ENC10	Timer ENC10 Timer ENC11
Motor control function	Output pin for 6-phase PWM mode	TMQ0 + TMQOP0 (+TMP0)	TMQ0 + TMQOP0 (+TMP0) TMQ1 + TMQOP1 (+TMP1)
A/D converters 0, 1	Analog input	Total of two circuits: 6 channels A/D converter 0: 2 channels A/D converter 1: 4 channels	Total of two circuits: 8 channels A/D converter 0: 4 channels A/D converter 1: 4 channels
	Operational amplifier for input level amplification	Total of two circuits: 5 channels A/D converter 0: 2 channels (ANI00, ANI01) A/D converter 1: 3 channels (ANI10 to ANI12)	Total of two circuits: 6 channels A/D converter 0: 3 channels (ANI00 to ANI02) A/D converter 1: 3 channels (ANI10 to ANI12)
	Overvoltage detection comparator	Total of two circuits: 5 channels A/D converter 0: 2 channels (ANI00, ANI01) A/D converter 1: 3 channels (ANI10 to ANI12)	Total of two circuits: 6 channels A/D converter 0: 3 channels (ANI00 to ANI02) A/D converter 1: 3 channels (ANI10 to ANI12)
A/D converter 2	Analog input	6 channels	8 channels
On-chip debug function		None	Provided (μPD70F3186 only)
Package		80-pin plastic QFP	100-pin plastic LQFP 100-pin plastic QFP

1.2 V850E/IA3

1.2.1 Features (V850E/IA3)

- Minimum instruction execution time:
  15.6 ns (at internal 64 MHz operation)

- General-purpose registers: 32 bits × 32

- CPU features
  Signed multiplication (16 bits × 16 bits → 32 bits or 32 bits × 32 bits → 64 bits):
  1 to 2 clocks
  Saturated operation instructions (with overflow/underflow detection function)
  32-bit shift instructions: 1 clock
  Bit manipulation instructions
  Load/store instructions with long/short format
  Signed load instructions

- Internal memory

Part Number	Internal ROM/Flash Memory	Internal RAM
μPD703183	128 KB (mask ROM)	6 KB
μPD70F3184	256 KB (flash memory)	12 KB

- ROM correction: Four places can be corrected.

- Interrupts/exceptions:
  Non-maskable interrupts: 1 source (external: none, internal: 1)
  Maskable interrupts: 55 sources (external: 7, internal: 48)
  Software exceptions: 32 sources
  Exception traps: 2 sources

- DMA controller: 4 channels
  Transfer unit: 8 bits/16 bits
  Maximum transfer count: 65,536 (2^{16})
  Transfer type: 2-cycle
  Transfer mode: Single/single step/block
  Transfer target: On-chip peripheral I/O ↔ internal RAM,
  on-chip peripheral I/O ↔ on-chip peripheral I/O
  Transfer request: On-chip peripheral I/O/software
  Next address setting function

- I/O lines: Total: 50 (Input-only ports: 6, I/O ports: 44)

○ Timer/counter function:
- 16-bit up/down counter/timer (TMENC) for 2-phase encoder input: 1 channel
- 16-bit interval timer M (TMM): 1 channel
- 16-bit timer/event counter Q (TMQ): 2 channels
- 16-bit timer/event counter P (TMP): 4 channels
- Motor control function (uses timer TMQ: 1 channel (TMQ0), TMP: 1 channel (TMP0))
  - 16-bit accuracy 6-phase PWM function with deadtime: 1 channel
  - High-impedance output control function
  - Timer tuning operation function
  - Arbitrary cycle setting function
  - Arbitrary deadtime setting function
- Watchdog timer: 1 channel

○ Serial interfaces:
- Asynchronous serial interface A (UARTA)
- Clocked serial interface B (CSIB)
  - CSIB0: 1 channel
  - UARTA0: 1 channel
  - CSIB1/UARTA1: 1 channel

○ A/D converter
- 10-bit resolution A/D converters (A/D converters 0 and 1): 2 channels + 4 channels (2 units)
  - The two A/D converter 0 channels and three of the four A/D converter 1 channels are provided with an operational amplifier for input level amplification (gain = ×2.5, ×5) and a comparator for overvoltage detection (input voltage range = $0.1AV_{DD}$ to $0.5AV_{DD}$).
- A/D converter 2, using first-order ΔΣ conversion method: 6 channels

○ Clock generator:
- 4 to 8 MHz resonator connectable (external clock input prohibited)
- Multiplication function by PLL clock synthesizer (fixed to multiplication by eight, f_{XX} = 32 to 64 MHz)
  - PLL operation specifiable by PLLSIN pin
- CPU clock division function (f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$)

○ Power-save function:
- HALT/IDLE/STOP mode

○ Package:
- 80-pin plastic QFP (14 × 14)

○ Operation supply voltage
- Internal unit: V_{DD} = 2.3 to 2.7 V
- Oscillation block: CV_{DD} = 2.3 to 2.7 V
- External pin: EV_{DD} = 4.0 to 5.5 V (4.5 to 5.5 V when using A/D converters 0 to 2)
- A/D converter block: AV_{DD} = 4.5 to 5.5 V

○ Operation ambient temperature
- T_A = −40 to +85°C

1.2.2 Applications (V850E/IA3)

- Commercial equipment (such as inverter air conditioners, washing machines, driers, refrigerators, etc.)
- Industrial equipment (such as motor control and general-purpose inverters, etc.)

1.2.3 Ordering information (V850E/IA3)

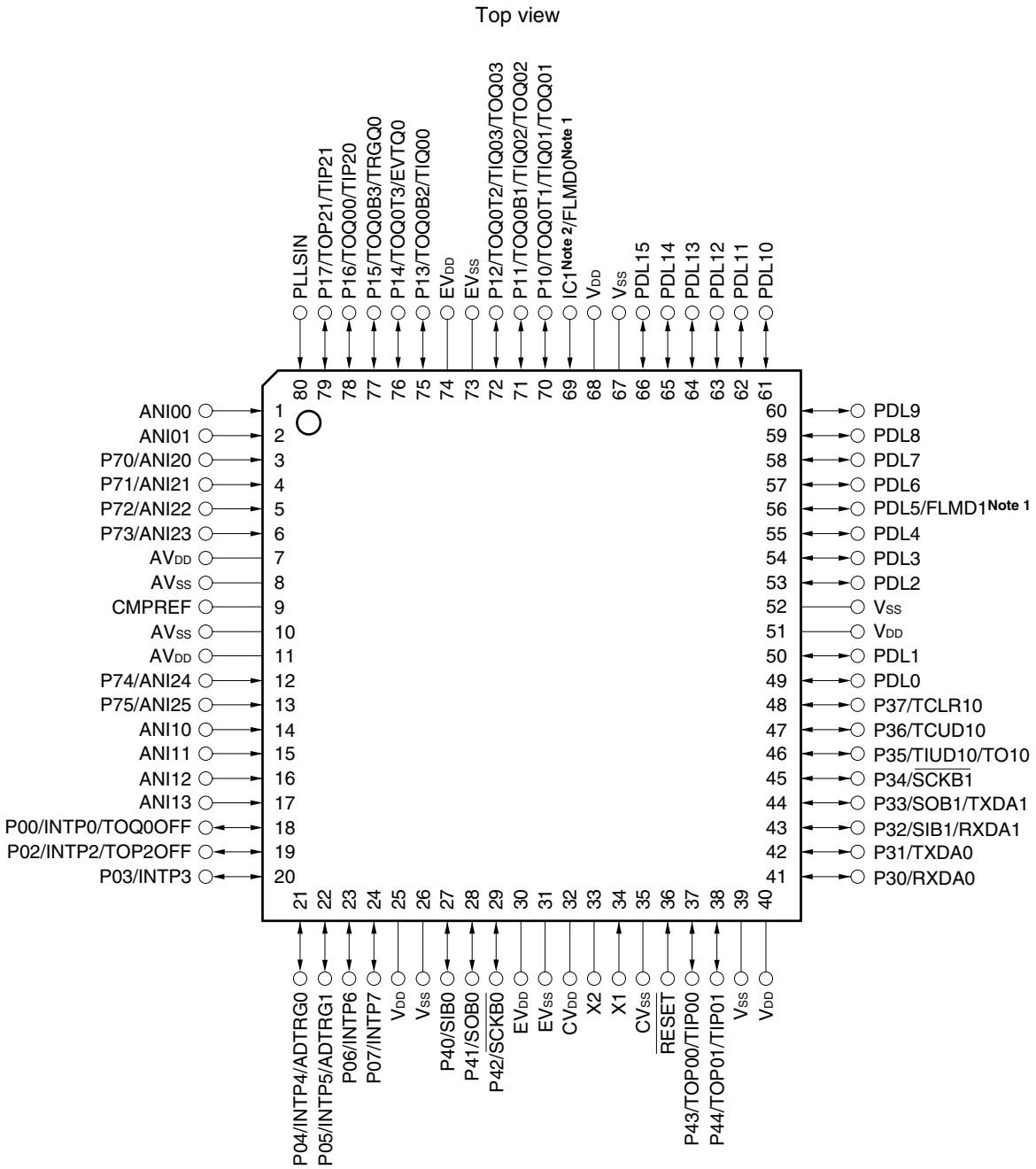
Part Number	Package	Internal ROM/Flash Memory
μPD703183GC-xxx-8BT-A	80-pin plastic QFP (14 × 14)	Mask ROM (128 KB)
μPD70F3184GC-8BT-A	80-pin plastic QFP (14 × 14)	Flash memory (256 KB)

Remarks 1. xxx indicates ROM code suffix.

2. Products with -A at the end of the part number are lead-free products.

1.2.4 Pin configuration (V850E/IA3)

- **80-pin plastic QFP (14 × 14)**
  μPD703183GC-xxx-8BT-A
  μPD70F3184GC-8BT-A



Notes 1. μPD70F3184 only

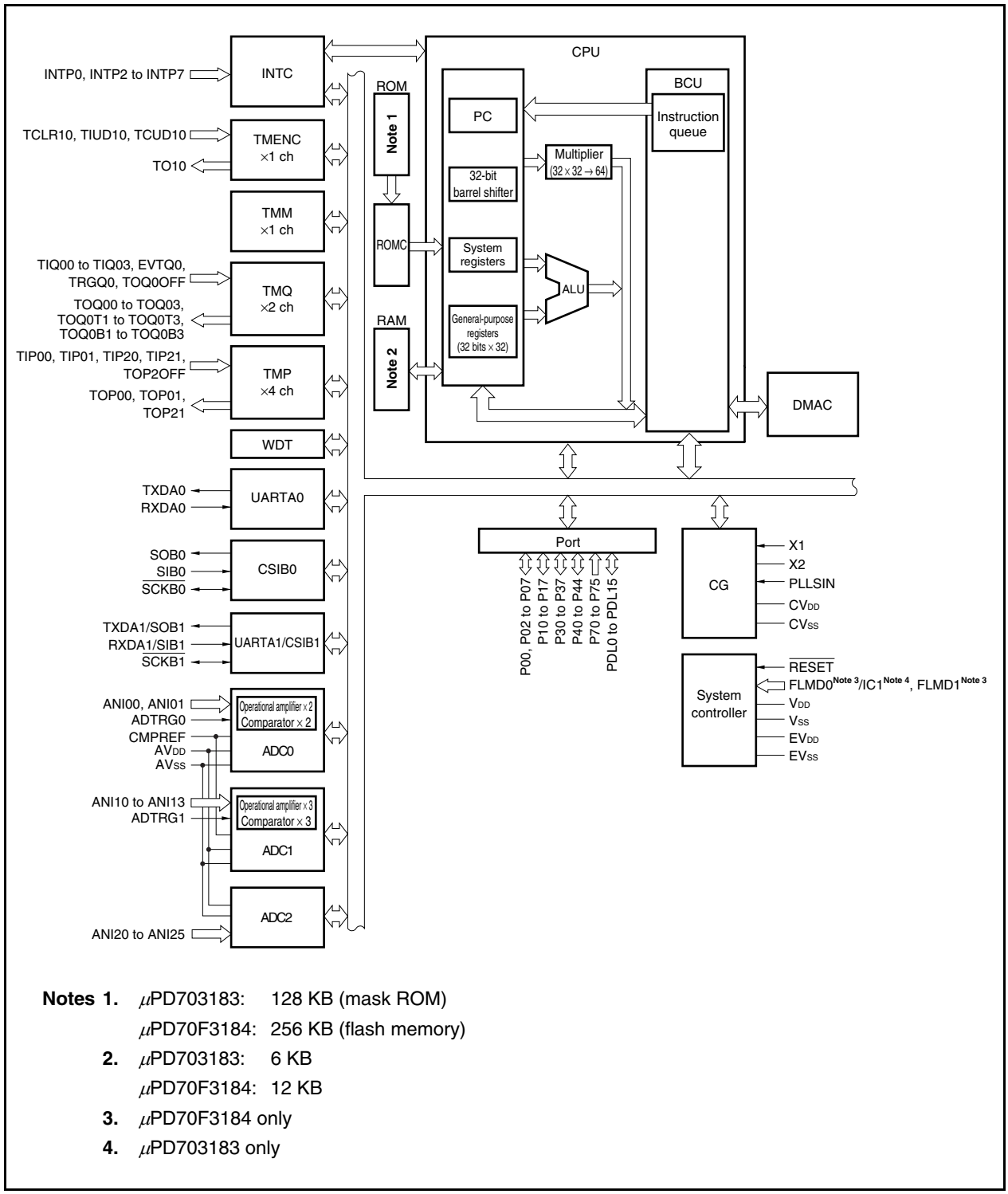
2. μPD703183 only

Pin Identification (V850E/IA3)

ADTRG0, ADTRG1: A/D trigger input
ANI00, ANI01, ANI10 to ANI13, ANI20 to ANI25: Analog input
AV_{DD}: Analog power supply
AV_{SS}: Analog ground
CMPREF: Comparator reference voltage
CV_{DD}: Power supply for clock generator
CV_{SS}: Ground for clock generator
EV_{DD}: Power supply for port
EV_{SS}: Ground for port
EVTQ0: Timer event count input
FLMD0, FLMD1: Flash programming mode
IC1: Internally connected
INTP0, INTP2 to INTP7: External interrupt input
P00, P02 to P07: Port 0
P10 to P17: Port 1
P30 to P37: Port 3
P40 to P44: Port 4
P70 to P75: Port 7
PDL0 to PDL15: Port DL
PLLSIN: PLL select input
$\overline{\text{RESET}}$: Reset
RXDA0, RXDA1: Receive data
$\overline{\text{SCKB0}}$, $\overline{\text{SCKB1}}$: Serial clock
SIB0, SIB1: Serial input
SOB0, SOB1: Serial output
TCLR10: Timer clear
TCUD10: Timer control pulse input
TIP00, TIP01, TIP20, TIP21, TIQ00 to TIQ03: Timer trigger input
TIUD10: Timer count pulse input
TO10, TOP00, TOP01, TOP21, TOQ0B1 to TOQ0B3, TOQ0T1 to TOQ0T3, TOQ00 to TOQ03: Timer output
TOP2OFF, TOQ0OFF: Timer output off
TRGQ0: Timer trigger input
TXDA0, TXDA1: Transmit data
V_{DD}: Power supply
V_{SS}: Ground
X1, X2: Clock oscillator pin

1.2.5 Function blocks (V850E/IA3)

(1) Internal block diagram



Notes 1. μPD703183: 128 KB (mask ROM)
μPD70F3184: 256 KB (flash memory)

2. μPD703183: 6 KB
μPD70F3184: 12 KB

3. μPD70F3184 only

4. μPD703183 only

(2) Internal units

(a) CPU

The CPU uses five-stage pipeline control to enable single-clock execution of address calculations, arithmetic logic operations, data transfers, and almost all other instruction processing.

Other dedicated on-chip hardware, such as a multiplier (32 bits × 32 bits → 64 bits) and a barrel shifter (32 bits), help accelerate complex processing.

(b) Bus control unit (BCU)

The BCU controls the internal bus.

(i) DMA controller (DMAC)

This controller controls data transfer between on-chip peripheral I/O and internal RAM or on-chip peripheral I/O and on-chip peripheral I/O instead of the CPU.

The transfer type is two-cycle transfer, and single transfer, single-step transfer, and block transfer are used in transfer mode.

(c) ROM

This is mask ROM or flash memory that is mapped from address 00000000H.

During instruction fetch, ROM/flash memory can be accessed from the CPU in 1-clock cycles. The internal ROM capacity and area differ as follows depending on the product.

Part Number	Internal ROM Capacity	Internal ROM Area
μPD703183	128 KB (mask ROM)	x0000000H to x001FFFFH
μPD70F3184	256 KB (flash memory)	x0000000H to x003FFFFH

(d) RAM

The internal RAM capacity and area differ as follows depending on the product.

During instruction fetch or data access, data can be accessed from the CPU in 1-clock cycles.

Part Number	Internal RAM Capacity	Internal RAM Area
μPD703183	6 KB	xFFFD800H to xFFFEFFFH
μPD70F3184	12 KB	xFFFC000H to xFFFEFFFH

(e) Interrupt controller (INTC)

This controller handles hardware interrupt requests (INTP0, INTP2 to INTP7) from on-chip peripheral hardware and external hardware. Eight levels of interrupt priorities can be specified for these interrupt requests, and multiple-interrupt servicing control can be performed.

(f) Clock generator (CG)

The clock generator includes two basic operation modes: PLL mode (fixed to multiplication by eight) and clock-through mode. It generates four types of clocks (f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$), and supplies one of them as the operating clock for the CPU (f_{CPU}).

(g) Timer/counter

This unit incorporates one 16-bit up/down counter/timer channel (TMENC1) for 2-phase encoder input, one 16-bit interval timer (TMM) channel, two 16-bit timer/event counter (TMQ) channels, and four 16-bit timer/event counter (TMP) channels, and can measure pulse interval widths or frequency, enable an inverter function for motor control, and output a programmable pulse.

(h) Watchdog timer (WDT)

A watchdog timer is equipped to detect program loops, system abnormalities, etc.

It generates a non-maskable interrupt request signal (INTWDT) or internal reset signal (WDTRES) after an overflow occurs.

(i) Serial interface

The V850E/IA3 includes four serial interface channels: for two asynchronous serial interface A (UARTA) channels and two clocked serial interface B (CSIB) channels. Of these, UARTA1 and CSIB1 share a pin.

For UARTA, data is transferred via the TXDAn and RXDAn pins (n = 0, 1).

For CSIB, data is transferred via the SOBn, SIBn, and $\overline{\text{SCKBn}}$ pins (n = 0, 1).

(j) A/D converter (ADC)

One channel is provided for each of the high-speed, high-resolution 10-bit A/D converters ADC0 and ADC1 (total of two channels), which have two and four analog input pins respectively, and one channel is provided for the low-speed first-order $\Delta\Sigma$ conversion method 8-/10-bit A/D converter ADC2, which has six analog input pins.

Both the ADC0 channels and three of the ADC1 channels include an operation amplifier and a comparator so that these A/D converters can amplify an analog input voltage and detect overvoltage input.

(k) ROM correction

A ROM correction function that replaces part of a program in the mask ROM or flash memory with a program in the internal RAM is provided. Up to four correction addresses can be specified.

(l) Ports

As shown below, the following ports have general-purpose port functions and control pin functions.

Port	I/O	Control Function
Port 0	7-bit I/O	Timer/counter input, external interrupt input, external trigger input of A/D converters 0, 1
Port 1	8-bit I/O	Timer/counter I/O
Port 3	8-bit I/O	Serial interface I/O, timer/counter I/O
Port 4	5-bit I/O	Serial interface I/O, timer/counter I/O
Port 7	6-bit input	A/D converter 2 input
Port DL	16-bit I/O	–

1.3 V850E/IA4

1.3.1 Features (V850E/IA4)

○ Minimum instruction execution time:
15.6 ns (at internal 64 MHz operation)

○ General-purpose registers: 32 bits × 32

○ CPU features: Signed multiplication (16 bits × 16 bits → 32 bits or 32 bits × 32 bits → 64 bits): 1 to 2 clocks
Saturated operation instructions (with overflow/underflow detection function)
32-bit shift instructions: 1 clock
Bit manipulation instructions
Load/store instructions with long/short format
Signed load instructions

○ Internal memory

Part Number	Internal ROM/Flash Memory	Internal RAM
μPD703185	128 KB (mask ROM)	6 KB
μPD703186	256 KB (mask ROM)	12 KB
μPD70F3186	256 KB (flash memory)	12 KB

○ ROM correction: Four places can be corrected.

○ On-chip debug function JTAG interface (μPD70F3186 only)

○ Interrupts/exceptions: Non-maskable interrupts: 1 source (external: none, internal: 1)
Maskable interrupts: 60 sources (external: 8, internal: 52)
Software exceptions: 32 sources
Exception traps: 2 sources

○ DMA controller: 4 channels
Transfer unit: 8 bits/16 bits
Maximum transfer count: 65,536 (2^{16})
Transfer type: 2-cycle
Transfer mode: Single/single step/block
Transfer target: On-chip peripheral I/O ↔ internal RAM,
on-chip peripheral I/O ↔ on-chip peripheral I/O
Transfer request: On-chip peripheral I/O/software
Next address setting function

○ I/O lines: Total: 64 (Input ports: 8, I/O ports: 56)

○ Timer/counter function:
16-bit up/down counter/timer (TMENC) for 2-phase encoder input: 2 channels
16-bit interval timer (TMM): 1 channel
16-bit timer/event counter (TMQ): 2 channels
16-bit timer/event counter (TMP): 4 channels
Motor control function (uses timer TMQ: 2 channels (TMQ0 and TMQ1), TMP: 2 channels (TMP0 and TMP1))
- 16-bit accuracy 6-phase PWM function with deadtime: 2 channels
- High-impedance output control function
- Timer tuning operation function
- Arbitrary cycle setting function
- Arbitrary deadtime setting function

Watchdog timer: 1 channel

○ Serial interfaces:
Asynchronous serial interface A (UARTA)
Clocked serial interface B (CSIB)
- CSIB0: 1 channel
- UARTA0: 1 channel
- CSIB1/UARTA1: 1 channel

○ A/D converter:
10-bit resolution A/D converters (A/D converters 0 and 1): 4 channels + 4 channels (2 units)
Three of the four A/D converter channels are provided with an operational amplifier for input level amplification (gain = ×2.5, ×5) and a comparator for overvoltage detection (input voltage range = $0.1AV_{DD}$ to $0.5AV_{DD}$) (2 units).
A/D converter 2, using first-order ΔΣ conversion method: 8 channels

○ Clock generator:
4 to 8 MHz resonator connectable (external clock input prohibited)
Multiplication function by PLL clock synthesizer (fixed to multiplication by eight, f_{XX} = 32 to 64 MHz)
PLL operation specifiable by PLLSIN pin
CPU clock division function (f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$)

○ Power-save function:
HALT/IDLE/STOP mode

○ Package:
100-pin plastic LQFP (fine pitch) (14 × 14)
100-pin plastic QFP (14 × 20)

○ Operation supply voltage
- Internal unit: V_{DD} = 2.3 to 2.7 V
- Oscillation block: CV_{DD} = 2.3 to 2.7 V
- External pin: EV_{DD} = 4.0 to 5.5 V (4.5 to 5.5 V when using A/D converters 0 to 2)
- A/D converter block: AV_{DD} = 4.5 to 5.5 V

○ Operation ambient temperature
T_A = −40 to +85°C

1.3.2 Applications (V850E/IA4)

- Commercial equipment (such as inverter air conditioners, washing machines, driers, refrigerators, etc.)
- Industrial equipment (such as motor control and general-purpose inverters, etc.)

1.3.3 Ordering information (V850E/IA4)

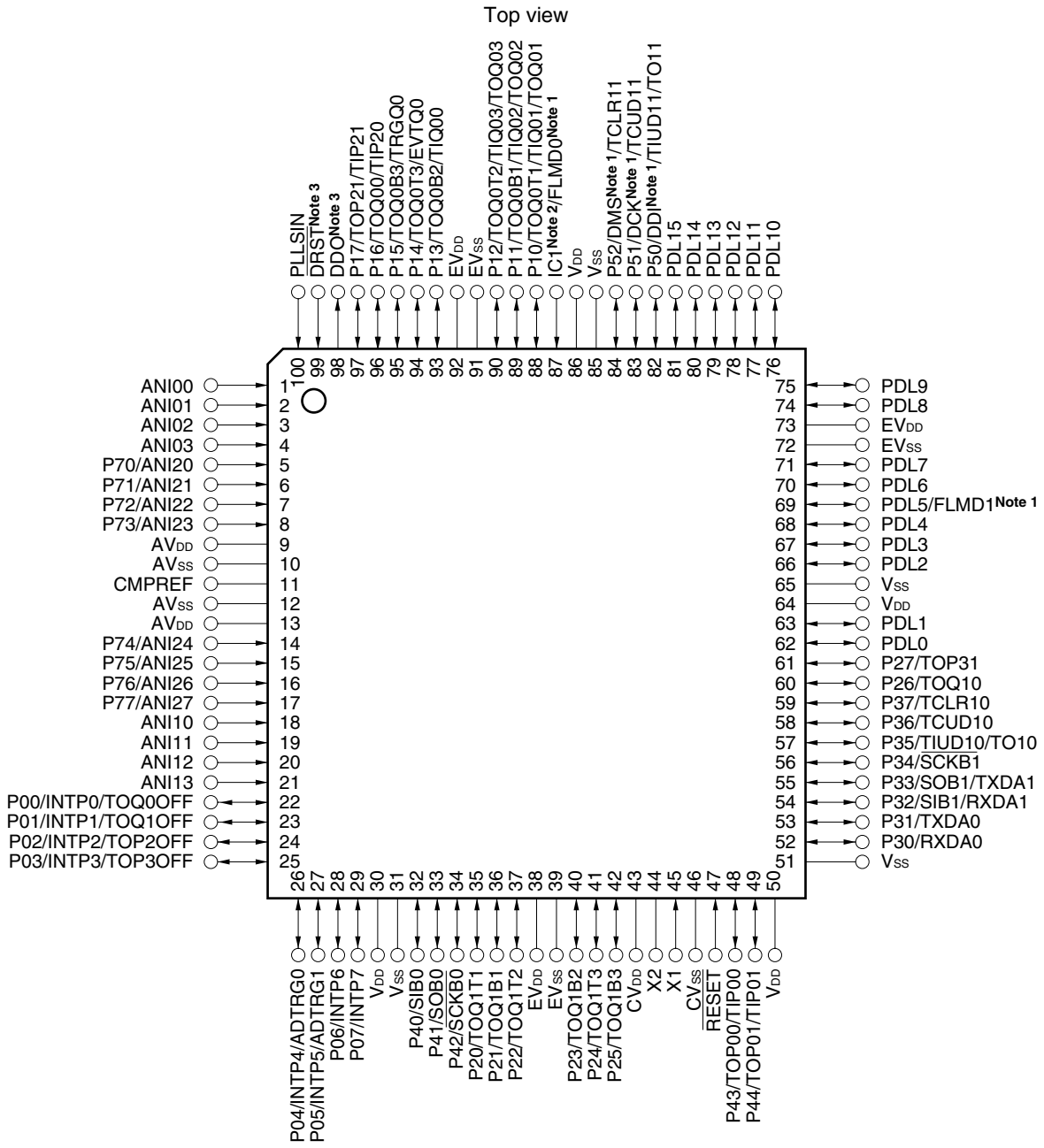
Part Number	Package	Internal ROM/Flash Memory
μPD703185GC-xxx-8EU-A	100-pin plastic LQFP (fine pitch) (14 × 14)	Mask ROM (128 KB)
μPD703185GF-xxx-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM (128 KB)
μPD703186GC-xxx-8EU-A	100-pin plastic LQFP (fine pitch) (14 × 14)	Mask ROM (128 KB)
μPD703186GF-xxx-3BA-A	100-pin plastic QFP (14 × 20)	Mask ROM (128 KB)
μPD70F3186GC-8EU-A	100-pin plastic LQFP (fine pitch) (14 × 14)	Flash memory (256 KB)
μPD70F3186GF-3BA-A	100-pin plastic QFP (14 × 20)	Flash memory (256 KB)

Remarks 1. xxx indicates ROM code suffix.

2. Products with -A at the end of the part number are lead-free products.

1.3.4 Pin configuration (V850E/IA4)

- **100-pin plastic LQFP (fine pitch) (14 × 14)**
  μPD703185GC-xxx-8EU-A
  μPD703186GC-xxx-8EU-A
  μPD70F3186GC-8EU-A



Notes 1. μPD70F3186 only

2. μPD703185, 703186 only

3. μPD70F3186 only
For the μPD703185 and 703186, read as follows.
DDO: IC2, $\overline{\text{DRST}}$: IC3

Caution When using the μPD703185 or 703186, leave the IC2 and IC3 pins open when they are not used.

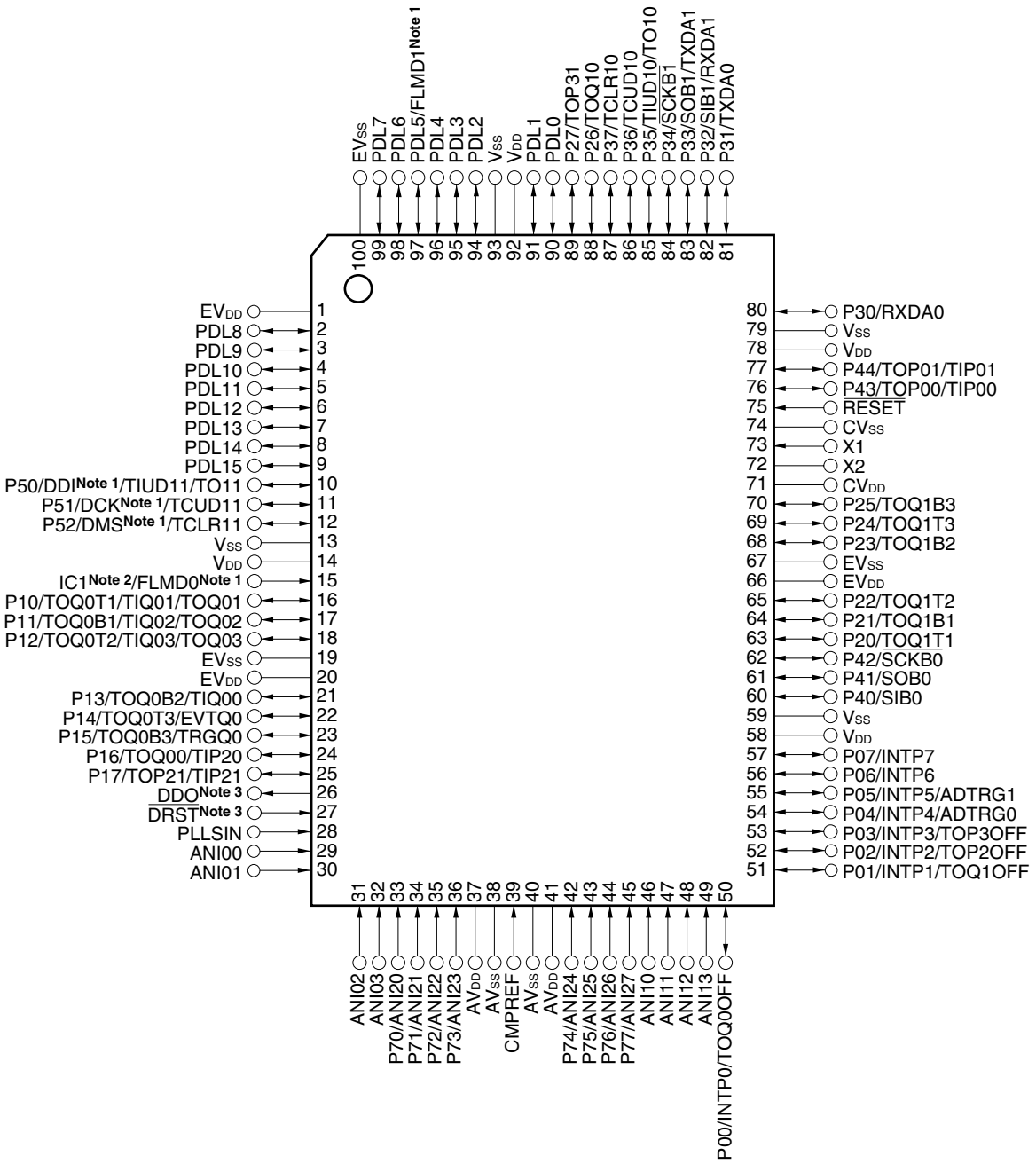
- **100-pin plastic QFP (14 × 20)**

  μPD703185GF-xxx-3BA-A

  μPD703186GF-xxx-3BA-A

  μPD70F3186GF-3BA-A

Top view



Notes 1. μPD70F3186 only

2. μPD703185, 703186 only

3. μPD70F3186 only

For the μPD703185 and 703186, read as follows.

DDO: IC2, $\overline{\text{DRST}}$: IC3

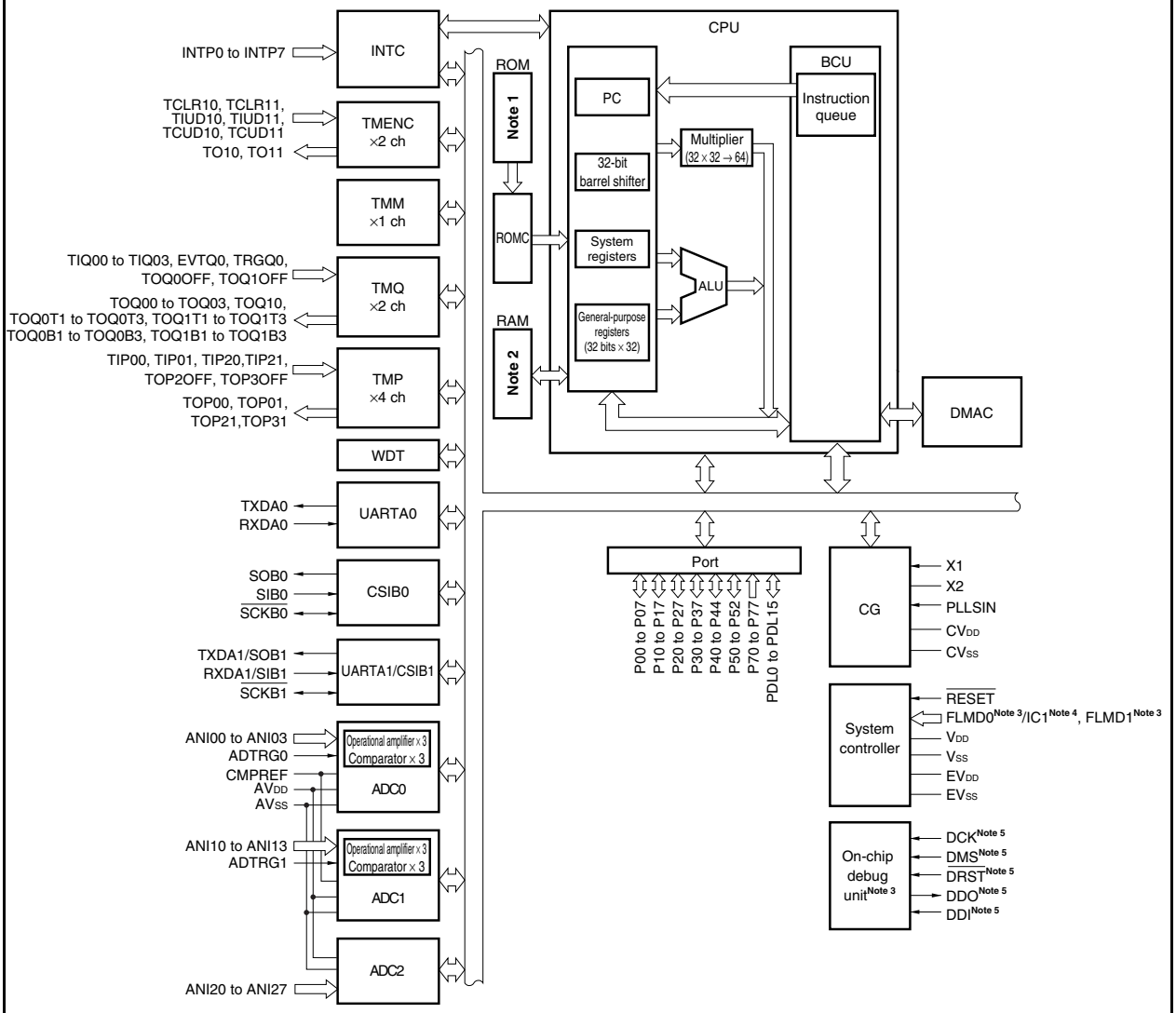
Caution When using the μPD703185 or 703186, leave the IC2 and IC3 pins open when they are not used.

Pin Identification (V850E/IA4)

ADTRG0, ADTRG1: A/D trigger input
ANI00 to ANI03,
ANI10 to ANI13,
ANI20 to ANI27: Analog input
AV_{DD}: Analog power supply
AV_{SS}: Analog ground
CMPREF: Comparator reference voltage
CV_{DD}: Power supply for clock generator
CV_{SS}: Ground for clock generator
DCK: Debug clock
DDI: Debug data input
DDO: Debug data output
DMS: Debug mode select
$\overline{DRST}$: Debug reset
EV_{DD}: Power supply for port
EV_{SS}: Ground for port
EVTQ0: Timer event count input
FLMD0, FLMD1: Flash programming mode
IC1 to IC3: Internally connected
INTP0 to INTP7: External interrupt input
P00 to P07: Port 0
P10 to P17: Port 1
P20 to P27: Port 2
P30 to P37: Port 3
P40 to P44: Port 4
P50 to P52: Port 5
P70 to P77: Port 7
PDL0 to PDL15: Port DL
PLLSIN: PLL select input
$\overline{RESET}$: Reset
RXDA0, RXDA1: Receive data
$\overline{SCKB0}$, $\overline{SCKB1}$: Serial clock
SIB0, SIB1: Serial input
SOB0, SOB1: Serial output
TCLR10, TCLR11: Timer clear
TCUD10, TCUD11: Timer control pulse input
TIP00, TIP01,
TIP20, TIP21,
TIQ00 to TIQ03: Timer trigger input
TIUD10, TIUD11: Timer count pulse input
TO10, TO11,
TOP00, TOP01,
TOP21, TOP31,
TOQ0B1 to TOQ0B3,
TOQ0T1 to TOQ0T3,
TOQ00 to TOQ03,
TOQ1B1 to TOQ1B3,
TOQ1T1 to TOQ1T3,
TOQ10: Timer output
TOP2OFF, TOP3OFF,
TOQ0OFF, TOQ1OFF: Timer output off
TRGQ0: Timer trigger input
TXDA0, TXDA1: Transmit data
V_{DD}: Power supply
V_{SS}: Ground
X1, X2: Clock oscillator pin

1.3.5 Function blocks (V850E/IA4)

(1) Internal block diagram



Notes 1. μPD703185: 128 KB (mask ROM)
μPD703186: 256 KB (mask ROM)
μPD70F3186: 256 KB (flash memory)

2. μPD703185: 6 KB
μPD703186, 70F3186: 12 KB

3. μPD70F3186 only

4. μPD703185, 703186 only

5. μPD70F3186 only
For the μPD703185 and 703186, read as follows.
DDO: IC2, $\overline{\text{DRST}}$: IC3

(2) Internal units

(a) CPU

The CPU uses five-stage pipeline control to enable single-clock execution of address calculations, arithmetic logic operations, data transfers, and almost all other instruction processing.

Other dedicated on-chip hardware, such as a multiplier (32 bits × 32 bits → 64 bits) and a barrel shifter (32 bits), help accelerate complex processing.

(b) Bus control unit (BCU)

The BCU controls the internal bus.

(i) DMA controller (DMAC)

This controller controls data transfer between on-chip peripheral I/O and internal RAM or on-chip peripheral I/O and on-chip peripheral I/O instead of the CPU.

The transfer type is two-cycle transfer, and single transfer, single-step transfer, and block transfer are used in transfer mode.

(c) ROM

This is mask ROM or flash memory that is mapped from address 00000000H.

During instruction fetch, ROM/flash memory can be accessed from the CPU in 1-clock cycles. The internal ROM capacity and area differ as follows depending on the product.

Part Number	Internal ROM Capacity	Internal ROM Area
μPD703185	128 KB (mask ROM)	x0000000H to x001FFFFH
μPD703186	256 KB (mask ROM)	x0000000H to x003FFFFH
μPD70F3186	256 KB (flash memory)	x0000000H to x003FFFFH

(d) RAM

The internal RAM capacity and area differ as follows depending on the product.

During instruction fetch or data access, data can be accessed from the CPU in 1-clock cycles.

Part Number	Internal RAM Capacity	Internal RAM Area
μPD703185	6 KB	xFFFD800H to xFFFEFFFH
μPD703186	12 KB	xFFFC000H to xFFFEFFFH
μPD70F3186	12 KB	xFFFC000H to xFFFEFFFH

(e) Interrupt controller (INTC)

This controller handles hardware interrupt requests (INTP0 to INTP7) from on-chip peripheral hardware and external hardware. Eight levels of interrupt priorities can be specified for these interrupt requests, and multiple-interrupt servicing control can be performed.

(f) Clock generator (CG)

The clock generator includes two basic operation modes: PLL mode (fixed to multiplication by eight) and clock-through mode. It generates four types of clocks (f_{XX}, $f_{XX}/2$, $f_{XX}/4$, $f_{XX}/8$), and supplies one of them as the operating clock for the CPU (f_{CPU}).

(g) Timer/counter

This unit incorporates two 16-bit up/down counter/timer (TMENC1) channels for 2-phase encoder input, one 16-bit interval timer (TMM) channel, two 16-bit timer/event counter (TMQ) channels, and four 16-bit timer/event counter (TMP) channels, and can measure pulse interval widths or frequency, enable an inverter function for motor control, and output a programmable pulse.

(h) Watchdog timer (WDT)

A watchdog timer is equipped to detect program loops, system abnormalities, etc.

It generates a non-maskable interrupt request signal (INTWDT) or internal reset signal (WDTRES) after an overflow occurs.

(i) Serial interface

The V850E/IA4 includes four serial interface channels: for two asynchronous serial interface A (UARTA) channels and two clocked serial interface B (CSIB) channels. Of these, UARTA1 and CSIB1 share a pin.

For UARTA, data is transferred via the TXDAn and RXDAn pins (n = 0, 1).

For CSIB, data is transferred via the SOBn, SIBn, and $\overline{\text{SCKBn}}$ pins (n = 0, 1).

(j) A/D converter (ADC)

Two channels are provided for high-speed, high-resolution 10-bit A/D converters ADC0 and ADC1, which have four analog input pins, and one channel is provided for a low-speed first-order $\Delta\Sigma$ conversion method 8-/10-bit A/D converter ADC2, which has eight analog input pins.

Three of the four channels of ADC0 and ADC1 include an operation amplifier and a comparator, so that these A/D converters can amplify an analog input voltage and detect overvoltage input.

(k) ROM correction

A ROM correction function that replaces part of a program in the mask ROM or flash memory with a program in the internal RAM is provided. Up to four correction addresses can be specified.

(l) On-chip debug function (μPD70F3186 only)

An on-chip debug function via an on-chip debug emulator using JTAG interface is provided.

(m) Ports

As shown below, the following ports have general-purpose port functions and control pin functions.

Port	I/O	Control Function
Port 0	8-bit I/O	Timer/counter input, external interrupt input, external trigger input of A/D converters 0, 1
Port 1	8-bit I/O	Timer/counter I/O
Port 2	8-bit I/O	Timer/counter output
Port 3	8-bit I/O	Serial interface I/O, timer/counter I/O
Port 4	5-bit I/O	Serial interface I/O, timer/counter I/O
Port 5	3-bit I/O	Timer/counter I/O, debug I/O
Port 7	8-bit input	A/D converter 2 input
Port DL	16-bit I/O	–

CHAPTER 2 PIN FUNCTIONS

The names and functions of the pins in the V850E/IA3 and V850E/IA4 are listed below. These pins can be divided into port pins and non-port pins according to their function.

2.1 List of Pin Functions

There are two power supplies for the I/O buffer of a pin: AV_{DD} and EV_{DD}. The relationship between each power supply and the pins is shown below.

Table 2-1. I/O Buffer Power Supplies for Each Pin

(a) V850E/IA3

Power Supply	Corresponding Pins
AV_{DD}	P70 to P75
EV_{DD}	P00, P02 to P07, P10 to P17, P30 to P37, P40 to P44, PDL0 to PDL15, $\overline{\text{RESET}}$

(b) V850E/IA4

Power Supply	Corresponding Pins
AV_{DD}	P70 to P77
EV_{DD}	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15, $\overline{\text{RESET}}$, DCK[Note], DMS[Note], DDI[Note], DDO[Note], $\overline{\text{DRST}}$[Note]

Note μPD70F3186 only

(1) Port pins

(1/3)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
P00	18	22	50	I/O	Port 0 V850E/IA3: 7-bit I/O port V850E/IA4: 8-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when the pins function as input pins of the alternate function).	INTP0/TOQ0OFF
P01Note	–	23	51			INTP1Note/TOQ1OFFNote
P02	19	24	52			INTP2/TOP2OFF
P03	20	25	53			INTP3/TOP3OFFNote
P04	21	26	54			INTP4/ADTRG0
P05	22	27	55			INTP5/ADTRG1
P06	23	28	56			INTP6
P07	24	29	57			INTP7
P10	70	88	16	I/O	Port 1 8-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when the pins function as input pin of the alternate function, and when TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, and TOP21 pins (output pins of the alternate function) go into a high-impedance state).	TOQ0T1/TIQ01/TOQ01
P11	71	89	17			TOQ0B1/TIQ02/TOQ02
P12	72	90	18			TOQ0T2/TIQ03/TOQ03
P13	75	93	21			TOQ0B2/TIQ00
P14	76	94	22			TOQ0T3/EVTQ0
P15	77	95	23			TOQ0B3/TRGQ0
P16	78	96	24			TOQ00/TIP20
P17	79	97	25			TOP21/TIP21
P20Note	–	35	63	I/O	Port 2 (V850E/IA4 only) 8-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when TOQ1T1 to TOQ1T3, TOQ1B1 to TOQ1B3, and TOP31 pins (output pins of the alternate function) go into a high-impedance state).	TOQ1T1Note
P21Note	–	36	64			TOQ1B1Note
P22Note	–	37	65			TOQ1T2Note
P23Note	–	40	68			TOQ1B2Note
P24Note	–	41	69			TOQ1T3Note
P25Note	–	42	70			TOQ1B3Note
P26Note	–	60	88			TOQ10Note
P27Note	–	61	89			TOP31Note
P30	41	52	80	I/O	Port 3 8-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when the pins function as input pins of the alternate function (including slave mode of $\overline{\text{SCKB1}}$ pin)).	RXDA0
P31	42	53	81			TXDA0
P32	43	54	82			SIB1/RXDA1
P33	44	55	83			SOB1/TXDA1
P34	45	56	84			$\overline{\text{SCKB1}}$
<R> P35	46	57	85			TIUD10/TO10
P36	47	58	86			TCUD10
P37	48	59	87			TCLR10

Note V850E/IA4 only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(2/3)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
P40	27	32	60	I/O	Port 4 5-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when the pins function as input pins of the alternate function (including slave mode of $\overline{\text{SCKB0}}$ pin)).	SIB0
P41	28	33	61			SOB0
P42	29	34	62			$\overline{\text{SCKB0}}$
P43	37	48	76			TOP00/TIP00
P44	38	49	77			TOP01/TIP01
P50[Note 1]	–	82	10	I/O	Port 5 (V850E/IA4 only) 3-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected when the pins are in the port mode and input mode, and when the pins function as input pins of the alternate function).	DDI[Note 2]/TIUD11[Note 1]/TO11[Note 1]
P51[Note 1]	–	83	11			DCK[Note 2]/TCUD11[Note 1]
P52[Note 1]	–	84	12			DMS[Note 2]/TCLR11[Note 1]
P70	3	5	33	Input	Port 7 V850E/IA3: 6-bit input-only port V850E/IA4: 8-bit input-only port	ANI20
P71	4	6	34			ANI21
P72	5	7	35			ANI22
P73	6	8	36			ANI23
P74	12	14	42			ANI24
P75	13	15	43			ANI25
P76[Note 1]	–	16	44			ANI26[Note 1]
P77[Note 1]	–	17	45			ANI27[Note 1]

<R>

Notes 1. V850E/IA4 only

2. μPD70F3186 (V850E/IA4) only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(3/3)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
PDL0	49	62	90	I/O	Port DL 16-bit I/O port Input data read/output data write is enabled in 1-bit units. An on-chip pull-up resistor can be specified in 1-bit units (the on-chip pull-up resistor can be connected only when the pins are in the port mode and input mode).	–
PDL1	50	63	91			–
PDL2	53	66	94			–
PDL3	54	67	95			–
PDL4	55	68	96			–
PDL5	56	69	97			FLMD1[Note]
PDL6	57	70	98			–
PDL7	58	71	99			–
PDL8	59	74	2			–
PDL9	60	75	3			–
PDL10	61	76	4			–
PDL11	62	77	5			–
PDL12	63	78	6			–
PDL13	64	79	7			–
PDL14	65	80	8			–
PDL15	66	81	9			–

Note μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(2) Non-port pins

(1/4)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
ADTRG0	21	26	54	Input	External trigger input for A/D converters 0, 1	INTP4/P04
ADTRG1	22	27	55			INTP5/P05
ANI00	1	1	29	Input	Analog input for A/D converters 0, 1	–
ANI01	2	2	30			–
ANI02Note 1	–	3	31			–
ANI03Note 1	–	4	32			–
ANI10	14	18	46			–
ANI11	15	19	47			–
ANI12	16	20	48			–
ANI13	17	21	49			–
ANI20	3	5	33	Input	Analog input for A/D converter 2	P70
ANI21	4	6	34			P71
ANI22	5	7	35			P72
ANI23	6	8	36			P73
ANI24	12	14	42			P74
ANI25	13	15	43			P75
ANI26Note 1	–	16	44			P76Note 1
ANI27Note 1	–	17	45			P77Note 1
AV_{DD}	Note 2	Note 2	Note 2	–	Positive power supply for A/D converters 0 to 2 (5 V power supply pin)Note 4	–
AV_{SS}	Note 3	Note 3	Note 3	–	Ground potential for A/D converters 0 to 2Note 4	–
CMPREF	9	11	39	–	Comparator reference power supply for A/D converters 0, 1	–
CV_{DD}	32	43	71	–	Power supply for oscillator and PLL (2.5 V power supply pin)	–
CV_{SS}	35	46	74	–	Ground potential for oscillator and PLL	–

Notes 1. V850E/IA4 only

2. GC (V850E/IA3): 7, 11
GC (V850E/IA4): 9, 13
GF (V850E/IA4): 37, 41

3. GC (V850E/IA3): 8, 10
GC (V850E/IA4): 10, 12
GF (V850E/IA4): 38, 40

4. For details on the power supply connection specifications, see **12.2 Configuration**.

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(2/4)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
DCK[Note 1]	–	83	11	Input	Debug clock input for on-chip debug emulator	TCUD11[Note 2]/P51[Note 2]
DDI[Note 1]	–	82	10	Input	Debug data input for on-chip debug emulator	TIUD11[Note 2]/TO11[Note 2]/P50[Note 2]
DDO[Note 1]	–	98	26	Output	Debug data output for on-chip debug emulator	–
DMS[Note 1]	–	84	12	Input	Debug mode select for on-chip debug emulator	TCLR11[Note 2]/P52[Note 2]
$\overline{\text{DRST}}$[Note 1]	–	99	27	Input	Debug reset input for on-chip debug emulator	–
EV_{DD}	Note 3	Note 3	Note 3	–	Positive power supply for external pin (5 V power supply pin)	–
EV_{SS}	Note 4	Note 4	Note 4	–	Ground potential for external pin	–
EVTQ0	76	94	22	Input	External event count input of TMQ0	TOQ0T3/P14
FLMD0	69	87	15	Input	Pin for setting flash memory programming mode (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only)	–
FLMD1	56	69	97	Input		PDL5
IC1	69	87	15	–	Internal connection pin (μPD703183 (V850E/IA3), μPD703185 (V850E/IA4), μPD703186 (V850E/IA4) only)	–
IC2	–	98	26	–	Internal connection pin (μPD703185 (V850E/IA4), μPD703186 (V850E/IA4) only)	–
IC3	–	99	27	–		–
INTP0	18	22	50	Input	External maskable interrupt request input	TOQ0OFF/P00
INTP1[Note 2]	–	23	51			TOQ1OFF[Note 2]/P01[Note 2]
INTP2	19	24	52			TOP2OFF/P02
INTP3	20	25	53			TOP3OFF[Note 2]/P03
INTP4	21	26	54			ADTRG0/P04
INTP5	22	27	55			ADTRG1/P05
INTP6	23	28	56			P06
INTP7	24	29	57			P07

Notes 1. μPD70F3186 (V850E/IA4) only

2. V850E/IA4 only

3. GC (V850E/IA3): 30, 74
GC (V850E/IA4): 38, 73, 92
GF (V850E/IA4): 1, 20, 66

4. GC (V850E/IA3): 31, 73
GC (V850E/IA4): 39, 72, 91
GF (V850E/IA4): 19, 67, 100

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(3/4)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
PLLSIN	80	100	28	Input	Output frequency select signal input in PLL mode	–
$\overline{\text{RESET}}$	36	47	75	Input	System reset input	–
RXDA0	41	52	80	Input	Serial receive data input of UARTA0, UARTA1	P30
RXDA1	43	54	82			SIB1/P32
$\overline{\text{SCKB0}}$	29	34	62	I/O	Serial clock I/O of CSIB0, CSIB1	P42
$\overline{\text{SCKB1}}$	45	56	84			P34
SIB0	27	32	60	Input	Serial receive data input of CSIB0, CSIB1	P40
SIB1	43	54	82			RXDA1/P32
SOB0	28	33	61	Output	Serial transmit data output of CSIB0, CSIB1	P41
SOB1	44	55	83			TXDA1/P33
TCLR10	48	59	87	Input	Clear signal input to TMENC10, TMENC11	P37
TCLR11[Note 1]	–	84	12			DMS[Note 2]/P52[Note 1]
TCUD10	47	58	86	Input	Count operation switching signal for TMENC10, TMENC11	P36
TCUD11[Note 1]	–	83	11			DCK[Note 2]/P51[Note 1]
TIP00	37	48	76	Input	External event count input/external trigger input/capture trigger input of TMP0	TOP00/P43
TIP01	38	49	77		Capture trigger input of TMP0	TOP01/P44
TIP20	78	96	24		External event count input/external trigger input/capture trigger input of TMP2	TOQ00/P16
TIP21	79	97	25		Capture trigger input of TMP2	TOP21/P17
TIQ00	75	93	21	Input	Capture trigger input of TMQ0	TOQ0B2/P13
TIQ01	70	88	16			TOQ01/TOQ0T1/P10
TIQ02	71	89	17			TOQ02/TOQ0B1/P11
TIQ03	72	90	18			TOQ03/TOQ0T2/P12
TIUD10	46	57	85	Input	External count clock input of TMENC10, TMENC11	TO10/P35
TIUD11[Note 1]	–	82	10			TO11[Note 1]/DDI[Note 2]/P50[Note 1]
TO10	46	57	85	Output	Pulse signal output of TMENC10, TMENC11	TIUD10/P35
TO11[Note 1]	–	82	10			DDI[Note 2]/TIUD11[Note 1]/P50[Note 1]
TOP00	37	48	76	Output	Pulse signal output of TMP0, TMP2	TIP00/P43
TOP01	38	49	77			TIP01/P44
TOP21	79	97	25			TIP21/P17
TOP2OFF	19	24	52	Input	High-impedance output control signal input	INTP2/P02
TOP31[Note 1]	–	61	89	Output	Pulse signal output of TMP3	P27[Note 1]
TOP3OFF[Note 1]	–	25	53	Input	High-impedance output control signal input	INTP3/P03

Notes 1. V850E/IA4 only

2. μPD70F3186 (V850E/IA4) only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(4/4)

Pin Name	Pin No.			I/O	Function	Alternate Function
	IA3	IA4				
	GC	GC	GF			
TOQ00	78	96	24	Output	Pulse signal output of TMQ0	TIP20/P16
TOQ01	70	88	16			TOQ0T1/TIQ01/P10
TOQ02	71	89	17			TOQ0B1/TIQ02/P11
TOQ03	72	90	18			TOQ0T2/TIQ03/P12
TOQ0B1	71	89	17	Output	Pulse signal output for 6-phase PWM0	TIQ02/TOQ02/P11
TOQ0B2	75	93	21			TIQ00/P13
TOQ0B3	77	95	23			TRGQ0/P15
TOQ0OFF	18	22	50	Input	High-impedance output control signal input	INTP0/P00
TOQ0T1	70	88	16	Output	Pulse signal output for 6-phase PWM0	TIQ01/TOQ01/P10
TOQ0T2	72	90	18			TIQ03/TOQ03/P12
TOQ0T3	76	94	22			EVTQ0/P14
TOQ10Note 1	–	60	88	Output	Pulse signal output of TMQ1	P26Note 1
TOQ1B1Note 1	–	36	64	Output	Pulse signal output for 6-phase PWM1	P21Note 1
TOQ1B2Note 1	–	40	68			P23Note 1
TOQ1B3Note 1	–	42	70			P25Note 1
TOQ1OFFNote 1	–	23	51	Input	High-impedance output control signal input	INTP1Note 1/P01Note 1
TOQ1T1Note 1	–	35	63	Output	Pulse signal output for 6-phase PWM1	P20Note 1
TOQ1T2Note 1	–	37	65			P22Note 1
TOQ1T3Note 1	–	41	69			P24Note 1
TRGQ0	77	95	23	Input	External trigger input of TMQ0	TOQ0B3/P15
TXDA0	42	53	81	Output	Serial transmit data output of UARTA0, UARTA1	P31
TXDA1	44	55	83			SOB1/P33
V_{DD}	Note 2	Note 2	Note 2	–	Positive power supply for internal unit (2.5 V power supply pin)	–
V_{SS}	Note 3	Note 3	Note 3	–	Ground potential for internal unit	–
X1	34	45	73	Input	Oscillator connection pin for system clock.	–
X2	33	44	72	–		–

Notes 1. V850E/IA4 only

2. GC (V850E/IA3): 25, 40, 51, 68
GC (V850E/IA4): 30, 50, 64, 86
GF (V850E/IA4): 14, 58, 78, 92

3. GC (V850E/IA3): 26, 39, 52, 67
GC (V850E/IA4): 31, 51, 65, 85
GF (V850E/IA4): 13, 59, 79, 93

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

2.2 Pin I/O Circuits and Recommended Connection of Unused Pins

It is recommended that 1 to 10 kΩ resistors be used when connecting to EV_{DD} or EV_{SS} via resistors.

(1/3)

Pin Name	Alternate-Function Pin Name	Pin No.			I/O Circuit Type	Recommended Connection
		IA3	IA4			
		GC	GC	GF		
P00	INTP0/TOQ0OFF	18	22	50	5-AH	Input: Independently connect to EV_{DD} or EV_{SS} via a resistor. Output: Leave open.
P01[Note]	INTP1[Note]/TOQ1OFF[Note]	–	23	51		
P02	INTP2/TOP2OFF	19	24	52		
P03	INTP3/TOP3OFF[Note]	20	25	53		
P04	INTP4/ADTRG0	21	26	54		
P05	INTP5/ADTRG1	22	27	55		
P06	INTP6	23	28	56		
P07	INTP7	24	29	57		
P10	TOQ0T1/TIQ01/TOQ01	70	88	16		
P11	TOQ0B1/TIQ02/TOQ02	71	89	17		
P12	TOQ0T2/TIQ03/TOQ03	72	90	18		
P13	TOQ0B2/TIQ00	75	93	21		
P14	TOQ0T3/EVTQ0	76	94	22		
P15	TOQ0B3/TRGQ0	77	95	23		
P16	TOQ00/TIP20	78	96	24		
P17	TOP21/TIP21	79	97	25		
P20[Note]	TOQ1T1[Note]	–	35	63	5-AG	
P21[Note]	TOQ1B1[Note]	–	36	64		
P22[Note]	TOQ1T2[Note]	–	37	65		
P23[Note]	TOQ1B2[Note]	–	40	68		
P24[Note]	TOQ1T3[Note]	–	41	69		
P25[Note]	TOQ1B3[Note]	–	42	70		
P26[Note]	TOQ10[Note]	–	60	88		
P27[Note]	TOP31[Note]	–	61	89		
P30	RXDA0	41	52	80	5-AH	
P31	TXDA0	42	53	81	5-AG	
P32	SIB1/RXDA1	43	54	82	5-AH	
P33	SOB1/TXDA1	44	55	83	5-AG	
P34	$\overline{\text{SCKB1}}$	45	56	84	5-AH	
P35	TIUD10/TO10	46	57	85		
P36	TCUD10	47	58	86		
P37	TCLR10	48	59	87		

Note V850E/IA4 only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(2/3)

Pin Name	Alternate-Function Pin Name	Pin No.			I/O Circuit Type	Recommended Connection
		IA3	IA4			
		GC	GC	GF		
P40	SIB0	27	32	60	5-AH	Input: Independently connect to EV_{DD} or EV_{SS} via a resistor. Output: Leave open.
P41	SOB0	28	33	61	5-AG	
P42	$\overline{\text{SCKB0}}$	29	34	62	5-AH	
P43	TOP00/TIP00	37	48	76		
P44	TOP01/TIP01	38	49	77		
P50[Note 1]	DDI[Note 2]/TIUD11[Note 1]/TO11[Note 1]	–	82	10		
P51[Note 1]	DCK[Note 2]/TCUD11[Note 1]	–	83	11		
P52[Note 1]	DMS[Note 2]/TCLR11[Note 1]	–	84	12		
P70	ANI20	3	5	33	9	Connect to AV_{DD} or AV_{SS}.
P71	ANI21	4	6	34		
P72	ANI22	5	7	35		
P73	ANI23	6	8	36		
P74	ANI24	12	14	42		
P75	ANI25	13	15	43		
P76[Note 1]	ANI26[Note 1]	–	16	44		
P77[Note 1]	ANI27[Note 1]	–	17	45		
PDL0	–	49	62	90	5-AG	Input: Independently connect to EV_{DD} or EV_{SS} via a resistor. Output: Leave open.
PDL1	–	50	63	91		
PDL2	–	53	66	94		
PDL3	–	54	67	95		
PDL4	–	55	68	96		
PDL5	FLMD1[Note 3]	56	69	97		
PDL6	–	57	70	98		
PDL7	–	58	71	99		
PDL8	–	59	74	2		
PDL9	–	60	75	3		
PDL10	–	61	76	4		
PDL11	–	62	77	5		
PDL12	–	63	78	6		
PDL13	–	64	79	7		
PDL14	–	65	80	8		
PDL15	–	66	81	9		

Notes 1. V850E/IA4 only

2. μPD70F3186 (V850E/IA4) only

3. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(3/3)

Pin Name	Alternate-Function Pin Name	Pin No.			I/O Circuit Type	Recommended Connection
		IA3	IA4			
		GC	GC	GF		
ANI00	–	1	1	29	7-C	Connect to AV_{DD} or AV_{SS}.
ANI01	–	2	2	30		
ANI02[Note 1]	–	–	3	31		
ANI03[Note 1]	–	–	4	32	7	
ANI10	–	14	18	46	7-C	
ANI11	–	15	19	47		
ANI12	–	16	20	48		
ANI13	–	17	21	49	7	
DDO[Note 2]	–	–	98	26	3-C	Leave open (always level output during reset).
$\overline{DRST}$[Note 2]	–	–	99	27	2-M	Leave open (on-chip pull-down resistor).
$\overline{RESET}$	–	36	47	75	2	–
FLMD0[Note 3]/ IC1[Note 4]	–	69	87	15		–
PLLSIN	–	80	100	28		–
CMPREF	–	9	11	39	–	Connect to AV_{SS}. Set the OPnCTL1 register to 00H (disable operation).

Notes 1. V850E/IA4 only

2. μPD70F3186 (V850E/IA4) only

3. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

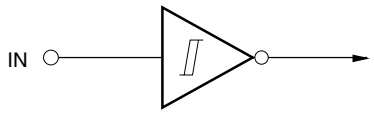
4. μPD703183 (V850E/IA3), μPD703185 (V850E/IA4), μPD703186 (V850E/IA4) only

Remarks 1. IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

2. n = 0, 1

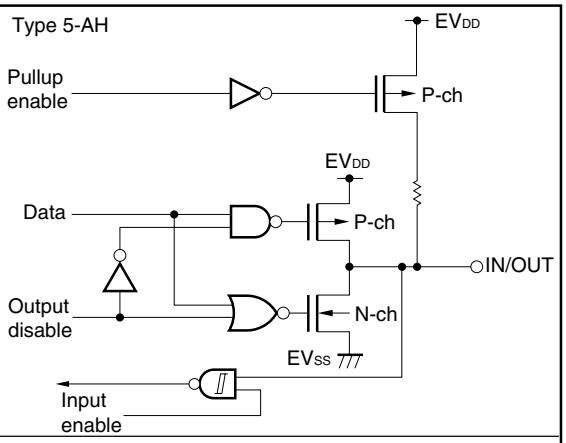
2.3 Pin I/O Circuits

Type 2

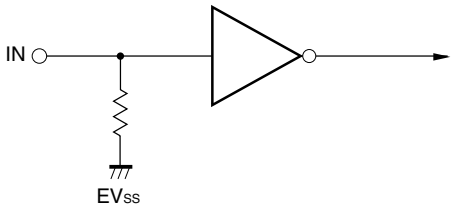


Schmitt-triggered input with hysteresis characteristics

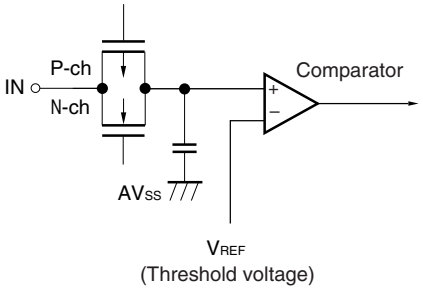
Type 5-AH



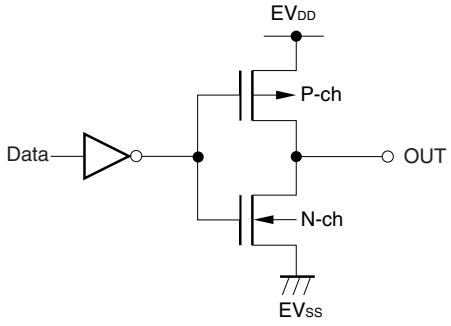
Type 2-M



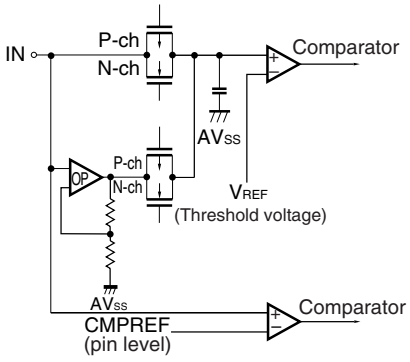
Type 7



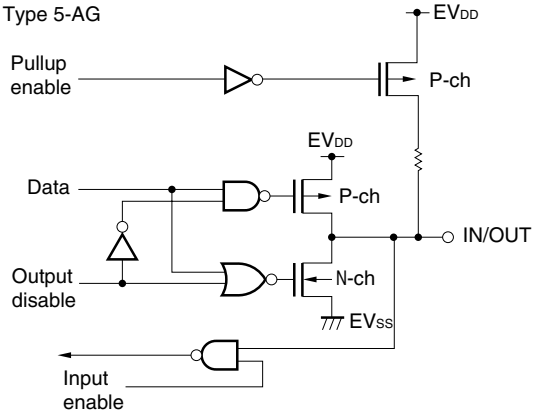
Type 3-C



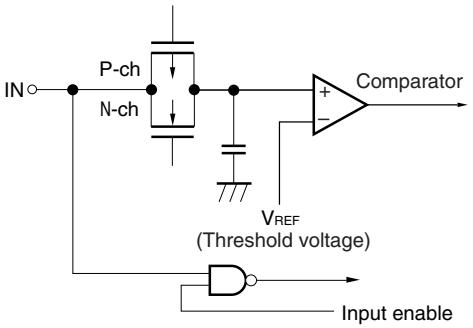
Type 7-C



Type 5-AG



Type 9



CHAPTER 3 CPU FUNCTION

The CPU of the V850E/IA3 and V850E/IA4 are based on RISC architecture and executes almost all the instructions in one clock cycle using 5-stage pipeline control.

3.1 Features

- Minimum instruction execution time: 15.6 ns (@ 64 MHz internal operation)
- Thirty-two 32-bit general-purpose registers
- Internal 32-bit architecture
- Five-stage pipeline control
- Multiply/divide instructions
- Saturated operation instructions
- One-clock 32-bit shift instruction
- Load/store instruction with long/short instruction format
- Four types of bit manipulation instructions
  - SET1
  - CLR1
  - NOT1
  - TST1

3.2 CPU Register Set

The registers of the V850E/IA3 and V850E/IA4 can be classified into two categories: a general-purpose program register set and a dedicated system register set. All the registers have a 32-bit width.

For details, refer to **V850E1 Architecture User's Manual**.

Figure 3-1. CPU Register Set

(1) Program register set

31	0
r0	(Zero register)
r1	(Assembler-reserved register)
r2	
r3	(Stack pointer (SP))
r4	(Global pointer (GP))
r5	(Text pointer (TP))
r6	
r7	
r8	
r9	
r10	
r11	
r12	
r13	
r14	
r15	
r16	
r17	
r18	
r19	
r20	
r21	
r22	
r23	
r24	
r25	
r26	
r27	
r28	
r29	
r30	(Element pointer (EP))
r31	(Link pointer (LP))

31	0
PC	(Program counter)

(2) System register set

31	0
EIPC	(Status saving register during interrupt)
EIPSW	(Status saving register during interrupt)
FEPC	(Status saving register during NMI)
FEPSW	(Status saving register during NMI)
ECR	(Interrupt source register)
PSW	(Program status word)
CTPC	(Status saving register during CALLT execution)
CTPSW	(Status saving register during CALLT execution)
DBPC	(Status saving register during exception/debug trap)
DBPSW	(Status saving register during exception/debug trap)
CTBP	(CALLT base pointer)

3.2.1 Program register set

The program register set includes general-purpose registers and a program counter.

(1) General-purpose registers (r0 to r31)

Thirty-two general-purpose registers, r0 to r31, are available. Any of these registers can be used as a data variable or address variable.

However, r0 and r30 are implicitly used by instructions, and care must be exercised when using these registers. r0 is a register that always holds 0, and is used for operations using 0 and offset 0 addressing. r30 is used, by means of the SLD and SST instructions, as a base pointer for when memory is accessed. Also, r1, r3 to r5, and r31 are implicitly used by the assembler and C compiler. Therefore, before using these registers, their contents must be saved so that they are not lost. The contents must be restored to the registers after the registers have been used. r2 may be used by the real-time OS. If the real-time OS does not use r2, it can be used as a variable register.

Table 3-1. General-Purpose Registers

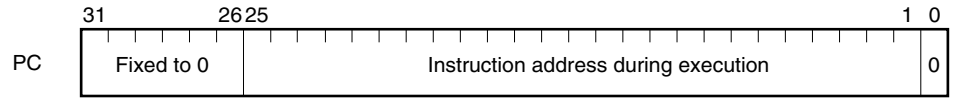
Name	Usage	Operation
r0	Zero register	Always holds 0
r1	Assembler-reserved register	Working register for generating 32-bit immediate data
r2	Address/data variable register (when r2 is not used by the real-time OS)	
r3	Stack pointer	Used to generate stack frame when function is called
r4	Global pointer	Used to access global variable in data area
r5	Text pointer	Register to indicate the start of the text area (where program code is located)
r6 to r29	Address/data variable registers	
r30	Element pointer	Base pointer when memory is accessed
r31	Link pointer	Used by compiler when calling function

(2) Program counter (PC)

This register holds the instruction address during program execution. The lower 26 bits of this register are valid, and bits 31 to 26 are fixed to 0. If a carry occurs from bit 25 to 26, it is ignored.

Bit 0 is fixed to 0, and branching to an odd address cannot be performed.

After reset: 00000000H



3.2.2 System register set

System registers control the status of the CPU and hold interrupt information.

To read/write these system registers, specify a system register number indicated below using the system register load/store instruction (LDSR or STSR instruction).

Table 3-2. System Register Numbers

System Register No.	System Register Name	Operand Specification	
		LDSR Instruction	STSR Instruction
0	Status saving register during interrupt (EIPC)[Note 1]	√	√
1	Status saving register during interrupt (EIPSW)[Note 1]	√	√
2	Status saving register during NMI (FEPC)	√	√
3	Status saving register during NMI (FEPSW)	√	√
4	Interrupt source register (ECR)	×	√
5	Program status word (PSW)	√	√
6 to 15	Reserved for future function expansion (operations that access these register numbers cannot be guaranteed).	×	×
16	Status saving register during CALLT execution (CTPC)	√	√
17	Status saving register during CALLT execution (CTPSW)	√	√
18	Status saving register during exception/debug trap (DBPC)	√[Note 2]	√[Note 2]
19	Status saving register during exception/debug trap (DBPSW)	√[Note 2]	√[Note 2]
20	CALLT base pointer (CTBP)	√	√
21 to 31	Reserved for future function expansion (operations that access these register numbers cannot be guaranteed).	×	×

Notes 1. Because this register has only one set, to enable multiple interrupts, it is necessary to save this register by program.

<R> **2.** Can be accessed only after the DBTRAP instruction or illegal opcode is executed and before the DBRET instruction is executed.

Caution Even if bit 0 of EIPC, FEPC, or CTPC is set to 1 by the LDSR instruction, bit 0 will be ignored when the program is returned by the RETI instruction after interrupt servicing (because bit 0 of the PC is fixed to 0). When setting the value of EIPC, FEPC, and CTPC, use an even value (bit 0 = 0).

Remark √: Access allowed
×: Access prohibited

(1) Interrupt status saving registers (EIPC, EIPSW)

There are two interrupt status saving registers, EIPC and EIPSW.

Upon occurrence of a software exception or a maskable interrupt, the contents of the program counter (PC) are saved to EIPC and the contents of the program status word (PSW) are saved to EIPSW (upon occurrence of a non-maskable interrupt (NMI), the contents are saved to the NMI status saving registers (FEPC, FEPSW)).

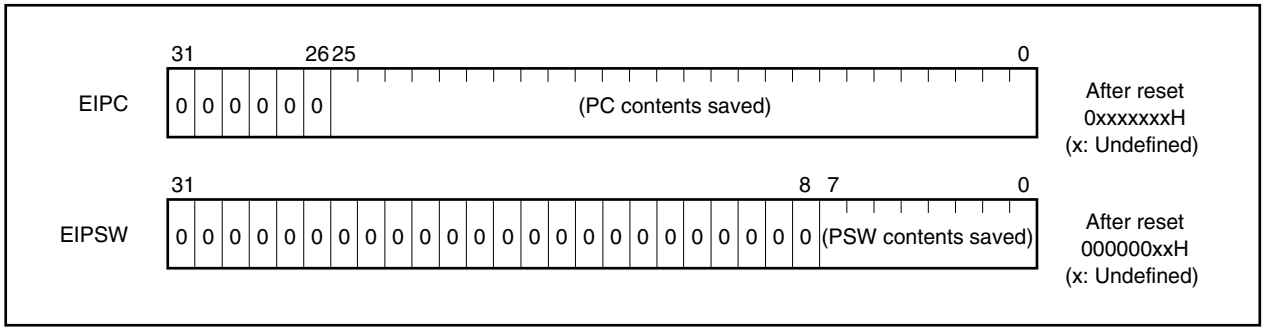
The address of the next instruction following the instruction executed when a software exception or maskable interrupt occurs is saved to EIPC, except for some instructions (see **17.9 Periods in Which CPU Does Not Acknowledge Interrupts**).

The current PSW contents are saved to EIPSW.

Since there is only one set of interrupt status saving registers, the contents of these registers must be saved by the program when multiple interrupt servicing is enabled.

Bits 31 to 26 of EIPC and bits 31 to 8 of EIPSW are reserved (fixed to 0) for future function expansion.

When the RETI instruction is executed, the values in EIPC and EIPSW are restored to the PC and PSW, respectively.



(2) NMI status saving registers (FEPC, FEPSW)

There are two NMI status saving registers, FEPC and FEPSW.

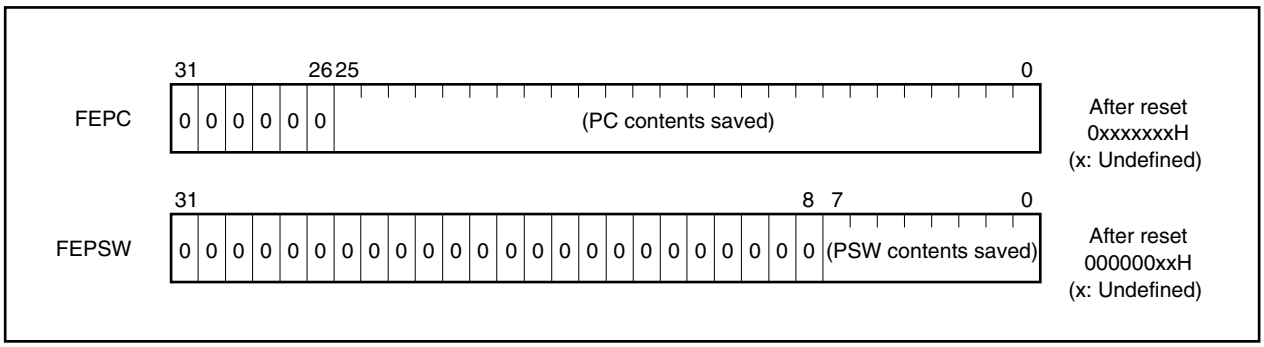
Upon occurrence of a non-maskable interrupt (NMI), the contents of the program counter (PC) are saved to FEPC and the contents of the program status word (PSW) are saved to FEPSW.

The address of the next instruction following the instruction executed when a non-maskable interrupt occurs is saved to FEPC, except for some instructions.

The current PSW contents are saved to FEPSW.

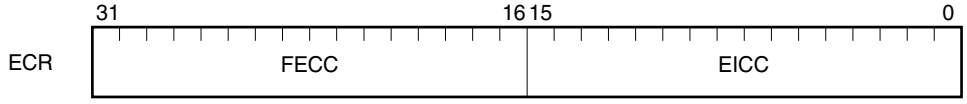
Bits 31 to 26 of FEPC and bits 31 to 8 of FEPSW are reserved (fixed to 0) for future function expansion.

When the RETI instruction has been executed, the values of FEPC and FEPSW are restored to the PC and PSW, respectively.



(3) Interrupt source register (ECR)

Upon occurrence of an interrupt or an exception, the interrupt source register (ECR) holds the source of an interrupt or an exception. The value held by ECR is the exception code coded for each interrupt source. This register is a read-only register, and thus data cannot be written to it using the LDSR instruction.



Bit position	Bit name	Description
31 to 16	FECC	Non-maskable interrupt (NMI) exception code
15 to 0	EICC	Exception, maskable interrupt exception code

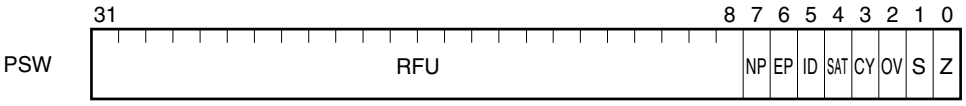
(4) Program status word (PSW)

The program status word (PSW) is a collection of flags that indicate the program status (instruction execution result) and the CPU status.

When the contents of this register are changed using the LDSR instruction, the new contents become valid immediately following completion of LDSR instruction execution. Interrupt request acknowledgment is held pending while a write to the PSW is being executed by the LDSR instruction.

Bits 31 to 8 are reserved (fixed to 0) for future function expansion.

(1/2)



Bit position	Flag name	Description
31 to 8	RFU	Reserved field. Fixed to 0.
7	NP	Indicates that non-maskable interrupt (NMI) servicing is in progress. This flag is set to 1 when an NMI request is acknowledged, and disables multiple interrupts. 0: NMI servicing not in progress 1: NMI servicing in progress
6	EP	Indicates that exception processing is in progress. This flag is set to 1 when an exception occurs. Moreover, interrupt requests can be acknowledged even when this bit is set. 0: Exception processing not in progress 1: Exception processing in progress
5	ID	Indicates whether maskable interrupt request acknowledgment is enabled. 0: Interrupt enabled (EI) 1: Interrupt disabled (DI)
4	SAT[Note]	Indicates that the result of executing a saturated operation instruction has overflowed and that the calculation result is saturated. Since this is a cumulative flag, it is set to 1 when the result of a saturated operation instruction becomes saturated, and it is not cleared to 0 even if the operation results of successive instructions do not become saturated. This flag is neither set nor cleared when arithmetic operation instructions are executed. 0: Not saturated 1: Saturated
3	CY	Indicates whether carry or borrow occurred as the result of an operation. 0: No carry or borrow occurred 1: Carry or borrow occurred
2	OV[Note]	Indicates whether overflow occurred during an operation. 0: No overflow occurred 1: Overflow occurred.
1	S[Note]	Indicates whether the result of an operation is negative. 0: Operation result is positive or 0. 1: Operation result is negative.
0	Z	Indicates whether operation result is 0. 0: Operation result is not 0. 1: Operation result is 0.

Remark **Note** is explained on the following page.

(2/2)

Note During saturated operation, the saturated operation results are determined by the contents of the OV flag and S flag. The SAT flag is set (to 1) only when the OV flag is set (to 1) during saturated operation.

Operation result status	Flag status			Saturated operation result
	SAT	OV	S	
Maximum positive value exceeded	1	1	0	7FFFFFFFH
Maximum negative value exceeded	1	1	1	80000000H
Positive (maximum value not exceeded)	Holds value before operation	0	0	Actual operation result
Negative (maximum value not exceeded)			1	

(5) CALLT execution status saving registers (CTPC, CTPSW)

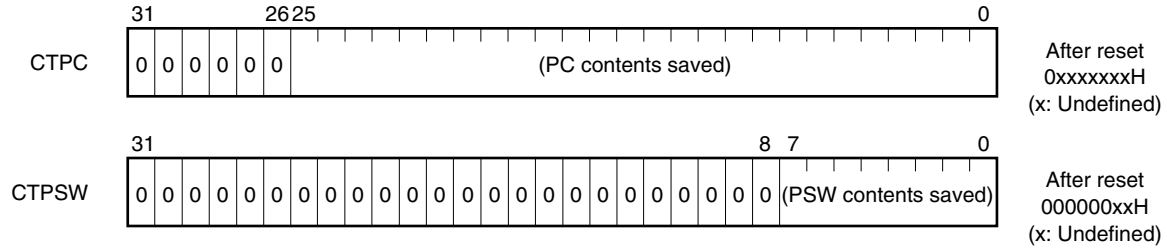
There are two CALLT execution status saving registers, CTPC and CTPSW.

When the CALLT instruction is executed, the contents of the program counter (PC) are saved to CTPC, and the program status word (PSW) contents are saved to CTPSW.

The contents saved to CTPC consist of the address of the next instruction after the CALLT instruction.

The current PSW contents are saved to CTPSW.

Bits 31 to 26 of CTPC and bits 31 to 8 of CTPSW are reserved (fixed to 0) for future function expansion.



(6) Exception/debug trap status saving registers (DBPC, DBPSW)

There are two exception/debug trap status saving registers, DBPC and DBPSW.

Upon occurrence of an exception trap or debug trap, the contents of the program counter (PC) are saved to DBPC, and the program status word (PSW) contents are saved to DBPSW.

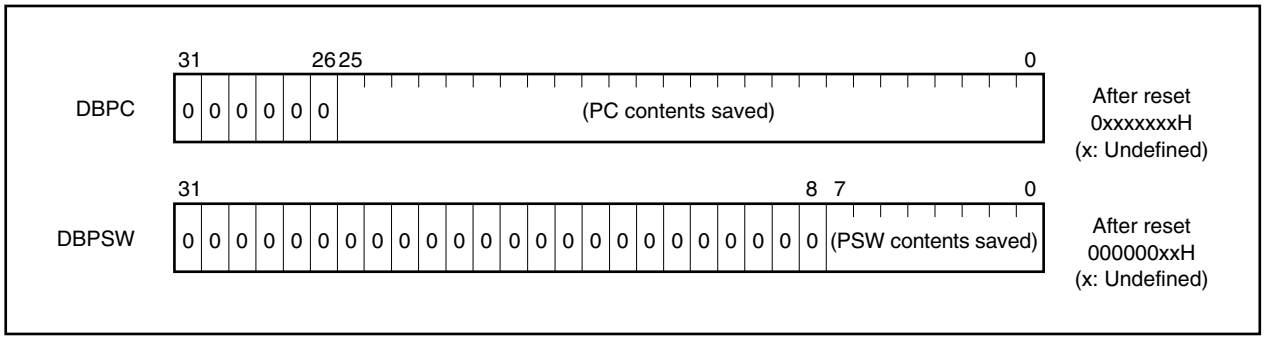
The contents saved to DBPC consist of the address of the next instruction after the instruction executed when an exception trap or debug trap occurs.

The current PSW contents are saved to DBPSW.

<R> These registers can be read or written only in the period between DBTRAP instruction or illegal opcode execution and DBRET instruction execution.

Bits 31 to 26 of DBPC and bits 31 to 8 of DBPSW are reserved (fixed to 0) for future function expansion.

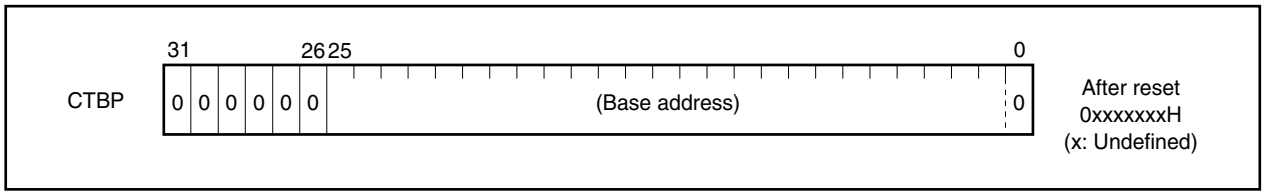
When the DBRET instruction has been executed, the values of DBPC and DBPSW are restored to the PC and PSW, respectively.



(7) CALLT base pointer (CTBP)

The CALLT base pointer (CTBP) is used to specify table addresses and generate target addresses (bit 0 is fixed to 0).

Bits 31 to 26 are reserved (fixed to 0) for future function expansion.



3.3 Operating Modes

3.3.1 Operating modes

The V850E/IA3 and V850E/IA4 have the following operating modes. Mode specification is carried out using the FLMD0 and FLMD1 pins.

(1) Normal operation mode

In this mode, execution branches to the reset entry address in the internal ROM and instruction processing is started when system reset is released.

(2) Flash memory programming mode (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only)

If this mode is specified, a program can be written to the internal flash memory by the flash memory programmer.

3.3.2 Operating mode specification

The operating mode is specified according to the status of the FLMD0 and FLMD1 pins.

FLMD1	FLMD0	Operating Mode	Remarks
×	L	Normal operation mode	Internal ROM area is allocated from address 000000H.
L	H	Flash memory programming mode[Note]	–
Other than above		Setting prohibited	

Note Valid only for the μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4)

Remark L: Low-level input
H: High-level input

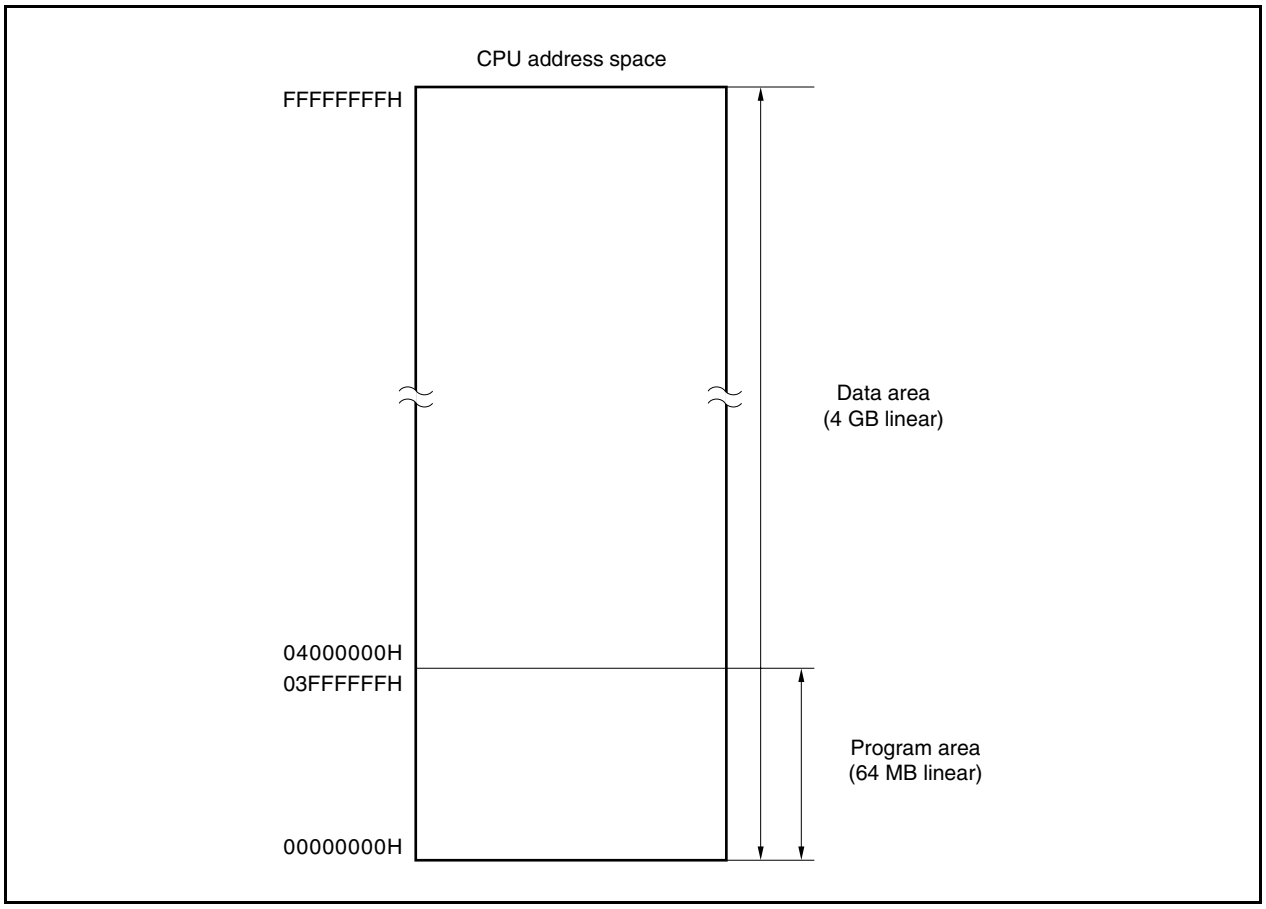
3.4 Address Space

3.4.1 CPU address space

The CPU of the V850E/IA3 and V850E/IA4 has 32-bit architecture and supports up to 4 GB of linear address space (data space) during operand addressing (data access). Also, in instruction address addressing, a maximum of 64 MB of linear address space (program space) is supported.

Figure 3-2 shows the CPU address space.

Figure 3-2. CPU Address Space

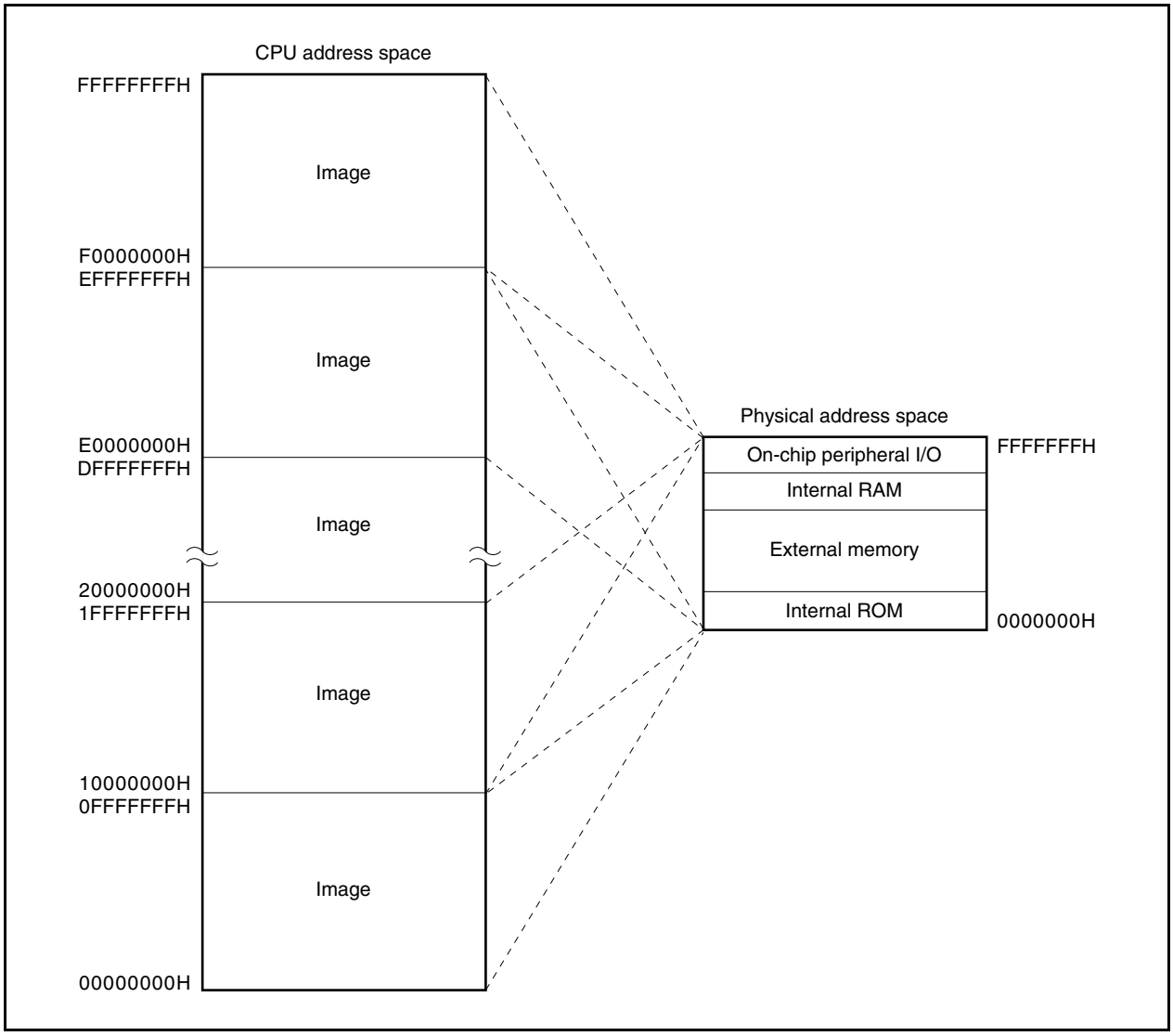


3.4.2 Image

A 256 MB physical address space is seen as 16 images in the 4 GB CPU address space. In actuality, the same 256 MB physical address space is accessed regardless of the values of bits 31 to 28 of the CPU address. Figure 3-3 shows the image of the virtual addressing space.

Physical address x0000000H can be seen as CPU address 00000000H, and in addition, can be seen as address 10000000H, address 20000000H, … , address E0000000H, or address F0000000H.

Figure 3-3. Images on Address Space



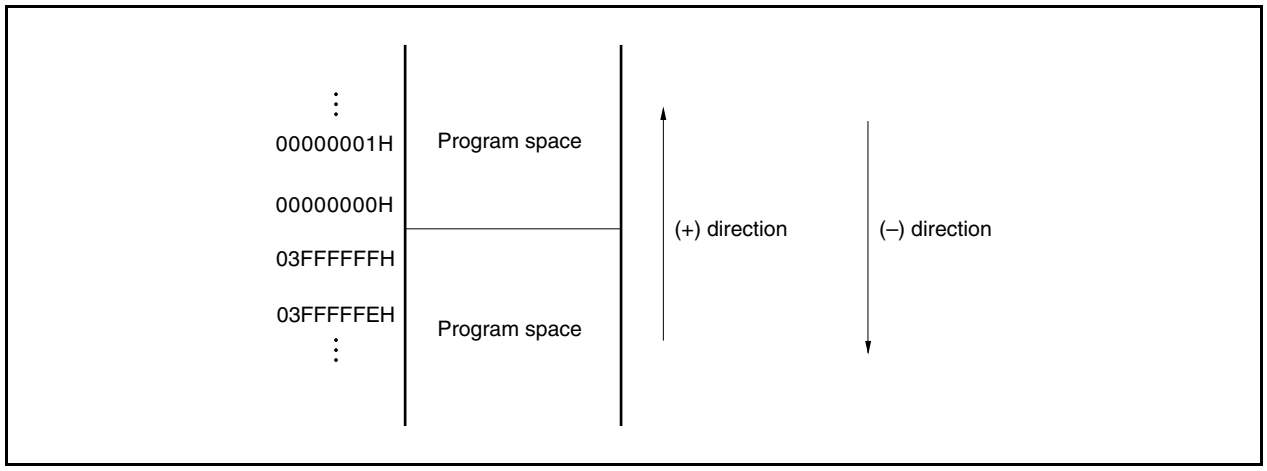
3.4.3 Wraparound of CPU address space

(1) Program space

Of the 32 bits of the PC (program counter), the higher 6 bits are fixed to 0, and only the lower 26 bits are valid. Even if a carry or borrow occurs from bit 25 to 26 as a result of a branch address calculation, the higher 6 bits ignore the carry or borrow.

Therefore, the upper-limit address of the program space, address 03FFFFFFH, and the lower-limit address 00000000H become contiguous addresses. Wraparound refers to a situation like this whereby the lower-limit address and upper-limit address become contiguous.

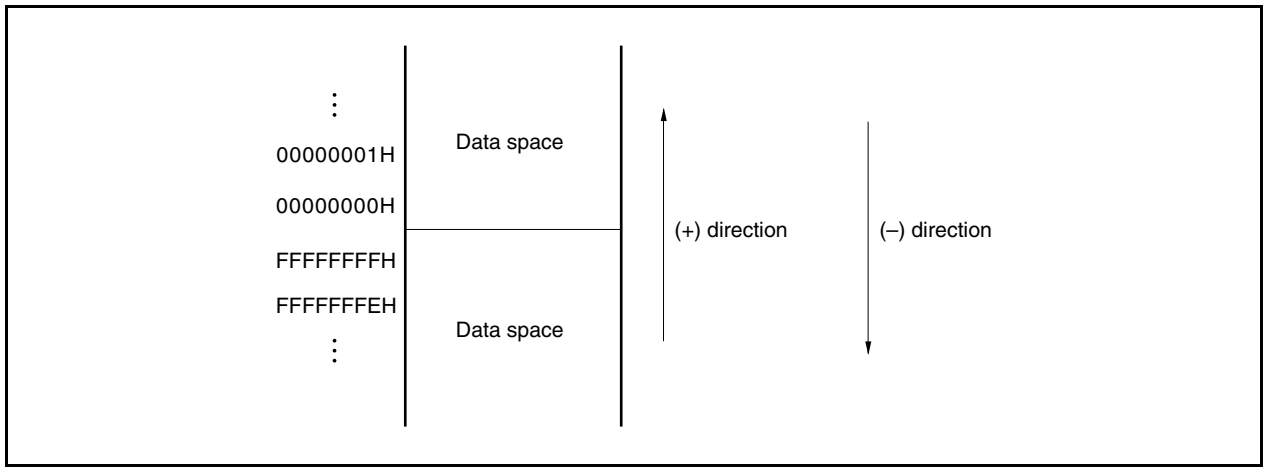
Caution The 4 KB area of 03FFF000H to 03FFFFFFH can be seen as an image of 0FFFF000H to 0FFFFFFFH. This area is access-prohibited. Therefore, do not execute any branch address calculation in which the result will reside in any part of this area.



(2) Data space

The result of an operand address calculation that exceeds 32 bits is ignored.

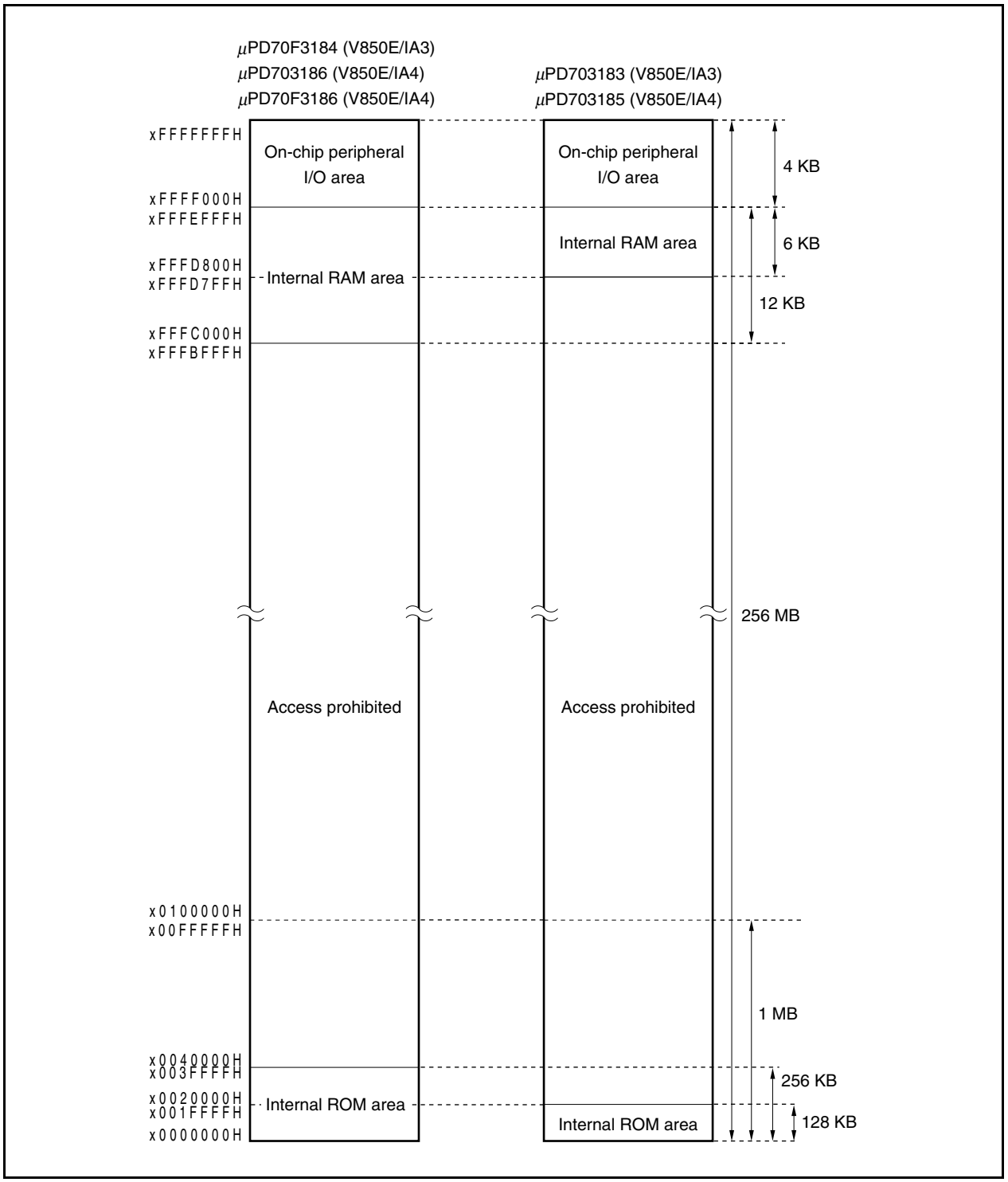
Therefore, the upper-limit address of the program space, address FFFFFFFFH, and the lower-limit address 00000000H are contiguous addresses, and the data space is wrapped around at the boundary of these addresses.



3.4.4 Memory map

The V850E/IA3 and V850E/IA4 reserve areas as shown in Figure 3-4.

Figure 3-4. Memory Map



3.4.5 Area

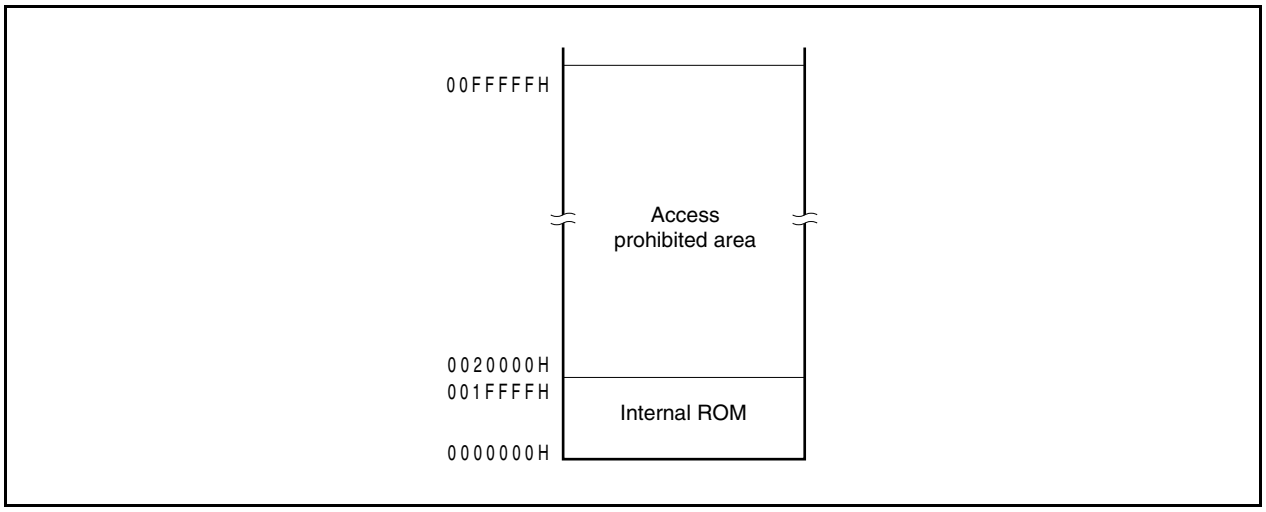
(1) Internal ROM area

1 MB of internal ROM area, addresses 00000H to FFFFFH, is reserved.

(a) μPD703183 (V850E/IA3), μPD703185 (V850E/IA4)

128 KB are provided at addresses 000000H to 01FFFFH as physical internal ROM.

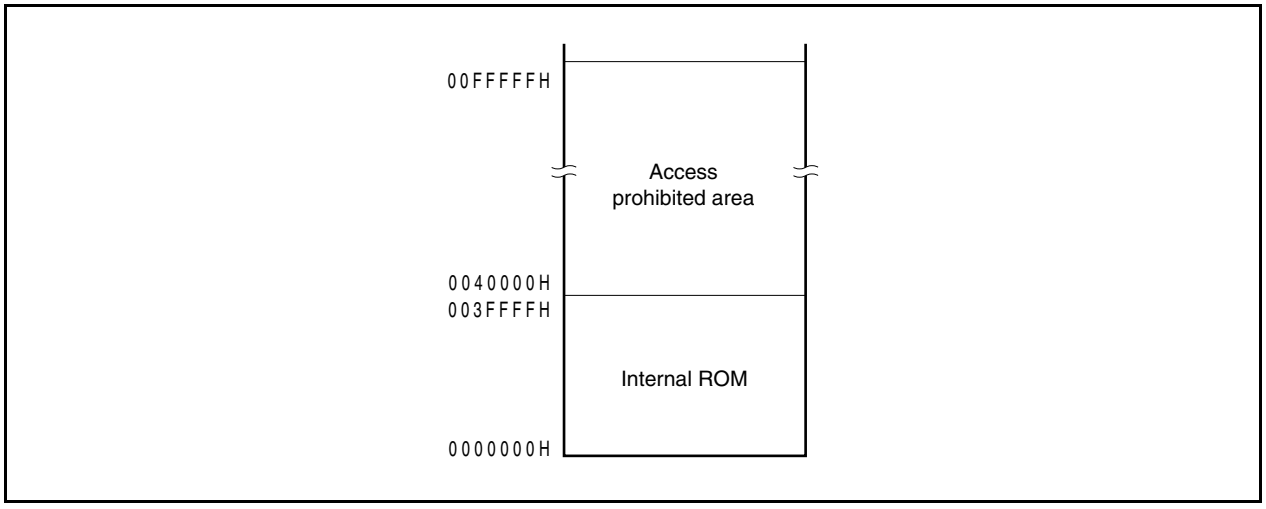
Figure 3-5. Internal ROM Area (128 KB)



(b) μPD70F3184 (V850E/IA3), μPD703186 (V850E/IA4), μPD70F3186 (V850E/IA4)

256 KB are provided at addresses 000000H to 03FFFFH as physical internal ROM.

Figure 3-6. Internal ROM Area (256 KB)



(2) Internal RAM area

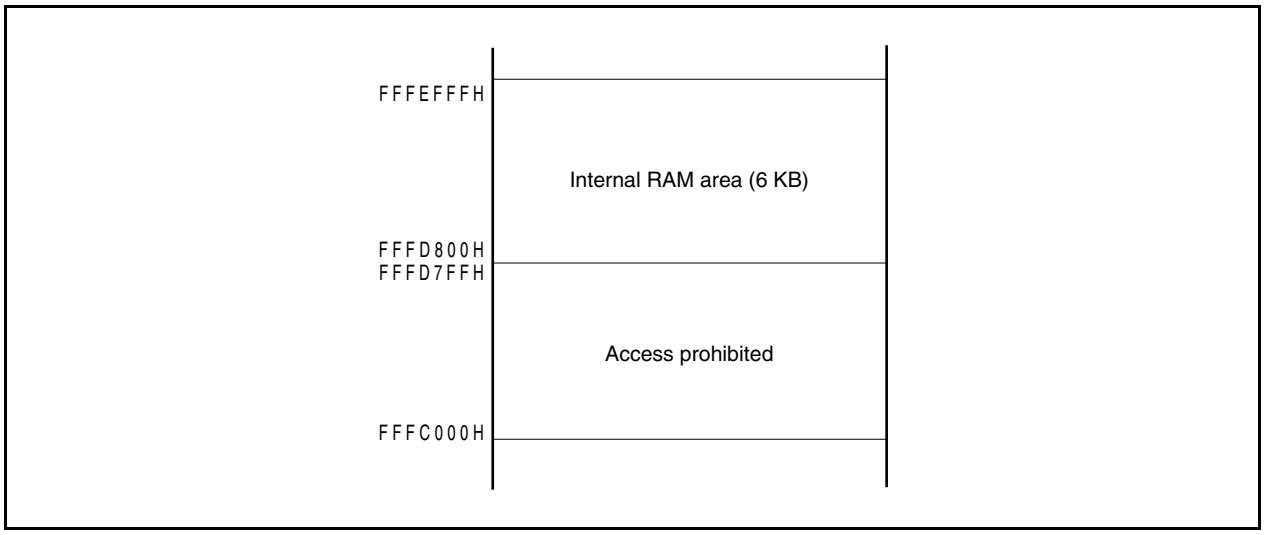
The 12 KB area of addresses FFFC000H to FFFEFFFH is reserved for the internal RAM area.

(a) μPD703183 (V850E/IA3), μPD703185 (V850E/IA4)

The 6 KB area of addresses FFFD800H to FFFEFFFH is provided as physical internal RAM.

Caution The following areas are access-prohibited.
Addresses FFFC000H to FFFD7FFH

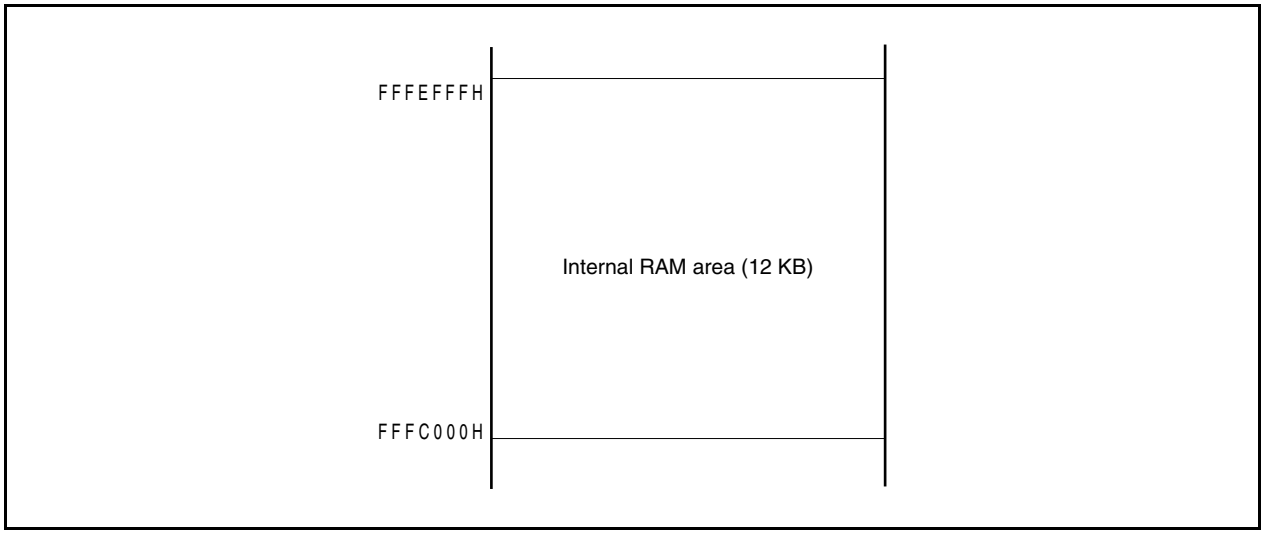
Figure 3-7. Internal RAM Area (6 KB)



(b) μPD70F3184 (V850E/IA3), μPD703186 (V850E/IA4), μPD70F3186 (V850E/IA4)

The 12 KB area of addresses FFFC000H to FFFEFFFH is provided as physical internal RAM.

Figure 3-8. Internal RAM Area (12 KB)



(c) Internal memory size switching register (IMS)

The IMS register is used to make the internal RAM areas of the μPD70F3184 (V850E/IA3)/μPD70F3186 (V850E/IA4) identical to those of the μPD703183 (V850E/IA3)/μPD703185 (V850E/IA4) when the operation of a program for the internal 6 KB RAM of the μPD703183 (V850E/IA3) and μPD703185 (V850E/IA4) is checked by using the internal 12 KB RAM of the μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4).

This register can be read or written in 8-bit units.

Reset sets this register to 00H.

Cautions 1. Write the IMS register before the internal RAM is accessed. This register can be written only once after reset has been released. If the 6 KB internal RAM is selected and if addresses FFFC000H to FFFD7FFH are accessed, this register cannot be written. When it is read, the CPU reads an undefined value.

2. The IMS register is provided only in the flash memory versions; V850E/IA3 (μPD70F3184) and V850E/IA4 (μPD70F3186). The IMS register is not provided in the mask ROM versions; V850E/IA3 (μPD703183) and V850E/IA4 (μPD703185, 703186), and mistakenly writing to the IMS register does not affect the internal RAM size.

3. The sample startup routine supplied with the CA850 includes a code that clears the internal RAM area to 0. Therefore, setting the IMS register is required before the zero-clear routine is executed.

When using the sample startup routine, add instructions <2> to <5> shown in [Description example] below immediately after the __START label in the startup routine.

"0x13" of instruction <2> is the set value of the VSWC register and "0x01" of instruction <4> is the set value of the IMS register. Be sure to set a value to the IMS register in accordance with the internal RAM size to be set (see Caution 2).

[Description example]

```
<1>_ _START:
<2>mov     0x13,      r13
<3>st.b    r13,       VSWC
<4>mov     0x01,      r12
<5>st.b    r12,       IMS
<6>mov     #_tp_TEXT, tp
           :
           :
```

Add

Remark When using a partner manufacturer's tool, make the settings in accordance with the contents of **Cautions 1** to **3.**

Moreover, enter the following to define the IMS register.

```
#define  IMS  (*((volatile unsigned char *)0xfffff9f0))
```

After reset: 00H R/W Address: FFFFF9F0H

	7	6	5	4	3	2	1	0
IMS	0	0	0	0	0	0	0	IRAM0

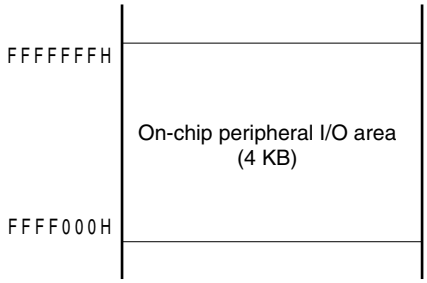
IRAM0	Specification of internal RAM size
0	6 KB (FFFD800H to FFFEFFFH)
1	12 KB (FFFC000H to FFFEFFFH)

(3) On-chip peripheral I/O area

4 KB of memory, addresses FFFF000H to FFFFFFFH, is provided as an on-chip peripheral I/O area.
An image of addresses FFFF000H to FFFFFFFH can be seen at addresses 3FFF000H to 3FFFFFFH[Note].

Note Addresses 3FFF000H to 3FFFFFFH are access-prohibited. To access the on-chip peripheral I/O, specify addresses FFFF000H to FFFFFFFH.

Figure 3-9. On-Chip Peripheral I/O Area



On-chip peripheral I/O registers associated with the operating mode specification and the state monitoring for the on-chip peripheral I/O are all memory-mapped to the on-chip peripheral I/O area. Program fetches cannot be executed from this area.

Cautions 1. In the V850E/IA3 and V850E/IA4, if a register is word accessed, halfword access is performed twice in the order of lower address, then higher address of the word area, disregarding the lower 2 bits of the address.

2. For registers in which byte access is possible, if halfword access is executed, the higher 8 bits become undefined during the read operation, and the lower 8 bits of data are written to the register during the write operation.

3. Addresses that are not defined as registers are reserved for future expansion. If these addresses are accessed, the operation is undefined and not guaranteed.
Addresses 3FFF000H to 3FFFFFFH cannot be specified as the source/destination address of DMA transfer. Be sure to use addresses FFFF000H to FFFFFFFH for the source/destination address of DMA transfer.

3.4.6 Recommended use of address space

The architecture of the V850E/IA3 and V850E/IA4 require that a register that serves as a pointer be secured for address generation in operand data accessing of data space. Operand data access from instruction can be directly executed at the address in this pointer register area ±32 KB. However, because the general-purpose registers that can be used as a pointer register are limited, by minimizing the deterioration of address calculation performance when changing the pointer value, the number of usable general-purpose registers for handling variables is maximized, and the program size can be saved.

(1) Program space

Of the 32 bits of the program counter (PC), the higher 6 bits are fixed to 0, and only the lower 26 bits are valid. Therefore, a contiguous 64 MB space, starting from address 00000000H, unconditionally corresponds to the memory map of the program space.

(2) Data space

With the V850E/IA3 and V850E/IA4, a 256 MB physical address space is seen as 16 images in the 4 GB CPU address space. The highest bit (bit 25) of this 26-bit address is assigned as an address sign-extended to 32 bits.

(a) Application examples using wraparound

When R = r0 (zero register) is specified by the LD/ST disp16 [R] instruction, an addressing range of 00000000H ±32 KB can be referenced by the sign-extended disp16.

The zero register (r0) is a register set to 0 by the hardware, and eliminates the need for additional registers for the pointer.

Example μPD70F3186 (V850E/IA4)

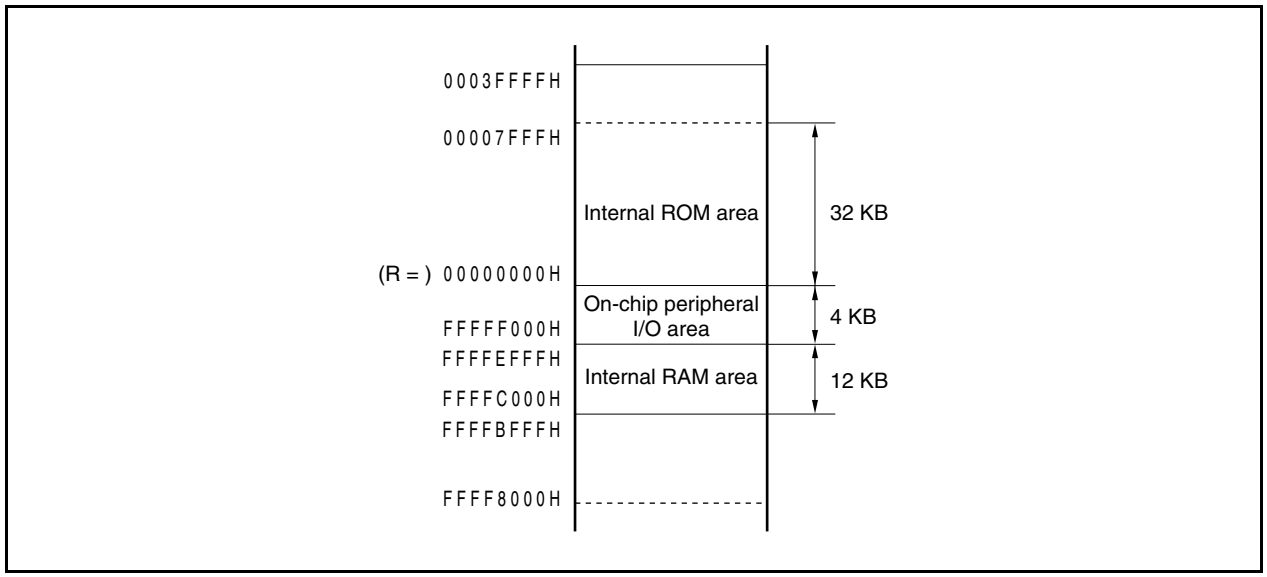
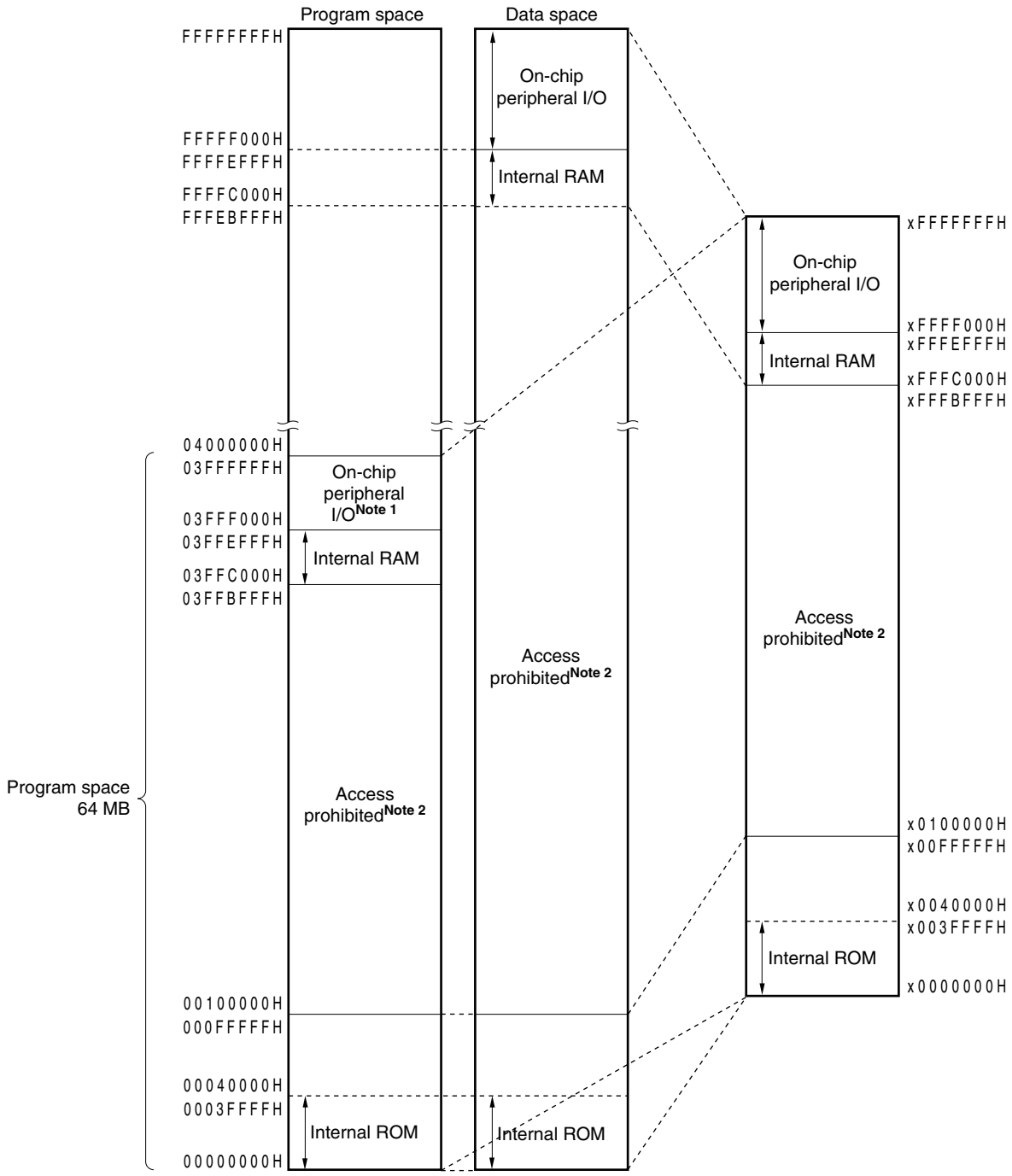


Figure 3-10. Recommended Memory Map



Notes 1. This area is access-prohibited. To access the on-chip peripheral I/O, specify addresses FFFF000H to FFFFFFFH.

2. The operation is not guaranteed if an access-prohibited area is accessed.

Remarks 1. The arrows indicate the recommended area.

2. This is a recommended memory map when the μPD70F3186 (V850E/IA4) is set to the normal operation mode.

3.4.7 On-chip peripheral I/O registers

(1/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF004H	Port DL register	PDL	R/W			√	Undefined
FFFFF004H	Port DL register L	PDLL		√	√		Undefined
FFFFF005H	Port DL register H	PDLH		√	√		Undefined
FFFFF024H	Port DL mode register	PMDL				√	FFFFH
FFFFF024H	Port DL mode register L	PMDLL		√	√		FFH
FFFFF025H	Port DL mode register H	PMDLH		√	√		FFH
FFFFF06EH	System wait control register	VSWC			√		77H
FFFFF080H	DMA source address register 0L	DSA0L				√	Undefined
FFFFF082H	DMA source address register 0H	DSA0H				√	Undefined
FFFFF084H	DMA destination address register 0L	DDA0L				√	Undefined
FFFFF086H	DMA destination address register 0H	DDA0H				√	Undefined
FFFFF088H	DMA source address register 1L	DSA1L				√	Undefined
FFFFF08AH	DMA source address register 1H	DSA1H				√	Undefined
FFFFF08CH	DMA destination address register 1L	DDA1L				√	Undefined
FFFFF08EH	DMA destination address register 1H	DDA1H				√	Undefined
FFFFF090H	DMA source address register 2L	DSA2L				√	Undefined
FFFFF092H	DMA source address register 2H	DSA2H				√	Undefined
FFFFF094H	DMA destination address register 2L	DDA2L				√	Undefined
FFFFF096H	DMA destination address register 2H	DDA2H				√	Undefined
FFFFF098H	DMA source address register 3L	DSA3L				√	Undefined
FFFFF09AH	DMA source address register 3H	DSA3H				√	Undefined
FFFFF09CH	DMA destination address register 3L	DDA3L				√	Undefined
FFFFF09EH	DMA destination address register 3H	DDA3H				√	Undefined
FFFFF0C0H	DMA transfer count register 0	DBC0				√	Undefined
FFFFF0C2H	DMA transfer count register 1	DBC1				√	Undefined
FFFFF0C4H	DMA transfer count register 2	DBC2				√	Undefined
FFFFF0C6H	DMA transfer count register 3	DBC3				√	Undefined
FFFFF0D0H	DMA addressing control register 0	DADC0				√	0000H
FFFFF0D2H	DMA addressing control register 1	DADC1				√	0000H
FFFFF0D4H	DMA addressing control register 2	DADC2				√	0000H
FFFFF0D6H	DMA addressing control register 3	DADC3				√	0000H
FFFFF0E0H	DMA channel control register 0	DCHC0		√	√		00H
FFFFF0E2H	DMA channel control register 1	DCHC1		√	√		00H
FFFFF0E4H	DMA channel control register 2	DCHC2		√	√		00H
FFFFF0E6H	DMA channel control register 3	DCHC3		√	√		00H
FFFFF100H	Interrupt mask register 0	IMR0				√	FFFFH
FFFFF100H	Interrupt mask register 0L	IMR0L		√	√		FFH
FFFFF101H	Interrupt mask register 0H	IMR0H		√	√		FFH
FFFFF102H	Interrupt mask register 1	IMR1				√	FFFFH
FFFFF102H	Interrupt mask register 1L	IMR1L		√	√		FFH
FFFFF103H	Interrupt mask register 1H	IMR1H		√	√		FFH

(2/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF104H	Interrupt mask register 2	IMR2	R/W			√	FFFFH
FFFFF104H	Interrupt mask register 2L	IMR2L		√	√		FFH
FFFFF105H	Interrupt mask register 2H	IMR2H		√	√		FFH
FFFFF106H	Interrupt mask register 3	IMR3				√	FFFFH
FFFFF106H	Interrupt mask register 3L	IMR3L		√	√		FFH
FFFFF107H	Interrupt mask register 3H	IMR3H		√	√		FFH
FFFFF110H	Interrupt control register	PIC0		√	√		47H
FFFFF112H	Interrupt control register	PIC1[Note]		√	√		47H
FFFFF114H	Interrupt control register	PIC2		√	√		47H
FFFFF116H	Interrupt control register	PIC3		√	√		47H
FFFFF118H	Interrupt control register	PIC4		√	√		47H
FFFFF11AH	Interrupt control register	PIC5		√	√		47H
FFFFF11CH	Interrupt control register	PIC6		√	√		47H
FFFFF11EH	Interrupt control register	PIC7		√	√		47H
FFFFF120H	Interrupt control register	CMPIC0		√	√		47H
FFFFF122H	Interrupt control register	CMPIC1		√	√		47H
FFFFF124H	Interrupt control register	TQ0OVIC		√	√		47H
FFFFF126H	Interrupt control register	TQ0CCIC0		√	√		47H
FFFFF128H	Interrupt control register	TQ0CCIC1		√	√		47H
FFFFF12AH	Interrupt control register	TQ0CCIC2		√	√		47H
FFFFF12CH	Interrupt control register	TQ0CCIC3		√	√		47H
FFFFF12EH	Interrupt control register	TQ1OVIC		√	√		47H
FFFFF130H	Interrupt control register	TQ1CCIC0		√	√		47H
FFFFF132H	Interrupt control register	TQ1CCIC1		√	√		47H
FFFFF134H	Interrupt control register	TQ1CCIC2		√	√		47H
FFFFF136H	Interrupt control register	TQ1CCIC3		√	√		47H
FFFFF138H	Interrupt control register	CC0IC0		√	√		47H
FFFFF13AH	Interrupt control register	CC0IC1		√	√		47H
FFFFF13CH	Interrupt control register	CM0IC0		√	√		47H
FFFFF13EH	Interrupt control register	CM0IC1		√	√		47H
FFFFF140H	Interrupt control register	CC1IC0[Note]		√	√		47H
FFFFF142H	Interrupt control register	CC1IC1[Note]		√	√		47H
FFFFF144H	Interrupt control register	CM1IC0[Note]		√	√		47H
FFFFF146H	Interrupt control register	CM1IC1[Note]		√	√		47H
FFFFF148H	Interrupt control register	TP0OVIC		√	√		47H
FFFFF14AH	Interrupt control register	TP0CCIC0		√	√		47H
FFFFF14CH	Interrupt control register	TP0CCIC1		√	√		47H
FFFFF14EH	Interrupt control register	TP1OVIC		√	√		47H
FFFFF150H	Interrupt control register	TP1CCIC0		√	√		47H
FFFFF152H	Interrupt control register	TP1CCIC1		√	√		47H
FFFFF154H	Interrupt control register	TP2OVIC		√	√		47H
FFFFF156H	Interrupt control register	TP2CCIC0		√	√		47H

Note V850E/IA4 only

(3/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF158H	Interrupt control register	TP2CCIC1	R/W	√	√		47H
FFFFF15AH	Interrupt control register	TP3OVIC		√	√		47H
FFFFF15CH	Interrupt control register	TP3CCIC0		√	√		47H
FFFFF15EH	Interrupt control register	TP3CCIC1		√	√		47H
FFFFF160H	Interrupt control register	DMAIC0		√	√		47H
FFFFF162H	Interrupt control register	DMAIC1		√	√		47H
FFFFF164H	Interrupt control register	DMAIC2		√	√		47H
FFFFF166H	Interrupt control register	DMAIC3		√	√		47H
FFFFF168H	Interrupt control register	UA0REIC		√	√		47H
FFFFF16AH	Interrupt control register	UA0RIC		√	√		47H
FFFFF16CH	Interrupt control register	UA0TIC		√	√		47H
FFFFF16EH	Interrupt control register	CB0REIC		√	√		47H
FFFFF170H	Interrupt control register	CB0RIC		√	√		47H
FFFFF172H	Interrupt control register	CB0TIC		√	√		47H
FFFFF174H	Interrupt control register	UA1REIC		√	√		47H
FFFFF176H	Interrupt control register	UA1RIC		√	√		47H
FFFFF178H	Interrupt control register	UA1TIC		√	√		47H
FFFFF17AH	Interrupt control register	CB1REIC		√	√		47H
FFFFF17CH	Interrupt control register	CB1RIC		√	√		47H
FFFFF17EH	Interrupt control register	CB1TIC		√	√		47H
FFFFF180H	Interrupt control register	AD0IC		√	√		47H
FFFFF182H	Interrupt control register	AD1IC		√	√		47H
FFFFF184H	Interrupt control register	AD2IC		√	√		47H
FFFFF186H	Interrupt control register	TM0EQIC0		√	√		47H
FFFFF1FAH	In-service priority register	ISPR	R	√	√		00H
FFFFF1FCH	Command register	PRCMD	W		√		Undefined
FFFFF1FEH	Power save control register	PSC	R/W	√	√		00H
FFFFF200H	A/D converter 0 mode register 0	ADA0M0		√	√		00H
FFFFF201H	A/D converter 0 mode register 1	ADA0M1		√	√		00H
FFFFF202H	A/D converter 0 channel specification register	ADA0S		√	√		00H
FFFFF203H	A/D converter 0 mode register 2	ADA0M2		√	√		00H
FFFFF210H	A/D0 conversion result register 0	ADA0CR0	R			√	Undefined
FFFFF211H	A/D0 conversion result register 0H	ADA0CR0H			√		Undefined
FFFFF212H	A/D0 conversion result register 1	ADA0CR1				√	Undefined
FFFFF213H	A/D0 conversion result register 1H	ADA0CR1H			√		Undefined
FFFFF214H	A/D0 conversion result register 2	ADA0CR2				√	Undefined
FFFFF215H	A/D0 conversion result register 2H	ADA0CR2H			√		Undefined
FFFFF216H	A/D0 conversion result register 3	ADA0CR3				√	Undefined
FFFFF217H	A/D0 conversion result register 3H	ADA0CR3H			√		Undefined
FFFFF218H	A/D0 conversion result register 4	ADA0CR4				√	Undefined
FFFFF219H	A/D0 conversion result register 4H	ADA0CR4H			√		Undefined
FFFFF21AH	A/D0 conversion result register 5	ADA0CR5				√	Undefined
FFFFF21BH	A/D0 conversion result register 5H	ADA0CR5H			√		Undefined

(4/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF21CH	A/D0 conversion result register 6	ADA0CR6	R			√	Undefined
FFFFF21DH	A/D0 conversion result register 6H	ADA0CR6H			√		Undefined
FFFFF21EH	A/D0 conversion result register 7	ADA0CR7				√	Undefined
FFFFF21FH	A/D0 conversion result register 7H	ADA0CR7H			√		Undefined
FFFFF220H	A/D converter 1 mode register 0	ADA1M0	R/W	√	√		00H
FFFFF221H	A/D converter 1 mode register 1	ADA1M1		√	√		00H
FFFFF222H	A/D converter 1 channel specification register	ADA1S		√	√		00H
FFFFF223H	A/D converter 1 mode register 2	ADA1M2		√	√		00H
FFFFF230H	A/D1 conversion result register 0	ADA1CR0	R			√	Undefined
FFFFF231H	A/D1 conversion result register 0H	ADA1CR0H			√		Undefined
FFFFF232H	A/D1 conversion result register 1	ADA1CR1				√	Undefined
FFFFF233H	A/D1 conversion result register 1H	ADA1CR1H			√		Undefined
FFFFF234H	A/D1 conversion result register 2	ADA1CR2				√	Undefined
FFFFF235H	A/D1 conversion result register 2H	ADA1CR2H			√		Undefined
FFFFF236H	A/D1 conversion result register 3	ADA1CR3				√	Undefined
FFFFF237H	A/D1 conversion result register 3H	ADA1CR3H			√		Undefined
FFFFF238H	A/D1 conversion result register 4	ADA1CR4				√	Undefined
FFFFF239H	A/D1 conversion result register 4H	ADA1CR4H			√		Undefined
FFFFF23AH	A/D1 conversion result register 5	ADA1CR5				√	Undefined
FFFFF23BH	A/D1 conversion result register 5H	ADA1CR5H			√		Undefined
FFFFF23CH	A/D1 conversion result register 6	ADA1CR6				√	Undefined
FFFFF23DH	A/D1 conversion result register 6H	ADA1CR6H			√		Undefined
FFFFF23EH	A/D1 conversion result register 7	ADA1CR7				√	Undefined
FFFFF23FH	A/D1 conversion result register 7H	ADA1CR7H			√		Undefined
FFFFF240H	A/D converter 2 control register 0	ADA2CTL0	R/W	√	√		00H
FFFFF241H	A/D converter 2 control register 1	ADA2CTL1		√	√		00H
FFFFF242H	A/D converter 2 control register 2	ADA2CTL2		√	√		00H
FFFFF243H	A/D converter 2 control register 3	ADA2CTL3		√	√		00H
FFFFF246H	A/D converter 2 status register	ADA2STR	R		√		00H
FFFFF250H	A/D2 conversion result register 0	ADA2CR0				√	0000H
FFFFF251H	A/D2 conversion result register 0H	ADA2CR0H			√		00H
FFFFF252H	A/D2 conversion result register 1	ADA2CR1				√	0000H
FFFFF253H	A/D2 conversion result register 1H	ADA2CR1H			√		00H
FFFFF254H	A/D2 conversion result register 2	ADA2CR2				√	0000H
FFFFF255H	A/D2 conversion result register 2H	ADA2CR2H			√		00H
FFFFF256H	A/D2 conversion result register 3	ADA2CR3				√	0000H
FFFFF257H	A/D2 conversion result register 3H	ADA2CR3H			√		00H
FFFFF258H	A/D2 conversion result register 4	ADA2CR4				√	0000H
FFFFF259H	A/D2 conversion result register 4H	ADA2CR4H			√		00H
FFFFF25AH	A/D2 conversion result register 5	ADA2CR5				√	0000H
FFFFF25BH	A/D2 conversion result register 5H	ADA2CR5H			√		00H
FFFFF25CH	A/D2 conversion result register 6	ADA2CR6				√	0000H
FFFFF25DH	A/D2 conversion result register 6H	ADA2CR6H			√		00H
FFFFF25EH	A/D2 conversion result register 7	ADA2CR7				√	0000H
FFFFF25FH	A/D2 conversion result register 7H	ADA2CR7H			√		00H

(5/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF260H	Operational amplifier 0 control register 0	OP0CTL0	R/W	√	√		00H
FFFFF261H	Operational amplifier 0 control register 1	OP0CTL1		√	√		00H
FFFFF268H	Operational amplifier 1 control register 0	OP1CTL0		√	√		00H
FFFFF269H	Operational amplifier 1 control register 1	OP1CTL1		√	√		00H
FFFFF310H	External interrupt noise elimination control register	INTPNRC		√	√		00H
FFFFF400H	Port 0 register	P0		√	√		Undefined
FFFFF402H	Port 1 register	P1		√	√		Undefined
FFFFF404H	Port 2 register	P2[Note]		√	√		Undefined
FFFFF406H	Port 3 register	P3		√	√		Undefined
FFFFF408H	Port 4 register	P4		√	√		Undefined
FFFFF40AH	Port 5 register	P5[Note]		√	√		Undefined
FFFFF40EH	Port 7 register	P7	R	√	√		Undefined
FFFFF420H	Port 0 mode register	PM0	R/W	√	√		FFH
FFFFF422H	Port 1 mode register	PM1		√	√		FFH
FFFFF424H	Port 2 mode register	PM2[Note]		√	√		FFH
FFFFF426H	Port 3 mode register	PM3		√	√		FFH
FFFFF428H	Port 4 mode register	PM4		√	√		FFH
FFFFF42AH	Port 5 mode register	PM5[Note]		√	√		FFH
FFFFF440H	Port 0 mode control register	PMC0		√	√		00H
FFFFF442H	Port 1 mode control register	PMC1		√	√		00H
FFFFF444H	Port 2 mode control register	PMC2[Note]		√	√		00H
FFFFF446H	Port 3 mode control register	PMC3		√	√		00H
FFFFF448H	Port 4 mode control register	PMC4		√	√		00H
FFFFF44AH	Port 5 mode control register	PMC5[Note]		√	√		00H
FFFFF44EH	Port 7 mode control register	PMC7		√	√		00H
FFFFF462H	Port 1 function control register	PFC1		√	√		00H
FFFFF466H	Port 3 function control register	PFC3		√	√		00H
FFFFF468H	Port 4 function control register	PFC4		√	√		00H
FFFFF46AH	Port 5 function control register	PFC5[Note]		√	√		00H
FFFFF540H	TMM0 control register 0	TM0CTL0		√	√		00H
FFFFF544H	TMM0 compare register 0	TM0CMP0				√	0000H
FFFFF580H	Timer ENC10	TMENC10				√	0000H
FFFFF582H	Compare register 100	CM100				√	0000H
FFFFF584H	Compare register 101	CM101				√	0000H
FFFFF586H	Capture/compare register 100	CC100				√	0000H
FFFFF588H	Capture/compare register 101	CC101				√	0000H
FFFFF58AH	Capture/compare control register 10	CCR10		√	√		00H
FFFFF58BH	Timer unit mode register 10	TUM10		√	√		00H
FFFFF58CH	Timer control register 10	TMC10		√	√		00H
FFFFF58DH	Valid edge select register 10	SESA10		√	√		00H
FFFFF58EH	Prescaler mode register 10	PRM10		√	√		07H
FFFFF58FH	Status register 10	STATUS10	R	√	√		00H

Note V850E/IA4 only

(6/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF596H	CC101 capture input select register	CSL10	R/W	√	√		00H
FFFFF598H	Noise elimination time select register 10	NRC10		√	√		00H
FFFFF5A0H	Timer ENC11	TMENC11Note				√	0000H
FFFFF5A2H	Compare register 110	CM110Note				√	0000H
FFFFF5A4H	Compare register 111	CM111Note				√	0000H
FFFFF5A6H	Capture/compare register 110	CC110Note				√	0000H
FFFFF5A8H	Capture/compare register 111	CC111Note				√	0000H
FFFFF5AAH	Capture/compare control register 11	CCR11Note		√	√		00H
FFFFF5ABH	Timer unit mode register 11	TUM11Note		√	√		00H
FFFFF5ACH	Timer control register 11	TMC11Note		√	√		00H
FFFFF5ADH	Valid edge select register 11	SESA11Note		√	√		00H
FFFFF5AEH	Prescaler mode register 11	PRM11Note		√	√		07H
FFFFF5AFH	Status register 11	STATUS11Note	R	√	√		00H
FFFFF5B6H	CC111 capture input select register	CSL11Note	R/W	√	√		00H
FFFFF5B8H	Noise elimination time select register 11	NRC11Note		√	√		00H
FFFFF5C0H	TMQ0 control register 0	TQ0CTL0		√	√		00H
FFFFF5C1H	TMQ0 control register 1	TQ0CTL1		√	√		00H
FFFFF5C2H	TMQ0 I/O control register 0	TQ0IOC0		√	√		00H
FFFFF5C3H	TMQ0 I/O control register 1	TQ0IOC1		√	√		00H
FFFFF5C4H	TMQ0 I/O control register 2	TQ0IOC2		√	√		00H
FFFFF5C5H	TMQ0 option register 0	TQ0OPT0		√	√		00H
FFFFF5C6H	TMQ0 capture/compare register 0	TQ0CCR0				√	0000H
FFFFF5C8H	TMQ0 capture/compare register 1	TQ0CCR1				√	0000H
FFFFF5CAH	TMQ0 capture/compare register 2	TQ0CCR2				√	0000H
FFFFF5CCH	TMQ0 capture/compare register 3	TQ0CCR3				√	0000H
FFFFF5CEH	TMQ0 counter read buffer register	TQ0CNT	R			√	0000H
FFFFF5E0H	TMQ0 option register 1	TQ0OPT1	R/W	√	√		00H
FFFFF5E1H	TMQ0 option register 2	TQ0OPT2		√	√		00H
FFFFF5E2H	TMQ0 I/O control register 3	TQ0IOC3		√	√		A8H
FFFFF5E3H	TMQ0 option register 3	TQ0OPT3		√	√		00H
FFFFF5E4H	TMQ0 deadtime compare register	TQ0DTC				√	0000H
FFFFF5F0H	High-impedance output control register 00	HZA0CTL0		√	√		00H
FFFFF5F1H	High-impedance output control register 01	HZA0CTL1		√	√		00H
FFFFF600H	TMQ1 control register 0	TQ1CTL0		√	√		00H
FFFFF601H	TMQ1 control register 1	TQ1CTL1		√	√		00H
FFFFF602H	TMQ1 I/O control register 0	TQ1IOC0Note		√	√		00H
FFFFF605H	TMQ1 option register 0	TQ1OPT0		√	√		00H
FFFFF606H	TMQ1 capture/compare register 0	TQ1CCR0				√	0000H
FFFFF608H	TMQ1 capture/compare register 1	TQ1CCR1				√	0000H
FFFFF60AH	TMQ1 capture/compare register 2	TQ1CCR2				√	0000H
FFFFF60CH	TMQ1 capture/compare register 3	TQ1CCR3				√	0000H
FFFFF60EH	TMQ1 counter read buffer register	TQ1CNT	R			√	0000H
FFFFF620H	TMQ1 option register 1	TQ1OPT1Note	R/W	√	√		00H

Note V850E/IA4 only

(7/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFF621H	TMQ1 option register 2	TQ1OPT2[Note]	R/W	√	√		00H
FFFFF622H	TMQ1 I/O control register 3	TQ1IOC3[Note]		√	√		A8H
FFFFF623H	TMQ1 option register 3	TQ1OPT3[Note]		√	√		00H
FFFFF624H	TMQ1 deadtime compare register	TQ1DTC[Note]				√	0000H
FFFFF630H	High-impedance output control register 10	HZA1CTL0[Note]		√	√		00H
FFFFF631H	High-impedance output control register 11	HZA1CTL1[Note]		√	√		00H
FFFFF638H	High-impedance output control register 20	HZA2CTL0		√	√		00H
FFFFF639H	High-impedance output control register 21	HZA2CTL1		√	√		00H
FFFFF640H	TMP0 control register 0	TP0CTL0		√	√		00H
FFFFF641H	TMP0 control register 1	TP0CTL1		√	√		00H
FFFFF642H	TMP0 I/O control register 0	TP0IOC0		√	√		00H
FFFFF643H	TMP0 I/O control register 1	TP0IOC1		√	√		00H
FFFFF644H	TMP0 I/O control register 2	TP0IOC2		√	√		00H
FFFFF645H	TMP0 option register 0	TP0OPT0		√	√		00H
FFFFF646H	TMP0 capture/compare register 0	TP0CCR0				√	0000H
FFFFF648H	TMP0 capture/compare register 1	TP0CCR1				√	0000H
FFFFF64AH	TMP0 counter read buffer register	TP0CNT	R			√	0000H
FFFFF660H	TMP1 control register 0	TP1CTL0	R/W	√	√		00H
FFFFF661H	TMP1 control register 1	TP1CTL1		√	√		00H
FFFFF665H	TMP1 option register 0	TP1OPT0		√	√		00H
FFFFF666H	TMP1 capture/compare register 0	TP1CCR0				√	0000H
FFFFF668H	TMP1 capture/compare register 1	TP1CCR1				√	0000H
FFFFF66AH	TMP1 counter read buffer register	TP1CNT	R			√	0000H
FFFFF680H	TMP2 control register 0	TP2CTL0	R/W	√	√		00H
FFFFF681H	TMP2 control register 1	TP2CTL1		√	√		00H
FFFFF682H	TMP2 I/O control register 0	TP2IOC0		√	√		00H
FFFFF683H	TMP2 I/O control register 1	TP2IOC1		√	√		00H
FFFFF684H	TMP2 I/O control register 2	TP2IOC2		√	√		00H
FFFFF685H	TMP2 option register 0	TP2OPT0		√	√		00H
FFFFF686H	TMP2 capture/compare register 0	TP2CCR0				√	0000H
FFFFF688H	TMP2 capture/compare register 1	TP2CCR1				√	0000H
FFFFF68AH	TMP2 counter read buffer register	TP2CNT	R			√	0000H
FFFFF6A0H	TMP3 control register 0	TP3CTL0	R/W	√	√		00H
FFFFF6A1H	TMP3 control register 1	TP3CTL1		√	√		00H
FFFFF6A2H	TMP3 I/O control register 0	TP3IOC0[Note]		√	√		00H
FFFFF6A5H	TMP3 option register 0	TP3OPT0		√	√		00H
FFFFF6A6H	TMP3 capture/compare register 0	TP3CCR0				√	0000H
FFFFF6A8H	TMP3 capture/compare register 1	TP3CCR1				√	0000H
FFFFF6AAH	TMP3 counter read buffer register	TP3CNT	R			√	0000H
FFFFF6C0H	Oscillation stabilization time select register	OSTS	R/W		√		04H
FFFFF6D0H	Watchdog timer mode register	WDTM			√		67H
FFFFF6D1H	Watchdog timer enable register	WDTE			√		1AH

Note V850E/IA4 only

(8/9)

	Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation				After Reset
					1	8	16	32	
	FFFFF702H	Port 1 function control expansion register	PFCE1	R/W	√	√			00H
	FFFFF802H	System status register	SYS		√	√			00H
	FFFFF810H	DMA trigger factor register 0	DTFR0		√	√			00H
	FFFFF812H	DMA trigger factor register 1	DTFR1		√	√			00H
	FFFFF814H	DMA trigger factor register 2	DTFR2		√	√			00H
	FFFFF816H	DMA trigger factor register 3	DTFR3		√	√			00H
	FFFFF820H	Power save mode register	PSMR		√	√			00H
	FFFFF828H	Processor clock control register	PCC		√	√			03H
	FFFFF82CH	PLL control register	PLLCTL		√	√			01H
	FFFFF840H	Correction address register 0	CORAD0					√	00000000H
	FFFFF840H	Correction address register 0L	CORAD0L				√		0000H
	FFFFF842H	Correction address register 0H	CORAD0H				√		0000H
	FFFFF844H	Correction address register 1	CORAD1					√	00000000H
	FFFFF844H	Correction address register 1L	CORAD1L				√		0000H
	FFFFF846H	Correction address register 1H	CORAD1H				√		0000H
	FFFFF848H	Correction address register 2	CORAD2					√	00000000H
	FFFFF848H	Correction address register 2L	CORAD2L				√		0000H
	FFFFF84AH	Correction address register 2H	CORAD2H				√		0000H
	FFFFF84CH	Correction address register 3	CORAD3					√	00000000H
	FFFFF84CH	Correction address register 3L	CORAD3L				√		0000H
	FFFFF84EH	Correction address register 3H	CORAD3H				√		0000H
	FFFFF870H	Clock monitor mode register	CLM		√	√			00H
	FFFFF880H	Correction control register	CORCN		√	√			00H
	FFFFF888H	Reset source flag register	RESF		√	√			00H/10H
	FFFFF9F0H	Internal memory size switching register	IMS[Note]			√			00H
	FFFFFA00H	UARTA0 control register 0	UA0CTL0		√	√			10H
<R>	FFFFFA01H	UARTA0 control register 1	UA0CTL1			√			00H
	FFFFFA02H	UARTA0 control register 2	UA0CTL2			√			FFH
	FFFFFA03H	UARTA0 option control register 0	UA0OPT0		√	√			14H
	FFFFFA04H	UARTA0 status register	UA0STR		√	√			00H
	FFFFFA06H	UARTA0 receive data register	UA0RX	R		√			FFH
	FFFFFA07H	UARTA0 transmit data register	UA0TX	R/W		√			FFH
	FFFFFA10H	UARTA1 control register 0	UA1CTL0		√	√			10H
<R>	FFFFFA11H	UARTA1 control register 1	UA1CTL1			√			00H
	FFFFFA12H	UARTA1 control register 2	UA1CTL2			√			FFH
	FFFFFA13H	UARTA1 option control register 0	UA1OPT0		√	√			14H
	FFFFFA14H	UARTA1 status register	UA1STR		√	√			00H
	FFFFFA16H	UARTA1 receive data register	UA1RX	R		√			FFH

Note μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4) only

(9/9)

Address	Function Register Name	Symbol	R/W	Bit Units for Manipulation			After Reset
				1	8	16	
FFFFFA17H	UARTA1 transmit data register	UA1TX	R/W		√		FFH
FFFFFC00H	External interrupt falling edge specification register 0	INTF0		√	√		00H
FFFFFC20H	External interrupt rising edge specification register 0	INTR0		√	√		00H
FFFFFC40H	Pull-up resistor option register 0	PU0		√	√		00H
FFFFFC42H	Pull-up resistor option register 1	PU1		√	√		00H
FFFFFC44H	Pull-up resistor option register 2	PU2[Note]		√	√		00H
FFFFFC46H	Pull-up resistor option register 3	PU3		√	√		00H
FFFFFC48H	Pull-up resistor option register 4	PU4		√	√		00H
FFFFFC4AH	Pull-up resistor option register 5	PU5[Note]		√	√		00H
FFFFFD00H	CSIB0 control register 0	CB0CTL0		√	√		01H
FFFFFD01H	CSIB0 control register 1	CB0CTL1		√	√		00H
FFFFFD02H	CSIB0 control register 2	CB0CTL2			√		00H
FFFFFD03H	CSIB0 status register	CB0STR		√	√		00H
FFFFFD04H	CSIB0 receive data register	CB0RX	R			√	0000H
FFFFFD04H	CSIB0 receive data register L	CB0RXL			√		00H
FFFFFD06H	CSIB0 transmit data register	CB0TX	R/W			√	0000H
FFFFFD06H	CSIB0 transmit data register L	CB0TXL			√		00H
FFFFFD10H	CSIB1 control register 0	CB1CTL0		√	√		01H
FFFFFD11H	CSIB1 control register 1	CB1CTL1		√	√		00H
FFFFFD12H	CSIB1 control register 2	CB1CTL2			√		00H
FFFFFD13H	CSIB1 status register	CB1STR		√	√		00H
FFFFFD14H	CSIB1 receive data register	CB1RX	R			√	0000H
FFFFFD14H	CSIB1 receive data register L	CB1RXL			√		00H
FFFFFD16H	CSIB1 transmit data register	CB1TX	R/W			√	0000H
FFFFFD16H	CSIB1 transmit data register L	CB1TXL			√		00H
FFFFFF44H	Pull-up resistor option register DL	PUDL				√	0000H
FFFFFF44H	Pull-up resistor option register DLL	PUDLL		√	√		00H
FFFFFF45H	Pull-up resistor option register DLH	PUDLH		√	√		00H

Note V850E/IA4 only

3.4.8 Special registers

Special registers are registers that are protected from being written with illegal data due to a program hang-up.

The V850E/IA3 and V850E/IA4 have the following four special registers.

- Power save control register (PSC)
- Processor clock control register (PCC)
- Reset source flag register (RESF)
- Clock monitor mode register (CLM)

In addition, a command register (PRCDM) is provided to protect against a write access to the special registers so that the application system does not inadvertently stop due to a program hang-up. A write access to the special registers is made in a specific sequence, and an illegal store operation is reported to the system status register (SYS).

(1) Setting data to special registers

Set data to the special registers in the following sequence.

<1> Prepare data to be set to the special register in a general-purpose register.
<2> Write the data prepared in <1> to the command register.
<3> Write the setting data to the special register (by using the following instructions).
- Store instruction (ST/SST instruction)
- Bit manipulation instruction (SET1/CLR1/NOT1 instruction)

(<4> to <8> Insert NOP instructions (5 instructions).)[Note]

[Example] With PSC register (setting standby mode)

```
    ST.B r11, PSMR[r0]   ; Set PSMR register (setting IDLE and STOP modes).
<1>MOV 0x02, r10
<2>ST.B r10, PRCMD[r0]   ; Write PRCMD register.
<3>ST.B r10, PSC[r0]     ; Set PSC register.
<4>NOP[Note]             ; Dummy instruction
<5>NOP[Note]             ; Dummy instruction
<6>NOP[Note]             ; Dummy instruction
<7>NOP[Note]             ; Dummy instruction
<8>NOP[Note]             ; Dummy instruction
 (next instruction)
```

There is no special sequence to read a special register.

Note Five NOP instructions or more must be inserted immediately after setting the IDLE mode or STOP mode (by setting the PSC.STB bit to 1).

Cautions 1. When a store instruction is executed to store data in the command register, interrupts are not acknowledged. This is because it is assumed that steps <2> and <3> above are performed by successive store instructions. If another instruction is placed between <2> and <3>, and if an interrupt is acknowledged by that instruction, the above sequence may not be established, causing malfunction.

2. Although dummy data is written to the command register, use the same general-purpose register used to set the special register (<3> in Example) by using the store instruction to write data to the command register (<2> in Example). The same applies when a general-purpose register is used for addressing.

An example of setting the special register (<3> in Example) by using the bit manipulation instruction is shown below.

```
CLR1 4, RESF[r0]
```

3. Before executing this processing, terminate all DMA transfer operations.

(2) Command register (PRCMD)

The PRCMD register is an 8-bit register that protects the registers that may seriously affect the application system from being written, so that the system does not inadvertently stop due to a program hang-up. The first write access to a special register is valid after data has been written in advance to the PRCMD register. In this way, the value of the special register can be rewritten only in a specific sequence, so as to protect the register from an illegal write access.

An illegal write operation to a special register can be checked by using the SYS.PRERR bit.

The PRCMD register is write-only, in 8-bit units (undefined data is read when this register is read).

Reset makes this register undefined.

After reset: Undefined W Address: FFFFF1FCH

	7	6	5	4	3	2	1	0
PRCMD	REG7	REG6	REG5	REG4	REG3	REG2	REG1	REG0

(3) System status register (SYS)

Status flags that indicate the operation status of the overall system are allocated to this register.

If this register is not written in the correct sequence including an access to the PRCMD register, data is not written to the intended register, a protection error occurs, and the PRERR flag is set. This register is cleared by writing "0" to it by an instruction from CPU.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF802H

	7	6	5	4	3	2	1	<0>
SYS	0	0	0	0	0	0	0	PRERR

PRERR	Protection error detection
0	Protection error did not occur.
1	Protection error occurred.

The PRERR flag operates under the following conditions.

(a) Set condition (PRERR flag = 1)

- When data is written to a special register without writing anything to the PRCMD register (when <3> is executed without executing <2> in **3.4.8 (1) Setting data to special registers**)
- When data is written to an on-chip peripheral I/O register other than a special register (including execution of a bit manipulation instruction) after writing data to the PRCMD register (if <3> in **3.4.8 (1) Setting data to special registers** is not the setting of a special register)

Remark Even if an on-chip peripheral I/O register is read (excluding execution of a bit manipulation instruction) between a write access to the PRCMD register and a write access to a special register (such as an access to the internal RAM), the PRERR flag is not set and data can be written to the special register.

(b) Clear condition (PRERR flag = 0)

(i) When 0 is written to the SYS.PRERR flag

(ii) When the system is reset

Cautions 1. If 0 is written to the PRERR bit of the SYS register which is not a special register, immediately after a write access to the PRCMD register, the PRERR bit is cleared to 0 (the write access takes precedence).

2. If data is written to the PRCMD register, which is not a special register, immediately after a write access to the PRCMD register, the PRERR bit is set to 1.

3.4.9 System wait control register (VSWC)

The VSWC register is a register that controls the bus access wait for the on-chip peripheral I/O registers.

Access to on-chip peripheral I/O registers of the V850E1 CPU core is basically made in 3 clocks; however, in the V850E/IA3 and V850E/IA4, a wait set by the VSWC register is required in addition to those 3 clocks. Set 13H (set wait for 4 clocks) to VSWC.

This register can be read or written in 8-bit units (address: FFFFF06EH, initial value: 77H).

CPU Clock Frequency (f_{CPU})	VSWC Set Value
500 kHz ≤ f_{CPU} ≤ 64 MHz	13H

Caution When using the V850E/IA3 and V850E/IA4, the VSWC register must be set first. Set other registers if necessary after setting the VSWC register.

Remark When a register includes status flags that indicate the statuses of the on-chip peripheral functions (register such as STATUS1n) or a register that indicates the count value of a timer (such as TMENC1n) is accessed, a register access retry operation takes place if the timing at which the flag and count value changes and the timing of the register access overlap. Consequently, access to the on-chip peripheral I/O register may take a long time.

3.4.10 Cautions

Use the processing in either (1) or (2) below to set the internal RAM size to 6 KB in the μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4).

(1) Compile using the DF703183 (V850E/IA3) or DF703185 (V850E/IA4) device file.

(2) Change the description of the address immediately after the SIDATA and DATA labels in the link directive file as follows.

[Description example]

```
SIDATA  : !LOAD ?RW V0fffd800  ← V0fffd800 is the internal RAM start address
              :
              :
DATA    : !LOAD ?RW V0fffd900  ← V0fffd900 is the internal RAM start address + 100 addresses
```

Remark When using a partner manufacturer's tool, define so that the area used by the internal RAM does not exceed the range from FFFD800H to FFFEFFFH in the file equivalent to the link directive file.

CHAPTER 4 PORT FUNCTIONS

4.1 Features

4.1.1 V850E/IA3

- ○ Input-only ports: 6
  I/O ports: 44
- ○ Input and output can be specified in 1-bit units.
- ○ On-chip pull-up resistor can be connected in 1-bit units (ports 0, 1, 3, 4, and DL only)
  However, an on-chip pull-up resistor can only be connected when the pins are in input mode in the port mode, or when the pins function as input pins in the alternate-function mode. Moreover, an on-chip pull-up resistor can be connected to the TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, and TOP21 pins, these are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ0OFF or TOP2OFF pin or software processing.

4.1.2 V850E/IA4

- ○ Input-only ports: 8
  I/O ports: 56
- ○ Input and output can be specified in 1-bit units.
- ○ On-chip pull-up resistor can be connected in 1-bit units (ports 0 to 5 and DL only)
  However, an on-chip pull-up resistor can only be connected when the pins are in input mode in the port mode, or when the pins function as input pins in the alternate-function mode. Moreover, an on-chip pull-up resistor can be connected to the TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, TOP21, TOQ1T1 to TOQ1T3, TOQ1B1 to TOQ1B3, and TOP31 pins, these are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ0OFF, TOQ1OFF, TOP2OFF, or TOP3OFF pin or software processing.

4.2 Port Configuration

4.2.1 V850E/IA3

The V850E/IA3 incorporates a total of 50 input/output ports (including 6 input-only ports) labeled ports 0, 1, 3, 4, 7, and DL. The port configuration is shown in Figure 4-1.

There are two power supplies for the I/O buffer of a pin: AV_{DD} and EV_{DD}. The relationship between each of these power supplies and the pin is shown in Table 4-1.

Figure 4-1. Port Configuration (V850E/IA3)

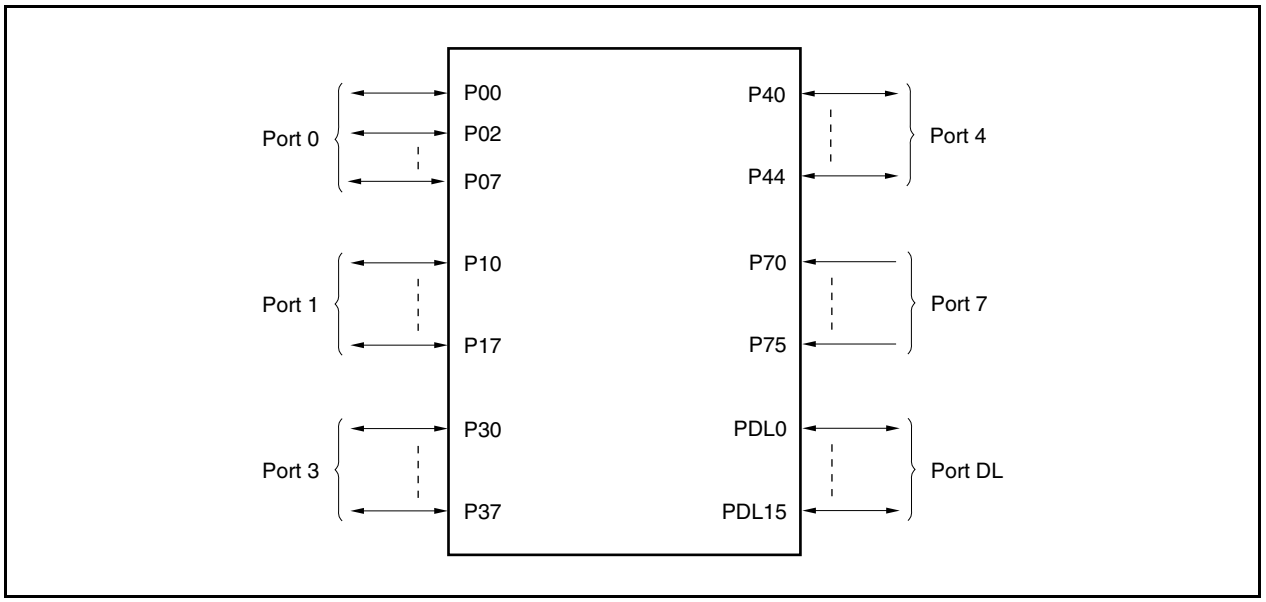


Table 4-1. Power Supplies for I/O Buffer of Each Pin (V850E/IA3)

Power Supply	Corresponding Pins
AV_{DD}	P70 to P75
EV_{DD}	P00, P02 to P07, P10 to P17, P30 to P37, P40 to P44, PDL0 to PDL15, $\overline{\text{RESET}}$

4.2.2 V850E/IA4

The V850E/IA4 incorporates a total of 64 input/output ports (including 8 input-only ports) labeled ports 0 to 5, 7, and DL. The port configuration is shown in Figure 4-2.

There are two power supplies for the I/O buffer of a pin: AV_{DD} and EV_{DD}. The relationship between each of these power supplies and the pin is shown in Table 4-2.

Figure 4-2. Port Configuration (V850E/IA4)

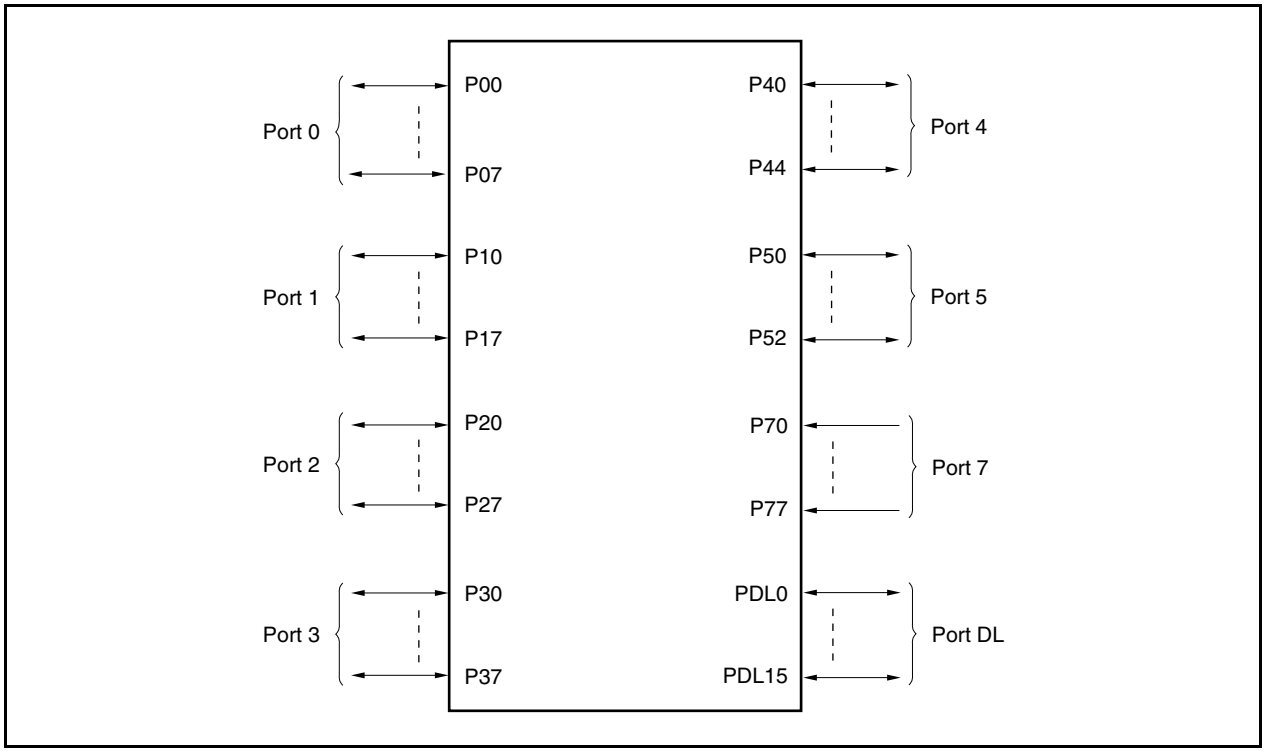


Table 4-2. Power Supplies for I/O Buffer of Each Pin (V850E/IA4)

Power Supply	Corresponding Pins
AV_{DD}	P70 to P77
EV_{DD}	P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15, $\overline{\text{RESET}}$, DCK[Note], DMS[Note], DDI[Note], DDO[Note], $\overline{\text{DRST}}$[Note]

Note μPD70F3186 (V850E/IA4) only

4.3 Port Configuration

Table 4-3. Port Configuration (V850E/IA3)

Item	Configuration
Control registers	Port n register (Pn: n = 0, 1, 3, 4, 7, DL) Port n mode register (PMn: n = 0, 1, 3, 4, DL) Port n mode control register (PMCn: n = 0, 1, 3, 4, 7) Port n function control register (PFCn: n = 1, 3, 4) Port 1 function control expansion register (PFCE1) Pull-up resistor option register (PUn: n = 0, 1, 3, 4, DL)
Ports	Input-only: 6, I/O: 44
Pull-up resistor	Software control: 44

Table 4-4. Port Configuration (V850E/IA4)

Item	Configuration
Control registers	Port n register (Pn: n = 0 to 5, 7, DL) Port n mode register (PMn: n = 0 to 5, DL) Port n mode control register (PMCn: n = 0 to 5, 7) Port n function control register (PFCn: n = 1, 3 to 5) Port 1 function control expansion register (PFCE1) Pull-up resistor option register (PUn: n = 0 to 5, DL)
Ports	Input-only: 8, I/O: 56
Pull-up resistor	Software control: 56

(1) Port n register (Pn)

Data is input from or output to an external device by writing or reading the Pn register.

The Pn register consists of a port latch that holds output data, and a circuit that reads the status of pins.

Each bit of the Pn register corresponds to one pin of port n, and can be read or written in 1-bit units.

After reset: Undefined R/W

	7	6	5	7	3	2	1	0
Pn	Pn7	Pn6	Pn5	Pn4	Pn3	Pn2	Pn1	Pn0

Pnm	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Data is written to or read from the Pn register as follows, regardless of the setting of the PMCn register.

Table 4-5. Writing/Reading Pn Register

Setting of PMn Register	Writing to Pn Register	Reading from Pn Register
Output mode (PMnm = 0)	Data is written to the output latch[Note 1]. In the port mode (PMCn = 0), the contents of the output latch are output from the pins.	The value of the output latch is read[Note 2].
Input mode (PMnm = 1)	Data is written to the output latch. The pin status is not affected[Note 1].	The pin status is read[Note 3].

Notes 1. The value written to the output latch is retained until a new value is written to the output latch.

2. Also, the value of the Pn register is read when the PMn register is in the output mode while the alternate function is set.

3. If the PMn register is in the input mode while the alternate function is set, the statuses of the pins at that time are read regardless of whether the alternate function is an input or output function.

(2) Port n mode register (PMn)

The PMn register specifies the input or output mode of the corresponding port pin.

Each bit of this register corresponds to one pin of port n, and the input or output mode can be specified in 1-bit units.

After reset: FFH R/W

	7	6	5	4	3	2	1	0
PMn	PMn7	PMn6	PMn5	PMn4	PMn3	PMn2	PMn1	PMn0

PMnm	Control of input/output mode
0	Output mode
1	Input mode

(3) Port n mode control register (PMCn)

The PMCn register specifies the port mode or alternate function.

Each bit of this register corresponds to one pin of port n, and the mode of the port can be specified in 1-bit units.

After reset: 00H R/W

	7	6	5	4	3	2	1	0
PMCn	PMCn7	PMCn6	PMCn5	PMCn4	PMCn3	PMCn2	PMCn1	PMCn0

PMCnm	Specification of operation mode
0	Port mode
1	Alternate function

(4) Port n function control register (PFCn)

The PFCn register specifies the alternate function of a port pin to be used if the pin has two alternate functions.

Each bit of this register corresponds to one pin of port n, and the alternate function of a port pin can be specified in 1-bit units.

After reset: 00H R/W

	7	6	5	4	3	2	1	0
PFCn	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0

PFCnm	Specification of alternate function
0	Alternate function 1
1	Alternate function 2

(5) Port n function control expansion register (PFCEn)

The PFCEn register specifies the alternate function of a port pin to be used if the pin has three or more alternate functions.

Each bit of this register corresponds to one pin of port n, and the alternate function of a port pin can be specified in 1-bit units.

After reset: 00H R/W

	7	6	5	4	3	2	1	0
PFCEn	PFCEn7	PFCEn6	PFCEn5	PFCEn4	PFCEn3	PFCEn2	PFCEn1	PFCEn0

	7	6	5	4	3	2	1	0
PFCn	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0

PFCEnm	PFCnm	Specification of alternate function
0	0	Alternate function 1
0	1	Alternate function 2
1	0	Alternate function 3
1	1	Alternate function 4

(6) Pull-up resistor option register (PUn)

PUn is a register that specifies the connection of an on-chip pull-up resistor.

Each bit of the pull-up resistor option register corresponds to one pin of port n and can be specified in 1-bit units.

After reset: 00H R/W

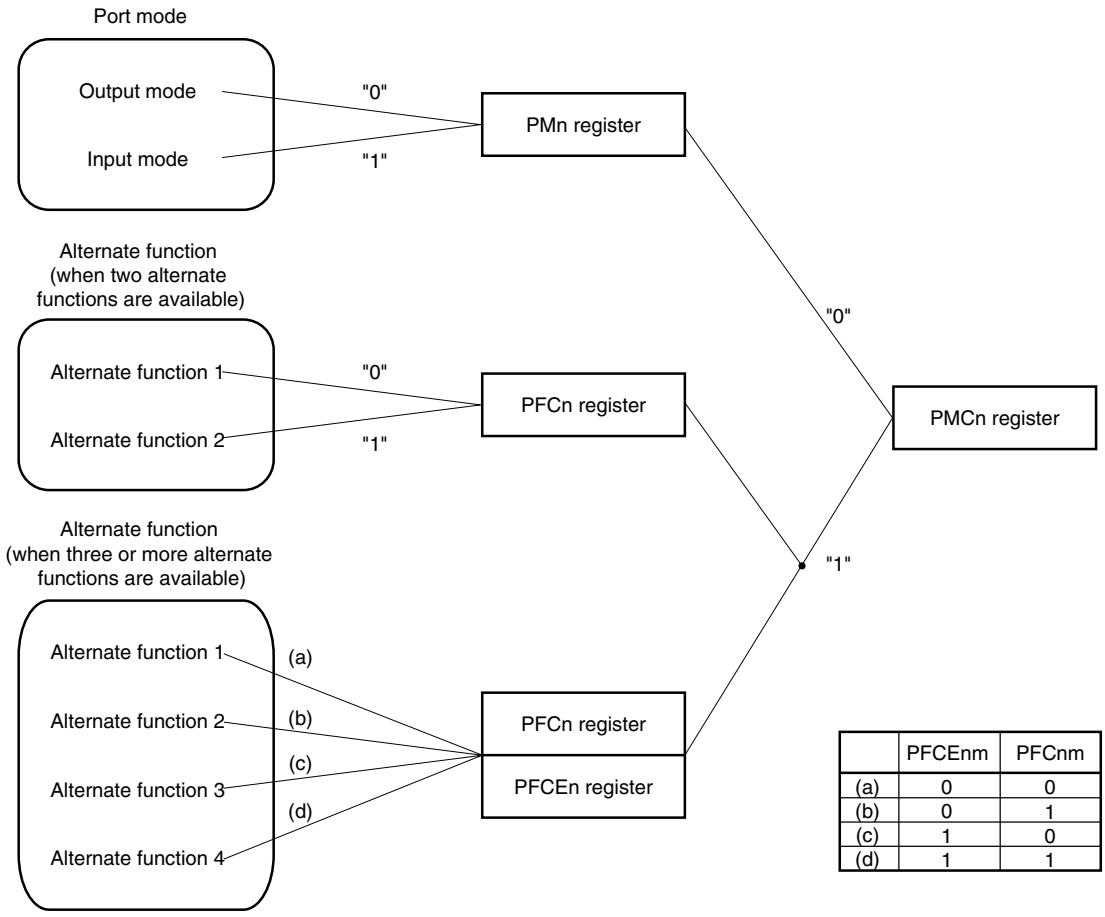
	7	6	5	4	3	2	1	0
PUn	PUn7	PUn6	PUn5	PUn4	PUn3	PUn2	PUn1	PUn0

PUnm	Control of on-chip pull-up resistor connection
0	Not connected
1	Connected

(7) Port settings

Set the ports as follows.

Figure 4-3. Register Settings and Pin Functions



	PFCEnm	PFCnm
(a)	0	0
(b)	0	1
(c)	1	0
(d)	1	1

Caution To switch to external interrupt input (INTPn) from the port mode (by changing the PMC0.PMC0n bit from 0 to 1), an external interrupt may be input if a wrong valid edge is detected. Therefore, be sure to disable edge detection (INTF0.INTF0n bit = 0 and INTR0.INTR0n bit = 0), select external interrupt input (INTPn), and then specify the valid edge (n = 0 to 7).

When switching to the port mode from external interrupt input (INTPn) (PMC0n bit = 1 → 0), an edge may be detected. Be sure to disable edge detection (INTF0n bit = 0, INTR0n bit = 0), and then select the port mode.

Remark Switch to the alternate function using the following procedure.

<1> Set the PFCn and PFCEn registers.

<2> Set the PMCn register.

<3> Set the INTRn/INTFn register (when external interrupt pin is set).

If the PMCn register is set before setting the PFCn and PFCEn registers, an unexpected peripheral function may be selected while the PFCn and PFCEn registers are being set.

4.3.1 Port 0

Port 0 can be set to the input or output mode in 1-bit units.

The number of I/O pins differs from one product to another.

Commercial Name	Number of I/O Pins
V850E/IA3	7-bit I/O port
V850E/IA4	8-bit I/O port

Port 0 has an alternate function as the following pins.

Table 4-6. Alternate-Function Pins of Port 0

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note 1]
	IA3	IA4				
	GC	GC	GF			
P00	18	22	50	INTP0/TOQ0OFF[Note 2]	Input	Provided
P01[Note 3]	–	23	51	INTP1[Note 3]/TOQ1OFF[Notes 2, 3]	Input	
P02	19	24	52	INTP2/TOP2OFF[Note 2]	Input	
P03	20	25	53	INTP3/TOP3OFF[Notes 2, 3]	Input	
P04	21	26	54	INTP4/ADTRG0[Note 2]	Input	
P05	22	27	55	INTP5/ADTRG1[Note 2]	Input	
P06	23	28	56	INTP6	Input	
P07	24	29	57	INTP7	Input	

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

Notes 1. Software pull-up function

2. The TOQ0OFF, TOQ1OFF, TOP2OFF, TOP3OFF (V850E/IA4 only), ADTRG0, and ADTRG1 signals are input to the high-impedance output controller (see **CHAPTER 10 MOTOR CONTROL FUNCTION**) and A/D converters 0 and 1 (see **CHAPTER 12 A/D CONVERTERS 0 AND 1**) after noise is eliminated by a port (analog delay). In addition, a signal whose edge was detected is input to the interrupt controller (INTC) as INTPn (V850E/IA3: n = 0, 2 to 5, V850E/IA4: n = 0 to 5). Edge detection is performed by the high-impedance output controller and A/D converters 0 and 1.

3. V850E/IA4 only

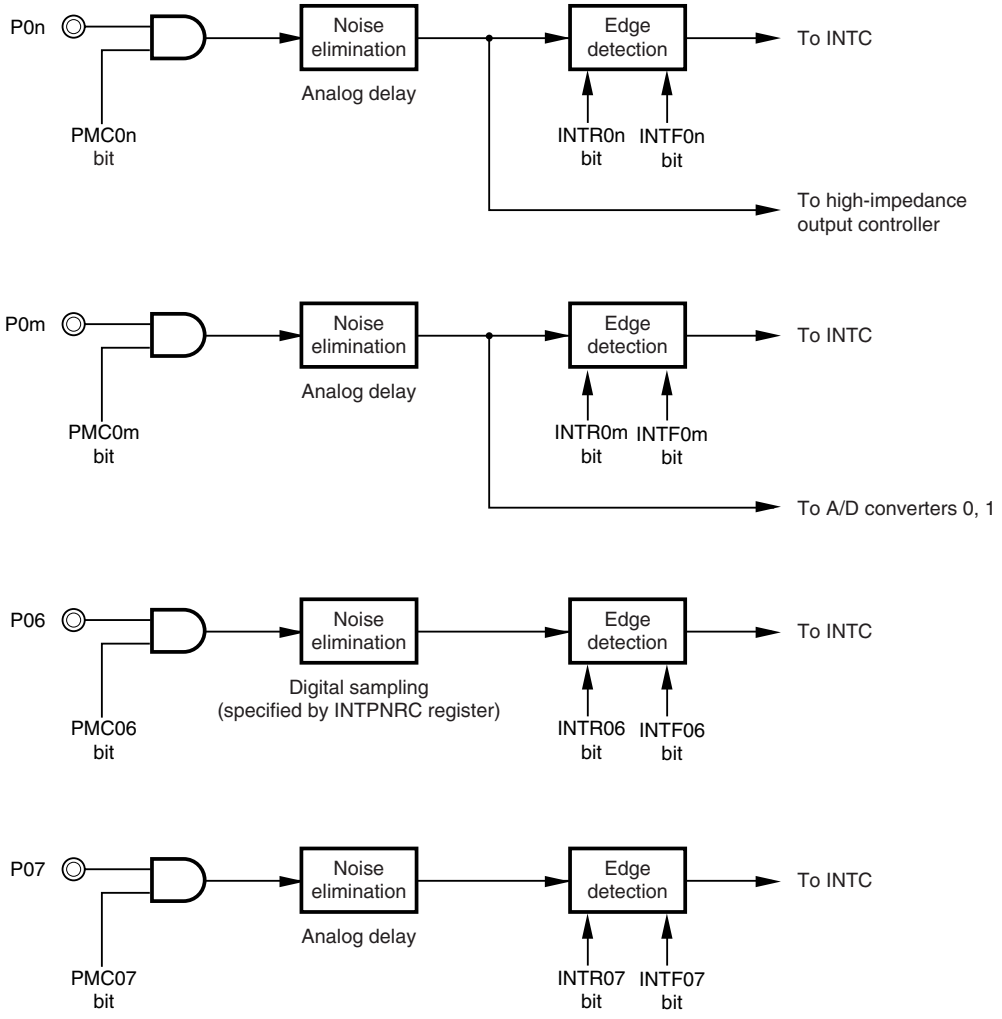
Cautions 1. To control the high-impedance output of a timer for motor control, be sure to set the PMC0.PMC0n bit to 1 and then specify the edge to be detected and enable the operation of the high-impedance output controller, because the output of the motor control timer may go into a high-impedance state if a wrong valid edge is detected (V850E/IA3: n = 0, 2, V850E/IA4: n = 0 to 3).

Cautions 2. To input an A/D trigger to A/D converter 0 or 1, be sure to set the PMC0.PMC0n bit to 1 and then specify the edge to be detected and enable the operation of A/D converter 0 or 1 because the trigger may be input if a wrong valid edge is detected (n = 4, 5).

3. To switch to external interrupt input (INTPn) from the port mode (by changing the PMC0.PMC0n bit from 0 to 1), an external interrupt may be input if a wrong valid edge is detected. Therefore, be sure to disable edge detection (INTF0.INTF0n bit = 0 and INTR0.INTR0n bit = 0), select external interrupt input (INTPn), and then specify the valid edge (n = 0 to 7).

When switching to the port mode from external interrupt input (INTPn) (PMC0n bit = 1 → 0), an edge may be detected. Be sure to disable edge detection (INTF0n bit = 0, INTR0n bit = 0), and then select the port mode.

Port 0 has a noise elimination function.



Caution To control high-impedance output of the external interrupt function and motor output control function, set the PMC0a bit to 1 (V850E/IA3: a = 0, 2, or 4 to 7, V850E/IA4: a = 0 to 7).

Remark n = 0, 2, 3 (n = 0, 2 only in PMC0n bit) (V850E/IA3)
n = 0 to 3 (V850E/IA4)
m = 4, 5

(1) Registers

(a) Port 0 register (P0)

After reset: Undefined R/W Address: FFFFF400H

	7	6	5	4	3	2	1	0
P0	P07	P06	P05	P04	P03	P02	P01[Note]	P00

P0n	Control of output data (in output mode)
0	Output 0.
1	Output 1

Note Valid only for the V850E/IA4.
With the V850E/IA3, the read value of this register is undefined.

Remark V850E/IA3: n = 0, 2 to 7
V850E/IA4: n = 0 to 7

(b) Port 0 mode register (PM0)

After reset: FFH R/W Address: FFFFF420H

	7	6	5	4	3	2	1	0
PM0	PM07	PM06	PM05	PM04	PM03	PM02	PM01[Note]	PM00

PM0n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Note Valid only for the V850E/IA4.
With the V850E/IA3, be sure to set this bit to 1.

Remark V850E/IA3: n = 0, 2 to 7
V850E/IA4: n = 0 to 7

(c) Port 0 mode control register (PMC0)

After reset: 00H R/W Address: FFFFF440H

	7	6	5	4	3	2	1	0
PMC0	PMC07	PMC06	PMC05	PMC04	PMC03	PMC02	PMC01[Note 1]	PMC00

PMC07	Specification of operating mode of P07 pin
0	I/O port
1	INTP7 input

PMC06	Specification of operating mode of P06 pin
0	I/O port
1	INTP6 input

PMC05	Specification of operating mode of P05 pin
0	I/O port
1	INTP5 input/ADTRG1 input

PMC04	Specification of operating mode of P04 pin
0	I/O port
1	INTP4 input/ADTRG0 input

PMC03	Specification of operating mode of P03 pin
0	I/O port
1	INTP3 input/TOP3OFF[Note 2] input

PMC02	Specification of operating mode of P02 pin
0	I/O port
1	INTP2 input/TOP2OFF input

PMC01	Specification of operating mode of P01 pin
0	I/O port
1	INTP1 input/TOQ1OFF input

PMC00	Specification of operating mode of P00 pin
0	I/O port
1	INTP0 input/TOQ0OFF input

Notes 1. Valid only in the V850E/IA4.
With the V850E/IA3, be sure to clear these bits to 0.

2. V850E/IA4 only

(d) Pull-up resistor option register 0 (PU0)

After reset: 00H R/W Address: FFFFFC40H

	7	6	5	4	3	2	1	0
PU0	PU07	PU06	PU05	PU04	PU03	PU02	PU01 Note 1	PU00

PU0n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect Note 2

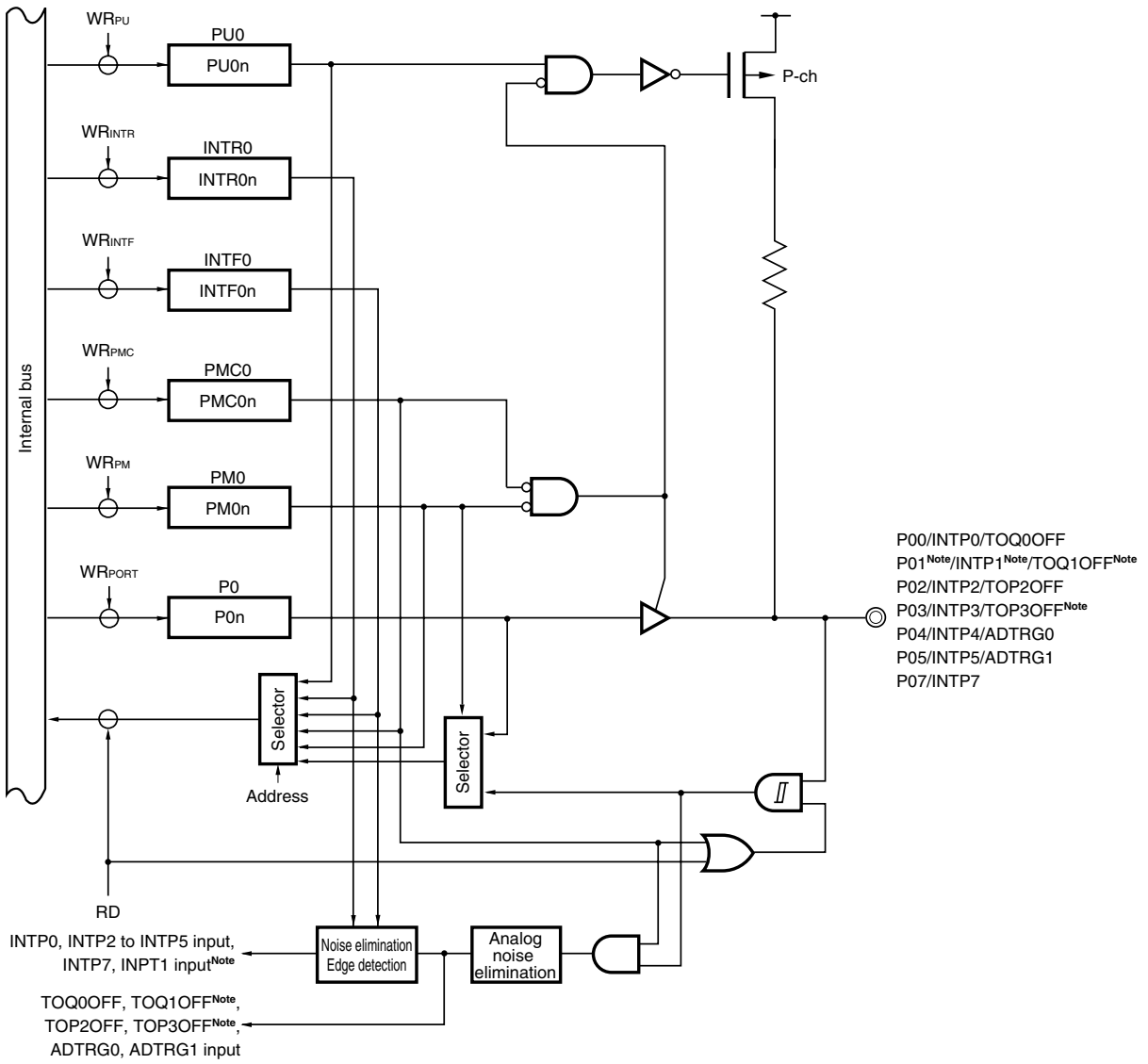
Notes 1. Valid only in the V850E/IA4.
With the V850E/IA3, be sure to clear this bit to 0.

2. An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode or when the pins function as alternate-function pins. An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark V850E/IA3: n = 0, 2 to 7
V850E/IA4: n = 0 to 7

(2) Block diagram

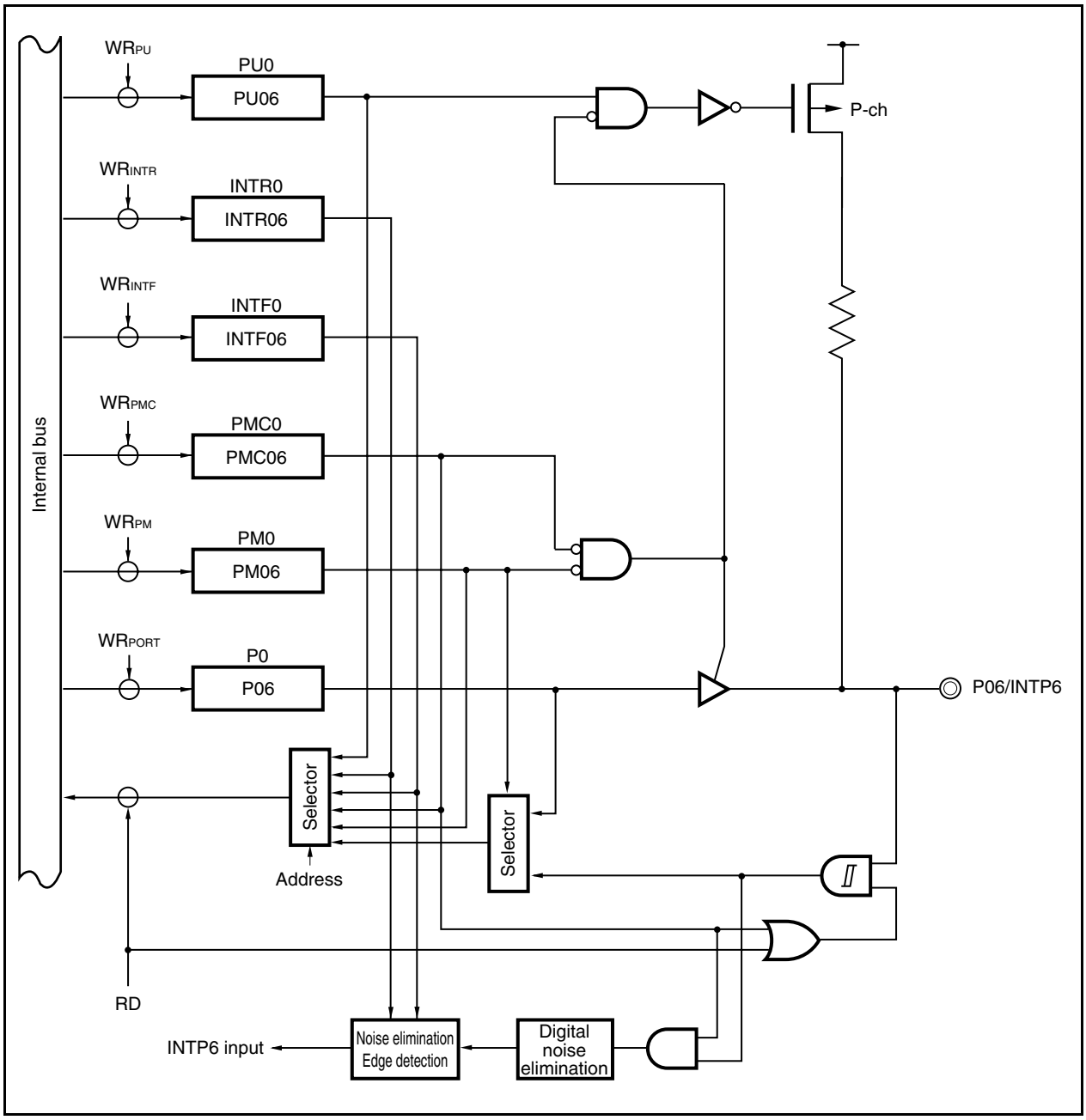
Figure 4-4. Block Diagram of P00 to P05 and P07 Pins



Note V850E/IA4 only

Remark V850E/IA3: n = 0, 2 to 5, 7
V850E/IA4: n = 0 to 5, 7

Figure 4-5. Block Diagram of P06 Pin



4.3.2 Port 1

Port 1 can be set to the input or output mode in 1-bit units.

Port 1 has an alternate function as the following pins.

Table 4-7. Alternate-Function Pins of Port 1

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note]
	IA3	IA4				
	GC	GC	GF			
P10	70	88	16	TOQ0T1/TIQ01/TOQ01	I/O	Provided
P11	71	89	17	TOQ0B1/TIQ02/TOQ02	I/O	
P12	72	90	18	TOQ0T2/TIQ03/TOQ03	I/O	
P13	75	93	21	TOQ0B2/TIQ00	I/O	
P14	76	94	22	TOQ0T3/EVTQ0	I/O	
P15	77	95	23	TOQ0B3/TRGQ0	I/O	
P16	78	96	24	TOQ00/TIP20	I/O	
P17	79	97	25	TOP21/TIP21	I/O	

Note Software pull-up function

Caution When P10 to P15 and P17 are used as TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, and TOP21, output is stopped when the following signals are asserted.

- **Output of high impedance setting signal from high impedance output controller**
- **Output of clock stop detection signal from clock monitor**

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 1 register (P1)

After reset: Undefined R/W Address: FFFFF402H

	7	6	5	4	3	2	1	0
P1	P17	P16	P15	P14	P13	P12	P11	P10

P1n	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Remark n = 0 to 7

(b) Port 1 mode register (PM1)

After reset: FFH R/W Address: FFFFF422H

	7	6	5	4	3	2	1	0
PM1	PM17	PM16	PM15	PM14	PM13	PM12	PM11	PM10

PM1n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Remark n = 0 to 7

(c) Port 1 mode control register (PMC1)

After reset: 00H R/W Address: FFFFF442H

	7	6	5	4	3	2	1	0
PMC1	PMC17	PMC16	PMC15	PMC14	PMC13	PMC12	PMC11	PMC10

PMC17	Specification of operating mode of P17 pin
0	I/O port
1	TOP21 output/TIP21 input

PMC16	Specification of operating mode of P16 pin
0	I/O port
1	TOQ00 output/TIP20 input

PMC15	Specification of operating mode of P15 pin
0	I/O port
1	TOQ0B3 output/TRGQ0 input

PMC14	Specification of operating mode of P14 pin
0	I/O port
1	TOQ0T3 output/EVTQ0 input

PMC13	Specification of operating mode of P13 pin
0	I/O port
1	TOQ0B2 output/TIQ00 input

PMC12	Specification of operating mode of P12 pin
0	I/O port
1	TOQ0T2 output/TIQ03 input/TOQ03 output

PMC11	Specification of operating mode of P11 pin
0	I/O port
1	TOQ0B1 output/TIQ02 input/TOQ02 output

PMC10	Specification of operating mode of P10 pin
0	I/O port
1	TOQ0T1 output/TIQ01 input/TOQ01 output

(d) Port 1 function control register (PFC1)

After reset: 00H R/W Address: FFFFF462H

	7	6	5	4	3	2	1	0
PFC1	PFC17	PFC16	PFC15	PFC14	PFC13	PFC12	PFC11	PFC10

Remark For the specification of alternate function, see **4.3.2 (1) (f) Setting of alternate function of port 1**.

(e) Port 1 function control function expansion register (PFCE1)

After reset: 00H R/W Address: FFFFF702H

	7	6	5	4	3	2	1	0
PFCE1	0	0	0	0	0	PFCE12	PFCE11	PFCE10

Remark For the specification of alternate function, see **4.3.2 (1) (f) Setting of alternate function of port 1**.

(f) Setting of alternate function of port 1

PFC17	Specification of Alternate Function of P17 Pin
0	TOP21 output
1	TIP21 input

PFC16	Specification of Alternate Function of P16 Pin
0	TOQ00 output
1	TIP20 input

PFC15	Specification of Alternate Function of P15 Pin
0	TOQ0B3 output
1	TRGQ0 input

PFC14	Specification of Alternate Function of P14 Pin
0	TOQ0T3 output
1	EVTQ0 input

PFC13	Specification of Alternate Function of P13 Pin
0	TOQ0B2 output
1	TIQ00 input

PFCE12	PFC12	Specification of Alternate Function of P12 Pin
0	0	TOQ0T2 output
0	1	TIQ03 input
1	0	TOQ03 output
1	1	Setting prohibited

PFCE11	PFC11	Specification of Alternate Function of P11 Pin
0	0	TOQ0B1 output
0	1	TIQ02 input
1	0	TOQ02 output
1	1	Setting prohibited

PFCE10	PFC10	Specification of Alternate Function of P10 Pin
0	0	TOQ0T1 output
0	1	TIQ01 input
1	0	TOQ01 output
1	1	Setting prohibited

(g) Pull-up resistor option register 1 (PU1)

After reset: 00H R/W Address: FFFFFC42H

	7	6	5	4	3	2	1	0
PU1	PU17	PU16	PU15	PU14	PU13	PU12	PU11	PU10

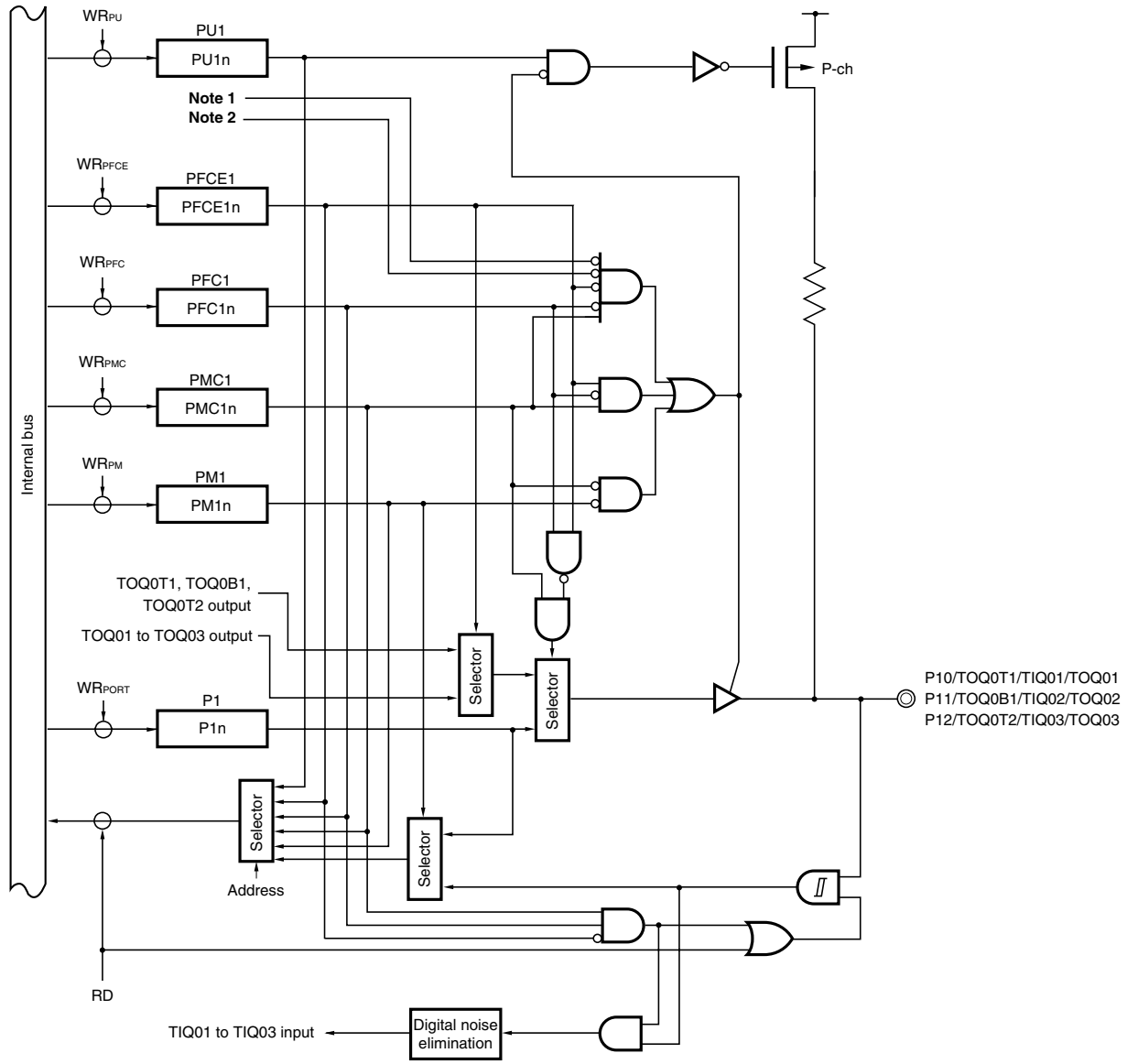
PU1n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note]

Note An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode or when the pins function as input pins in the alternate-function mode. Moreover, an on-chip pull-up resistor can be connected to the TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, and TOP21 pins, these are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ0OFF or TOP2OFF pin, or software processing. An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark n = 0 to 7

(2) Block diagram

Figure 4-6. Block Diagram of P10 to P12 Pins

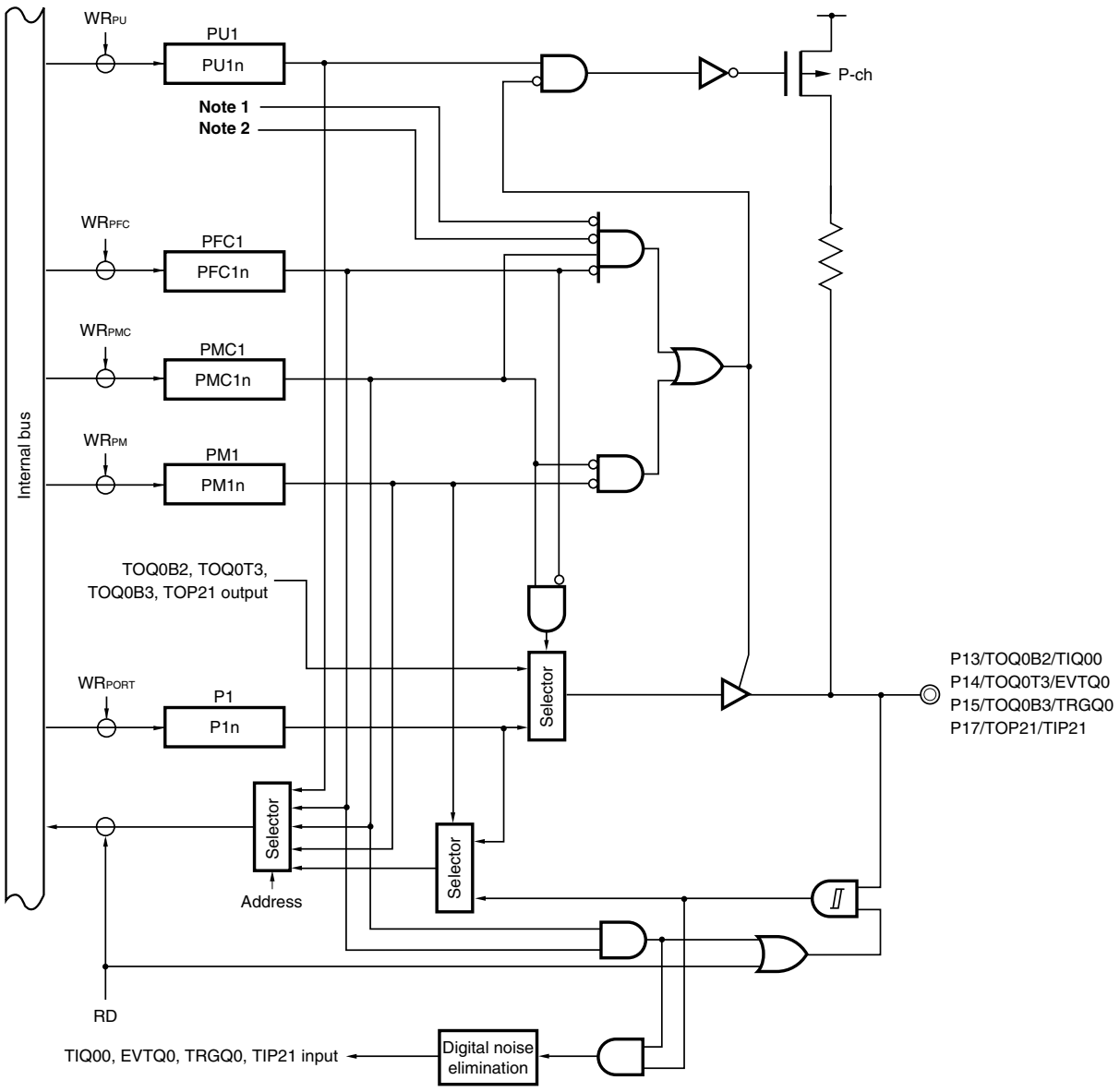


Notes 1. Output of high impedance setting signal from high impedance output controller

2. Output of clock stop detection signal from clock monitor

Remark n = 0 to 2

Figure 4-7. Block Diagram of P13 to P15 and P17 Pins

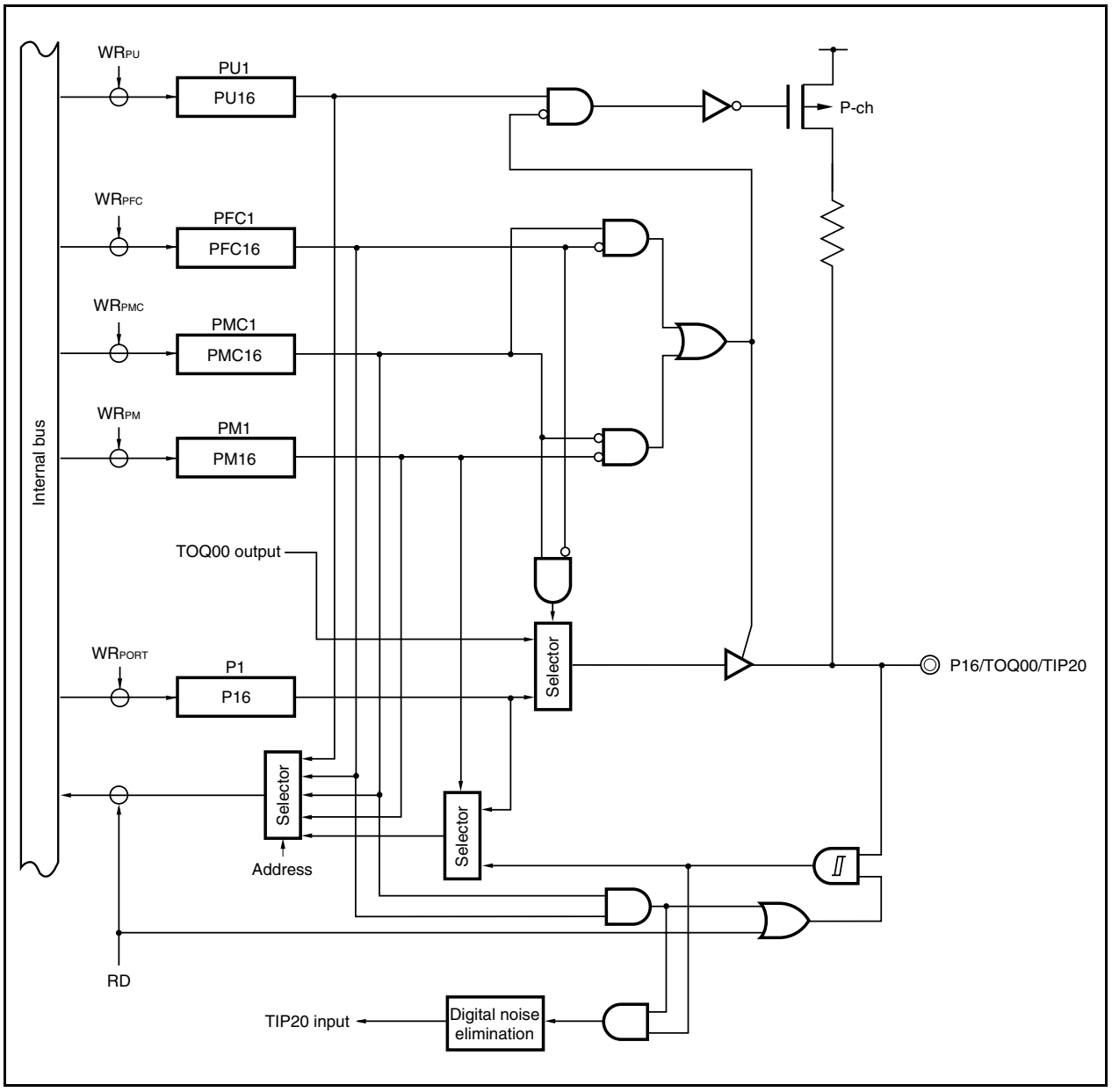


Notes 1. Output of high impedance setting signal from high impedance output controller

2. Output of clock stop detection signal from clock monitor

Remark n = 3 to 5, 7

Figure 4-8. Block Diagram of P16 Pin



4.3.3 Port 2 (V850E/IA4 only)

Port 2 can be set to the input or output mode in 1-bit units.

Port 2 has an alternate function as the following pins.

Table 4-8. Alternate-Function Pins of Port 2

Port	Pin No.		Alternate-Function Pin	I/O	Pull-up[Note]
	GC	GF			
P20	35	63	TOQ1T1	Output	Provided
P21	36	64	TOQ1B1	Output	
P22	37	65	TOQ1T2	Output	
P23	40	68	TOQ1B2	Output	
P24	41	69	TOQ1T3	Output	
P25	42	70	TOQ1B3	Output	
P26	60	88	TOQ10	Output	
P27	61	89	TOP31	Output	

Note Software pull-up function

Caution When P20 to P25 and P27 are used as TOQ1T1 to TOQ1T3, TOQ1B1 to TOQ1B3, and TOP31, output is stopped when the following signals are asserted.

- **Output of high impedance setting signal from high impedance output controller**
- **Output of clock stop detection signal from clock monitor**

Remark GC: 100-pin plastic LQFP (fine pitch) (14 × 14)
GF: 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 2 register (P2)

After reset: Undefined R/W Address: FFFFF404H

	7	6	5	4	3	2	1	0
P2	P27	P26	P25	P24	P23	P22	P21	P20

P2n	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Remark n = 0 to 7

(b) Port 2 mode register (PM2)

After reset: FFH R/W Address: FFFFF424H

	7	6	5	4	3	2	1	0
PM2	PM27	PM26	PM25	PM24	PM23	PM22	PM21	PM20

PM2n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Remark n = 0 to 7

(c) Port 2 mode control register (PMC2)

After reset: 00H R/W Address: FFFFF444H

	7	6	5	4	3	2	1	0
PMC2	PMC27	PMC26	PMC25	PMC24	PMC23	PMC22	PMC21	PMC20

PMC27	Specification of operating mode of P27 pin
0	I/O port
1	TOP31 output

PMC26	Specification of operating mode of P26 pin
0	I/O port
1	TOQ10 output

PMC25	Specification of operating mode of P25 pin
0	I/O port
1	TOQ1B3 output

PMC24	Specification of operating mode of P24 pin
0	I/O port
1	TOQ1T3 output

PMC23	Specification of operating mode of P23 pin
0	I/O port
1	TOQ1B2 output

PMC22	Specification of operating mode of P22 pin
0	I/O port
1	TOQ1T2 output

PMC21	Specification of operating mode of P21 pin
0	I/O port
1	TOQ1B1 output

PMC20	Specification of operating mode of P20 pin
0	I/O port
1	TOQ1T1 output

(d) Pull-up resistor option register 2 (PU2)

After reset: 00H R/W Address: FFFFFC44H

	7	6	5	4	3	2	1	0
PU2	PU27	PU26	PU25	PU24	PU23	PU22	PU21	PU20

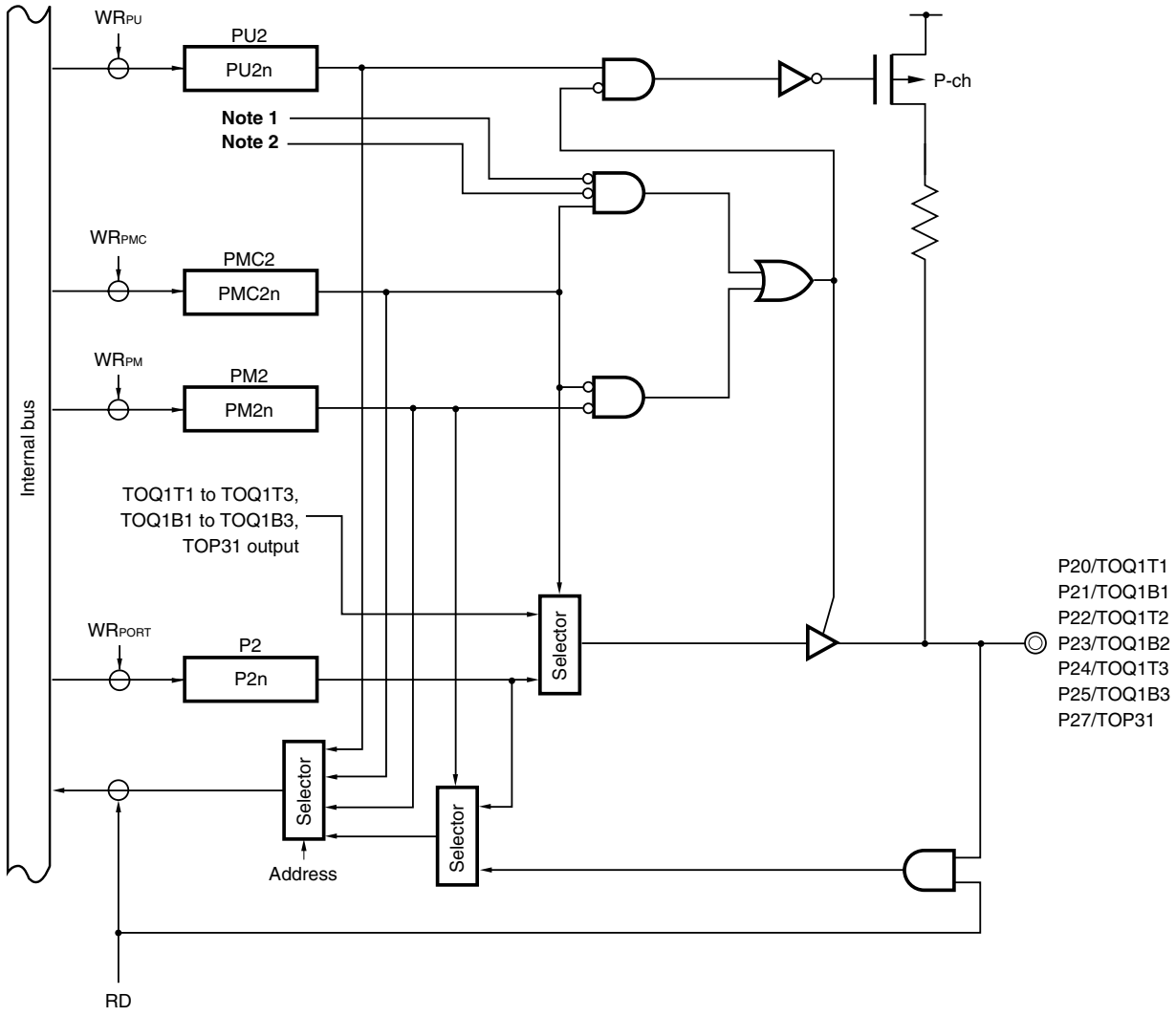
PU2n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note]

Note An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode. Moreover, an on-chip pull-up resistor can only be connected to the TOQ1T1 to TOQ1T3, TOQ1B1 to TOQ1B3, and TOP31 pins, these pins are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ1OFF or TOP3OFF pin, or software processing. An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark n = 0 to 7

(2) Block diagram

Figure 4-9. Block Diagram of P20 to P25, and P27 Pins

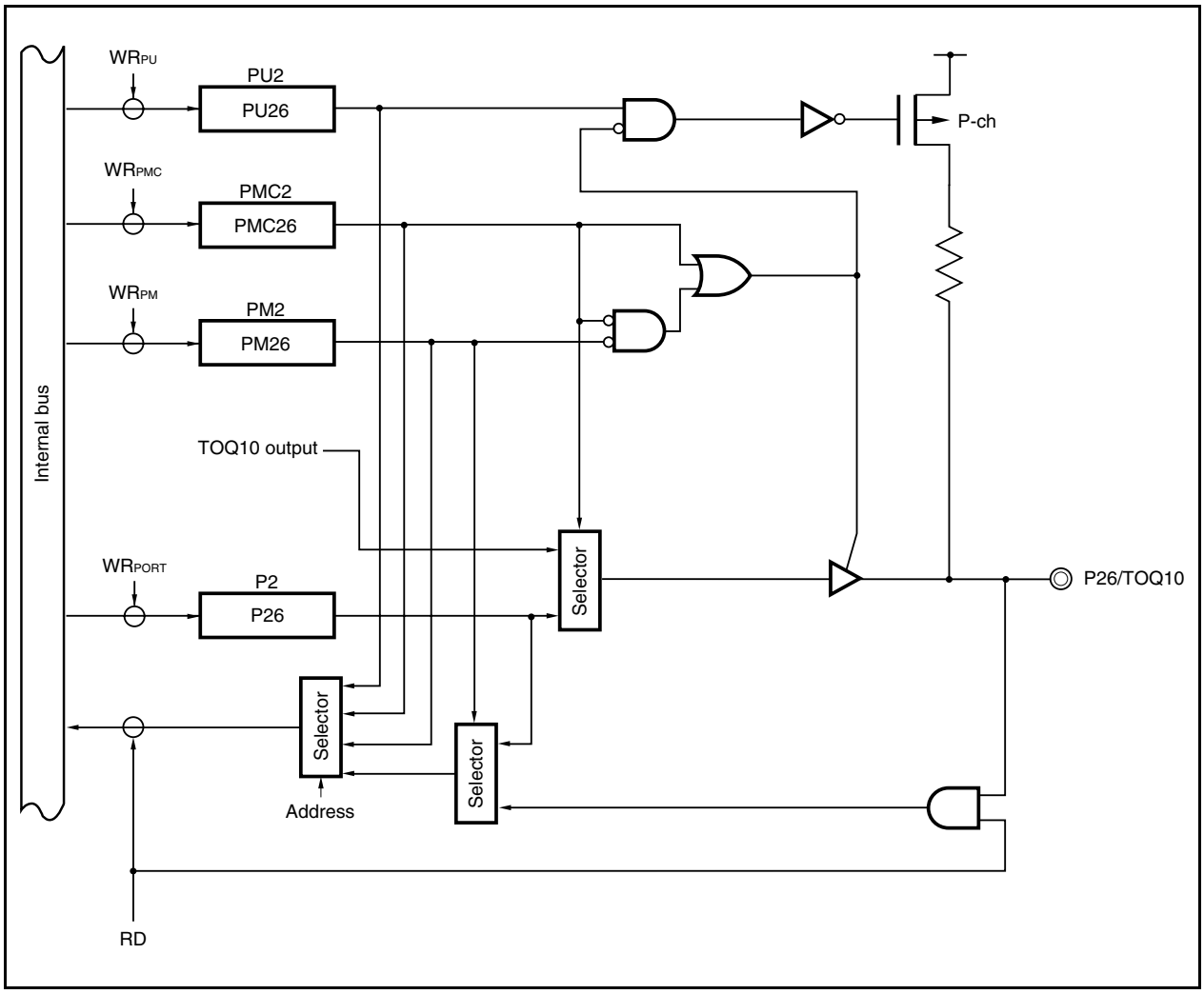


Notes 1. Output of high impedance setting signal from high impedance output controller

2. Output of clock stop detection signal from clock monitor

Remark n = 0 to 5, 7

Figure 4-10. Block Diagram of P26 Pin



4.3.4 Port 3

Port 3 can be set to the input or output mode in 1-bit units.

Port 3 has an alternate function as the following pins.

Table 4-9. Alternate-Function Pins of Port 3

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note]
	IA3	IA4				
	GC	GC	GF			
P30	41	52	80	RXDA0	Input	Provided
P31	42	53	81	TXDA0	Output	
P32	43	54	82	SIB1/RXDA1	Input	
P33	44	55	83	SOB1/TXDA1	Output	
P34	45	56	84	$\overline{\text{SCKB1}}$	I/O	
P35	46	57	85	TIUD10/TO10	I/O	
P36	47	58	86	TCUD10	Input	
P37	48	59	87	TCLR10	Input	

Note Software pull-up function

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 3 register (P3)

After reset: Undefined R/W Address: FFFFF406H

	7	6	5	4	3	2	1	0
P3	P37	P36	P35	P34	P33	P32	P31	P30

P3n	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Remark n = 0 to 7

(b) Port 3 mode register (PM3)

After reset: FFH R/W Address: FFFFF426H

	7	6	5	4	3	2	1	0
PM3	PM37	PM36	PM35	PM34	PM33	PM32	PM31	PM30

PM3n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Remark n = 0 to 7

(c) Port 3 mode control register (PMC3)

After reset: 00H R/W Address: FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

PMC37	Specification of operating mode of P37 pin
0	I/O port
1	TCLR10 input

PMC36	Specification of operating mode of P36 pin
0	I/O port
1	TCUD10 input

PMC35	Specification of operating mode of P35 pin
0	I/O port
1	TIUD10 input/TO10 output

PMC34	Specification of operating mode of P34 pin
0	I/O port
1	$\overline{\text{SCKB1}}$ I/O

PMC33	Specification of operating mode of P33 pin
0	I/O port
1	SOB1 output/TXDA1 output

PMC32	Specification of operating mode of P32 pin
0	I/O port
1	SIB1 input/RXDA1 input

PMC31	Specification of operating mode of P31 pin
0	I/O port
1	TXDA0 output

PMC30	Specification of operating mode of P30 pin
0	I/O port
1	RXDA0 input

(d) Port 3 function control register (PFC3)

After reset: 00H R/W Address: FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	0	0	PFC35	0	PFC33	PFC32	0	0

PFC35	Specification of alternate function of P35 pin
0	TIUD10 input
1	TO10 output

PFC33	Specification of alternate function of P33 pin
0	SOB1 output
1	TXDA1 output

PFC32	Specification of alternate function of P32 pin
0	SIB1 input
1	RXDA1 input

(e) Pull-up resistor option register 3 (PU3)

After reset: 00H R/W Address: FFFFFC46H

	7	6	5	4	3	2	1	0
PU3	PU37	PU36	PU35	PU34	PU33	PU32	PU31	PU30

PU3n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note]

Note An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode or <R> when the pins function as input pins in the alternate-function mode (including the slave mode of the $\overline{\text{SCKB1}}$ pin). An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark n = 0 to 7

(2) Block diagram

Figure 4-11. Block Diagram of P30 Pin

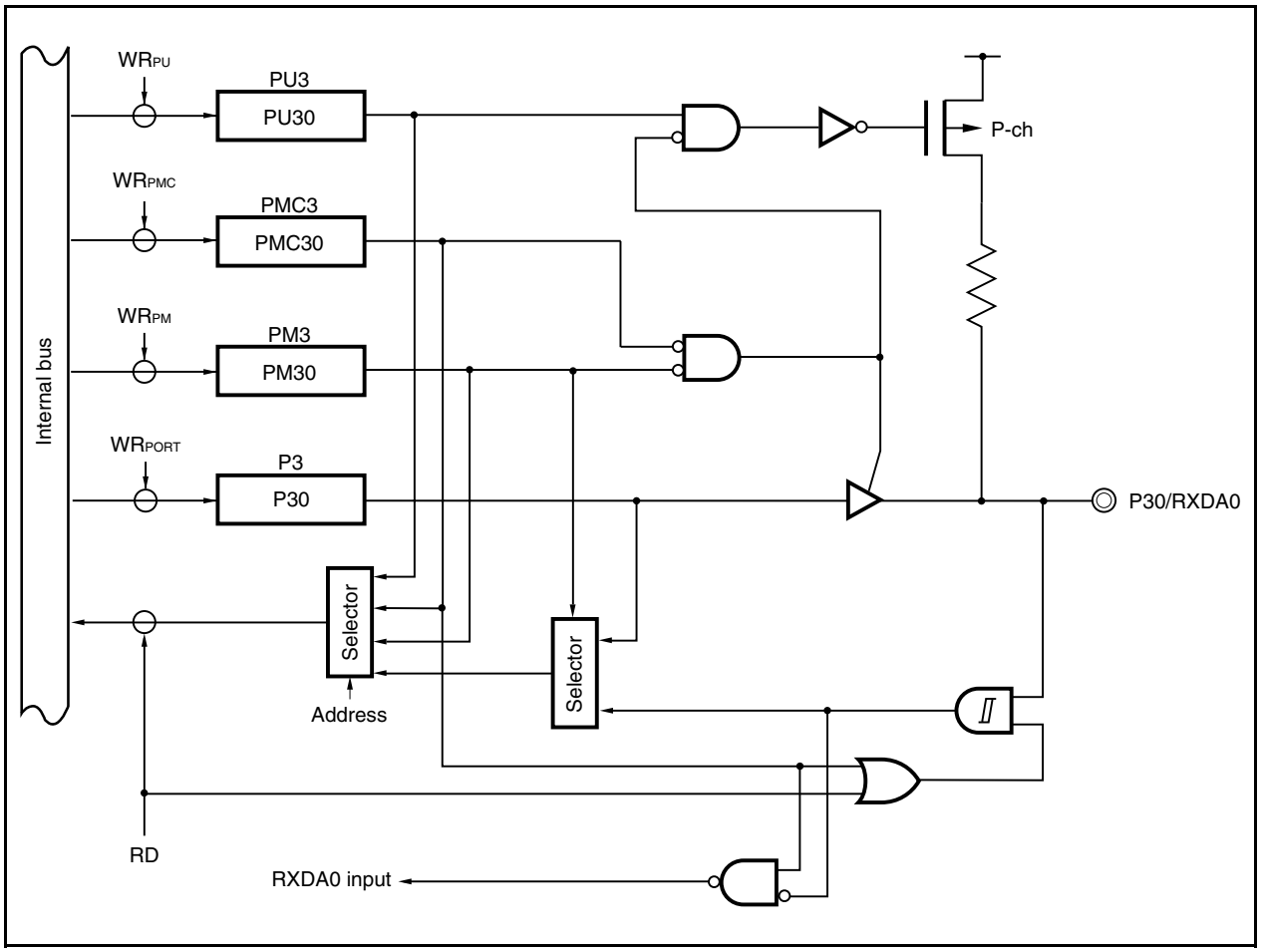


Figure 4-12. Block Diagram of P31 Pin

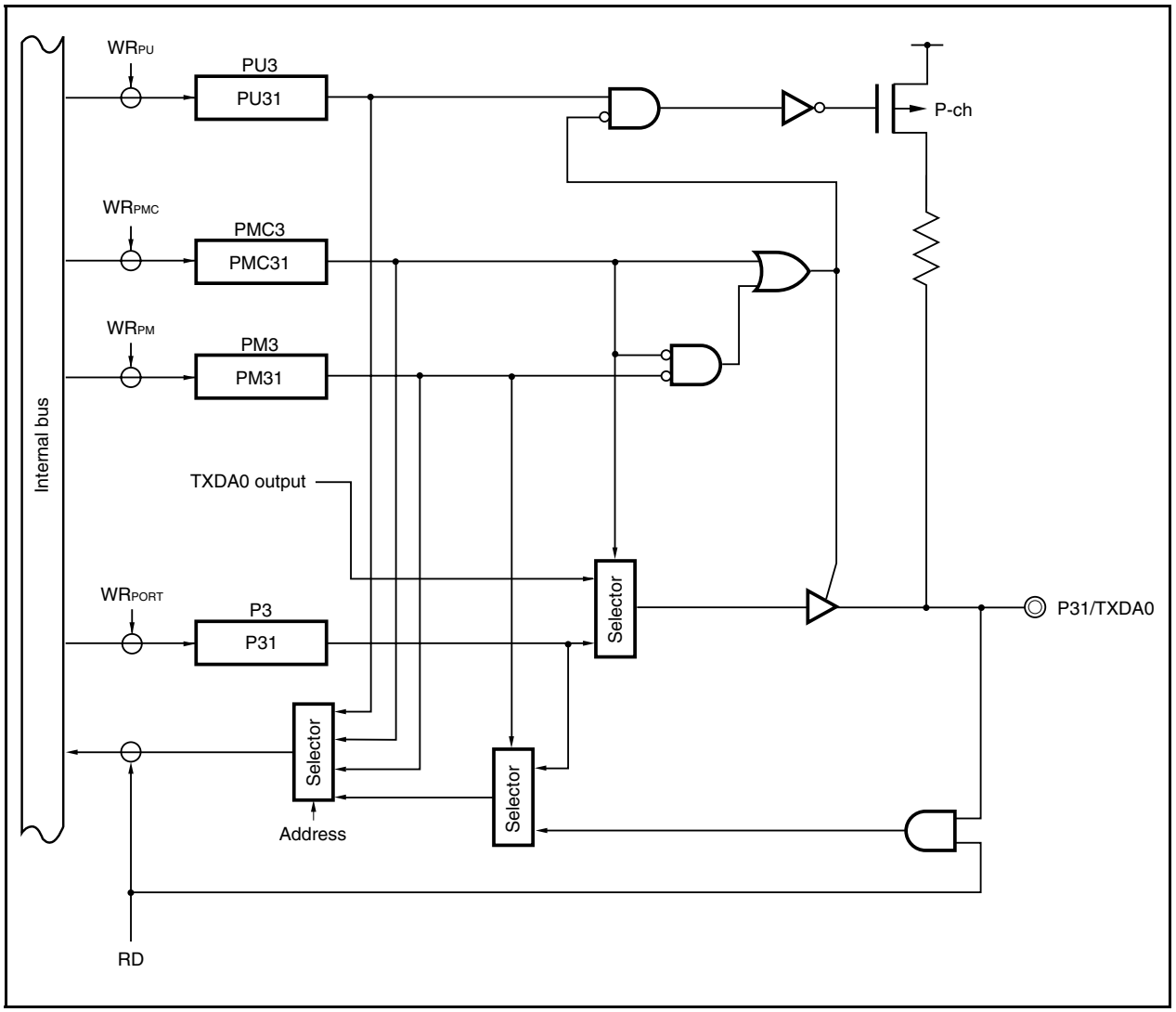


Figure 4-13. Block Diagram of P32 Pin

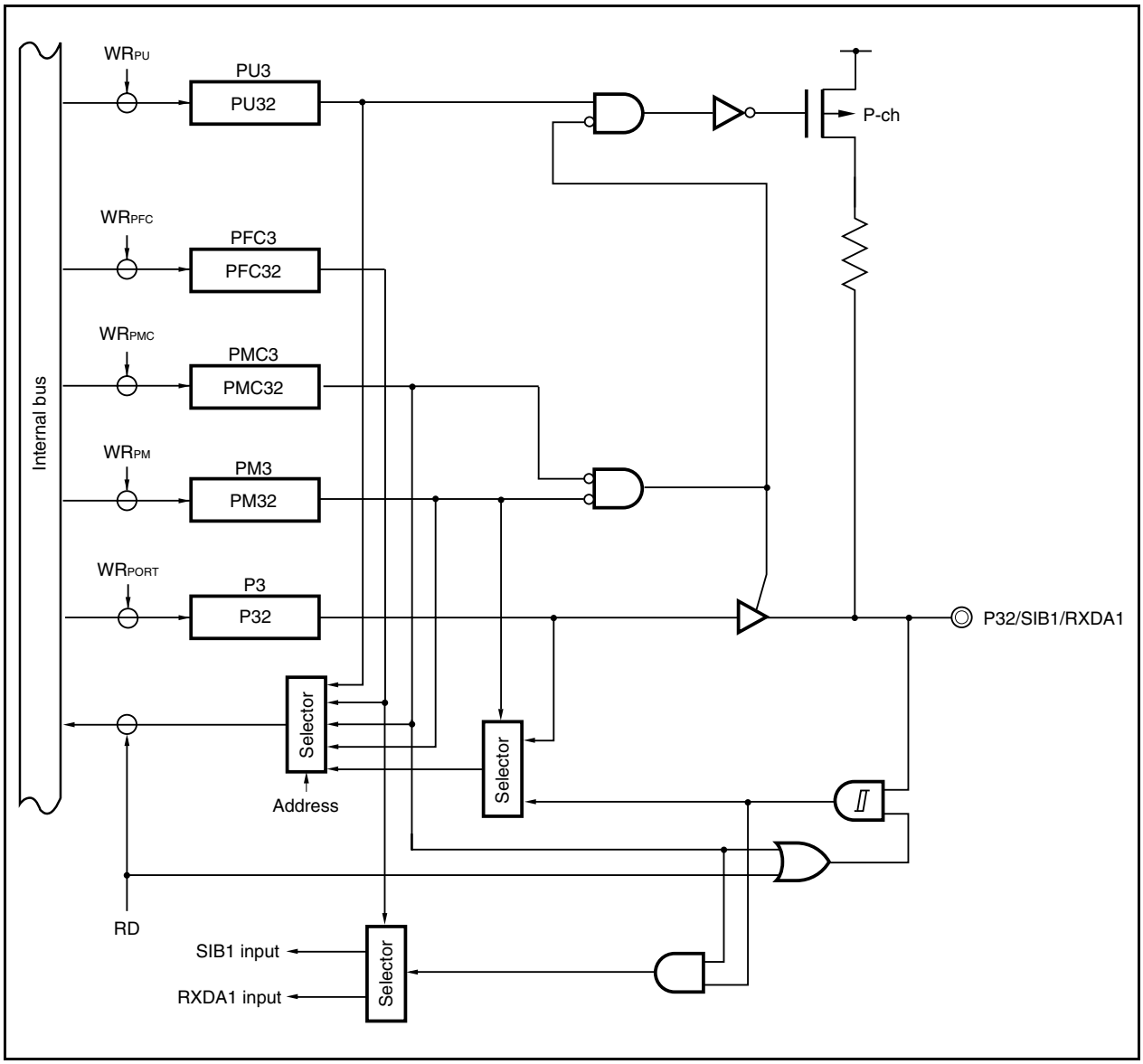


Figure 4-14. Block Diagram of P33 Pin

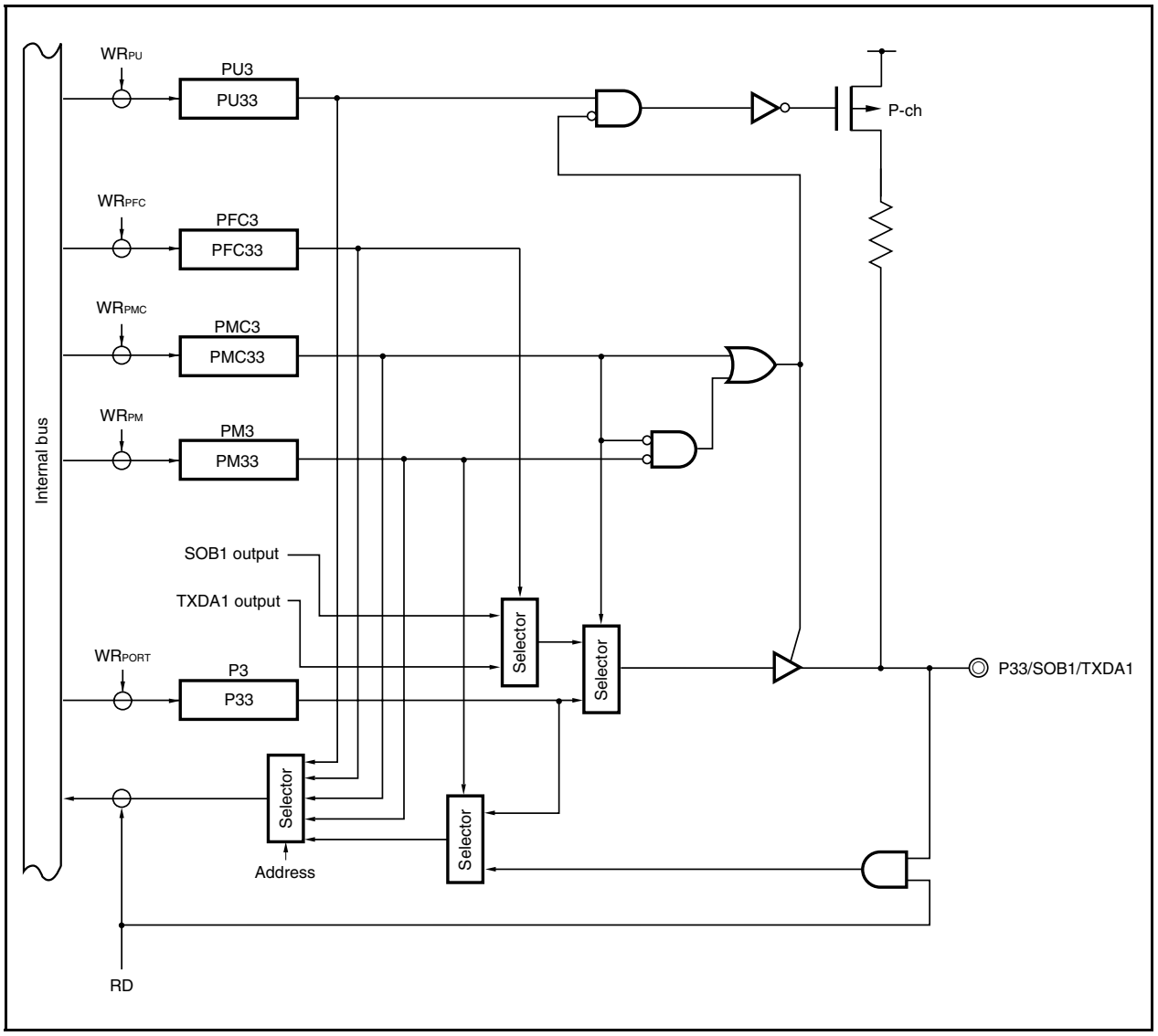


Figure 4-15. Block Diagram of P34 Pin

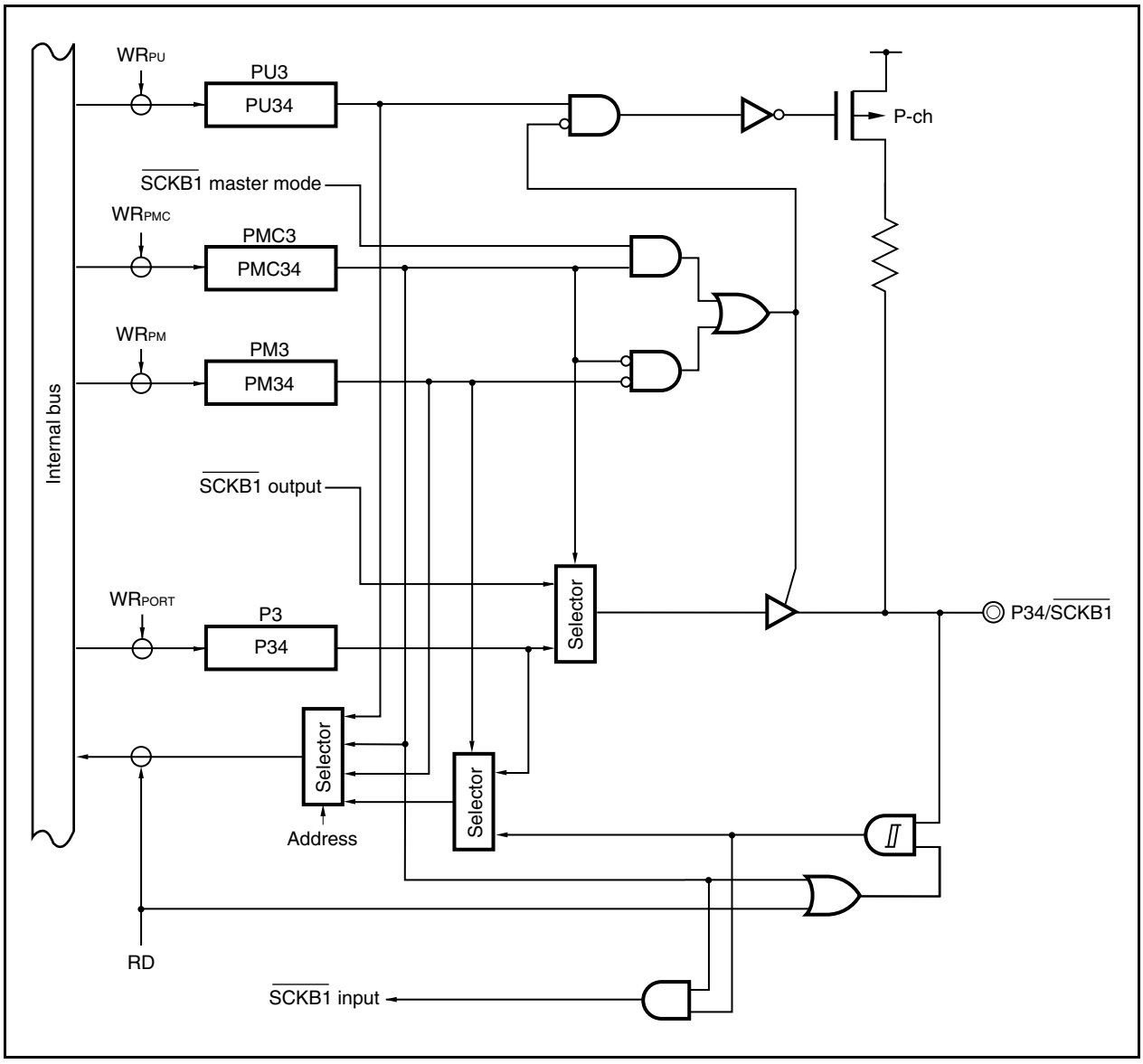


Figure 4-16. Block Diagram of P35 Pin

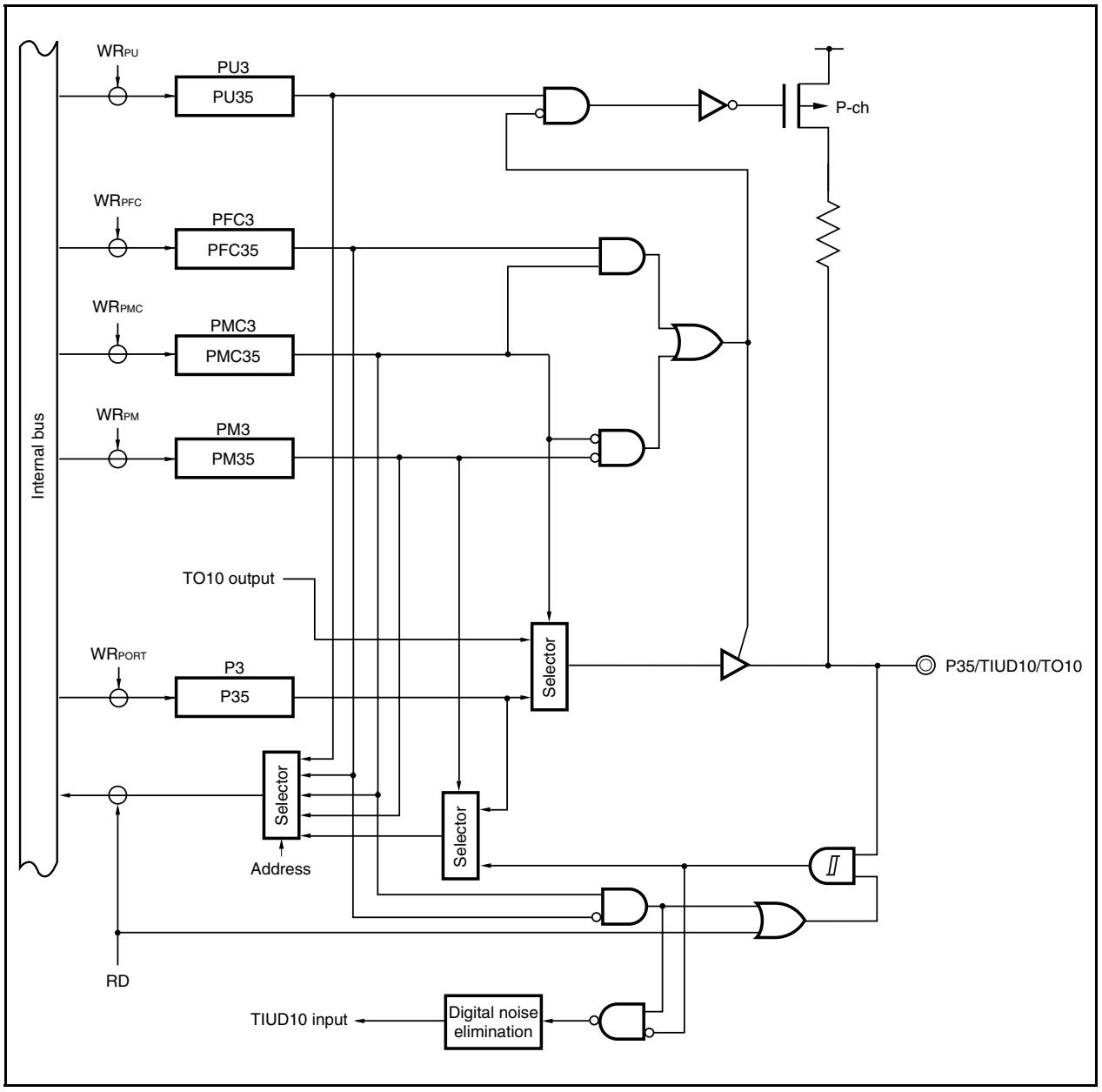
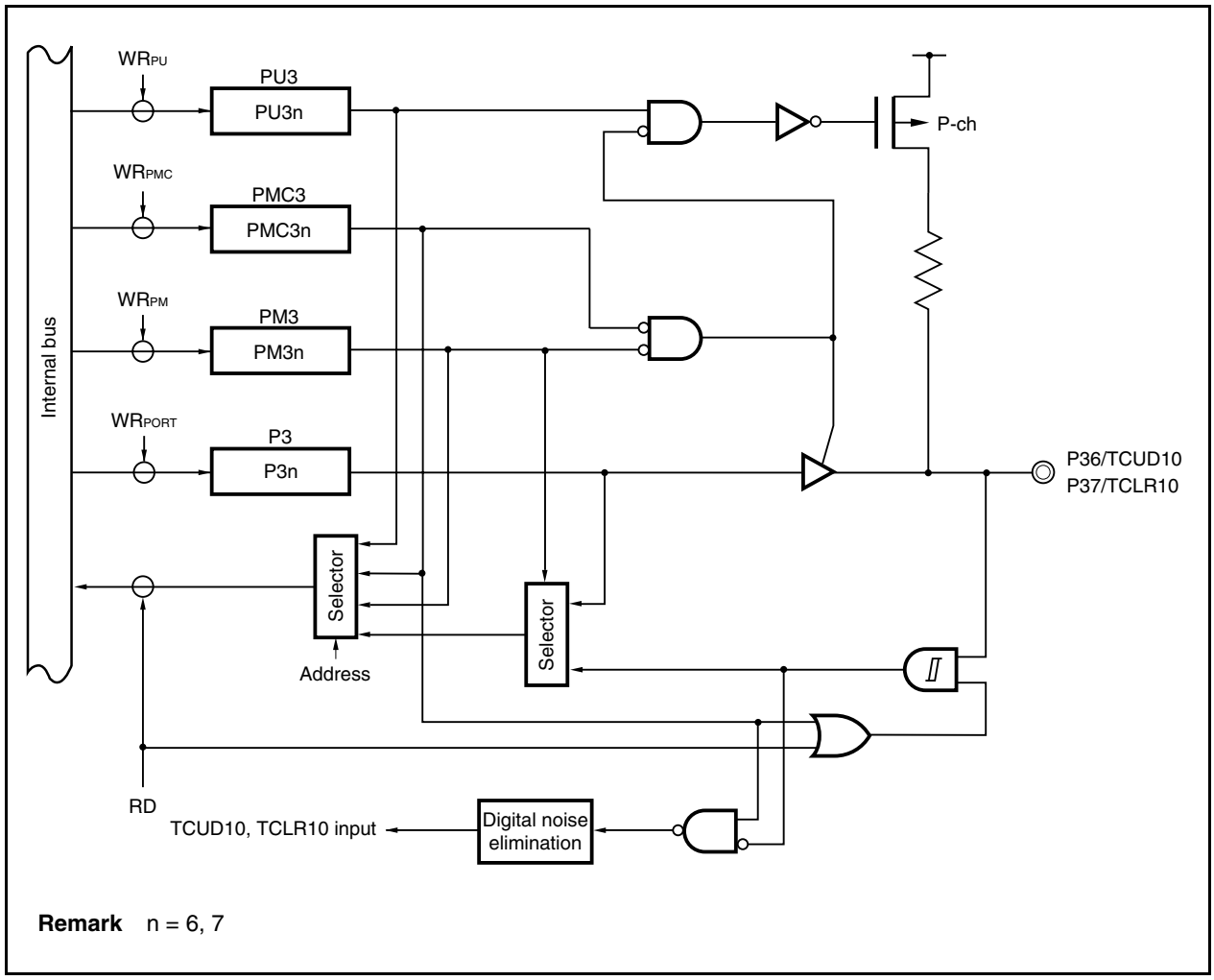


Figure 4-17. Block Diagram of P36 and P37 Pins



Remark n = 6, 7

4.3.5 Port 4

Port 4 can be set to the input or output mode in 1-bit units.

Port 4 has an alternate function as the following pins.

Table 4-10. Alternate-Function Pins of Port 4

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note]
	IA3	IA4				
	GC	GC	GF			
P40	27	32	60	SIB0	Input	Provided
P41	28	33	61	SOB0	Output	
P42	29	34	62	$\overline{\text{SCKB0}}$	I/O	
P43	37	48	76	TOP00/TIP00	I/O	
P44	38	49	77	TOP01/TIP01	I/O	

Note Software pull-up function

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 4 register (P4)

After reset: Undefined R/W Address: FFFFF408H

	7	6	5	4	3	2	1	0
P4	0	0	0	P44	P43	P42	P41	P40

P4n	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Remark n = 0 to 4

(b) Port 4 mode register (PM4)

After reset: FFH R/W Address: FFFFF428H

	7	6	5	4	3	2	1	0
PM4	1	1	1	PM44	PM43	PM42	PM41	PM40

PM4n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Remark n = 0 to 4

(c) Port 4 mode control register (PMC4)

After reset: 00H R/W Address: FFFFF448H

	7	6	5	4	3	2	1	0
PMC4	0	0	0	PMC44	PMC43	PMC42	PMC41	PMC40

PMC44	Specification of operating mode of P44 pin
0	I/O port
1	TOP01 output/TIP01 input

PMC43	Specification of operating mode of P43 pin
0	I/O port
1	TOP00 output/TIP00 input

PMC42	Specification of operating mode of P42 pin
0	I/O port
1	$\overline{\text{SCKB0}}$ I/O

PMC41	Specification of operating mode of P41 pin
0	I/O port
1	SOB0 output

PMC40	Specification of operating mode of P40 pin
0	I/O port
1	SIB0 input

(d) Port 4 function control register (PFC4)

After reset: 00H R/W Address: FFFFF468H

	7	6	5	4	3	2	1	0
PFC4	0	0	0	PFC44	PFC43	0	0	0

PFC44	Specification of alternate function of P44 pin
0	TOP01 output
1	TIP01 input

PFC43	Specification of alternate function of P43 pin
0	TOP00 output
1	TIP00 input

(e) Pull-up resistor option register 4 (PU4)

After reset: 00H R/W Address: FFFFFC48H

	7	6	5	4	3	2	1	0
PU4	0	0	0	PU44	PU43	PU42	PU41	PU40

PU4n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note]

Note An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode or when the pins function as input pins in the alternate-function mode (including when in the $\overline{\text{SCKB0}}$ pin slave mode). An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark n = 0 to 4

(2) Block diagram

Figure 4-18. Block Diagram of P40 Pin

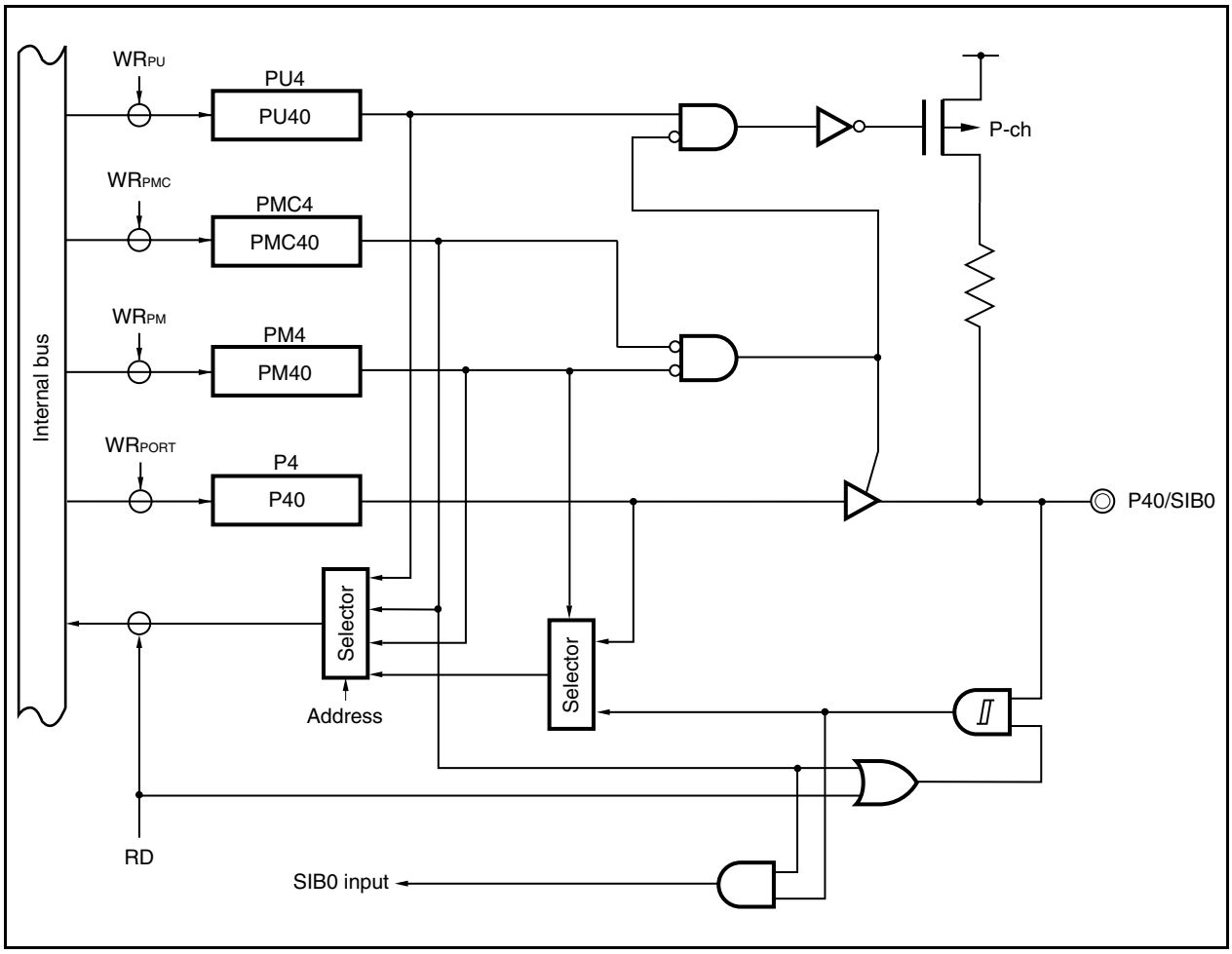


Figure 4-19. Block Diagram of P41 Pin

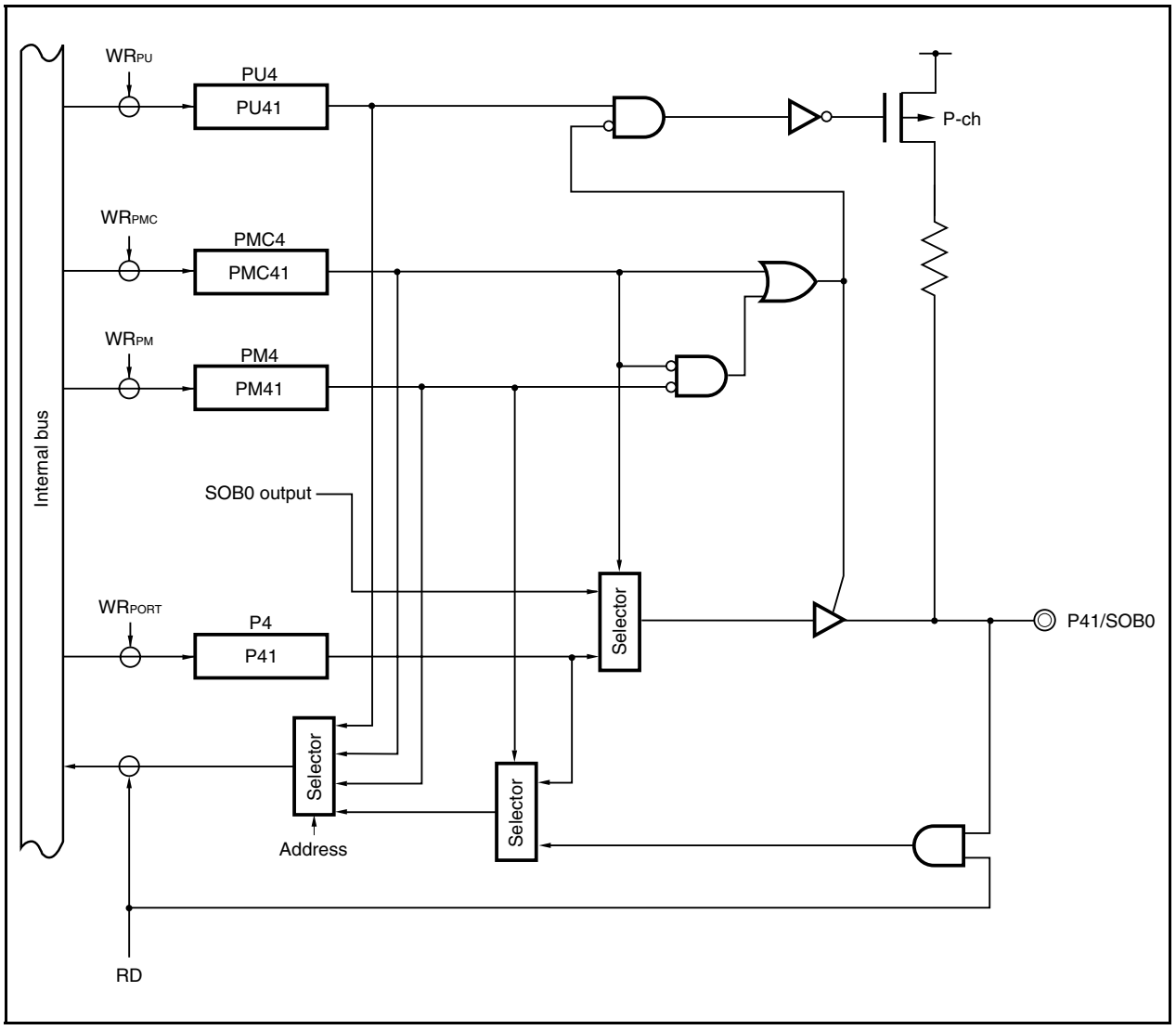


Figure 4-20. Block Diagram of P42 Pin

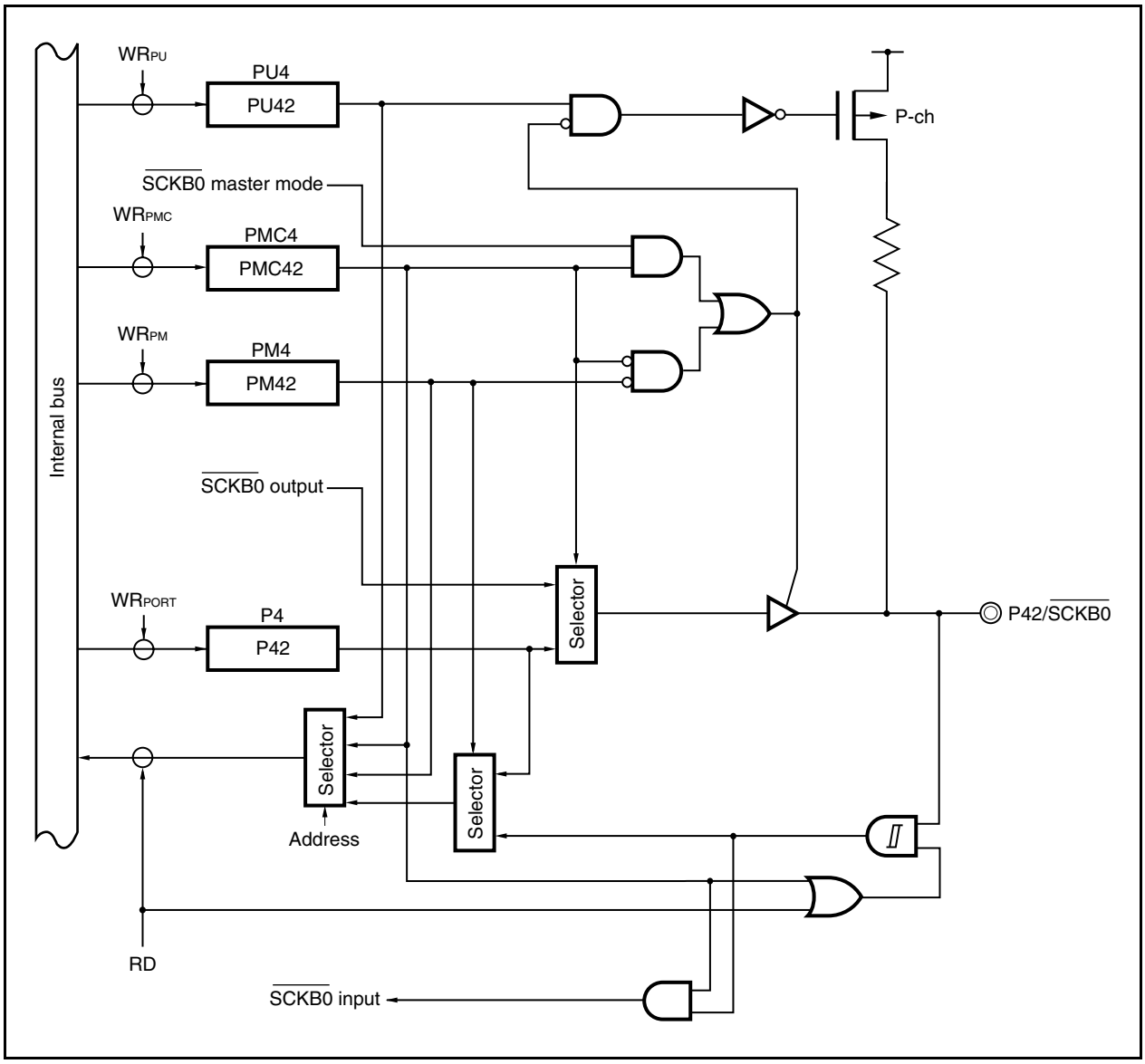
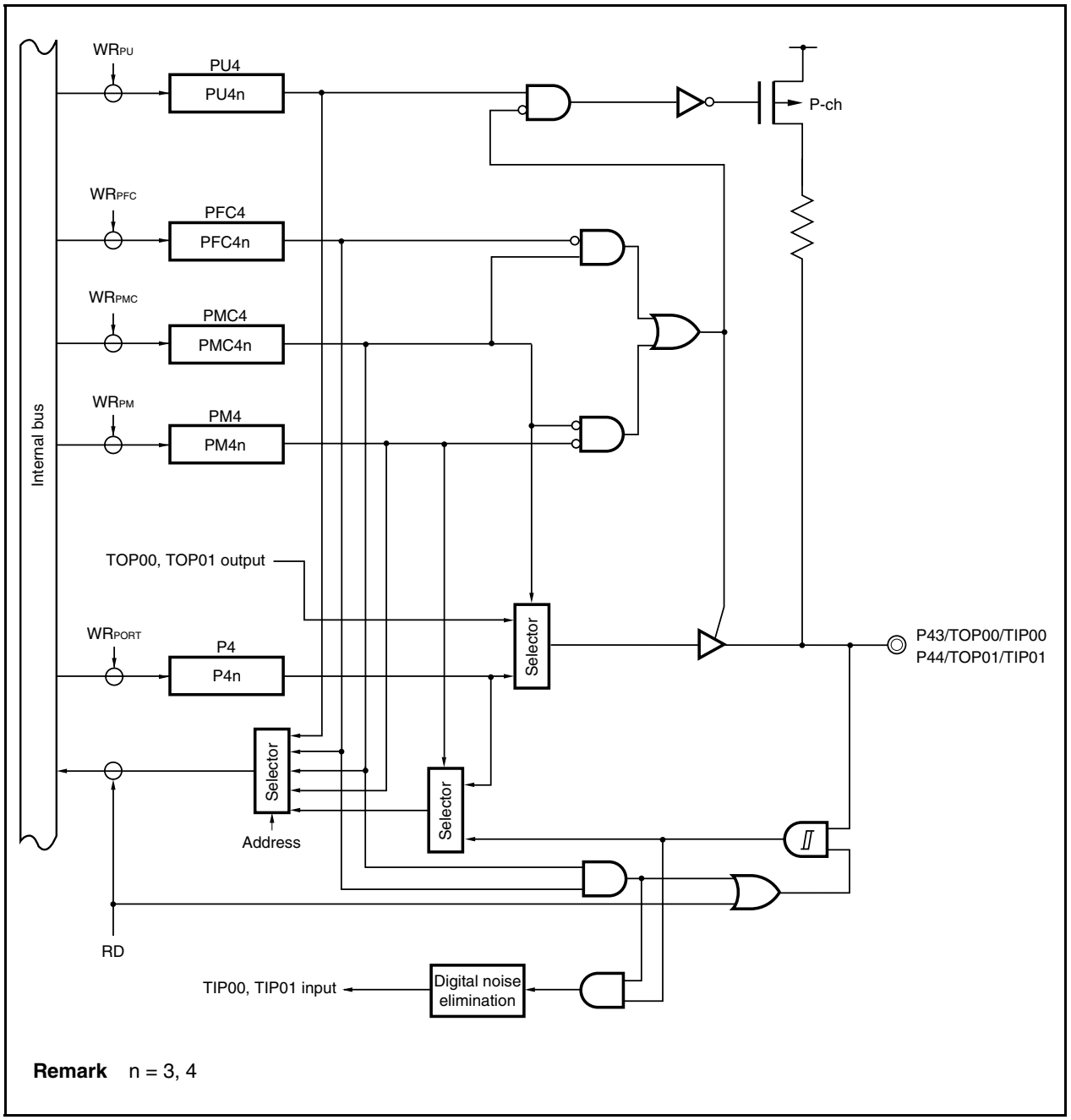


Figure 4-21. Block Diagram of P43 and P44 Pins



Remark n = 3, 4

4.3.6 Port 5 (V850E/IA4 only)

Port 5 can be set to the input or output mode in 1-bit units.

Port 5 has an alternate function as the following pins.

Table 4-11. Alternate-Function Pins of Port 5

Port	Pin No.		Alternate-Function Pin	I/O	Pull-up[Note 1]
	GC	GF			
P50	82	10	DDI[Notes 2, 3]/TIUD11/TO11	I/O	Provided
P51	83	11	DCK[Notes 2, 3]/TCUD11	Input	
P52	84	12	DMS[Notes 2, 3]/TCLR11	Input	

Notes 1. Software pull-up function

2 μPD70F3186 only

3. The P50 to P52 pins also function as on-chip debug pins. The on-chip debug function or port function (including the alternate functions) can be selected by using the level of the $\overline{\text{DRST}}$ pin, as shown in the table below.

Port 5 Functions	
$\overline{\text{DRST}}$ Pin Low-Level Input	$\overline{\text{DRST}}$ Pin High-Level Input
P50/TIUD11/TO11	DDI
P51/TCUD11	DCK
P52/TCLR11	DMS

Remark GC: 100-pin plastic LQFP (fine pitch) (14 × 14)
GF: 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 5 register (P5)

After reset: Undefined R/W Address: FFFFF40AH

	7	6	5	4	3	2	1	0
P5	Undefined	0	0	0	0	P52	P51	P50

P5n	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Remark n = 0 to 2

(b) Port 5 mode register (PM5)

After reset: FFH R/W Address: FFFFF42AH

	7	6	5	4	3	2	1	0
PM5	1	1	1	1	1	PM52	PM51	PM50

PM5n	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Remark n = 0 to 2

(c) Port 5 mode control register (PMC5)

After reset: 00H R/W Address: FFFFF44AH

	7	6	5	4	3	2	1	0
PMC5	0	0	0	0	0	PMC52	PMC51	PMC50

PMC52	Specification of operating mode of P52 pin
0	I/O port
1	TCLR11 input

PMC51	Specification of operating mode of P51 pin
0	I/O port
1	TCUD11 input

PMC50	Specification of operating mode of P50 pin
0	I/O port
1	TIUD11 input/TO11 output

(d) Port 5 function control register (PFC5)

After reset: 00H R/W Address: FFFFF46AH

	7	6	5	4	3	2	1	0
PFC5	0	0	0	0	0	0	0	PFC50

PFC50	Specification of alternate function of P50 pin
0	TIUD11 input
1	TO11 output

(e) Pull-up resistor option register 5 (PU5)

After reset: 00H R/W Address: FFFFFC4AH

	7	6	5	4	3	2	1	0
PU5	0	0	0	0	0	PU52	PU51	PU50

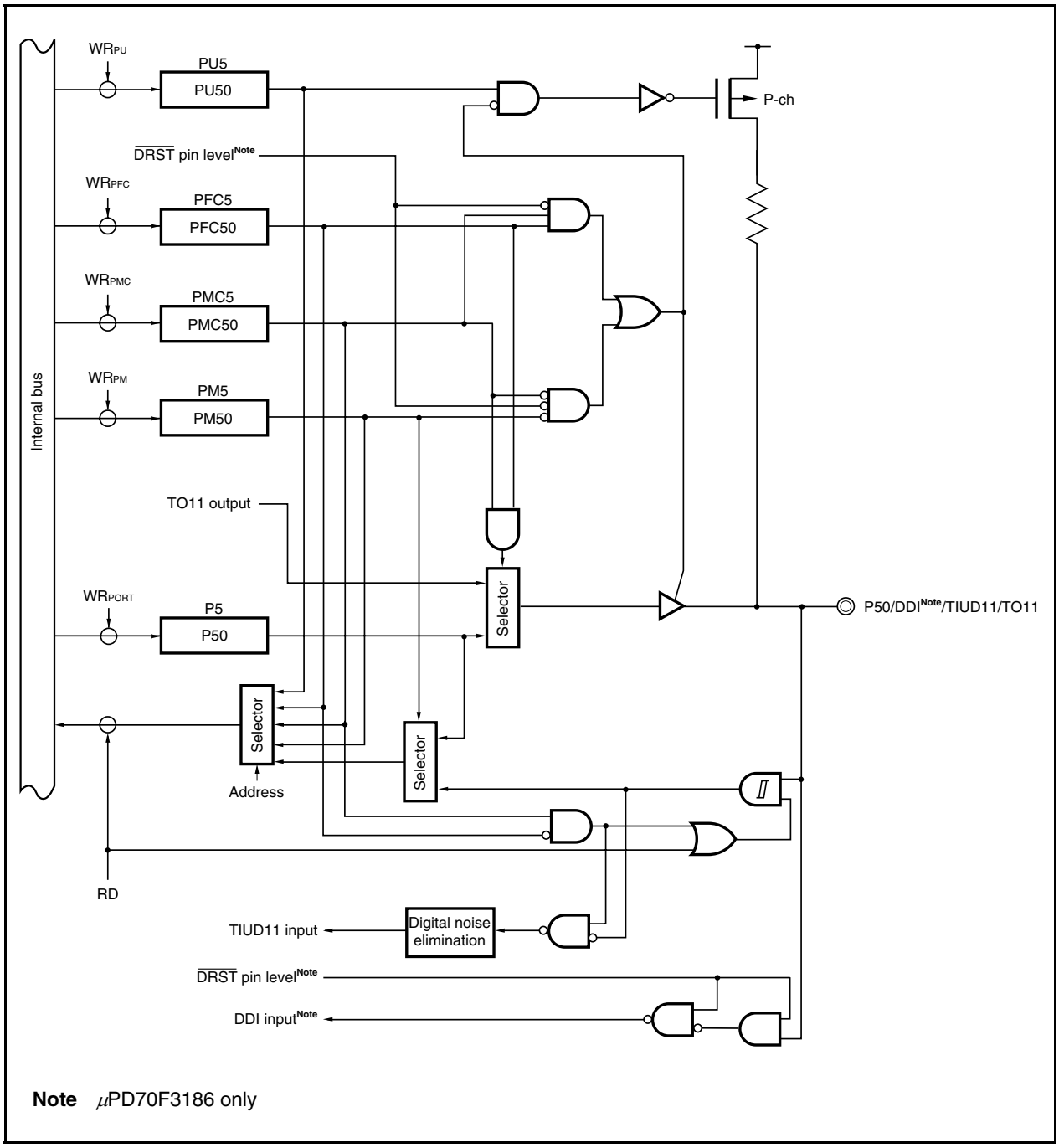
PU5n	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note]

Note An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode or when the pins function as input pins in the alternate-function mode. An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remark n = 0 to 2

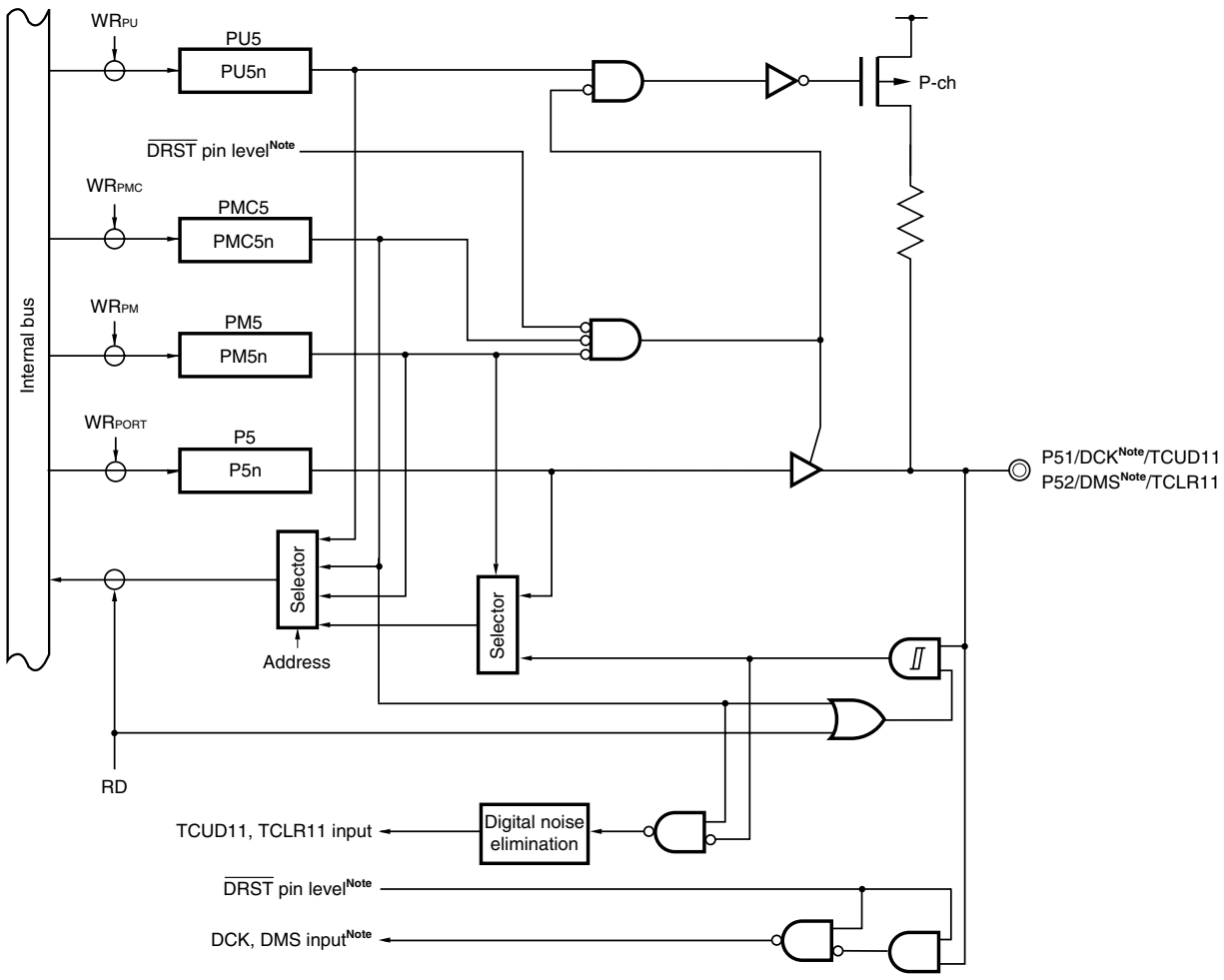
(2) Block diagram

Figure 4-22. Block Diagram of P50 Pin



Note μPD70F3186 only

Figure 4-23. Block Diagram of P51 and P52 Pins



Note μPD70F3186 only

Remark n = 1, 2

4.3.7 Port 7

Port 7 is an input port with all its pins fixed to the input mode.

The number of input port pins differs depending on the product.

Commercial Name	Number of Input-Only Port Pins
V850E/IA3	6-bit input-only port
V850E/IA4	8-bit input-only port

Port 7 has an alternate function as the following pins.

Table 4-12 Alternate-Function Pins of Port 7

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note 1]
	IA3	IA4				
	GC	GC	GF			
P70	3	5	33	ANI20	Input	None
P71	4	6	34	ANI21	Input	
P72	5	7	35	ANI22	Input	
P73	6	8	36	ANI23	Input	
P74	12	14	42	ANI24	Input	
P75	13	15	43	ANI25	Input	
P76[Note 2]	–	16	44	ANI26[Note 2]	Input	
P77[Note 2]	–	17	45	ANI27[Note 2]	Input	

Notes 1. Software pull-up function

2. V850E/IA4 only

Remark IA3: V850E/IA3

IA4: V850E/IA4

GC (V850E/IA3): 80-pin plastic QFP (14 × 14)

GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)

GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port 7 register (P7)

After reset: Undefined R Address: FFFFF40EH

	7	6	5	4	3	2	1	0
P7	P77Note	P76Note	P75	P74	P73	P72	P71	P70

P7n	Control of input data
0	Input low level.
1	Input high level.

Note Valid only in the V850E/IA4.
With the V850E/IA3, the read value of this register is undefined.

Caution When using a port input pin and analog input pin (ANI2n) together, be sure to set (1) the bit (PMC7n) of the PMC7 register to be used as the ANI2n pin.

Remark V850E/IA3: n = 0 to 5
V850E/IA4: n = 0 to 7

(b) Port 7 mode control register (PMC7)

After reset: 00H R/W Address: FFFFF44EH

	7	6	5	4	3	2	1	0
PMC7	PMC77Note	PMC76Note	PMC75	PMC74	PMC73	PMC72	PMC71	PMC70

PMC7n	Specification of operating mode of P7n pin
0	Input port (reading P7n enabled. Input buffer is on when this bit is read)
1	ANI2n input (reading P7n disabled. Input buffer is off when this bit is read)

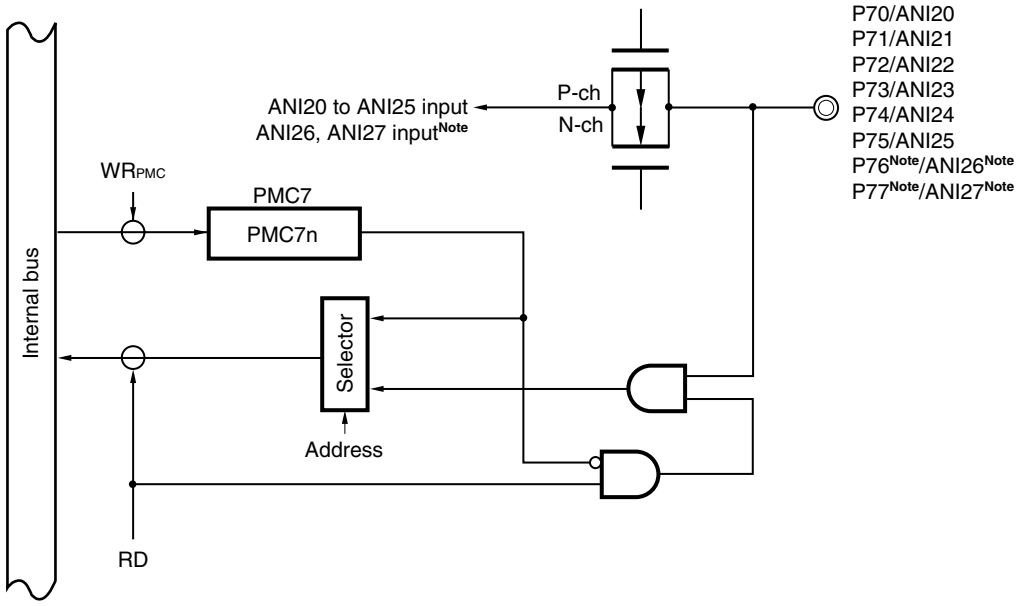
Note Valid only in the V850E/IA4.
With the V850E/IA3, be sure to clear these bits to 0.

Cautions 1. Do not change to the port mode using A/D converter 2 during A/D conversion.
2. The PMC7 register enables or disables reading of the P7 register. When the PMC7n bit = 1, the input buffer does not turn on even when the P7 register is read. In this case, the read value of the P7n bit is fixed to the low level (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7). This is to prevent through-current that may flow when the ANI2n input (intermediate level) is read.

Remark V850E/IA3: n = 0 to 5
V850E/IA4: n = 0 to 7

(2) Block diagram

Figure 4-24. Block Diagram of P70 to P77 Pins



Note V850E/IA4 only

Remark V850E/IA3: n = 0 to 5
V850E/IA4: n = 0 to 7

4.3.8 Port DL

Port DL can be set to the input or output mode in 1-bit units.

Port DL has an alternate function as the following pins.

Table 4-13. Alternate-Function Pins of Port DL

Port	Pin No.			Alternate-Function Pin	I/O	Pull-up[Note 1]
	IA3	IA4				
	GC	GC	GF			
PDL0	49	62	90	–	–	Provided
PDL1	50	63	91	–	–	
PDL2	53	66	94	–	–	
PDL3	54	67	95	–	–	
PDL4	55	68	96	–	–	
PDL5	56	69	97	FLMD1[Notes 2, 3]	Input	
PDL6	57	70	98	–	–	
PDL7	58	71	99	–	–	
PDL8	59	74	2	–	–	
PDL9	60	75	3	–	–	
PDL10	61	76	4	–	–	
PDL11	62	77	5	–	–	
PDL12	63	78	6	–	–	
PDL13	64	79	7	–	–	
PDL14	65	80	8	–	–	
PDL15	66	81	9	–	–	

Notes 1. Software pull-up function

2. This pin is used in the flash programming mode and does not have to be manipulated by a port control register. For details, see **CHAPTER 22 FLASH MEMORY**.

3. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(1) Registers

(a) Port DL register (PDL)

After reset: Undefined R/W Address: PDL FFFFF004H
PDLL FFFFF004H, PDLH FFFFF005H

	15	14	13	12	11	10	9	8
PDL (PDLH[Note])	PDL15	PDL14	PDL13	PDL12	PDL11	PDL10	PDL9	PDL8

	7	6	5	4	3	2	1	0
(PDLL)	PDL7	PDL6	PDL5	PDL4	PDL3	PDL2	PDL1	PDL0

PDLn	Control of output data (in output mode)
0	Output 0.
1	Output 1.

Note To read/write bits 8 to 15 of the PDL register in 8-bit or 1-bit units, specify them as bits 0 to 7 of the PDLH register.

Remarks 1. The PDL register can be read or written in 16-bit units.
When the higher 8 bits of the PDL register are used as the PDLH register, and the lower 8 bits, as the PDLL register, these registers can be read or written in 8-bit or 1-bit units.
2. n = 0 to 15

(b) Port DL mode register (PMDL)

After reset: FFFFH R/W Address: PMDL FFFFF024H
PMDLL FFFFF024H, PMDLH FFFFF025H

	15	14	13	12	11	10	9	8
PMDL (PMDLH[Note])	PMDL15	PMDL14	PMDL13	PMDL12	PDAL11	PDAL10	PMDL9	PMDL8

	7	6	5	4	3	2	1	0
(PMDLL)	PMDL7	PMDL6	PMDL5	PMDL4	PMDL3	PMDL2	PMDL1	PMDL0

PMDLn	Control of input/output mode (in port mode)
0	Output mode
1	Input mode

Note To read/write bits 8 to 15 of the PMDL register in 8-bit or 1-bit units, specify them as bits 0 to 7 of the PMDLH register.

Remarks 1. The PMDL register can be read or written in 16-bit units.
When the higher 8 bits of the PMDL register are used as the PMDLH register, and the lower 8 bits, as the PMDLL register, these registers can be read or written in 8-bit or 1-bit units.
2. n = 0 to 15

(c) Pull-up resistor option register DL (PUDL)

After reset: 0000H R/W Address: PUDL FFFFFF44H
PUDLL FFFFFF44H, PUDLH FFFFFF45H

	15	14	13	12	11	10	9	8
PUDL (PUDLH[Note 1])	PUDL15	PUDL14	PUDL13	PUDL12	PUDL11	PUDL10	PUDL9	PUDL8

	7	6	5	4	3	2	1	0
(PUDLL)	PUDL7	PUDL6	PUDL5	PUDL4	PUDL3	PUDL2	PUDL1	PUDL0

PUDLn	Control of on-chip pull-up resistor connection
0	Do not connect
1	Connect[Note 2]

Notes 1. To read/write bits 8 to 15 of the PUDL register in 8-bit or 1-bit units, specify them as bits 0 to 7 of the PUDLH register.

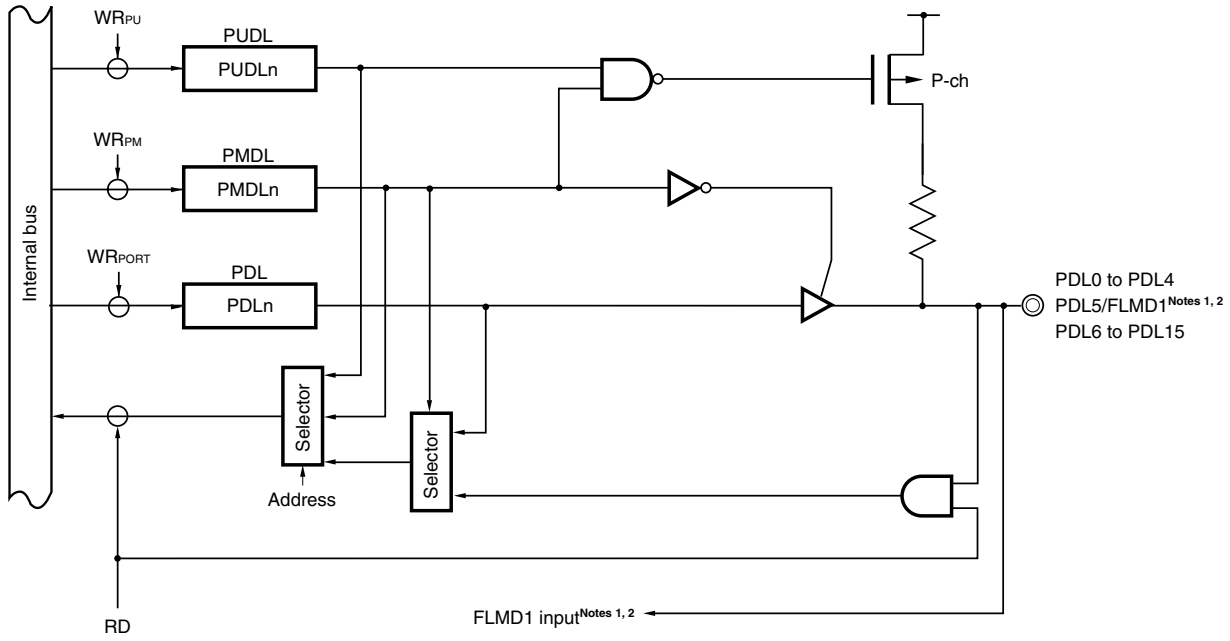
2. An on-chip pull-up resistor can be connected only when the pins are in input mode in the port mode. An on-chip pull-up resistor cannot be connected when the pins are in output mode.

Remarks 1. The PUDL register can be read or written in 16-bit units.
When the higher 8 bits of the PUDL register are used as the PUDLH register, and the lower 8 bits, as the PUDLL register, these registers can be read or written in 8-bit or 1-bit units.

2. n = 0 to 15

(2) Block diagram

Figure 4-25. Block Diagram of PDL0 to PDL15 Pins



Notes 1. This pin is used in the flash programming mode and does not have to be manipulated by a port control register. For details, see **CHAPTER 22 FLASH MEMORY**.

2. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

4.4 Output Data and Port Read Value for Each Setting

Table 4-14 shows the values used to select the alternate function of the respective pins, output data and port read values for each setting. In addition to the settings shown in Table 4-14, the setting of each peripheral function control register is required.

Table 4-14. Output Data and Port Read Value for Each Setting (1/5)

Port Name	Function	PMCmn	PFCEmn	PFCmn	PMmn	Output Data	Pmn Read Value	Remark
P00, P01[Note 1], P02 to P07	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	INTP0, INTP1[Note 1], INTP2 to INTP7[Note 2]	1	None	None	0	–	Port latch	Necessary to specify valid edge
					1		Pin level	
P10 to P12	Output port	0	×	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TOQ0T1, TOQ0B1, TOQ0T2	1	0	0	0	Alternate output 1 (timer output)	Port latch	
					1		Pin level	
	TIQ01 to TIQ03	1	0	1	0	–	Port latch	Alternate input (timer input)
					1		Pin level	
	TOQ01 to TOQ03	1	1	0	0	Alternate output 2 (timer output)	Port latch	
					1		Pin level	
P13 to P17	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TOQ0B2, TOQ0T3, TOQ0B3, TOQ00, TOP21	1	None	0	0	Alternate output 1 (timer output)	Port latch	
					1		Pin level	
	TIQ00, EVTQ0, TRGQ0 TIP20, TIP21	1	None	1	0	–	Port latch	Alternate input (timer input)
					1		Pin level	

Notes 1. V850E/IA4 only

2. Including TOQ0OFF, TOQ1OFF (V850E/IA4 only), TOP2OFF, TOP3OFF (V850E/IA4 only), ADTRG0, and ADTRG1.

Remark ×: don't care

Table 4-14. Output Data and Port Read Value for Each Setting (2/5)

Port Name	Function	PMCmn	PFCEmn	PFCmn	PMmn	Output Data	Pmn Read Value	Remark
P20 to P27[Note]	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TOQ1T1[Note], TOQ1B1[Note], TOQ1T2[Note], TOQ1B2[Note], TOQ1T3[Note], TOQ1B3[Note], TOQ10[Note], TOP31[Note]	1	None	None	0	Alternate output (timer output)	Port latch	
					1		Pin level	
P30, P36, P37	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	RXDA0, TCUD10, TCLR10	1	None	None	0	–	Port latch	Alternate input (serial input, timer input)
					1		Pin level	
P31	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TXDA0	1	None	None	0	Alternate output (serial output)	Port latch	
					1		Pin level	
P32	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	SIB1	1	None	0	0	–	Port latch	Alternate input (serial input)
					1		Pin level	
	RXDA1	1	None	1	0	–	Port latch	Alternate input (serial input)
					1		Pin level	

Note V850E/IA4 only

Remark ×: don't care

Table 4-14. Output Data and Port Read Value for Each Setting (3/5)

Port Name	Function	PMCmn	PFCEmn	PFCmn	PMmn	Output Data	Pmn Read Value	Remark
P33	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	SOB1	1	None	0	0	Alternate output 1	Port latch	
					1	(serial output)	Pin level	
	TXDA1	1	None	1	0	Alternate output 2	Port latch	
					1	(serial output)	Pin level	
P34	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	$\overline{\text{SCKB1}}$	1	None	None	0	Alternate I/O	Port latch	Output in master mode
					1	(serial)	Pin level	Input in slave mode
P35	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TIUD10	1	None	0	0	–	Port latch	Alternate input (timer input)
					1		Pin level	
	TO10	1	None	1	0	Alternate output	Port latch	
					1	(timer output)	Pin level	
P40	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	SIB0	1	None	None	0	–	Port latch	Alternate input (serial input)
					1		Pin level	
P41	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	SOB0	1	None	None	0	Alternate output	Port latch	
					1	(serial output)	Pin level	

Remark ×: don't care

Table 4-14. Output Data and Port Read Value for Each Setting (4/5)

Port Name	Function	PMCmn	PFCEmn	PFCmn	PMmn	Output Data	Pmn Read Value	Remark
P42	Output port	0	None	None	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	$\overline{\text{SCKB0}}$	1	None	None	0	Alternate I/O	Port latch	Output in master mode
					1	(serial)	Pin level	Input in slave mode
P43, P44	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TOP00, TOP01	1	None	0	0	Alternate output	Port latch	
					1	(timer output)	Pin level	
	TIP00, TIP01	1	None	1	0	–	Port latch	Alternate input (timer input)
					1		Pin level	
P50[Notes 1, 2]	Output port	0	None	×	0	Port latch	Port latch	
	Input port				1	–	Pin level	
	TIUD11[Note 1]	1	None	0	0	–	Port latch	Alternate input (timer input)
					1		Pin level	
	TO11[Note 1]	1	None	1	0	Alternate output	Port latch	
					1	(timer output)	Pin level	

Notes 1. V850E/IA4 only.

2. The P50 pin also functions as an on-chip debug pin (μPD70F3186 only). The on-chip debug function or port function (including the alternate function) can be selected by using the $\overline{\text{DRST}}$ pin level, as shown in the table below.

Port 5 Function	
$\overline{\text{DRST}}$ Pin Low-Level Input	$\overline{\text{DRST}}$ Pin High-Level Input
P50/TIUD11/TO11	DDI

Remark ×: don't care

Table 4-14. Output Data and Port Read Value for Each Setting (5/5)

Port Name	Function	PMCmn	PFCEmn	PFCmn	PMmn	Output Data	Pmn Read Value	Remark
P51[Notes 1, 2]	Output port	0	None	None	0	Port latch	Port latch	
P52[Notes 1, 2]	Input port				1	–	Pin level	
	TCUD11, TCLR11	1	None	None	0	–	Port latch	Alternate input (timer input)
					1		Pin level	
P70 to P77	Input port	0	None	None	None	–	Pin level	Input-only port
	ANI20 to ANI27	1				–	Low level	
PDL0 to	Output port	None	None	None	0	Port latch	Port latch	
PDL15[Note 3]	Input port				1	–	Pin level	

Notes 1. V850E/IA4 only

2. The P51 and P52 pins also function as on-chip debug pins (μPD70F3186 only). The on-chip debug function or port function (including the alternate function) can be selected by using the $\overline{\text{DRST}}$ pin level, as shown in the table below.

Port 5 Function	
$\overline{\text{DRST}}$ Pin Low-Level Input	$\overline{\text{DRST}}$ Pin High-Level Input
P51/TCUD11	DCK
P52/TCLR11	DMS

3. The PDL5 pin is also used in flash programming mode (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only). This pin does not have to be manipulated by a port control register. For details, see **CHAPTER 22 FLASH MEMORY**.

4.5 Port Register Settings When Alternate Function Is Used

The following shows the port register settings when each port is used for an alternate function. When using a port pin as an alternate-function pin, refer to the description of each pin.

Table 4-15. Using Port Pin as Alternate-Function Pin (1/5)

Pin Name	Alternate Pin		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bit (Register)
	Name	I/O						
P00	INTP0	Input	P00 = Setting not required	PM00 = Setting not required	PMC00 = 1	–	–	
	TOQ0OFF	Input	P00 = Setting not required	PM00 = Setting not required	PMC00 = 1	–	–	
P01[Note]	INTP1[Note]	Input	P01 = Setting not required	PM01 = Setting not required	PMC01 = 1	–	–	
	TOQ1OFF[Note]	Input	P01 = Setting not required	PM01 = Setting not required	PMC01 = 1	–	–	
P02	INTP2	Input	P02 = Setting not required	PM02 = Setting not required	PMC02 = 1	–	–	
	TOP2OFF	Input	P02 = Setting not required	PM02 = Setting not required	PMC02 = 1	–	–	
P03	INTP3	Input	P03 = Setting not required	PM03 = Setting not required	PMC03 = 1	–	–	
	TOP3OFF[Note]	Input	P03 = Setting not required	PM03 = Setting not required	PMC03 = 1	–	–	
P04	INTP4	Input	P04 = Setting not required	PM04 = Setting not required	PMC04 = 1	–	–	
	ADTRG0	Input	P04 = Setting not required	PM04 = Setting not required	PMC04 = 1	–	–	
P05	INTP5	Input	P05 = Setting not required	PM05 = Setting not required	PMC05 = 1	–	–	
	ADTRG1	Input	P05 = Setting not required	PM05 = Setting not required	PMC05 = 1	–	–	
P06	INTP6	Input	P06 = Setting not required	PM06 = Setting not required	PMC06 = 1	–	–	
P07	INTP7	Input	P07 = Setting not required	PM07 = Setting not required	PMC07 = 1	–	–	
P10	TOQ0T1	Output	P10 = Setting not required	PM10 = Setting not required	PMC10 = 1	PFCE10 = 0	PFC10 = 0	
	TIQ01	Input	P10 = Setting not required	PM10 = Setting not required	PMC10 = 1	PFCE10 = 0	PFC10 = 1	
	TOQ01	Output	P10 = Setting not required	PM10 = Setting not required	PMC10 = 1	PFCE10 = 1	PFC10 = 0	
P11	TOQ0B1	Output	P11 = Setting not required	PM11 = Setting not required	PMC11 = 1	PFCE11 = 0	PFC11 = 0	
	TIQ02	Input	P11 = Setting not required	PM11 = Setting not required	PMC11 = 1	PFCE11 = 0	PFC11 = 1	
	TOQ02	Output	P11 = Setting not required	PM11 = Setting not required	PMC11 = 1	PFCE11 = 1	PFC11 = 0	
P12	TOQ0T2	Output	P12 = Setting not required	PM12 = Setting not required	PMC12 = 1	PFCE12 = 0	PFC12 = 0	
	TIQ03	Input	P12 = Setting not required	PM12 = Setting not required	PMC12 = 1	PFCE12 = 0	PFC12 = 1	
	TOQ03	Output	P12 = Setting not required	PM12 = Setting not required	PMC12 = 1	PFCE12 = 1	PFC12 = 0	

Note V850E/IA4 only

Table 4-15. Using Port Pin as Alternate-Function Pin (2/5)

Pin Name	Alternate Pin		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bit (Register)
	Name	I/O						
P13	TOQ0B2	Output	P13 = Setting not required	PM13 = Setting not required	PMC13 = 1	–	PFC13 = 0	
	TIQ00	Input	P13 = Setting not required	PM13 = Setting not required	PMC13 = 1	–	PFC13 = 1	
P14	TOQ0T3	Output	P14 = Setting not required	PM14 = Setting not required	PMC14 = 1	–	PFC14 = 0	
	EVTQ0	Input	P14 = Setting not required	PM14 = Setting not required	PMC14 = 1	–	PFC14 = 1	
P15	TOQ0B3	Output	P15 = Setting not required	PM15 = Setting not required	PMC15 = 1	–	PFC15 = 0	
	TRGQ0	Input	P15 = Setting not required	PM15 = Setting not required	PMC15 = 1	–	PFC15 = 1	
P16	TOQ00	Output	P16 = Setting not required	PM16 = Setting not required	PMC16 = 1	–	PFC16 = 0	
	TIP20	Input	P16 = Setting not required	PM16 = Setting not required	PMC16 = 1	–	PFC16 = 1	
P17	TOP21	Output	P17 = Setting not required	PM17 = Setting not required	PMC17 = 1	–	PFC17 = 0	
	TIP21	Input	P17 = Setting not required	PM17 = Setting not required	PMC17 = 1	–	PFC17 = 1	
P20[Note]	TOQ1T1[Note]	Output	P20 = Setting not required	PM20 = Setting not required	PMC20 = 1	–	–	
P21[Note]	TOQ1B1[Note]	Output	P21 = Setting not required	PM21 = Setting not required	PMC21 = 1	–	–	
P22[Note]	TOQ1T2[Note]	Output	P22 = Setting not required	PM22 = Setting not required	PMC22 = 1	–	–	
P23[Note]	TOQ1B2[Note]	Output	P23 = Setting not required	PM23 = Setting not required	PMC23 = 1	–	–	
P24[Note]	TOQ1T3[Note]	Output	P24 = Setting not required	PM24 = Setting not required	PMC24 = 1	–	–	
P25[Note]	TOQ1B3[Note]	Output	P25 = Setting not required	PM25 = Setting not required	PMC25 = 1	–	–	
P26[Note]	TOQ10[Note]	Output	P26 = Setting not required	PM26 = Setting not required	PMC26 = 1	–	–	
P27[Note]	TOP31[Note]	Output	P27 = Setting not required	PM27 = Setting not required	PMC27 = 1	–	–	
P30	RXDA0	Input	P30 = Setting not required	PM30 = Setting not required	PMC30 = 1	–	–	
P31	TXDA0	Output	P31 = Setting not required	PM31 = Setting not required	PMC31 = 1	–	–	
P32	SIB1	Input	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	–	PFC32 = 0	
	RXDA1	Input	P32 = Setting not required	PM32 = Setting not required	PMC32 = 1	–	PFC32 = 1	

Note V850E/IA4 only

Table 4-15. Using Port Pin as Alternate-Function Pin (3/5)

Pin Name	Alternate Pin		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bit (Register)
	Name	I/O						
P33	SOB1	Output	P33 = Setting not required	PM33 = Setting not required	PMC33 = 1	–	PFC33 = 0	
	TXDA1	Output	P33 = Setting not required	PM33 = Setting not required	PMC33 = 1	–	PFC33 = 1	
P34	$\overline{\text{SCKB1}}$	I/O	P34 = Setting not required	PM34 = Setting not required	PMC34 = 1	–	–	
P35	TIUD10	Input	P35 = Setting not required	PM35 = Setting not required	PMC35 = 1	–	PFC35 = 0	
	TO10	Output	P35 = Setting not required	PM35 = Setting not required	PMC35 = 1	–	PFC35 = 1	
P36	TCUD10	Input	P36 = Setting not required	PM36 = Setting not required	PMC36 = 1	–	–	
P37	TCLR10	Input	P37 = Setting not required	PM37 = Setting not required	PMC37 = 1	–	–	
P40	SIB0	Input	P40 = Setting not required	PM40 = Setting not required	PMC40 = 1	–	–	
P41	SOB0	Output	P41 = Setting not required	PM41 = Setting not required	PMC41 = 1	–	–	
P42	$\overline{\text{SCKB0}}$	I/O	P42 = Setting not required	PM42 = Setting not required	PMC42 = 1	–	–	
P43	TOP00	Output	P43 = Setting not required	PM43 = Setting not required	PMC43 = 1	–	PFC43 = 0	
	TIP00	Input	P43 = Setting not required	PM43 = Setting not required	PMC43 = 1	–	PFC43 = 1	
P44	TOP01	Output	P44 = Setting not required	PM44 = Setting not required	PMC44 = 1	–	PFC44 = 0	
	TIP01	Input	P44 = Setting not required	PM44 = Setting not required	PMC44 = 1	–	PFC44 = 1	

Table 4-15. Using Port Pin as Alternate-Function Pin (4/5)

Pin Name	Alternate Pin		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bit (Register)
	Name	I/O						
P50[Note 1]	DDI[Notes 2, 3]	Input	P50 = Setting not required	PM50 = Setting not required	PMC50 = Setting not required	–	–	
	TIUD11[Note 1]	Input	P50 = Setting not required	PM50 = Setting not required	PMC50 = 1	–	PFC50 = 0	
	TO11[Note 1]	Output	P50 = Setting not required	PM50 = Setting not required	PMC50 = 1	–	PFC50 = 1	
P51[Note 1]	DCK[Notes 2, 3]	Input	P51 = Setting not required	PM51 = Setting not required	PMC51 = Setting not required	–	–	
	TCUD11[Note 1]	Input	P51 = Setting not required	PM51 = Setting not required	PMC51 = 1	–	–	
P52[Note 1]	DMS[Notes 2, 3]	Input	P52 = Setting not required	PM52 = Setting not required	PMC52 = Setting not required	–	–	
	TCLR11[Note 1]	Input	P52 = Setting not required	PM52 = Setting not required	PMC52 = 1	–	–	
P70	ANI20	Input	P70 = Setting not required	–	PMC70 = 1	–	–	
P71	ANI21	Input	P71 = Setting not required	–	PMC71 = 1	–	–	
P72	ANI22	Input	P72 = Setting not required	–	PMC72 = 1	–	–	
P73	ANI23	Input	P73 = Setting not required	–	PMC73 = 1	–	–	
P74	ANI24	Input	P74 = Setting not required	–	PMC74 = 1	–	–	
P75	ANI25	Input	P75 = Setting not required	–	PMC75 = 1	–	–	

Notes 1. V850E/IA4 only

2. μPD70F3186 (V850E/IA4) only

3. The P50 to P52 pins are also used for on-chip debugging (μPD70F3186 only). Switching between on-chip debug function and port function (including alternate function) can be set by setting the $\overline{\text{DRST}}$ pin level. The following shows the setting method.

Port 5 Functions	
Low-Level Input to $\overline{\text{DRST}}$ Pin	High-Level Input to $\overline{\text{DRST}}$ Pin
P50/TIUD11/TO11	DDI
P51/TCUD11	DCK
P52/TCLR11	DMS

Table 4-15. Using Port Pin as Alternate-Function Pin (5/5)

Pin Name	Alternate Pin		Pnx Bit of Pn Register	PMnx Bit of PMn Register	PMCnx Bit of PMCn Register	PFCEnx Bit of PFCEn Register	PFCnx Bit of PFCn Register	Other Bit (Register)
	Name	I/O						
P76[Note 1]	ANI26[Note 1]	Input	P76 = Setting not required	–	PMC76 = 1	–	–	
P77[Note 1]	ANI27[Note 1]	Input	P77 = Setting not required	–	PMC77 = 1	–	–	
PDL0	–	–	PDL0 = Setting not required	PMDL0 = Setting not required	–	–	–	
PDL1	–	–	PDL1 = Setting not required	PMDL1 = Setting not required	–	–	–	
PDL2	–	–	PDL2 = Setting not required	PMDL2 = Setting not required	–	–	–	
PDL3	–	–	PDL3 = Setting not required	PMDL3 = Setting not required	–	–	–	
PDL4	–	–	PDL4 = Setting not required	PMDL4 = Setting not required	–	–	–	
PDL5	FLMD1[Notes 2, 3]	Input	PDL5 = Setting not required	PMDL5 = Setting not required	–	–	–	
PDL6	–	–	PDL6 = Setting not required	PMDL6 = Setting not required	–	–	–	
PDL7	–	–	PDL7 = Setting not required	PMDL7 = Setting not required	–	–	–	
PDL8	–	–	PDL8 = Setting not required	PMDL8 = Setting not required	–	–	–	
PDL9	–	–	PDL9 = Setting not required	PMDL9 = Setting not required	–	–	–	
PDL10	–	–	PDL10 = Setting not required	PMDL10 = Setting not required	–	–	–	
PDL11	–	–	PDL11 = Setting not required	PMDL11 = Setting not required	–	–	–	
PDL12	–	–	PDL12 = Setting not required	PMDL12 = Setting not required	–	–	–	
PDL13	–	–	PDL13 = Setting not required	PMDL13 = Setting not required	–	–	–	
PDL14	–	–	PDL14 = Setting not required	PMDL14 = Setting not required	–	–	–	
PDL15	–	–	PDL15 = Setting not required	PMDL15 = Setting not required	–	–	–	

Notes 1. V850E/IA4 only

2. The PDL5 pin also functions as a pin to be set in the flash programming mode (μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only). This pin does not need to be manipulated using the port control register. For details, see **CHAPTER 22 FLASH MEMORY**.

3. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

4.6 Noise Eliminator

A timing controller used to secure the noise elimination time is provided for the following pins. Input signals that change within the noise elimination time are not internally acknowledged.

<R>

Table 4-16. Noise Eliminator

Unit	Target Pin	Delay Type	Noise Elimination Width	Sampling Clock
Reset	$\overline{\text{RESET}}$	Analog delay	Several 10 ns (TYP.)	–
On-chip debug	$\overline{\text{DRST}}$[Note 1]			
Mode pin	FLMD0[Note 2]/IC1[Note 3]			
Clock generator (CG)	PLLSIN		About 10 ns (TYP.)	
• Interrupt (INTC)[Note 4] • High-impedance output control function of timer for motor control • A/D converter (ADC)	INTP0/TOQ0OFF INTP1[Note 5]/TOQ1OFF[Note 5] INTP2/TOP2OFF INTP3/TOP3OFF[Note 5] INTP4/ADTRG0 INTP5/ADTRG1 INTP7		500 ns (MIN.)	
	INTP6	Digital delay	4 to 5 clocks (250 ns (at 64 MHz)) (500 ns (at 64 MHz)) (1 μs (at 64 MHz)) (2 μs (at 64 MHz))	fxx/4 (62.5 ns (at 64 MHz)) fxx/8 (125 ns (at 64 MHz)) fxx/16 (250 ns (at 64 MHz)) fxx/32 (500 ns (at 64 MHz)) selectable
Timer ENC (TMENC)	TIUD11[Note 5] TCUD11[Note 5] TCLR11[Note 5] TIUD10 TCUD10 TCLR10			
Timer Q (TMQ)	TIQ01 TIQ02 TIQ03 TIQ00 EVTQ0 TRGQ0		4 to 5 clocks (125 ns (at 64 MHz))	fxx/2 (31.25 ns (at 64 MHz))
Timer P (TMP)	TIP20 TIP21 TIP00 TIP01			

Notes 1. μPD70F3186 (V850E/IA4) only

2. μPD70F3184 (V850E/IA3), μPD70F3186 (V850E/IA4) only

3. μPD703183 (V850E/IA3), μPD703185 (V850E/IA4), μPD703186 (V850E/IA4) only

4. A maskable interrupt input other than INTP6 can be used as the release source of IDLE or STOP mode.

5. V850E/IA4 only

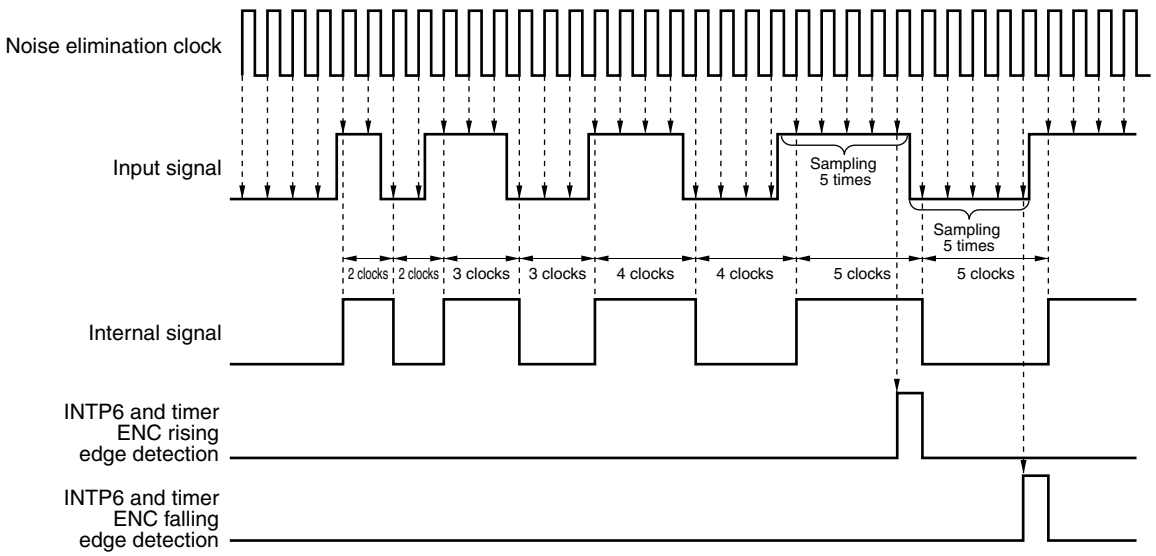
Cautions 1. The maskable interrupt pins are used to release the standby mode.

2. The noise filter of the digital delay pin uses clock sampling and therefore cannot acknowledge an input signal when the peripheral clock (fxx) is stopped.

3. The noise eliminator is valid only in the alternate-function mode.

The timing example of digital noise elimination at an input pin of INTP6 and the timer ENC.

<R> **Figure 4-26. Example of Noise Elimination Timing**



Caution If there are four or less noise elimination clocks while INTP6 and timer ENC input signals are high level (or low level), the input pulse is eliminated as noise. If it is sampled at least five times, the edge is detected as valid input.

(1) External interrupt noise elimination control register (INTPNRC)

The INTPNRC register is used to select the sampling clock that is used to eliminate digital noise on the INTP6 pin. If the same level is not detected five times in a row, the signal is eliminated as noise.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. If the input pulse lasts for the duration of 4 to 5 clocks, it is undefined whether the pulse is detected as a valid edge or eliminated as noise. So that the pulse is actually detected as a valid edge, the same pulse level must be input for the duration of 5 clocks or more.

2. If noise is generated in synchronization with the sampling clock, eliminate the noise by attaching a filter to the input pin.

3. Noise is not eliminated if the pin is used as a normal input port pin.

After reset: 00H R/W Address: FFFFF310H

	7	6	5	4	3	2	1	0
INTPNRC	0	0	0	0	0	0	INTPNRC1	INTPNRC0

INTPNRC1	INTPNRC0	Selection of sampling clock
0	0	$f_{XX}/32$
0	1	$f_{XX}/16$
1	0	$f_{XX}/8$
1	1	$f_{XX}/4$

(2) Noise elimination time select register 1n (NRC1n) (V850E/IA3: n = 0, V850E/IA4: n = 0, 1)

The NRC1n register is used to select the sampling clock that is used to eliminate digital noise on the TIUD1n, TCUD1n, or TCLR1n pin. If the same level is not detected on these pins five times in a row using the clock selected by the NRC1n register, the signal is eliminated as noise.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. If the input pulse lasts for the duration of 4 to 5 clocks, it is undefined whether the pulse is detected as a valid edge or eliminated as noise. So that the pulse is actually detected as a valid edge, the same pulse level must be input for the duration of 5 clocks or more.

2. If noise is generated in synchronization with the sampling clock, eliminate the noise by attaching a filter to the input pin.

3. Noise is not eliminated if the pin is used as a normal input port pin.

4. The noise elimination function starts operating when the TMC1n.TM1CEn bit is set to 1 (enabling count operations).

After reset: 00H R/W Address: NRC10 FFFFF598H, NRC11[Note] FFFFF5B8H

NRC1n

	7	6	5	4	3	2	1	0
V850E/IA3 n = 0	0	0	0	0	0	0	NRC1n1	NRC1n0

V850E/IA4 n = 0, 1

NRC1n1	NRC1n0	Selection of sampling clock
0	0	f_{XX}/32
0	1	f_{XX}/16
1	0	f_{XX}/8
1	1	f_{XX}/4

Note V850E/IA4 only

4.7 Cautions

4.7.1 Cautions on setting port pins

(1) Set the registers of a port in the following sequence.

<1> Set PFCn and PFCEn registers.
<2> Set PMCn register.
<3> Set INTFn and INTRn registers.

If the PMCn register is set before setting the PFCn and PFCEn registers, an unexpected peripheral function may be selected while the PFCn and PFCEn registers are being set.

(2) An on-chip pull-up resistor can only be connected when the pins are in input mode in the port mode, or when the pins function as input pins in the alternate-function mode.
Moreover, for the V850E/IA3, an on-chip pull-up resistor can be connected to the TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, and TOP21 pins, these are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ0OFF and TOP2OFF pins or software processing.
For the V850E/IA4, an on-chip pull-up resistor can be connected to the TOQ0T1 to TOQ0T3, TOQ0B1 to TOQ0B3, TOP21, TOQ1T1 to TOQ1T3, TOQ1B1 to TOQ1B3 and TOP31 pins, these are output pins in the alternate-function mode, when these pins go into a high-impedance state due to the TOQ0OFF, TOP2OFF, TOQ1OFF, or TOP3OFF pin or software processing.

4.7.2 Cautions on bit manipulation instruction for port n register (Pn)

When a 1-bit manipulation instruction is executed on a port that provides both input and output functions, the value of the output latch 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 P20 pin is an output port, P21 to P27 pins are input ports (all pin statuses are high level), and the value of the port latch is 00H, if the output of P20 pin is changed from low level to high level via a bit manipulation instruction, the value of the port latch 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 bit manipulation instruction is executed in the following order in the V850E/IA3 and V850E/IA4.

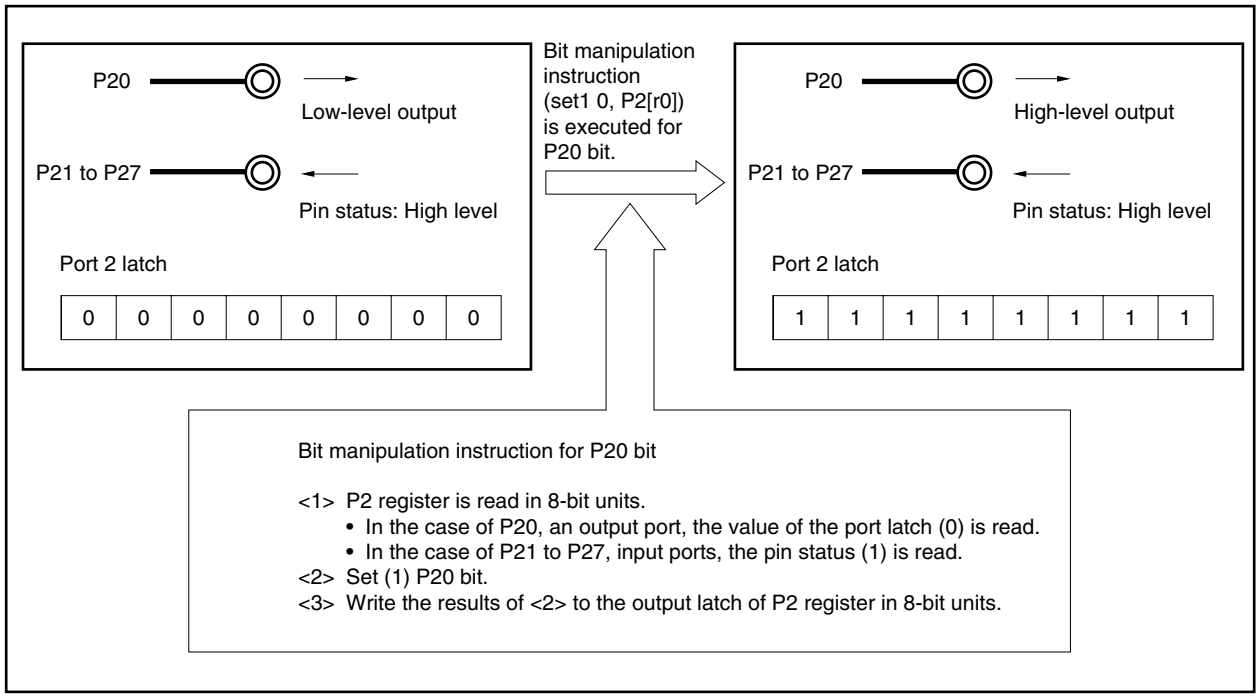
<1> The Pn register is read in 8-bit units.
<2> The targeted one bit is manipulated.
<3> The Pn register is written in 8-bit units.

In step <1>, the value of the output latch (0) of P20 pin, which is an output port, is read, while the pin statuses of P21 to P27 pins, which are input ports, are read. If the pin statuses of P21 to P27 pins 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 4-27. Bit Manipulation Instruction (P20 Pin)



CHAPTER 5 CLOCK GENERATOR

5.1 Overview

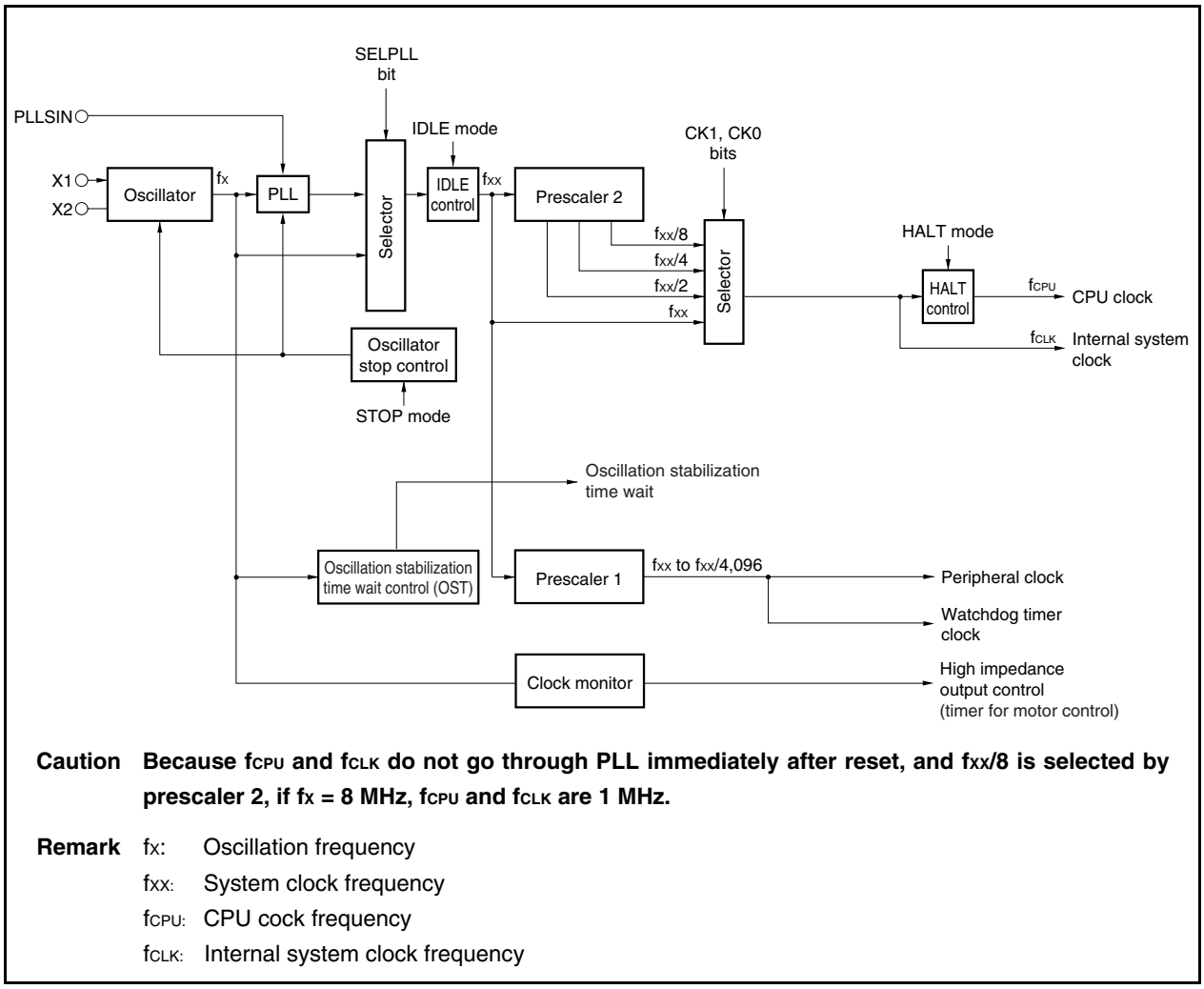
The features of clock generator are as follows.

- ○ Oscillator
  - In PLL mode: f_X = 4 to 8 MHz (f_{XX} = 32 to 64 MHz)
  - In clock-through mode: f_X = 4 to 8 MHz (f_{XX} = 4 to 8 MHz)
- ○ Multiply (×8 fixed) function by PLL (Phase Locked Loop)
  - Clock-through mode/PLL mode selectable
- ○ Internal system clock generation
  - 4 steps (f_{XX}, f_{XX}/2, f_{XX}/4, f_{XX}/8)
- ○ Peripheral clock generation
- ○ Oscillation stabilization time selection

Remark f_X: Oscillation frequency
f_{XX}: System clock

5.2 Configuration

Figure 5-1. Clock Generator



Caution Because f_{CPU} and f_{CLK} do not go through PLL immediately after reset, and $f_{XX}/8$ is selected by prescaler 2, if f_X = 8 MHz, f_{CPU} and f_{CLK} are 1 MHz.

Remark f_X: Oscillation frequency
f_{XX}: System clock frequency
f_{CPU}: CPU cock frequency
f_{CLK}: Internal system clock frequency

Table 5-1. Operation Clock of Each Function Block

Function Block	Operation Clock
CPU	f_{CPU} (Selected from f_{XX} to $f_{XX}/8$ by PCC register)
DMA, interrupt controller	f_{CLK} (Selected from f_{XX} to $f_{XX}/8$ by PCC register)
Timer (excluding watchdog timer)	$f_{XX}/2$
Watchdog timer	$f_{XX}/1,024$
UARTA	f_{UCLK} (Selected from $f_{XX}/2$ to $f_{XX}/4,096$ by UAnCTL1 register)
CSIB	f_{CCLK} (Selected from $f_{XX}/4$ to $f_{XX}/256$ by CBnCTL1 register)
A/D converters 0, 1	$f_{XX}/2$
A/D converter 2	$f_{XX}/4$

Remarks 1. f_{XX}: Peripheral clock
2. n = 0, 1

(1) Oscillator

The main resonator oscillates the following frequencies (f_X):

- In PLL mode (×8 fixed): f_X = 4 to 8 MHz (f_{XX} = 32 to 64 MHz)
- In clock-through mode: f_X = 4 to 8 MHz (f_{XX} = 4 to 8 MHz)

(2) IDLE control

All functions other than the oscillator, PLL, clock monitor operation, and CSIB in slave mode are stopped.

(3) HALT control

Only the CPU clock (f_{CPU}) is stopped.

(4) PLL

This circuit multiplies the clock generated by the oscillator (f_X) by 8.

It operates in two modes: clock-through mode in which f_X is output as is by setting the SELPLL bit of the PLL control register (PLLCTL), and PLL mode in which a multiplied clock is output.

The output frequency of PLL is 32 to 64 MHz in the PLL mode. When using the frequency in a range of 32 to 55 MHz (f_X = 4 to 6.875 MHz), fix the PLLSIN pin to the low level. When using the frequency in a range of 55 to 64 MHz (f_X = 6.876 to 8 MHz), fix the PLLSIN pin to the high level.

(5) Prescaler 1

<R> This prescaler generates the clock (f_{XX} to f_{XX}/4,096) to be supplied to on-chip peripheral functions.

(6) Prescaler 2

This circuit divides the system clock (f_{XX}).

The clock (f_{XX} to f_{XX}/8) to be supplied to the CPU clock (f_{CPU}) and internal system clock (f_{CLK}) is generated.

(7) Oscillation stabilization time wait control (OST)

This unit measures the time from when the clock generated by the oscillator was input until oscillation is stabilized. It also counts the PLL lockup time.

The count clock can be selected from $2^{14}/f_X$ to $2^{18}/f_X$.

(8) Clock monitor

The clock monitor samples the clock generated by the oscillator (f_X), by using the internal oscillation clock. When it detects stop of oscillation, output of the timer for motor control goes into a high-impedance state (for details, see **CHAPTER 10 MOTOR CONTROL FUNCTION)**.

5.3 Control Registers

The clock generator is controlled by the following six registers.

- PLL control register (PLLCTL)
- Processor clock control register (PCC)
- Power save control register (PSC)
- Power save mode register (PSMR)
- Oscillation stabilization time select register (OSTS)
- Clock monitor mode register (CLM)

(1) PLL control register (PLLCTL)

The PLLCTL register selects CPU operation clock.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 01H.

After reset: 01H R/W Address: FFFFF82CH

	7	6	5	4	3	2	<1>	0
PLLCTL	0	0	0	0	0	0	SELPLL	1

SELPLL	CPU operation clock selection
0	Clock-through mode
1	PLL mode

Caution Be sure to clear bits 7 to 2 to “0” and set bit 0 to “1”.

(2) Processor clock control register (PCC)

The PCC register is a special register. Data can be written to this register only in a combination of specific sequences (see **3.4.8 Special registers**).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 03H.

After reset: 03H R/W Address: FFFFF828H

	7	6	5	4	3	2	1	0
PCC	0	0	0	0	0	0	CK1	CK0

CK1	CK0	Clock selection (f_{CLK}/f_{CPU})
0	0	f_{XX}
0	1	$f_{XX}/2$
1	0	$f_{XX}/4$
1	1	$f_{XX}/8$

Cautions 1. Be sure to clear bits 2 to 7 to "0".

2. Set the PCC register after the PLL mode is selected (PLLCTL.SELPLL bit = 1).

(3) Power save control register (PSC)

The PSC register is a special register. Data can be written to this register only in a combination of specific sequences (see **3.4.8 Special registers**).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF1FEH

	7	6	5	<4>	3	2	<1>	0
PSC	0	0	0	INTM	0	0	STB	0

INTM	Standby mode control by maskable interrupt request (INTxx[Note])
0	Standby mode release by INTxx request enabled
1	Standby mode release by INTxx request disabled

STB	Sets operation mode
0	Normal mode
1	Standby mode

Note For details, see **Table 17-1 Interrupt Source List**.

Cautions 1. Be sure to clear bits 0, 2, 3, and 5 to 7 to "0".

2. Before setting a standby mode by setting the STB bit to 1, be sure to set the PCC register to 03H and then set the STB bit to 1. Otherwise, the standby mode may not be set or released. After releasing the standby mode, change the value of the PCC register to the desired value.

3. To set the IDLE mode or STOP mode, set the PCC register to 03H, and the PSMR.PSM0 bit in that order and then set the STB bit to 1.

(4) Power save mode register (PSMR)

The PSMR register is an 8-bit register that controls the operation in the software standby mode.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF820H

	7	6	5	4	3	2	1	<0>
PSMR	0	0	0	0	0	0	0	PSM0

PSM0	Specifies operation in software standby mode
0	IDLE mode
1	STOP mode

Cautions 1. Be sure to clear bits 1 to 7 to "0".

2. The PSM0 bit is valid only when the PSC.STB bit is 1.

(5) Oscillation stabilization time select register (OSTS)

The OSTS register selects the oscillation stabilization time until the oscillation stabilizes after the STOP mode is released by interrupt request.

This register can be read or written in 8-bit units.

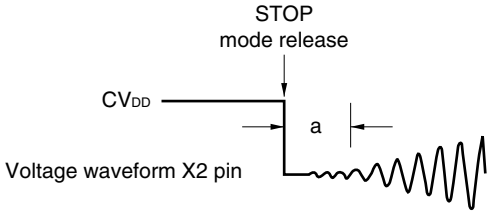
Reset sets this register to 04H.

After reset: 04H R/W Address: FFFFF6C0H

	7	6	5	4	3	2	1	0
OSTS	0	0	0	0	OSTS3	OSTS2	OSTS1	OSTS0

OSTS3	OSTS2	OSTS1	OSTS0	Selection of oscillation stabilization time (f_X = 8 MHz)
0	1	0	0	$2^{14}/f_X$ (2.05 ms)
0	1	0	1	$2^{15}/f_X$ (4.10 ms)
0	1	1	0	$2^{16}/f_X$ (8.19 ms)
0	1	1	1	$2^{17}/f_X$ (16.4 ms)
1	0	0	0	$2^{18}/f_X$ (32.8 ms)
Other than above				Setting prohibited

Cautions 1. The wait time does not include the time until the clock oscillation starts ("a" in the figure below) following release of the STOP mode.



2. The default value of the OSTS register after reset is 04H. If an 8 MHz resonator is used, therefore, the oscillation stabilization time is about 2 ms. Half the oscillation stabilization time is consumed by waiting for the lockup of PLL. Therefore, the actual stabilization time of the resonator is about 1 ms. When releasing reset, therefore, make sure that the oscillation stabilization time is secured during the active period of the reset signal. To release the STOP mode by an interrupt input other than $\overline{\text{RESET}}$ input, the oscillation stabilization time is determined by the set value of the OSTS register. Therefore, set a time twice as long as that required for the resonator to stabilize to the OSTS register (because half the oscillation stabilization time is the stabilization time of PLL).

3. Be sure to clear bits 4 to 7 to "0".

Remark fx: Oscillation frequency

(6) Clock monitor mode register (CLM)

The CLM register sets clock monitor operation mode. The CLM register is a special register. It can be written only in a combination of specific sequences (see **3.4.8 Special registers**).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF870H

	7	6	5	4	3	2	1	0
CLM	0	0	0	0	0	0	0	CLME

CLME	Clock monitor operation control
0	Clock monitor operation disabled
1	Clock monitor operation enabled

Cautions 1. The CLME bit is cleared to 0 only after reset.

2. When the CLME bit = 1, the clock monitor function is forcibly stopped if the following conditions are satisfied.

- **During oscillation stabilization time count after release of STOP mode**
- **During break (on-chip debug emulator)**

3. When the CLME bit = 1, output of the timer for motor control goes into a high-impedance state if oscillation (f_X) stop is detected. See Figure 10-4 for the target timer output.

5.4 PLL Function

5.4.1 Overview

The CPU and the operating clock of the peripheral macro can be switched between output of the oscillation frequency multiplied by 8, and clock-through mode.

When PLL function is used: Input clock (f_X) = 4 to 8 MHz, output clock (f_{XX}) = 32 to 64 MHz
Clock-through mode: Input clock (f_X) = 4 to 8 MHz, output clock (f_{XX}) = 4 to 8 MHz

5.4.2 Setting PLL output frequency

With the V850E/IA3 and V850E/IA4, the PLL output frequency range must be set as follows via signal input to the PLLSIN pin.

Table 5-2. Setting PLL Output Frequency

PLLSIN	Input Clock Frequency (f_X)	Output Clock Frequency (f_{XX})
L	4.0 MHz to 6.875 MHz	32.0 MHz to 55.0 MHz
H	6.876 MHz to 8.0 MHz	55.0 MHz to 64.0 MHz

Caution Fix the input levels of the PLLSIN pin during the reset period and do not change the levels during operation. Otherwise, the operation is not guaranteed.

5.4.3 PLL mode

In the PLL mode, the oscillation frequency (f_X) is multiplied by 8 with the PLL to generate a system clock (f_{XX}).

Fix the input level of the PLLSIN pin to the high or low level according to the value of f_X.

In the PLL mode, the clock is input from the oscillator to the PLL. A clock at a stable frequency must be supplied to the internal circuit after the lapse of the lockup time (frequency stabilization time) during which the phase is locked at a specific frequency and oscillation is stabilized. In the V850E/IA3 and V850E/IA4, the lockup time after release of reset is secured automatically.

Caution When a resonator of f_X = 8 MHz is used and if the oscillation stabilization time of that resonator must be 3 ms (MAX.), the reset input ($\overline{\text{RESET}}$ active) width must be 2 ms (MIN.).

5.4.4 Clock-through mode

In the clock-through mode, a system clock (f_{XX}) of the same frequency as the oscillation frequency (f_X) is generated.

5.5 Operation

5.5.1 Operation of each clock

The following table shows the operation status of each clock.

Table 5-3. Operation Status of Each Clock

<R>

Power Save Mode	Oscillator (f_X)	PLL	Internal System Clock (f_{CLK})	Peripheral Clock (f_{XX} to $f_{XX}/4,096$)	CPU Clock (f_{CPU})	Watchdog Timer Clock[Note 1]
Normal operation	√	√	√	√	√	√
HALT mode	√	√	√	√	×	√
IDLE mode	√	√	×	×	×	×
STOP mode	×[Note 2]	×[Note 2]	×	×	×	×
During oscillation stabilization time count after release of STOP mode	√	×[Note 2] → √	×	×	×	×
During $\overline{\text{RESET}}$ pin input and subsequent oscillation stabilization time count	√	× → √	√	×[Note 3]	√	×

Notes 1. The peripheral clock ($f_{XX}/1,024$) is used as the watchdog timer clock.

2. Operation continues during on-chip debugging.

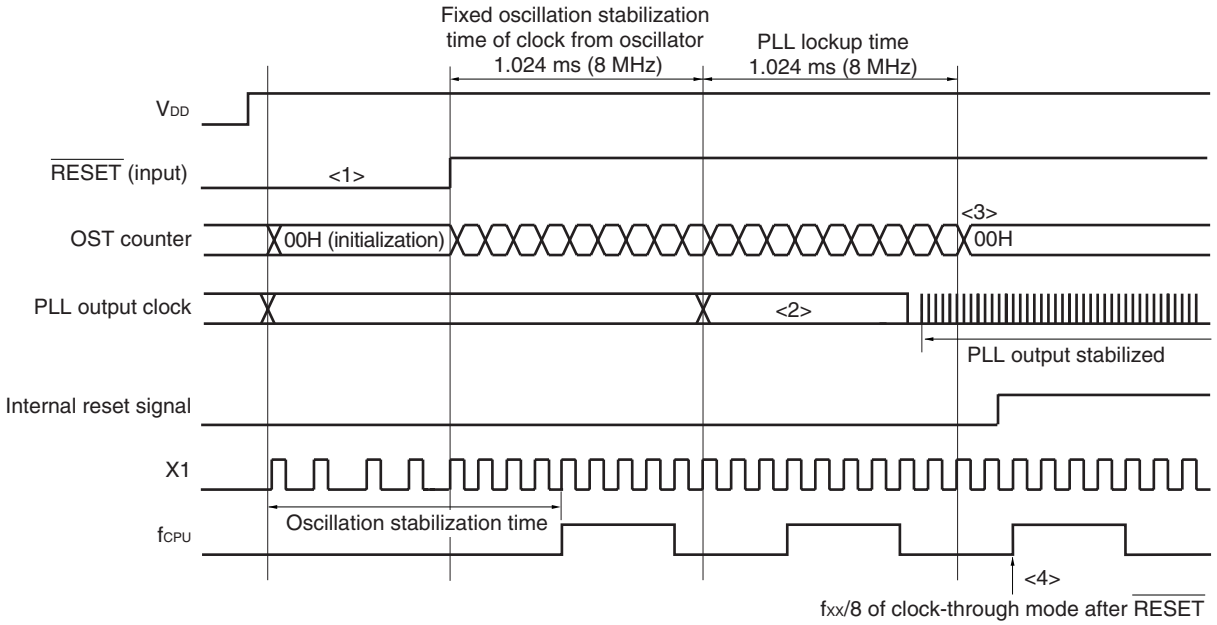
3. The watchdog timer clock is not output from the prescaler (PRS).

Remark √: Operating

×: Stopped

5.5.2 Operation timing

(1) Power on (power-on reset)



<1> The oscillator is activated during the $\overline{\text{RESET}}$ period that follows power application.
Make sure that the low-level width of the $\overline{\text{RESET}}$ signal is “Oscillation stabilization time of the resonator − Fixed oscillation stabilization time” or more, taking the oscillation stabilization time into consideration.
PLL stops during the $\overline{\text{RESET}}$ period and fixed oscillation stabilization time.

<2> When the fixed oscillation stabilization time that elapses after the $\overline{\text{RESET}}$ signal is released expires, PLL stop is released, and counting the lockup time starts.

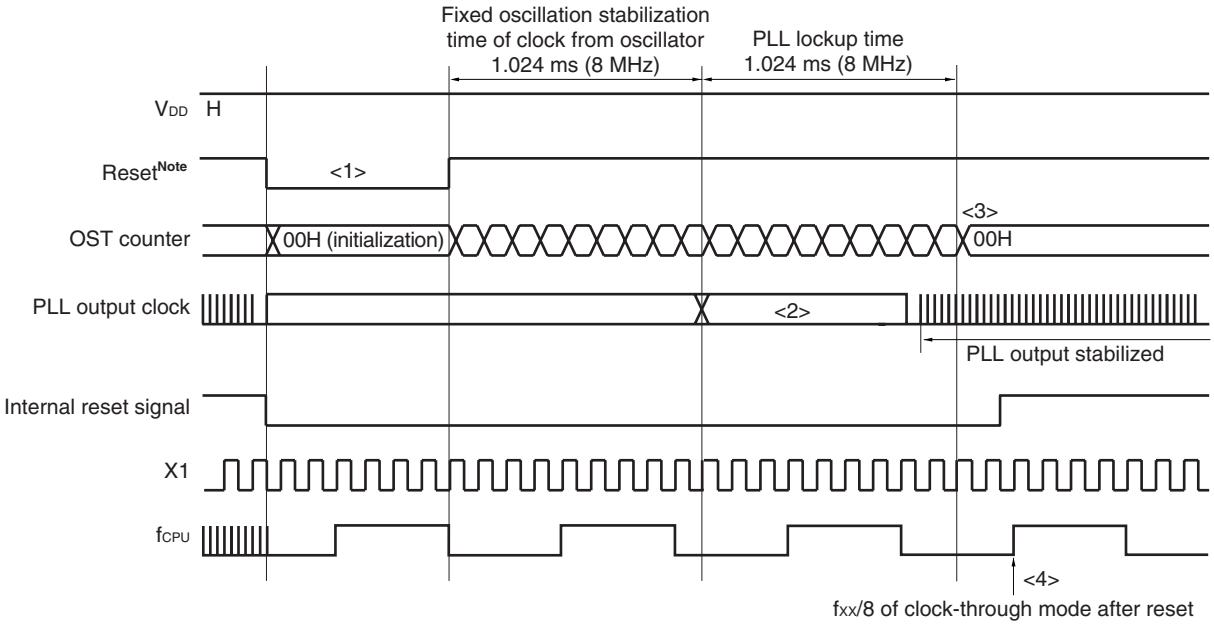
<3> PLL is locked when counting of the lockup time is over. The OST counter is initialized to 00H.

<4> When the lockup time expires, the CPU releases the reset signal and operates in the clock-through mode (f_X). The CPU operation clock (f_{CPU}) is $f_{XX}/8$. The PLL mode can be set by software.

Cautions 1. The clock generated by the oscillator starts oscillating during the $\overline{\text{RESET}}$ period. After the $\overline{\text{RESET}}$ signal is released, a specific wait time (fixed oscillation stabilization time) elapses.

2. To avoid malfunction due to noise, do not change the division ratio of the CPU operation clock (f_{CPU}) by using the PCC register before setting the PLL mode. Before changing the division ratio, be sure to select the PLL mode.

(2) Reset input with power on



<1> The oscillator continues operating during the reset period.
PLL stops during the reset period and fixed oscillation stabilization time.

<2> When the fixed oscillation stabilization time that elapses after the reset signal is released expires, PLL stop is released, and counting the lockup time starts.

<3> PLL is locked when counting of the lockup time is over. The OST counter is initialized to 00H.

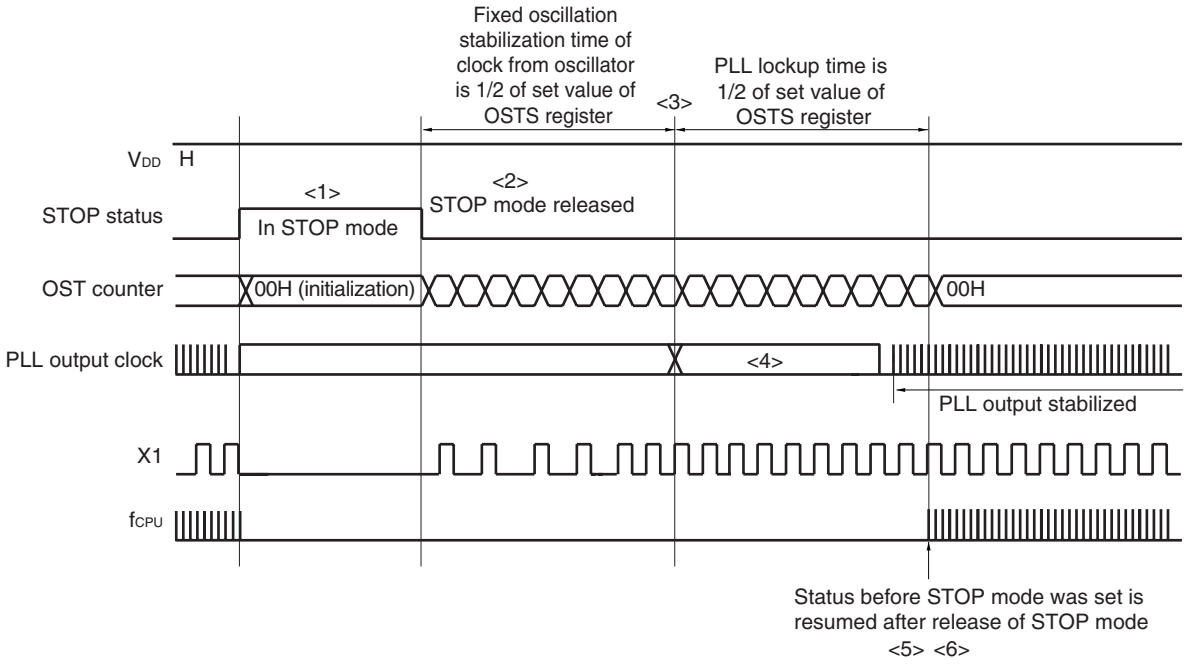
<4> When the lockup time expires, the CPU releases the reset signal and operates in the clock-through mode (f_X). The CPU operation clock (f_{CPU}) is $f_{XX}/8$. The PLL mode can be set by software.

Note $\overline{\text{RESET}}$ pin input or WDTRES signal generation

Cautions 1. The clock generated by the oscillator continues operating during a reset.
After the reset signal is released, a specific wait time (fixed oscillation stabilization time) elapses.

2. To avoid malfunction due to noise, do not change the division ratio of the CPU operation clock (f_{CPU}) by using the PCC register before setting the PLL mode. Before changing the division ratio, be sure to select the PLL mode.

(3) When releasing STOP mode by interrupt request



<1> When the STOP mode is set, both the oscillator and PLL stop.
At this time, PLL is stopped in the STOP mode. The OST counter is initialized.

<2> When the STOP mode is released, the oscillator is activated and the OST counter starts counting the oscillation stabilization time. At this time, PLL remains stopped.

<3> When half the oscillation stabilization time set to the OSTS has elapsed, PLL starts operating. The clock generated by the oscillator must be stabilized before PLL starts operating. The actual oscillation stabilization time is “1/2 the oscillation stabilization time”. Take this into consideration when setting a value to the OSTS register.

<4> After half the oscillation stabilization time has elapsed, the lockup wait time starts. The remaining count time of the OST counter is the lockup wait time.

<5> When the lockup time of PLL is over, clock supply to the internal circuitry is started. At this time, the status before the STOP mode was set is recovered.

<6> The operation to be performed when the STOP mode is released by $\overline{\text{RESET}}$ input is the same as that in **(1) Power on (power-on reset)** and **(2) Reset input with power on**.

5.6 Clock Monitor

(1) Clock monitor function

The clock monitor samples the clock generated by the oscillator, by using the internal oscillation clock. When it detects stop of oscillation, output of the timer for motor control goes into a high-impedance state (for details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**).

CHAPTER 6 16-BIT TIMER/EVENT COUNTER P (TMP)

Timer P (TMP) is a 16-bit timer/event counter.

The V850E/IA3 and V850E/IA4 incorporate TMP0 to TMP3.

6.1 Overview

The TMPn of channels are outlined below (n = 0 to 3).

Table 6-1. TMPn Overview

Item	TMP0	TMP1	TMP2	TMP3
Clock selection	8 ways	8 ways	8 ways	8 ways
Capture trigger input pin	2	None	2	None
External event count input pin	1	None	1	None
External trigger input pin	1	None	1	None
Timer counter	1	1	1	1
Capture/compare register	2	2 Note 1	2	2 Note 1
Capture/compare match interrupt request signal	2	2 Note 1	2	2 Note 1
<R> Overflow interrupt request signal	1	1	1	1
Timer output pin	2	None	1	**Note 2**

Notes 1. Compare function only

2. V850E/IA3: None

V850E/IA4: 1

6.2 Functions

The functions of TMPn that can be realized differ from one channel to another, as shown in the table below (n = 0 to 3).

Table 6-2. TMPn Functions

Function	TMP0	TMP1	TMP2	TMP3
Interval timer	√	√	√	√
External event counter	√	×	√	×
External trigger pulse output	√	×	√	**Note 1**
One-shot pulse output	√	×	√	**Note 1**
PWM output	√	×	√	**Note 2**
Free-running timer	√	√	√	√
Pulse width measurement	√	×	√	×
Timer tuning operation	√ (TMQ0)	**Note 3**	×	×

Notes 1. V850E/IA3: ×

V850E/IA4: √ (software trigger only, external trigger input cannot be used)

2. V850E/IA3: ×

V850E/IA4: √

3. V850E/IA3: ×

V850E/IA4: √ (TMQ1)

6.3 Configuration

TMPn includes the following hardware.

Table 6-3. Configuration of TMPn

Item	Configuration
Timer register	16-bit counter × 1
Registers	TMPn capture/compare registers 0, 1 (TPnCCR0, TPnCCR1) TMPn counter read buffer register (TPnCNT) CCR0 and CCR1 buffer registers
Timer input	Total 4 (TIP00[Note1], TIP01, TIP20[Note 1], TIP21 pins)[Note 2]
Timer output	Total 4 (TOP00, TOP01, TOP21, TOP31[Note 3] pins)[Note 4]
Control registers	TMPn control registers 0, 1 (TPnCTL0, TPnCTL1) TMPm I/O control register 0 (TPmIOC0) TMPk I/O control registers 1, 2 (TPkIOC1, TPkIOC2) TMPn option register 0 (TPnOPT0)

Notes 1. The TIP00 and TIP20 pins function alternately as a capture input signal, external event count input signal, and external trigger input signal.

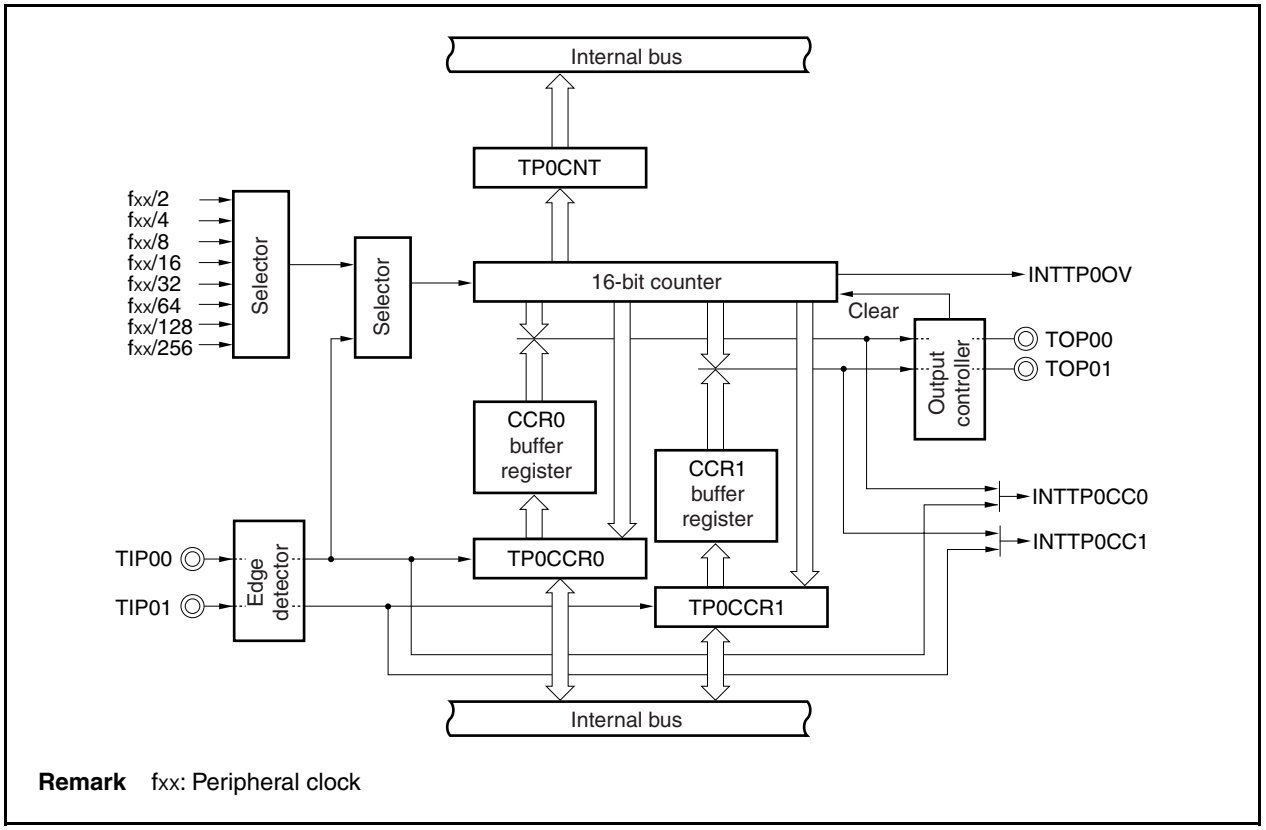
2. Not provided for TMP1 and TMP3

3. V850E/IA4 only

4. Not provided for TMP1

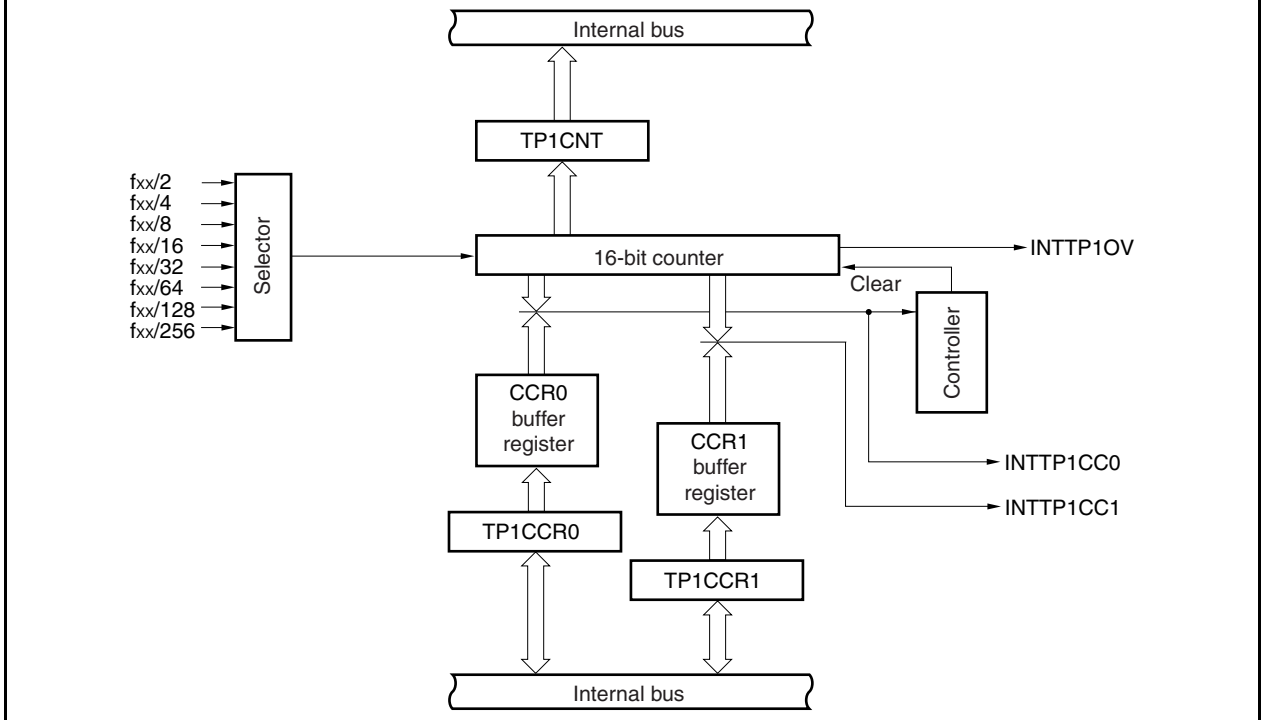
Remark V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2

Figure 6-1. TMP0 Block Diagram



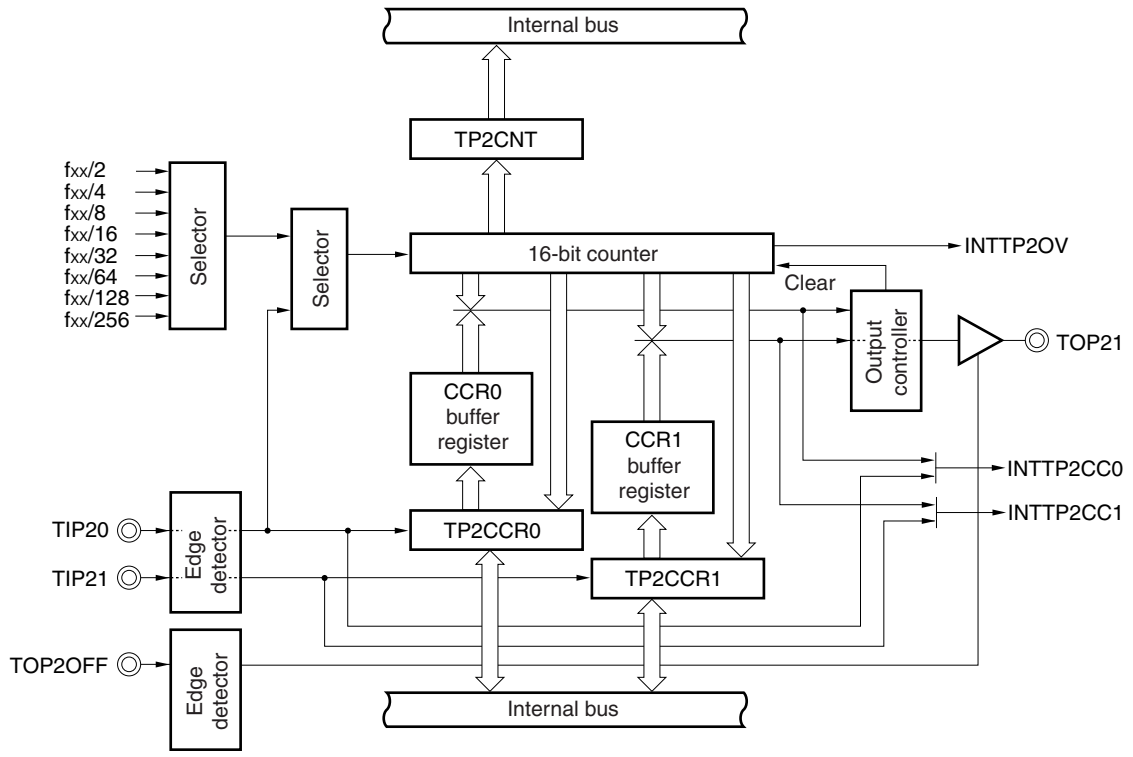
Remark fxx: Peripheral clock

Figure 6-2. TMP1 Block Diagram



Remark fxx: Peripheral clock

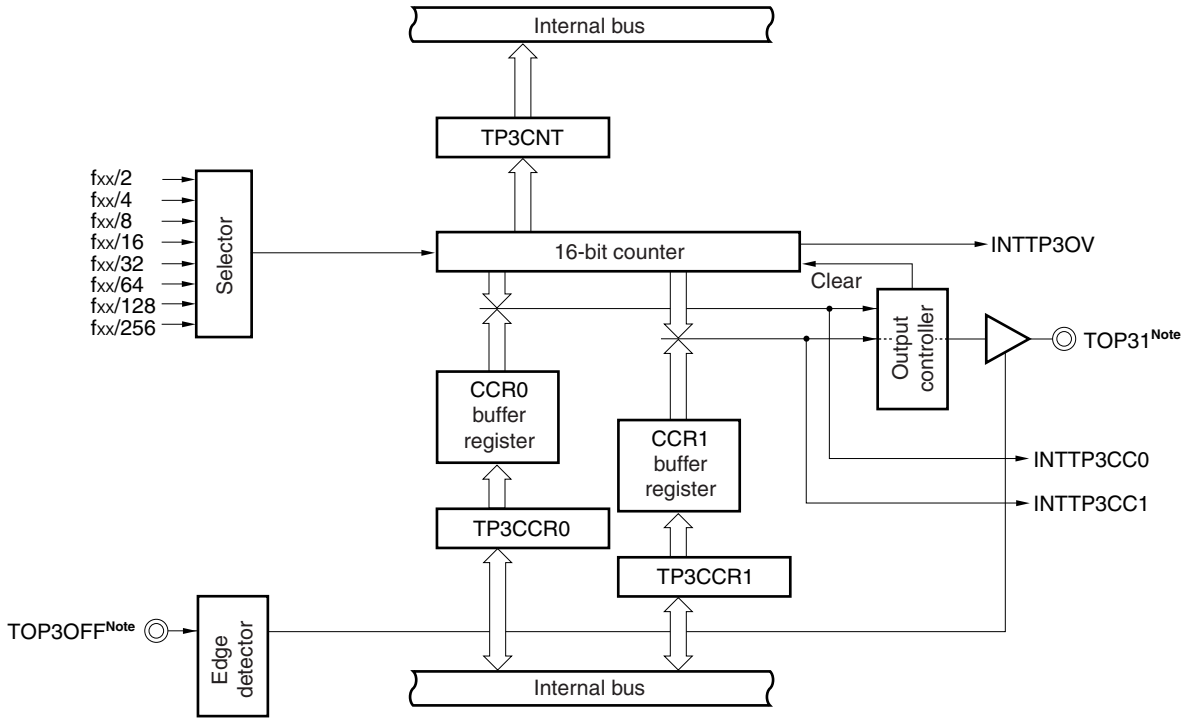
Figure 6-3. TMP2 Block Diagram



Remarks 1. fxx: Peripheral clock

2. For the TOP2OFF pin, see **10.3 (6) High-impedance output control registers 00, 01, 10, 11, 20, 21 (HZAmCTL0, HZAmCTL1)**.

Figure 6-4. TMP3 Block Diagram



Note V850E/IA4 only

Remarks 1. fxx: Peripheral clock

2. For the TOP3OFF pin, see **10.3 (6) High-impedance output control registers 00, 01, 10, 11, 20, 21 (HZAmCTL0, HZAmCTL1)**.

(1) 16-bit counter

This 16-bit counter can count internal clocks or external events.

The count value of this counter can be read by using the TPnCNT register.

When the TPnCTL0.TPnCE bit = 0, the value of the 16-bit counter is FFFFH. If the TPnCNT register is read at this time, 0000H is read.

The TPnCE bit is cleared to 0 after reset.

(2) CCR0 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TPnCCR0 register is used as a compare register, the value written to the TPnCCR0 register is transferred to the CCR0 buffer register. When the count value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTPnCC0) is generated.

The CCR0 buffer register cannot be read or written directly.

The TPnCCR0 register is cleared to 0000H after reset, and the CCR0 buffer register is cleared to 0000H.

(3) CCR1 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TPnCCR1 register is used as a compare register, the value written to the TPnCCR1 register is transferred to the CCR1 buffer register. When the count value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTPnCC1) is generated.

The CCR1 buffer register cannot be read or written directly.

The TPnCCR1 register is cleared to 0000H after reset, and the CCR1 buffer register is cleared to 0000H.

(4) Edge detector

This circuit detects the valid edges input to the TIP00, TIP01, TIP20, and TIP21 pins. No edge, rising edge, falling edge, or both the rising and falling edges can be selected as the valid edge by using the TP0IOC1, TP2IOC1, TP0IOC2, and TP2IOC2 registers.

(5) Output controller

This circuit controls the output of the TOP00, TOP01, TOP21, and TOP31 (V850E/IA4 only) pins. The output of the TOP00, TOP01, TOP21, and TOP31 (V850E/IA4 only) pins is controlled by the TP0IOC0, TP2IOC0, and TP3IOC0 (V850E/IA4 only) registers.

(6) Selector

This selector selects the count clock for the 16-bit counter. Eight types of internal clocks or an external event can be selected as the count clock.

6.4 Registers

(1) TMPn control register 0 (TPnCTL0)

The TPnCTL0 register is an 8-bit register that controls the operation of TMPn.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

The same value can always be written to the TPnCTL0 register by software.

After reset: 00H R/W Address: TP0CTL0 FFFFF640H, TP1CTL0 FFFFF660H, TP2CTL0 FFFFF680H, TP3CTL0 FFFFF6A0H

	<7>	6	5	4	3	2	1	0
TPnCTL0	TPnCE	0	0	0	0	TPnCKS2	TPnCKS1	TPnCKS0

V850E/IA3
n = 0 to 3
m = 0, 2

V850E/IA4
n = 0 to 3
m = 0, 2, 3

TPnCE	TMPn operation control
0	TMPn operation disabled (TMPn reset asynchronously[Note])
1	TMPn operation enabled. TMPn operation start

TPnCKS2	TPnCKS1	TPnCKS0	Internal count clock selection
0	0	0	$f_{XX}/2$
0	0	1	$f_{XX}/4$
0	1	0	$f_{XX}/8$
0	1	1	$f_{XX}/16$
1	0	0	$f_{XX}/32$
1	0	1	$f_{XX}/64$
1	1	0	$f_{XX}/128$
1	1	1	$f_{XX}/256$

Note The TPnOPT0.TPnOVF bit and 16-bit counter are reset simultaneously. Moreover, timer outputs (TOP00, TOP01, TOP21, and TOP31 (V850E/IA4 only) pins) as the 16-bit counter are reset to the TPmIOC0 register set status at the same time.

Cautions 1. Set the TPnCKS2 to TPnCKS0 bits when the TPnCE bit = 0.
When the value of the TPnCE bit is changed from 0 to 1, the TPnCKS2 to TPnCKS0 bits can be set simultaneously.
2. Be sure to clear bits 3 to 6 to "0".

Remark f_{XX}: Peripheral clock

(2) TMPn control register 1 (TPnCTL1)

The TPnCTL1 register is an 8-bit register that controls the TMPn operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

(1/2)

After reset: 00H R/W Address: TP0CTL1 FFFFF641H, TP1CTL1 FFFFF661H, TP2CTL1 FFFFF681H, TP3CTL1 FFFFF6A1H

	7	6	5	4	3	2	1	0
TPnCTL1	TPaSYE Note 1	TPmEST Note 2	TPkEEE Note 3	0	0	TPnMD2	TPnMD1	TPnMD0

V850E/IA3
n = 0 to 3
m = 0, 2
k = 0, 2
a = 0, 1

V850E/IA4
n = 0 to 3
m = 0, 2, 3
k = 0, 2
a = 0, 1

TPaSYE Note 1	Operation mode selection
0	TMPa single mode
1	Tuning operation mode (see **10.4.5**)
TMPa can be used only as an A/D conversion start trigger factor of A/D converters 0 and 1 during the tuning operation. In the tuning operation mode, this bit always operates in synchronization with TMQa.	

TPmEST Note 2	Software trigger control
0	–
1	Generate a valid signal for external trigger input. • In one-shot pulse output mode: A one-shot pulse is output with writing 1 to the TPmEST bit as the trigger. • In external trigger pulse output mode: A PWM waveform is output with writing 1 to the TPmEST bit as the trigger.
Read value of the TPmEST bit is always 0.	

TPkEEE	Count clock selection
0	Disable operation with external event count input (TIPk0 pin). (Perform counting with the count clock selected by the TPkCTL0.TPkCKS0 to TPkCTL0.TPkCKS2 bits.)
1	Enable operation with external event count input (TIPk0 pin) Note 4. (Perform counting at the valid edge of the external event count input signal.)
The TPkEEE bit selects whether counting is performed with the internal count clock or the valid edge of the external event count input.	

Notes 1. This bit can only be set in TMP0 and TMP1. Be sure to clear bit 7 of TMP2 and TMP3 to 0. For details of tuning operation mode, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

2. This bit can only be set in TMP0 and TMP2 in the V850E/IA3. Be sure to clear bit 6 of TMP1 and TMP3 to 0.

This bit can only be set in TMP0, TMP2, and TMP3 in the V850E/IA4. Be sure to clear bit 6 of TMP1 to 0.

3. This bit can only be set in TMP0 and TMP2. Be sure to clear bit 5 of TMP1 and TMP3 to 0.

4. Set the valid edge selection of capture trigger input (TIPk0 pin) to “No edge detection”.

(2/2)

TPnMD2	TPnMD1	TPnMD0	Timer mode selection[Note]
0	0	0	Interval timer mode
0	0	1	External event count mode
0	1	0	External trigger pulse output mode
0	1	1	One-shot pulse output mode
1	0	0	PWM output mode
1	0	1	Free-running timer mode
1	1	0	Pulse width measurement mode
1	1	1	Setting prohibited

Note The settings that can be realized differ from one channel to another. For details, see **Tables 6-8** to **6-11**.

Cautions 1. The TPmEST bit is valid only in the external trigger pulse output mode or one-shot pulse output mode. In any other mode, writing 1 to this bit is ignored.

2. External event count input is selected in the external event count mode regardless of the value of the TPkEEE bit.

<R> **3. Set the TPaSYE, TPkEEE, and TPnMD2 to TPnMD0 bits when the TPnCTL0.TPnCE bit = 0. (The same value can be written when the TPnCE bit = 1.) The operation is not guaranteed when rewriting is performed with the TPnCE bit = 1. If rewriting was mistakenly performed, clear the TPnCE bit to 0 and then set the bits again.**

4. Be sure to clear bits 3 and 4 to "0".

(3) TMPm I/O control register 0 (TPmIOC0)

The TPmIOC0 register is an 8-bit register that controls the timer output (TOP00, TOPm1 pins).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Remark TMP1 and TMP3 do not have the TP1IOC0 and TP3IOC0 registers in the V850E/IA3.
TMP1 does not have the TP1IOC0 register in the V850E/IA4.

(1/2)

After reset: 00H R/W Address: TP0IOC0 FFFFF642H, TP2IOC0 FFFFF682H, TP3IOC0 FFFFF6A2H[Note 1]

TPmIOC0

V850E/IA3 m = 0, 2

V850E/IA4 m = 0, 2, 3

7	6	5	4	3	<2>	1	<0>
0	0	0	0	TPmOL1	TPmOE1	TP0OL0[Note 2]	TP0OE0[Note 2]

TPmOL1	TOPm1 pin output level setting[Note 3]
0	TOPm1 pin starts output at high level.
1	TOPm1 pin starts output at low level.

TPmOE1	TOPm1 pin output setting
0	Timer output prohibited • Low level is output from the TOPm1 pin when the TPmOL1 bit = 0. • High level is output from the TOPm1 pin when the TPmOL1 bit = 1.
1	Timer output enabled (A pulse is output from the TOPm1 pin.)

TP0OL0[Note 2]	TOP00 pin output level setting[Note 3]
0	TOP00 pin starts output at high level.
1	TOP00 pin starts output at low level.

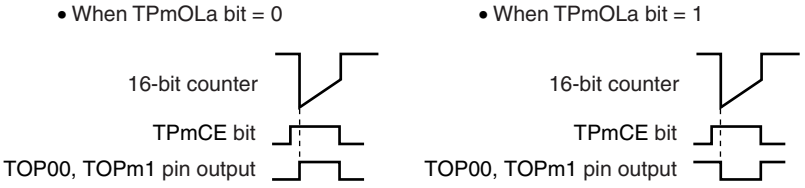
TP0OE0[Note 2]	TOP00 pin output setting
0	Timer output prohibited • Low level is output from the TOP00 pin when the TP0OL0 bit = 0. • High level is output from the TOP00 pin when the TP0OL0 bit = 1.
1	Timer output enabled (A pulse is output from the TOP00 pin.)

Notes 1. V850E/IA4 only

2. Valid only for TMP0. Be sure to clear bits 1 and 0 of TMP2 and TMP3 to 0.

3. The output level of the timer output pins (TOP00 and TOPm1) specified by the TPmOLa (a = 0, 1) bit is shown below (a = 0, 1).

<R>



<R>

Caution 1. If the setting of the TPmIOC0 register is changed when TOP00 and TOPm1 are set in the output mode, the output of the pins change. Set the port in the input mode and make the pin output go into a high-impedance state, noting changes in the pin status.

(2/2)

Cautions 2. Rewrite the TPmOL1, TPmOE1, TP0OL0, and TP0OE0 bits when the TPmCTL0.TPnCE bit = 0. (The same value can be written when the TPmCE bit = 1.) If rewriting was mistakenly performed, clear the TPmCE bit to 0 and then set the bits again.

3. Even if the TP0OL0 or TPmOL1 bit is manipulated when the TPmCE, TP0OE0, and TPmOE1 bits are 0, the output level of the TOP00 and TOPm1 pins changes.

(4) TMPk I/O control register 1 (TPkIOC1)

The TPkIOC1 register is an 8-bit register that controls the valid edge for the capture trigger input signals (TIPk0, TIPk1 pins).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Remark TMP1 and TMP3 do not have the TP1IOC1 and TP3IOC1 registers.

After reset: 00H R/W Address: TP0IOC1 FFFFF643H, TP2IOC1 FFFFF683H

	7	6	5	4	3	2	1	0
TPkIOC1	0	0	0	0	TPkIS3	TPkIS2	TPkIS1	TPkIS0

(k = 0, 2)

TPkIS3	TPkIS2	Capture trigger input signal (TIPk1 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TPkIS1	TPkIS0	Capture trigger input signal (TIPk0 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

Cautions 1. Rewrite the TPkIS3 to TPkIS0 bits when the TPkCTL0.TPkCE bit = 0.
(The same value can be written when the TPkCE bit = 1.) If rewriting was mistakenly performed, clear the TPkCE bit to 0 and then set the bits again.

2. The TPkIS3 to TPkIS0 bits are valid only in the free-running timer mode (only when the TPnOPT0.TPkCCS1, TPkCCS0 bits = 11) and the pulse width measurement mode. In all other modes, a capture operation is not possible (TMP0, TMP2 only).

(5) TMPk I/O control register 2 (TPkIOC2)

The TPkIOC2 register is an 8-bit register that controls the valid edge for the external event count input signal (TIPk0 pin) and external trigger input signal (TIPk0 pin).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Remark TMP1 and TMP3 do not have the TP1IOC2 and TP3IOC2 registers.

After reset: 00H R/W Address: TP0IOC2 FFFFF644H, TP2IOC2 FFFFF684H

	7	6	5	4	3	2	1	0
TPkIOC2	0	0	0	0	TPkEES1	TPkEES0	TPkETS1	TPkETS0

(k = 0, 2)

TPkEES1	TPkEES0	External event count input signal (TIPk0 pin) valid edge setting
0	0	No edge detection (external event count invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TPkETS1	TPkETS0	External trigger input signal (TIPk0 pin) valid edge setting
0	0	No edge detection (external trigger invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

Cautions 1. Rewrite the TPkEES1, TPkEES0, TPkETS1, and TPkETS0 bits when the TPkCTL0.TPkCE bit = 0. (The same value can be written when the TPkCE bit = 1.) If rewriting was mistakenly performed, clear the TPkCE bit to 0 and then set the bits again.

2. The TPkEES1 and TPkEES0 bits are valid only when the TPkCTL1.TPkEEE bit = 1 or when the external event count mode (TPkCTL1.TPkMD2 to TPkCTL1.TPkMD0 bits = 001) has been set.

3. The TPkETS1 and TPkETS0 bits are valid only in the external trigger pulse output mode or one-shot pulse output mode.

(6) TMPn option register 0 (TPnOPT0)

The TPnOPT0 register is an 8-bit register that sets the capture/compare operation and detects overflow.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: TP0OPT0 FFFFF645H, TP1OPT0 FFFFF665H, TP2OPT0 FFFFF685H, TP3OPT0 FFFFF6A5H

TPnOPT0 (n = 0 to 3, k = 0, 2)

7	6	5	4	3	2	1	<0>
0	0	TPkCCS1[Note]	TPkCCS0[Note]	0	0	0	TPnOVF

TPkCCS1[Note]	TPkCCR1 register capture/compare selection
0	Compare register selected
1	Capture register selected (Cleared by TPkCTL0.TPkCE bit = 0)
The TPkCCS1 bit setting is valid only in the free-running timer mode.	

TPkCCS0[Note]	TPkCCR0 register capture/compare selection
0	Compare register selected
1	Capture register selected (Cleared by TPkCTL0.TPkCE bit = 0)
The TPkCCS0 bit setting is valid only in the free-running timer mode.	

TPnOVF	TMPn overflow detection flag
Set (1)	Overflow occurred
Reset (0)	0 written to TPnOVF bit or TPnCTL0.TPnCE bit = 0

- The TPnOVF bit is set to 1 when the 16-bit counter value overflows from FFFFH to 0000H in the free-running timer mode or the pulse width measurement mode.
- An overflow interrupt request signal (INTTPnOV) is generated at the same time that the TPnOVF bit is set to 1. The INTTPnOV signal is not generated in modes other than the free-running timer mode and the pulse width measurement mode.
- The TPnOVF bit is not cleared to 0 even when the TPnOVF bit or the TPnOPT0 register is read when the TPnOVF bit = 1.
- Before clearing the TPnOVF bit to 0 after generation of the INTTPnOV signal, be sure to confirm (by reading) that the TPnOVF bit is set to 1.
- The TPnOVF bit can be both read and written, but the TPnOVF bit cannot be set to 1 by software. Writing 1 has no effect on the operation of TMPn.

Note Valid only for TMP0 and TMP2. Be sure to clear bits 5 and 4 of TMP1 and TMP3 to 0.

Cautions 1. Rewrite the TPkCCS1 and TPkCCS0 bits when the TPkCE bit = 0. (The same value can be written when the TPkCE bit = 1.) If rewriting was mistakenly performed, clear the TPkCE bit to 0 and then set the bits again.

2. Be sure to clear bits 1 to 3, 6, and 7 to "0".

(7) TMPn capture/compare register 0 (TPnCCR0)

The TP0CCR0 and TP2CCR0 registers are 16-bit registers that can be used as capture registers or compare registers depending on the mode. The TP1CCR0 and TP3CCR0 registers are 16-bit registers that can only be used as compare registers.

The TP0CCR0 and TP2CCR0 registers can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TP0OPT0.TP0CCS0 or TP2OPT0.TP2CCS0 bit. In the pulse width measurement mode, the TPnCCR0 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TPnCCR0 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TP0CCR0 FFFFF646H, TP1CCR0 FFFFF666H, TP2CCR0 FFFFF686H, TP3CCR0 FFFFF6A6H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPnCCR0																

(n = 0 to 3)

(a) Function as compare register

The TPnCCR0 register can be rewritten even when the TPnCTL0.TPnCE bit = 1.

The set value of the TPnCCR0 register is transferred to the CCR0 buffer register. When the value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTPnCC0) is generated. If TOP00 pin output is enabled at this time, the output of the TOP00 pin is inverted (TOP10, TOP20, and TOP30 pins are not provided).

When the TPnCCR0 register is used as a cycle register in the interval timer mode, external event count mode, external trigger pulse output mode, one-shot pulse output mode, or PWM output mode, the value of the 16-bit counter is cleared (0000H) if its count value matches the value of the CCR0 buffer register.

The compare register is not cleared by the TPnCTL0.TPnCE bit = 0.

(b) Function as capture register (TP0CCR0 and TP2CCR0 registers only)

When the TPkCCR0 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TPkCCR0 register if the valid edge of the capture trigger input pin (TIPk0 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TPkCCR0 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIPk0 pin) is detected.

Even if the capture operation and reading the TPkCCR0 register conflict, the correct value of the TPkCCR0 register can be read.

The capture register is cleared by the TPkCTL0.TPkCE bit = 0.

Remark k = 0, 2

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 6-4. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 2]	Compare register	Batch write[Note 4]
One-shot pulse output[Note 2]	Compare register	Anytime write
PWM output[Note 3]	Compare register	Batch write[Note 4]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMP0 and TMP2 only

2. TMP0 and TMP2 only (also TMP3 in software trigger mode in the V850E/IA4)

3. TMP0 and TMP2 only (also TMP3 in the V850E/IA4)

4. Writing to the TPnCCR1 register is the trigger.

Remark For anytime write and batch write, see **6.6 (2) Anytime write and batch write**.

(8) TMPn capture/compare register 1 (TPnCCR1)

The TP0CCR1 and TP2CCR1 registers are 16-bit registers that can be used as capture registers or compare registers depending on the mode. The TP1CCR1 and TP3CCR1 registers are 16-bit registers that can only be used as compare registers.

The TP0CCR1 and TP2CCR1 registers can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TP0OPT0.TP0CCS1 or TP2OPT0.TP2CCS1 bit. In the pulse width measurement mode, the TPnCCR1 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TPnCCR1 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TP0CCR1 FFFFF648H, TP1CCR1 FFFFF668H, TP2CCR1 FFFFF688H, TP3CCR1 FFFFF6A8H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPnCCR1																

(n = 0 to 3)

(a) Function as compare register

The TPnCCR1 register can be rewritten even when the TPnCTL0.TPnCE bit = 1.

The set value of the TPnCCR1 register is transferred to the CCR1 buffer register. When the value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTPnCC1) is generated. If TOPm1 pin output is enabled at this time, the output of the TOPm1 pin is inverted (the TOP11 pin is not provided).

The compare register is not cleared by the TPnCTL0.TPnCE bit = 0.

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

(b) Function as capture register (TP0CCR1 and TP2CCR1 registers only)

When the TPkCCR1 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TPkCCR1 register if the valid edge of the capture trigger input pin (TIPk1 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TPkCCR1 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIPk1 pin) is detected.

Even if the capture operation and reading the TPkCCR1 register conflict, the correct value of the TPkCCR1 register can be read.

The capture register is cleared by the TPkCTL0.TPkCE bit = 0.

Remark k = 0, 2

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 6-5. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 2]	Compare register	Batch write[Note 4]
One-shot pulse output[Note 2]	Compare register	Anytime write
PWM output[Note 3]	Compare register	Batch write[Note 4]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMP0 and TMP2 only

2. TMP0 and TMP2 only (also TMP3 in software trigger mode in the V850E/IA4)

3. TMP0 and TMP2 only (also TMP3 in the V850E/IA4)

4. Writing to the TPnCCR1 register is the trigger.

Remark For anytime write and batch write, see **6.6 (2) Anytime write and batch write**.

(9) TMPn counter read buffer register (TPnCNT)

The TPnCNT register is a read buffer register that can read the count value of the 16-bit counter.

If this register is read when the TPnCTL0.TPnCE bit = 1, the count value of the 16-bit counter can be read.

This register is read-only, in 16-bit units.

The value of the TPnCNT register is cleared to 0000H when the TPnCE bit = 0. If the TPnCNT register is read at this time, the value of the 16-bit counter (FFFFH) is not read, but 0000H is read.

The value of the TPnCE bit is cleared to 0 after reset, and the TPnCNT register is cleared to 0000H.

After reset: 0000H R Address: TP0CNT FFFFF64AH, TP1CNT FFFFF66AH, TP2CNT FFFFF68AH, TP3CNT FFFFF6AAH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TPnCNT																

(n = 0 to 3)

6.5 Timer Output Operations

The following table shows the operations and output levels of the TOP00 and TOPm1 pins.

Table 6-6. Timer Output Control in Each Mode

Operation Mode	TOPm1 Pin	TOP00 Pin
Interval timer mode	PWM output	
External event count mode	None	
External trigger pulse output mode	External trigger pulse output	PWM output
One-shot pulse output mode	One-shot pulse output	
PWM output mode	PWM output	
Free-running timer mode	PWM output (only when compare function is used)	
Pulse width measurement mode	None	

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

Table 6-7. Truth Table of TOP00 and TOPm1 Pins Under Control of Timer Output Control Bits

TPmIOC0.TPmOLa Bit	TPmIOC0.TPmOEa Bit	TPmCTL0.TPmCE Bit	Level of TOPma Pin
0	0	×	Low-level output
	1	0	Low-level output
		1	Low level immediately before counting, high level after counting is started
1	0	×	High-level output
	1	0	High-level output
		1	High level immediately before counting, low level after counting is started

Remark V850E/IA3: a = 0 or 1 when m = 0
a = 1 when m = 2
V850E/IA4: a = 0 or 1 when m = 0
a = 1 when m = 2 or 3

6.6 Operation

The functions of TMPn that can be realized differ from one channel to another. The functions of each channel are shown below.

Table 6-8. TMP0 Specifications in Each Mode

Operation	TP0CTL1.TP0EST Bit (Software Trigger Bit)	TIP00 Pin (External Trigger Input)	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode[Note 1]	Invalid	Invalid	Compare only	Anytime write
External trigger pulse output mode[Note 2]	Valid	Valid	Compare only	Batch write
One-shot pulse output mode[Note 2]	Valid	Valid	Compare only	Anytime write
PWM output mode	Invalid	Invalid	Compare only	Batch write
Free-running timer mode	Invalid	Invalid	Switchable	Anytime write
Pulse width measurement mode[Note 2]	Invalid	Invalid	Capture only	Not applicable

Notes 1. When using the external event count mode, set the TIP00 pin capture trigger input valid edge selection to “No edge detection”. (Clear the TP0IOC1.TP0IS1 and TP0IOC1.TP0IS0 bits to 00.)

2. When using the external trigger pulse output mode, one-shot pulse output mode, and pulse width measurement mode, select the internal clock as the count clock (by clearing the TP0CTL1.TP0EEE bit to 0).

Remarks 1. The TIP00 pin functions alternately as a capture trigger input, external event count input, and external trigger input.

2. TMP0 has a function to execute tuning with TMQ0. For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

Table 6-9. TMP1 Specifications in Each Mode

Operation	Software Trigger Bit	External Trigger Input	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode	None			
External trigger pulse output mode	None			
One-shot pulse output mode	None			
PWM output mode	None			
Free-running timer mode	Invalid	Invalid	Compare only	Anytime write
Pulse width measurement mode	None			

Remarks 1. TMP1 does not have timer input pins (TIP10, TIP11) and timer output pins (TOP10, TOP11). It has interrupt request signals (INTTP1CC0, INTTP1CC1) indicating a match between the value of the 16-bit counter and the values of the TP1CCR0 and TP1CCR1 registers.

2. TMP1 has a function to execute tuning with TMQ1 (V850E/IA4 only). For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

Table 6-10. TMP2 Specifications in Each Mode

Operation	TP2CTL1.TP2EST Bit (Software Trigger Bit)	TIP20 Pin (External Trigger Input)	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode[Note 1]	Invalid	Invalid	Compare only	Anytime write
External trigger pulse output mode[Note 2]	Valid	Valid	Compare only	Batch write
One-shot pulse output mode[Note 2]	Valid	Valid	Compare only	Anytime write
PWM output mode	Invalid	Invalid	Compare only	Batch write
Free-running timer mode	Invalid	Invalid	Switchable	Anytime write
Pulse width measurement mode[Note 2]	Invalid	Invalid	Capture only	Not applicable

Notes 1. When using the external event count mode, set the TIP20 pin capture trigger input valid edge selection to “No edge detection”. (Clear the TP2IOC1.TP2IS1 and TO2IOC1.TP2IS0 bits to 00.)

2. When using the external trigger pulse output mode and one-shot pulse output mode, select the internal clock as the count clock (by clearing the TP2CTL1.TP2EEE bit to 0).

Remark The TIP20 pin functions alternately as a capture trigger input, external event count input, and external trigger input.

Table 6-11. TMP3 Specifications in Each Mode

Operation	TP3CTL1.TP3EST Bit (Software Trigger Bit)	External Trigger Input	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode	None			
External trigger pulse output mode[Notes 1, 2]	Valid	Invalid	Compare only	Batch write
One-shot pulse output mode[Notes 1, 2]	Valid	Invalid	Compare only	Anytime write
PWM output mode[Note 1]	Invalid	Invalid	Compare only	Batch write
Free-running mode	Invalid	Invalid	Compare only	Anytime write
Pulse width measurement mode	None			

Notes 1. V850E/IA4 only

2. When using the external trigger pulse output mode, one-shot pulse output mode, an external trigger cannot be input. Only a software trigger (set by the TP3CTL1.TP3EST bit) can be used.

Remark TMP3 does not have timer input pins (TIP30, TIP31) and timer output pin (V850E/IA3: TOP30, TOP31, V850E/IA4: TOP30). The match interrupt request signals (INTTP3CC0, INTTP3CC1) of the 16-bit counter and the TP3CCR0 and TP3CCR1 registers are provided.

(1) Counter basic operation

This section explains the basic operation of the 16-bit counter. For details, refer to the description of the operation in each mode.

Remark n = 0 to 3, k = 0, 2

<R> **(a) Counter start operation**

- In external event count mode
  When the TPkCTL0.TPkCE bit is set from 0 to 1, the 16-bit counter is set to 0000H.
  After that, it counts up from 0001H to 0002H, 0003H, and so on, each time the valid edge of an external event count input (TIPk0) is detected.
- In modes other than the above
  The 16-bit counter of TMPn starts counting from the default value FFFFH.
  It counts up from FFFFH to 0000H, 0001H, 0002H, 0003H, and so on.

(b) Clear operation

The 16-bit counter is cleared to 0000H when its value matches the value of the compare register and is cleared, and when its value is captured and cleared. The counting operation from FFFFH to 0000H that takes place immediately after the counter has started counting or when the counter overflows is not a clearing operation. Therefore, the INTTPnCC0 and INTTPnCC1 interrupt signals are not generated.

(c) Overflow operation

The 16-bit counter overflows when the counter counts up from FFFFH to 0000H in the free-running timer mode or pulse width measurement mode. If the counter overflows, the TPnOPT0.TPnOVF bit is set to 1 and an interrupt request signal (INTTPnOV) is generated. Note that the INTTPnOV signal is not generated under the following conditions.

- Immediately after a counting operation has been started
- If the counter value matches the compare value FFFFH and is cleared
- When FFFFH is captured and cleared in the pulse width measurement mode and the counter counts up from FFFFH to 0000H

Caution After the overflow interrupt request signal (INTTPnOV) has been generated, be sure to check that the overflow flag (TPnOVF bit) is set to 1.

(d) Counter read operation during counting operation

The value of the 16-bit counter of TMPn can be read by using the TPnCNT register during the count operation. When the TPnCTL0.TPnCE bit = 1, the value of the 16-bit counter can be read by reading the TPnCNT register. However, when the TPnCTL0.TPnCE bit = 0, the 16-bit counter is FFFFH and the TPnCNT register is 0000H.

(e) Interrupt operation

TMPn generates the following three types of interrupt request signals.

- INTTPnCC0 interrupt: This signal functions as a match interrupt request signal of the CCR0 buffer register and as a capture interrupt request signal to the TPnCCR0 register.
- INTTPnCC1 interrupt: This signal functions as a match interrupt request signal of the CCR1 buffer register and as a capture interrupt request signal to the TPnCCR1 register.
- INTTPnOV interrupt: This signal functions as an overflow interrupt request signal.

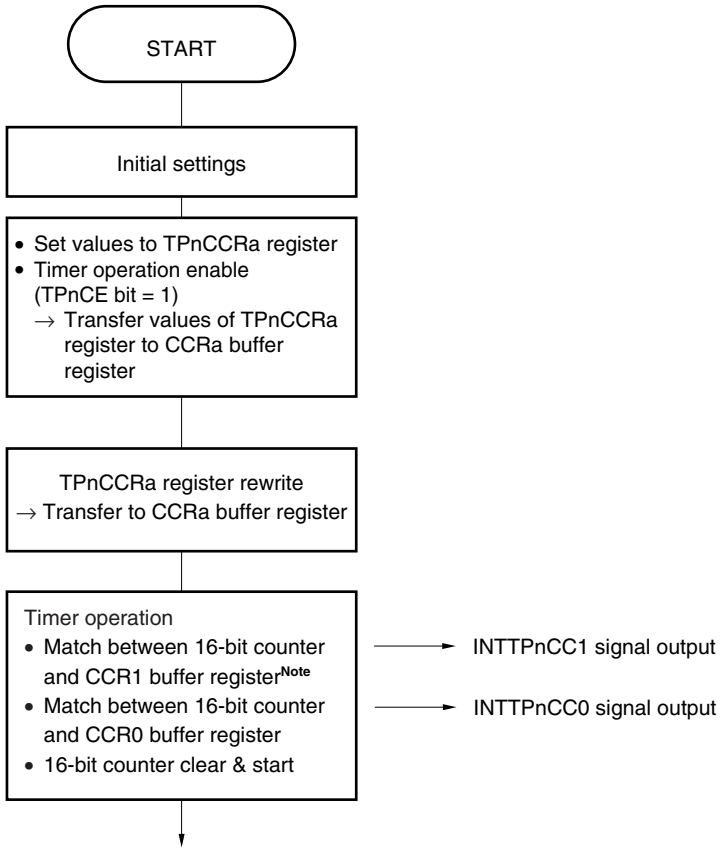
(2) Anytime write and batch write

The TPnCCR0 and TPnCCR1 registers in TMPn can be rewritten during timer operation (TPnCTL0.TPnCE bit = 1), but the write method (anytime write, batch write) of the CCR0 and CCR1 buffer registers differs depending on the mode.

(a) Anytime write

In this mode, data is transferred at any time from the TPnCCR0 and TPnCCR1 registers to the CCR0 and CCR1 buffer registers during timer operation (n = 0 to 3).

Figure 6-5. Flowchart of Basic Operation for Anytime Write



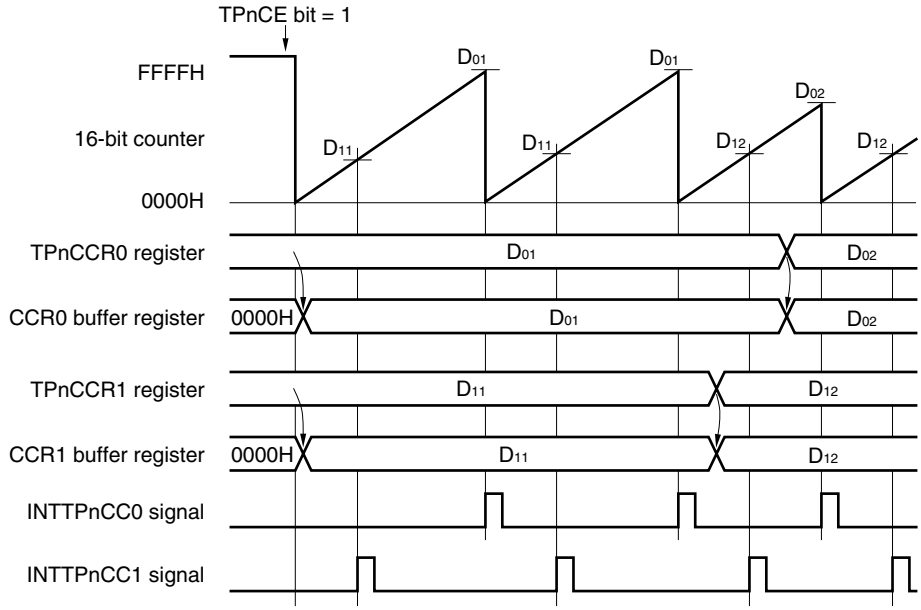
Note The 16-bit counter is not cleared upon a match between the value of the16-bit counter and the value of the CCR1 buffer register. It is cleared upon a match between the value of the 16-bit counter and the value of the CCR0 buffer register.

Remarks 1. The above flowchart illustrates an example of the operation in the interval timer mode.

2. n = 0 to 3

a = 0, 1

Figure 6-6. Timing of Anytime Write



Remarks 1. D_{01}, D_{02}: Setting values of TPnCCR0 register
D_{11}, D_{12}: Setting values of TPnCCR1 register

2. The above timing chart illustrates an example of the operation in the interval timer mode.

3. n = 0 to 3

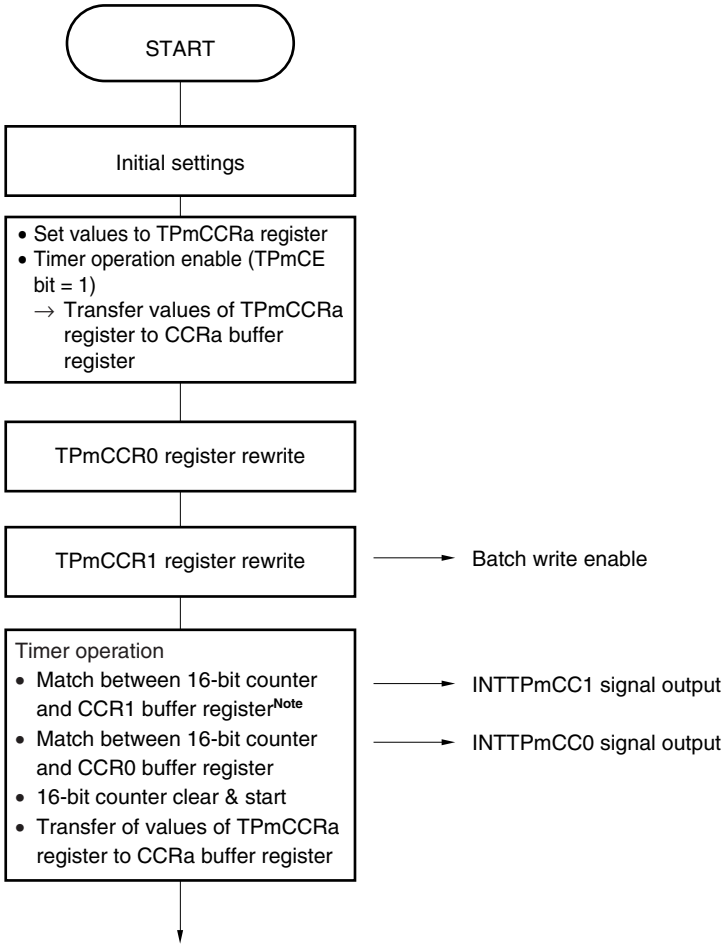
(b) Batch write

In this mode, data is transferred all at once from the TPmCCR0 and TPmCCR1 registers to the CCR0 and CCR1 buffer registers during timer operation. This data is transferred upon a match between the value of the CCR0 buffer register and the value of the 16-bit counter. Transfer is enabled by writing to the TPmCCR1 register.

Whether to enable or disable the next transfer timing is controlled by writing or not writing to the TPmCCR1 register.

In order for the setting value when the TPmCCR0 and TPmCCR1 registers are rewritten to become the 16-bit counter comparison value (in other words, in order for this value to be transferred to the CCR0 and CCR1 buffer registers), it is necessary to rewrite the TPmCCR0 register and then write to the TPmCCR1 register before the 16-bit counter value and the CCR0 buffer register value match. Therefore, the values of the TPmCCR0 and TPmCCR1 registers are transferred to the CCR0 and CCR1 buffer registers upon a match between the count value of the 16-bit counter and the value of the CCR0 buffer register. Thus even when wishing only to rewrite the value of the TPmCCR0 register, also write the same value (same as preset value of the TPmCCR1 register) to the TPmCCR1 register.

Figure 6-7. Flowchart of Basic Operation for Batch Write



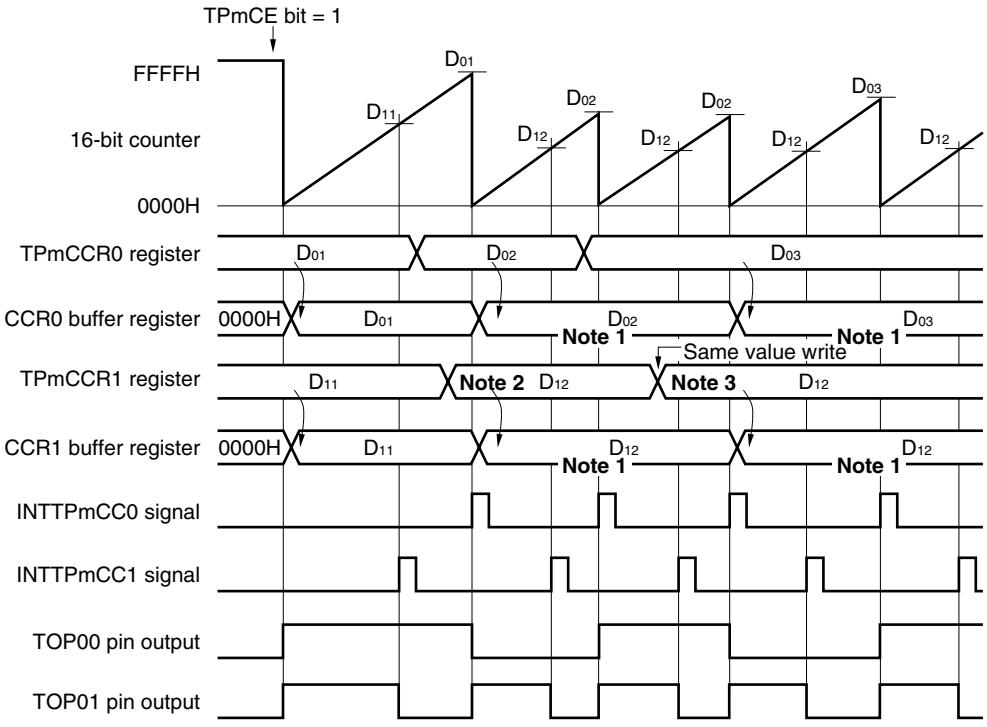
Note The 16-bit counter is not cleared upon a match between the value of the16-bit counter and the value of the CCR1 buffer register. It is cleared upon a match between the value of the 16-bit counter and the value of the CCR0 buffer register.

Caution Writing to the TPmCCR1 register includes enabling of batch write. Thus, rewrite the TPmCCR1 register after rewriting the TPmCCR0 register.

Remarks 1. The above flowchart illustrates an example of the operation in the PWM output mode.

2. V850E/IA3: m = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, a = 0, 1

Figure 6-8. Timing of Batch Write



Notes 1. Because the TPmCCR1 register was not rewritten, D_{03} is not transferred.

2. Because the TPmCCR1 register has been written (D_{12}), data is transferred to the CCR1 buffer register upon a match between the value of the 16-bit counter and the value of the TPmCCR0 register (D_{01}).

3. Because the TPmCCR1 register has been written (D_{12}), data is transferred to the CCR1 buffer register upon a match between the value of the 16-bit counter and the value of the TPmCCR0 register (D_{02}).

Remarks 1. D_{01}, D_{02}, D_{03}: Setting values of TPmCCR0 register
D_{11}, D_{12}: Setting values of TPmCCR1 register

2. The above timing chart illustrates an example of the operation in the PWM output mode.

3. V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

6.6.1 Interval timer mode (TPnMD2 to TPnMD0 bits = 000)

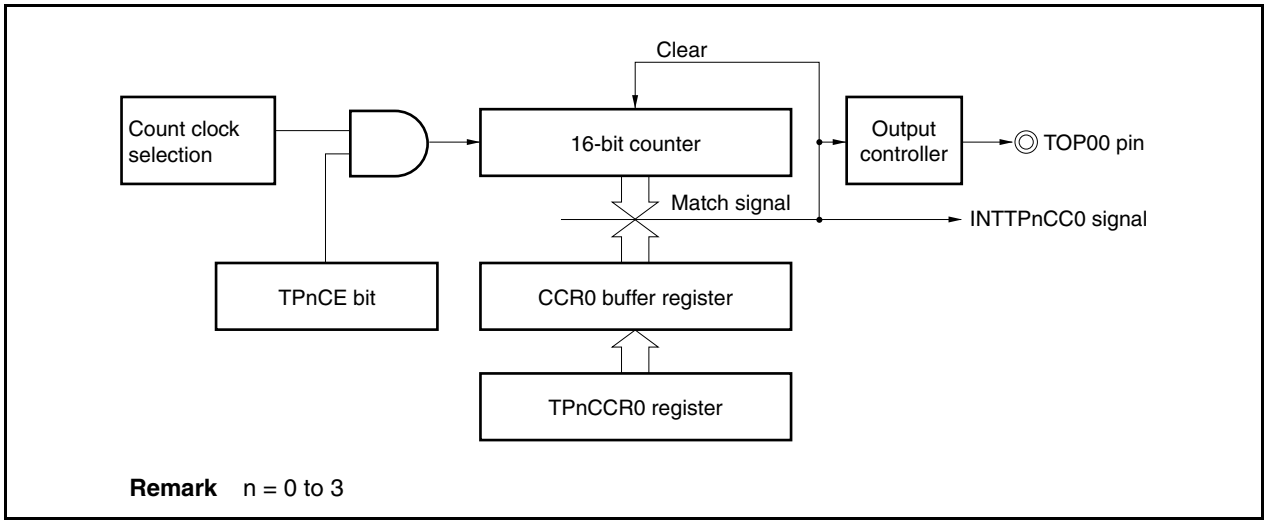
In the interval timer mode, an interrupt request signal (INTTPnCC0) is generated at the interval set by the TPnCCR0 register if the TPnCTL0.TPnCE bit is set to 1. A PWM waveform with a duty factor of 50% whose half cycle is equal to the interval can be output from the TOP00 pin (TMP0 only).

The TPnCCR1 register is not used in the interval timer mode. However, the set value of the TPnCCR1 register is transferred to the CCR1 buffer register, and when the count value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTPnCC1) is generated. In addition, a PWM waveform with a duty factor of 50%, which is inverted when the INTTPmCC1 signal is generated, can be output from the TOPm1 pin.

The value of the TPnCCR0 and TPnCCR1 registers can be rewritten even while the timer is operating.

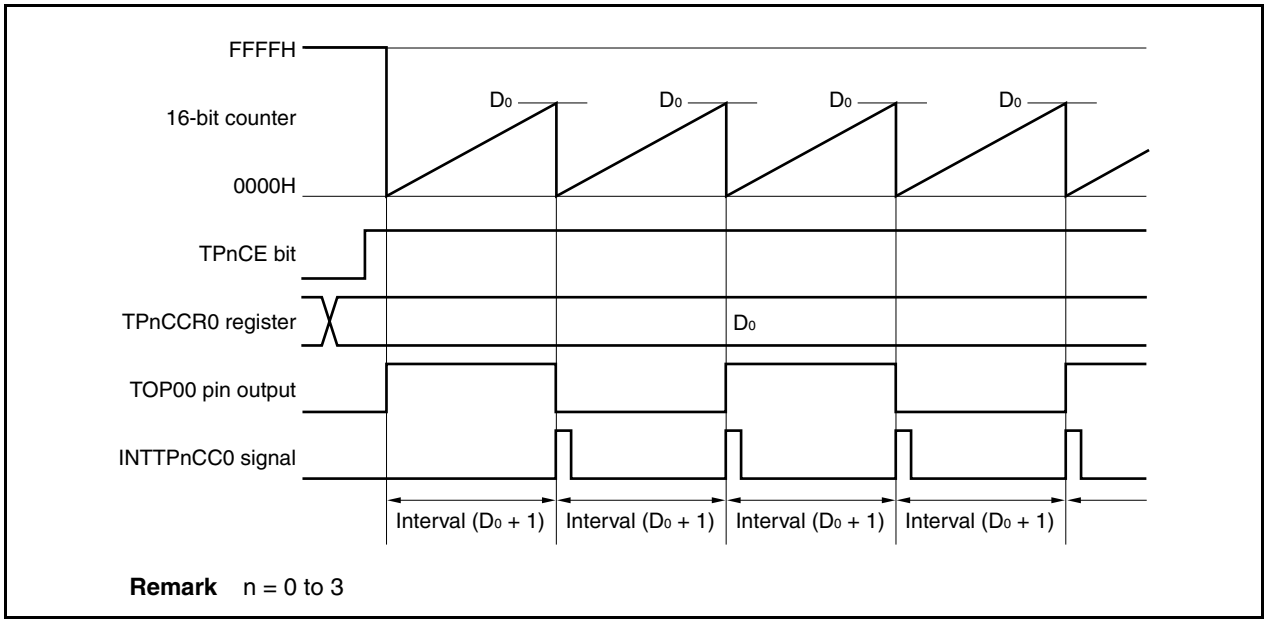
Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

Figure 6-9. Configuration of Interval Timer



Remark n = 0 to 3

Figure 6-10. Basic Timing of Operation in Interval Timer Mode



Remark n = 0 to 3

When the TPnCE bit is set to 1, the value of the 16-bit counter is cleared from FFFFH to 0000H in synchronization with the count clock, and the counter starts counting. At this time, the output of the TOP00 pin is inverted. Additionally, the set value of the TPnCCR0 register is transferred to the CCR0 buffer register.

When the count value of the 16-bit counter matches the value of the CCR0 buffer register, the 16-bit counter is cleared to 0000H, the output of the TOP00 pin is inverted, and a compare match interrupt request signal (INTTPnCC0) is generated.

The interval can be calculated by the following expression.

Interval = (Set value of TPnCCR0 register + 1) × Count clock cycle

Remark n = 0 to 3

Figure 6-11. Register Setting for Interval Timer Mode Operation (1/2)

(a) TMPn control register 0 (TPnCTL0)

	TPnCE					TPnCKS2	TPnCKS1	TPnCKS0
TPnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

Select count clock

0: Stop counting
1: Enable counting

(b) TMPn control register 1 (TPnCTL1)

	TPaSYE	TPmEST	TPkEEE			TPnMD2	TPnMD1	TPnMD0
<R> TPnCTL1	0	0	0/1 Note	0	0	0	0	0

0, 0, 0:
Interval timer mode

0: Operate at count clock selected in TPkCKS0 to TPkCKS2 bits
1: Count operation by external event count input signal

<R> **Note** Enable setting of the TPkEEE bit to 1 only when timer output (TOPk1) is used. In this case, set the TPkCCR0 and TPkCCR1 registers to the same value.

(c) TMPm I/O control register 0 (TPmIOC0)

					TPmOL1	TPmOE1	TP0OL0	TP0OE0
TPmIOC0	0	0	0	0	0/1	0/1	0/1	0/1

0: Disable TOP00 pin output
1: Enable TOP00 pin output

Setting of TOP00 pin output level before count operation
0: Low level
1: High level

0: Disable TOPm1 pin output
1: Enable TOPm1 pin output

Setting of TOPm1 pin output level before count operation
0: Low level
1: High level

Figure 6-11. Register Setting for Interval Timer Mode Operation (2/2)

<R> **(d) TMPk I/O control register 2 (TPkIOC2)**

					TPkEES1	TPkEES0	TPkETS1	TPkETS0
TPkIOC2	0	0	0	0	0/1[Note]	0/1[Note]	0	0

Selection of valid edge of external event count input (TIPk0 pin)

Note Enable setting of the TPkEES1 and TPkEES0 bits only when timer output (TOPk1) is used. In this case, set the TPkCCR0 and TPkCCR1 registers to the same value.

(e) TMPn counter read buffer register (TPnCNT)

By reading the TPnCNT register, the count value of the 16-bit counter can be read.

(f) TMPn capture/compare register 0 (TPnCCR0)

If the TPnCCR0 register is set to D_0, the interval is as follows.

Interval = $(D_0 + 1) \times$ Count clock cycle

(g) TMPn capture/compare register 1 (TPnCCR1)

The TPnCCR1 register is not used in the interval timer mode. However, the set value of the TPnCCR1 register is transferred to the CCR1 buffer register. When the count value of the 16-bit counter matches the value of the CCR1 buffer register, the TOPm1 pin output is inverted and a compare match interrupt request signal (INTTPnCC1) is generated.

By setting this register to the same value as the value set in the TPmCCR0 register, a PWM waveform with a duty factor of 50% can be output from the TOPm1 pin.

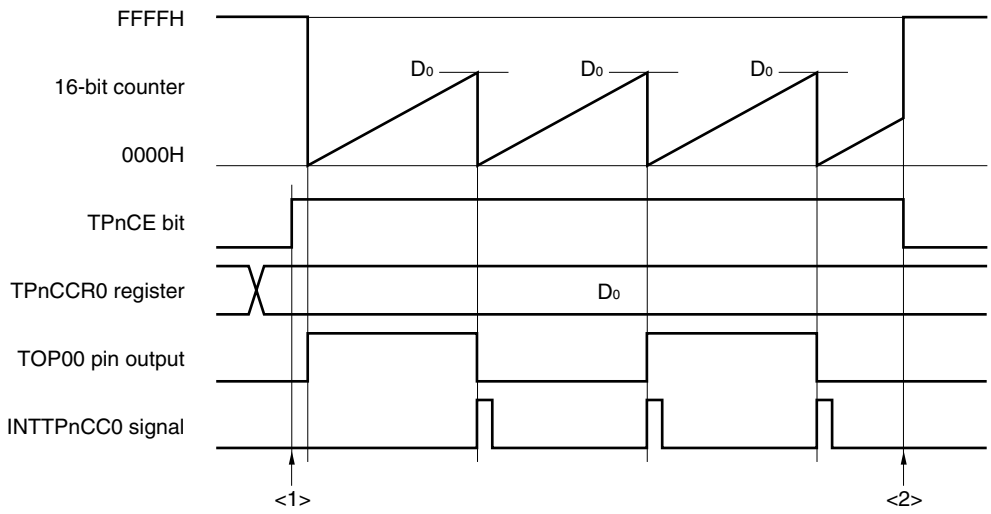
When the TPnCCR1 register is not used, it is recommended to set its value to FFFFH. Also mask the register by the interrupt mask flag (TPnCCIC1.TPnCCMK1).

<R> **Remarks 1.** TMPk I/O control register 1 (TPkIOC1) and TMPn option register 0 (TPnOPT0) are not used in the interval timer mode.

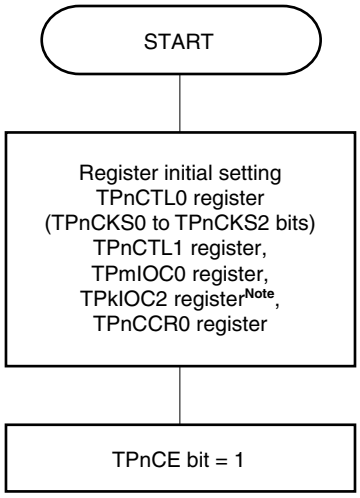
2. V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2, a = 0, 1

(1) Interval timer mode operation flow

Figure 6-12. Software Processing Flow in Interval Timer Mode (1/2)



<1> Count operation start flow



Initial setting of these registers is performed before setting the TPnCE bit to 1.

The TPnCKS0 to TPnCKS2 bits can be set at the same time when counting has been started (TPnCE bit = 1).

<R>

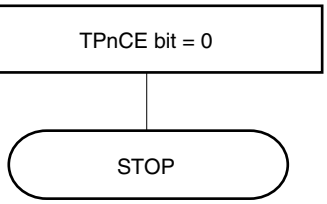
<R>

Note Enable setting of the TPkEES1 and TPkEES0 bits only when timer output (TOPk1) is used. In this case, set the TPkCCR0 and TPkCCR1 registers to the same value.

Remark V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2

Figure 6-12. Software Processing Flow in Interval Timer Mode (2/2)

<2> Count operation stop flow



The counter is initialized and counting is stopped by clearing the TPnCE bit to 0. The output level of the TOP00 pin is as specified by the TPmIOC0 register.

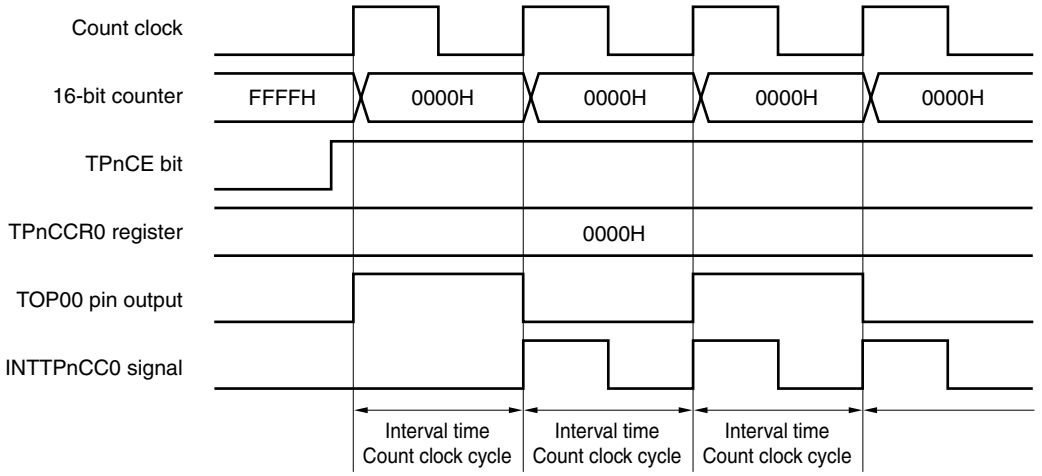
Remark V850E/IA3: n = 0 to 3, m = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3

(2) Interval timer mode operation timing

(a) Operation if TPnCCR0 register is set to 0000H

If the TPnCCR0 register is set to 0000H, the INTTPnCC0 signal is generated at each count clock, and the output of the TOP00 pin is inverted.

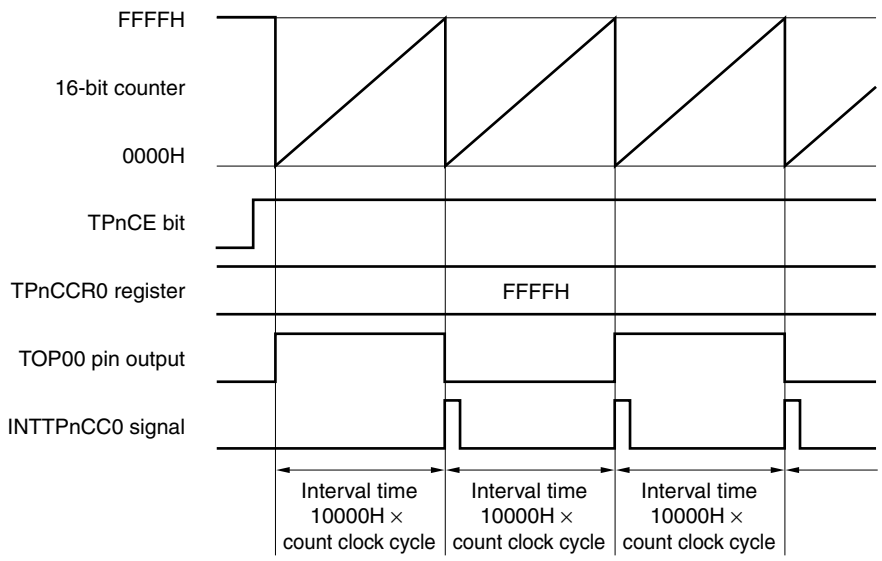
The value of the 16-bit counter is always 0000H.



Remark n = 0 to 3

(b) Operation if TPnCCR0 register is set to FFFFH

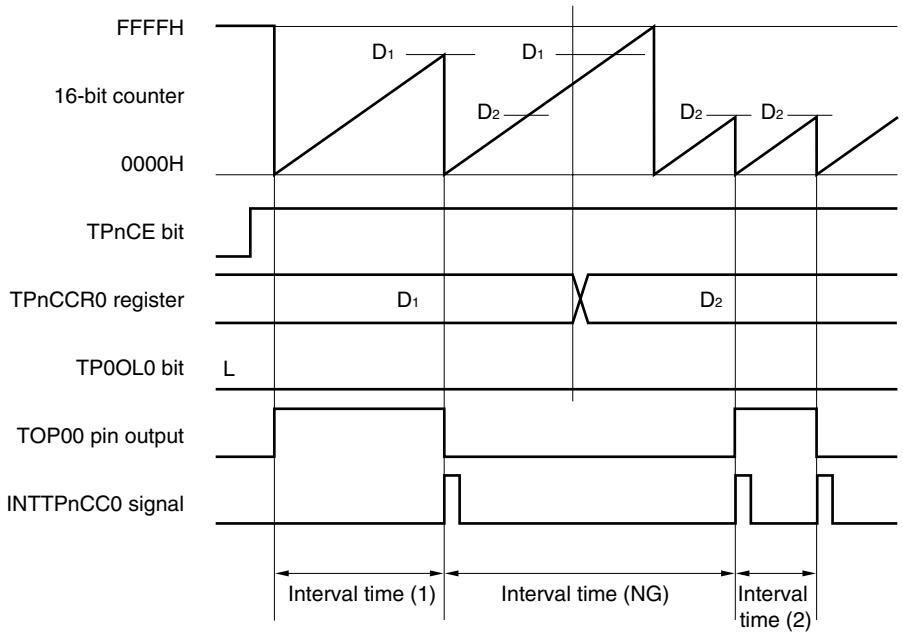
If the TPnCCR0 register is set to FFFFH, the 16-bit counter counts up to FFFFH. The counter is cleared to 0000H in synchronization with the next count-up timing. The INTTPnCC0 signal is generated and the output of the TOP00 pin is inverted. At this time, an overflow interrupt request signal (INTTPnOV) is not generated, nor is the overflow flag (TPnOPT0.TPnOVF bit) set to 1.



Remark n = 0 to 3

(c) Notes on rewriting TPnCCR0 register

If the value of the TPnCCR0 register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When an overflow may occur, stop counting and then change the set value.



Remarks 1. Interval time (1): $(D_1 + 1) \times$ Count clock cycle
Interval time (NG): $(10000H + D_2 + 1) \times$ Count clock cycle
Interval time (2): $(D_2 + 1) \times$ Count clock cycle

2. n = 0 to 3

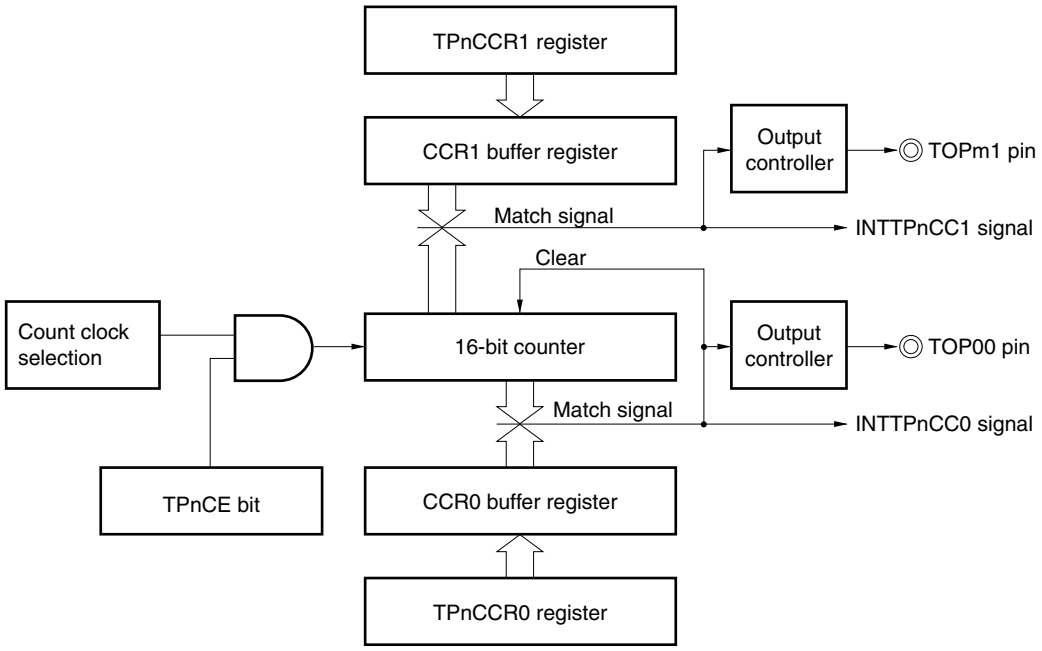
If the value of the TPnCCR0 register is changed from D_1 to D_2 while the count value is greater than D_2 but less than D_1, the count value is transferred to the CCR0 buffer register as soon as the TPnCCR0 register has been rewritten. Consequently, the value of the 16-bit counter that is compared is D_2.

Because the count value has already exceeded D_2, however, the 16-bit counter counts up to FFFFH, overflows, and then counts up again from 0000H. When the count value matches D_2, the INTTPnCC0 signal is generated and the output of the TOP00 pin is inverted.

Therefore, the INTTPnCC0 signal may not be generated at the interval time "$(D_1 + 1) \times$ Count clock cycle" or "$(D_2 + 1) \times$ Count clock cycle" originally expected, but may be generated at an interval of "$(10000H + D_2 + 1) \times$ Count clock cycle".

(d) Operation of TPnCCR1 register

Figure 6-13. Configuration of TPnCCR1 Register



Remark V850E/IA3: n = 0 to 3, m = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3

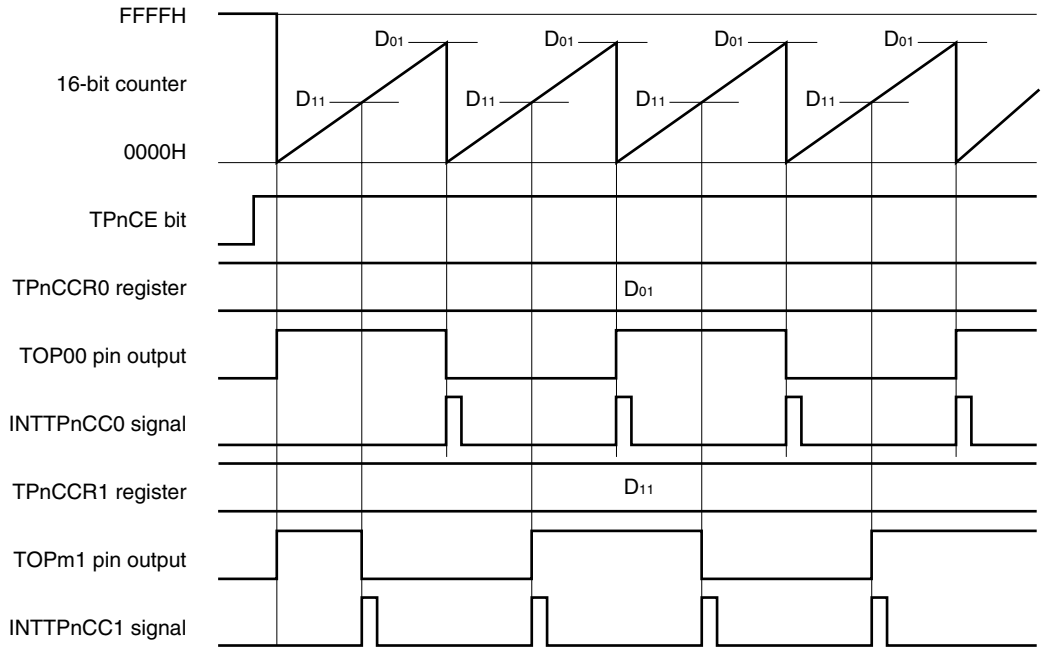
When the TPnCCR1 register is set to the same value as the TPnCCR0 register, the INTTPnCC1 signal is generated at the same timing as the INTTPnCC0 signal and the TOPm1 pin output is inverted. In other words, a PWM waveform with a duty factor of 50% can be output from the TOPm1 pin.

The following shows the operation when the TPnCCR1 register is set to other than the value set in the TPnCCR0 register.

If the set value of the TPnCCR1 register is less than the set value of the TPnCCR0 register, the INTTPnCC1 signal is generated once per cycle. At the same time, the output of the TOPm1 pin is inverted.

The TOPm1 pin outputs a PWM waveform with a duty factor of 50% after outputting a short-width pulse.

Figure 6-14. Timing Chart When $D_{01} \geq D_{11}$

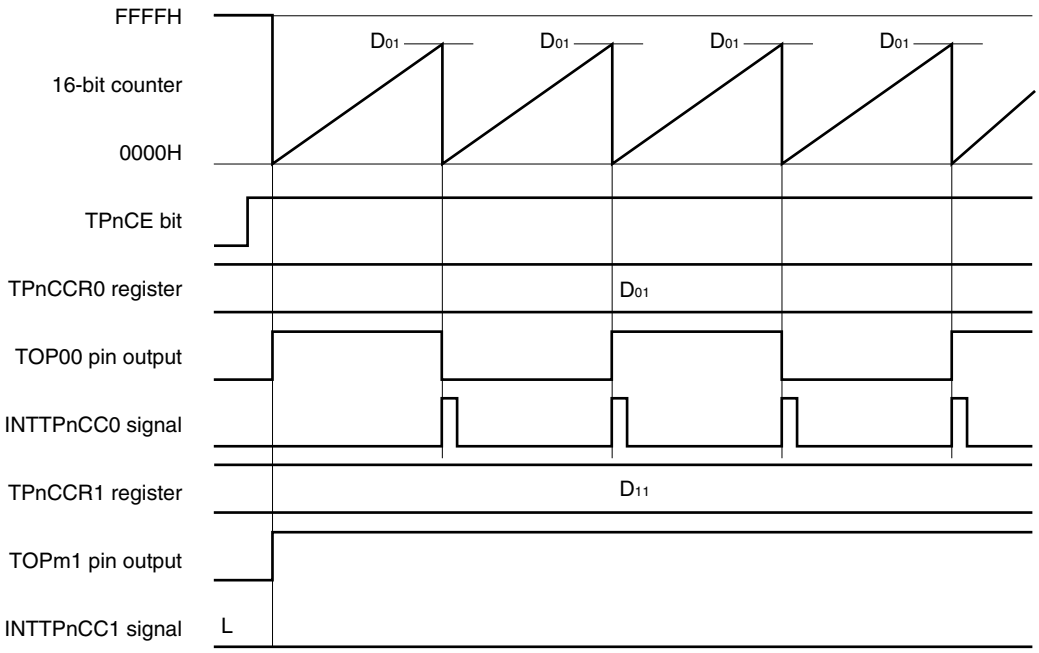


Remark V850E/IA3: n = 0 to 3, m = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3

If the set value of the TPnCCR1 register is greater than the set value of the TPnCCR0 register, the count value of the 16-bit counter does not match the value of the TPnCCR1 register. Consequently, the INTTPnCC1 signal is not generated, nor is the output of the TOPm1 pin changed.

When the TPnCCR1 register is not used, it is recommended to set its value to FFFFH.

Figure 6-15. Timing Chart When $D_{01} < D_{11}$



Remark V850E/IA3: n = 0 to 3, m = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3

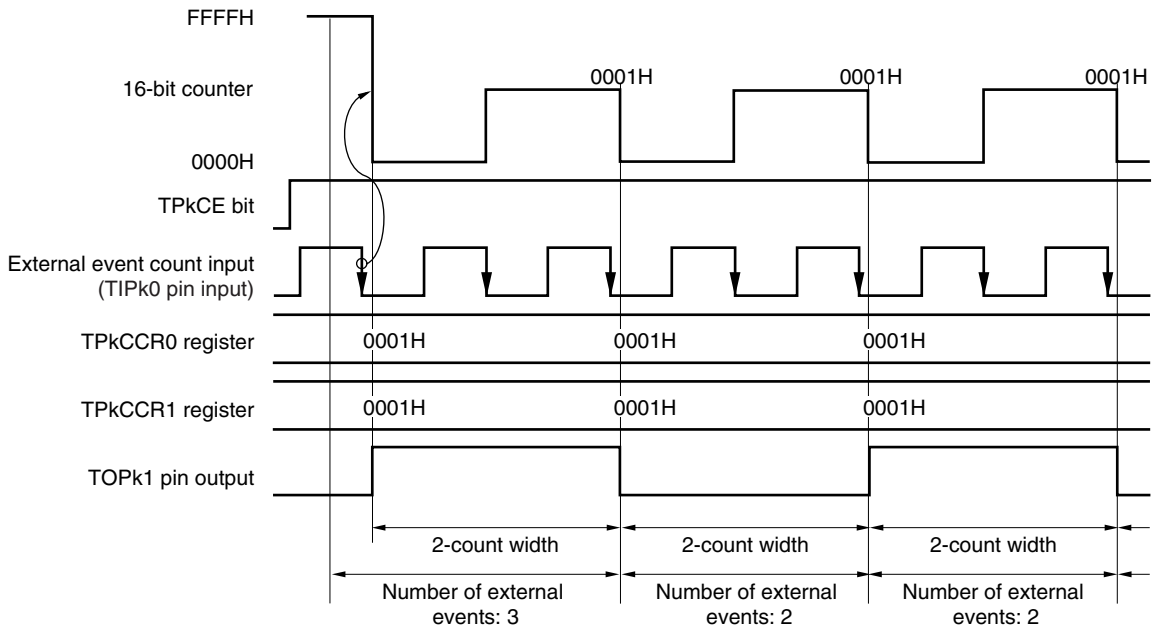
<R> (3) Operation by external event count input (TIPk0)

(a) Operation

To count the 16-bit counter at the valid edge of external event count input (TIPk0) in the interval timer mode, clear the 16-bit counter from FFFFH to 0000H at the valid edge of the first external event count input after the TPkCE bit is set from 0 to 1.

When both the TPkCCR0 and TPkCCR1 registers are set to 0001H, the TOPk1 pin output is inverted each time the 16-bit counter counts twice.

The TPkCTL1.TPkEEE bit can be set to 1 in the interval timer mode only when the timer output (TOPk1) is used with the external event count input.



Remark k = 0, 2

6.6.2 External event count mode (TPkMD2 to TPkMD0 bits = 001)

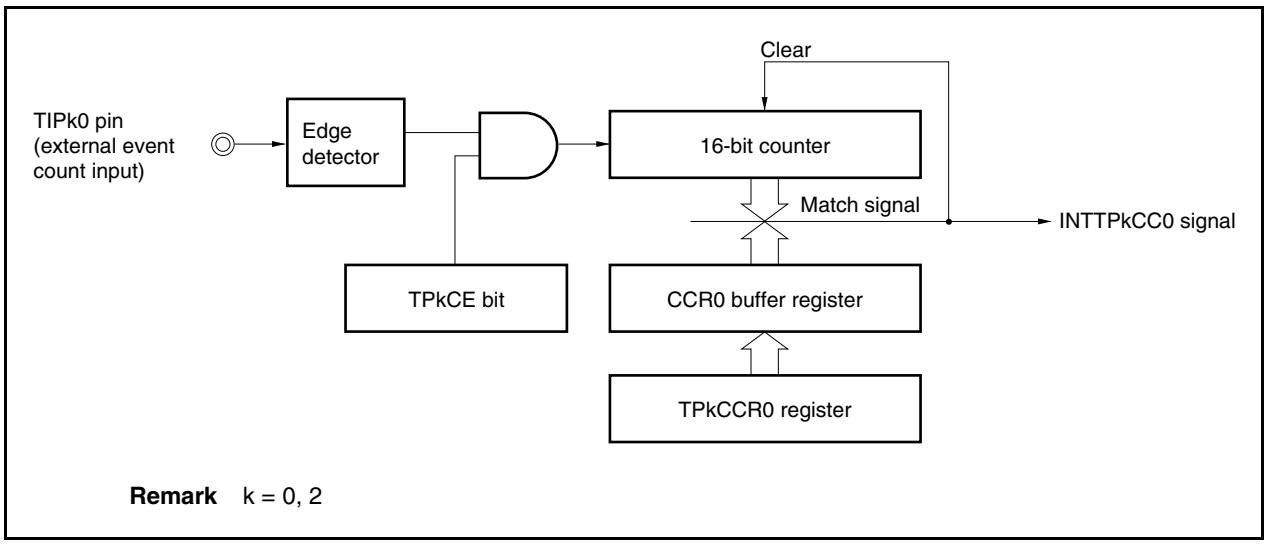
This mode is valid only in TMP0 and TMP2.

In the external event count mode, the valid edge of the external event count input (TIPk0) is counted when the TPkCTL0.TPkCE bit is set to 1, and an interrupt request signal (INTTPkCC0) is generated each time the number of <R> edges set by the TPkCCR0 register have been counted. The TOP00 and TOPk1 pins cannot be used. When using the TOPk1 pin in the external event count input mode, set the TPkCTL1.TPkEEE bit to 1 in the interval timer mode (see **6.6.1 (3) Operation by external event count input (TIPk0)**).

The TPkCCR1 register is not used in the external event count mode.

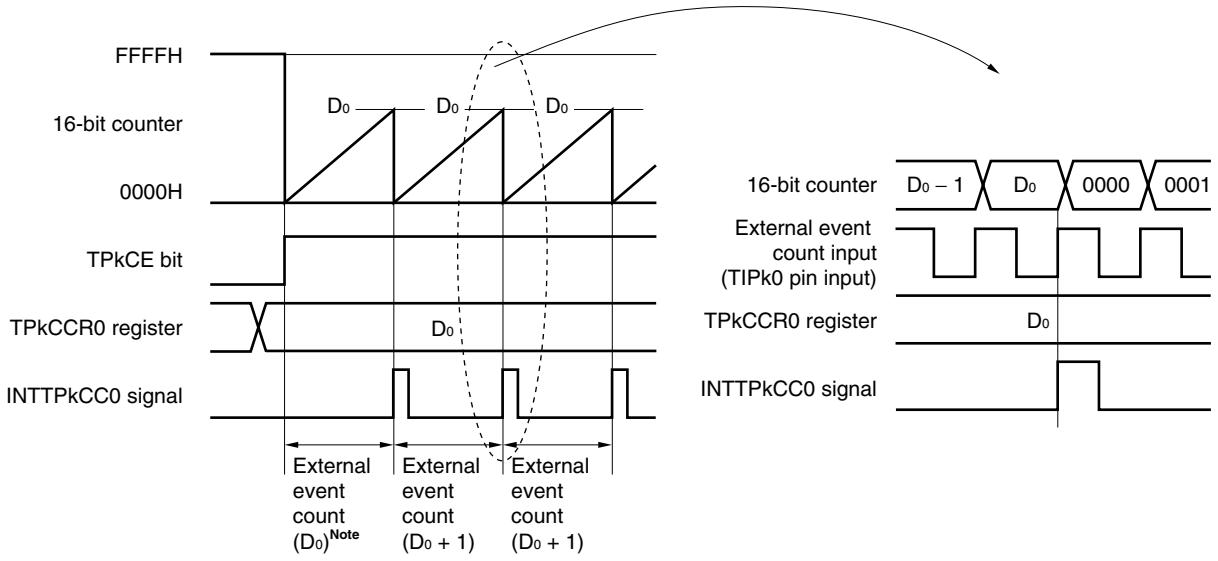
Caution In the external event count mode, the TPkCCR0 and TPkCCR1 registers must not be cleared to 0000H.

Figure 6-16. Configuration in External Event Count Mode



Remark k = 0, 2

Figure 6-17. Basic Timing in External Event Count Mode



Note In the external event count mode, when the TPkCTL0.TPkCE bit is set to 1 (operation starts), the 16-bit counter is cleared from FFFFH to 0000H at the same time. The first count operation starts from the 0001H every time the valid edge of the external event count input is detected. Therefore, the count of the first count operation is one number smaller than the count of second or subsequent count operation.

Remarks 1. This figure shows the basic timing when the rising edge is specified as the valid edge of the external event count input.

2. k = 0, 2

When the TPkCE bit is set to 1, the value of the 16-bit counter is cleared from FFFFH to 0000H. The counter counts each time the valid edge of external event count input is detected. Additionally, the set value of the TPkCCR0 register is transferred to the CCR0 buffer register.

When the count value of the 16-bit counter matches the value of the CCR0 buffer register, the 16-bit counter is cleared to 0000H, and a compare match interrupt request signal (INTTPkCC0) is generated.

The INTTPkCC0 signal is generated for the first time when the valid edge of the external event count input has been detected "value set to TPkCCR0 register" times. After that, the INTTPkCC0 signal is generated each time the valid edge of the external event count has been detected "value set to TPkCCR0 register + 1" times.

Figure 6-18. Register Setting for Operation in External Event Count Mode (1/2)

(a) TMPk control register 0 (TPkCTL0)

	TPkCE					TPkCKS2	TPkCKS1	TPkCKS0
TPkCTL0	0/1	0	0	0	0	0	0	0

0: Stop counting
1: Enable counting

(b) TMPk control register 1 (TPkCTL1)

	TP0SYE	TPkEST	TPkEEE			TPkMD2	TPkMD1	TPkMD0
TPkCTL1	0	0	0	0	0	0	0	1

0, 0, 1:
External event count mode

(c) TMPk I/O control register 2 (TPkIOC2)

					TPkEES1	TPkEES0	TPkETS1	TPkETS0
TPkIOC2	0	0	0	0	0/1	0/1	0	0

Select valid edge
of external event
count input (TIPk0 pin)

(d) TMPk counter read buffer register (TPkCNT)

The count value of the 16-bit counter can be read by reading the TPkCNT register.

(e) TMPk capture/compare register 0 (TPkCCR0)

If the TPkCCR0 register is set to D_0, the count is cleared when the number of external event counts has reached (D_0) and the first compare match interrupt request signal (INTTPkCC0) is generated. The second compare match interrupt request signal (INTTPkCC0) is generated when the number of external event counts has reached (D_0 + 1).

Figure 6-18. Register Setting for Operation in External Event Count Mode (2/2)

(f) TMPk capture/compare register 1 (TPkCCR1)

The TPkCCR1 register is not used in the external event count mode. However, the set value of the TPkCCR1 register is transferred to the CCR1 buffer register. When the count value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTPkCC1) is generated.

When the TPkCCR1 register is not used, it is recommended to set its value to FFFFH. Also mask the register by the interrupt mask flag (TPkCCIC1.TPkCCMK1).

<R> **Cautions 1. Set the TPkIOC0 register to 00H.**

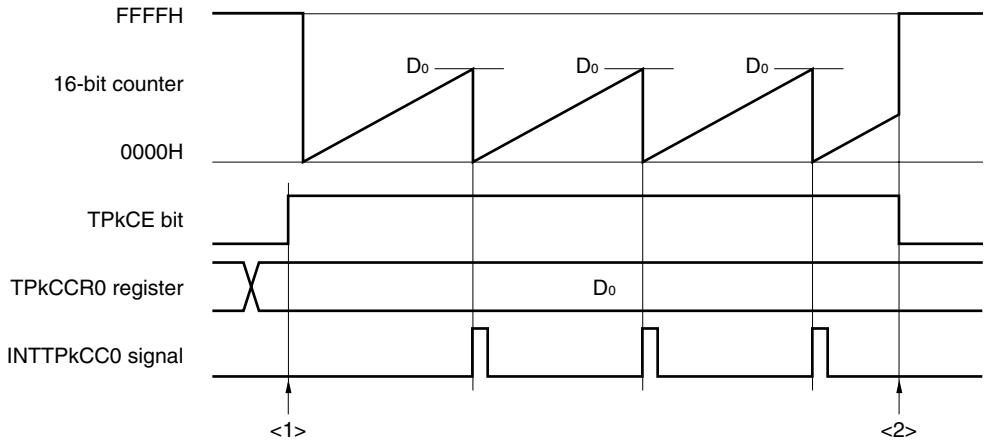
<R> **2. When an external clock is used as the count clock, the external clock can be input only from the TIPk0 pin. At this time, set the TPkIOC1.TPkIS1 and TPkIOC1.TPkIS0 bits to 00 (capture trigger input (TIPk0 pin): no edge detection).**

Remarks 1. TMPk I/O control register 1 (TPkIOC1) and TMPk option register 0 (TPkOPT0) are not used in the external event count mode.

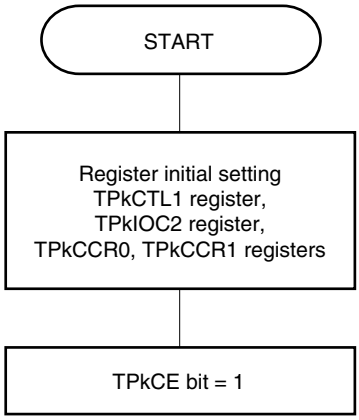
2. k = 0, 2

(1) External event count mode operation flow

Figure 6-19. Flow of Software Processing in External Event Count Mode

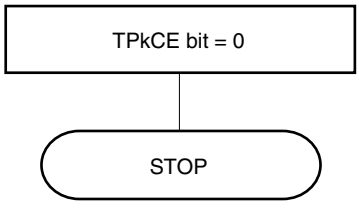


<1> Count operation start flow



Initial setting of these registers is performed before setting the TPkCE bit to 1.

<2> Count operation stop flow



The counter is initialized and counting is stopped by clearing the TPkCE bit to 0.

Remark k = 0, 2

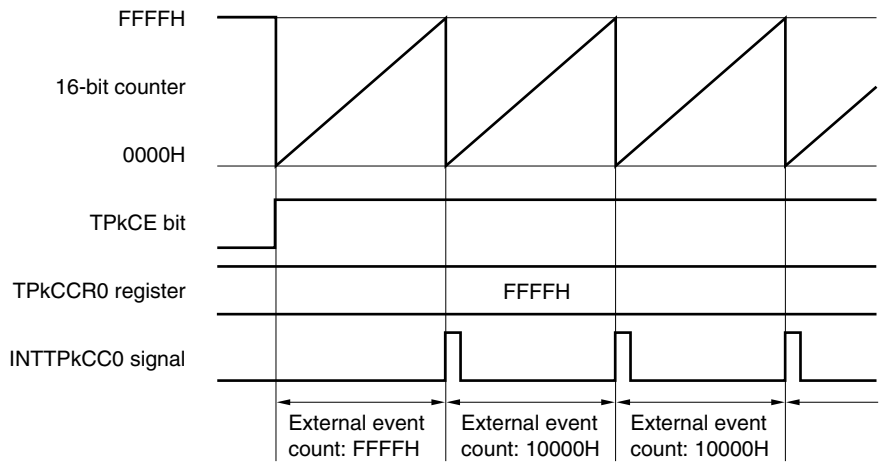
(2) Operation timing in external event count mode

Cautions 1. In the external event count mode, the TPkCCR0 and TPkCCR1 registers must not be cleared to 0000H.

<R> **2. In the external event count mode, use of the timer output (TOP00, TOPk1) is disabled. If performing timer output (TOPk1) using external event count input (TIPk0), set the interval timer mode, and enable the count clock operation with the external event count input (TPkCTL1.TPkEEE bit = 1) (see 6.6.1 (3) Operation by external event count input (TIPk0)).**

(a) Operation if TPkCCR0 register is set to FFFFH

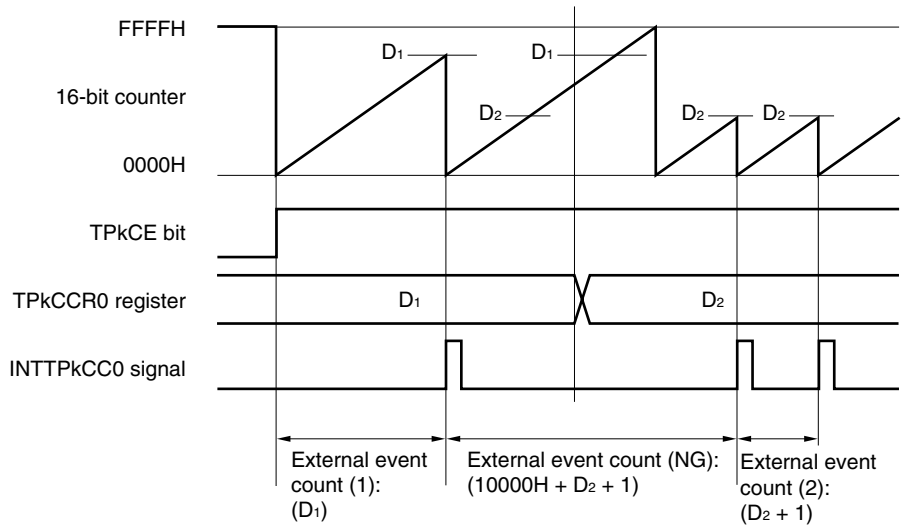
If the TPkCCR0 register is set to FFFFH, the 16-bit counter counts to FFFFH each time the valid edge of the external event count signal has been detected. The 16-bit counter is cleared to 0000H in synchronization with the next count-up timing, and the INTTPkCC0 signal is generated. At this time, the TPkOPT0.TPkOVF bit is not set.



Remark k = 0, 2

(b) Notes on rewriting the TPkCCR0 register

If the value of the TPkCCR0 register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When the overflow may occur, stop counting once and then change the set value.



Remark k = 0, 2

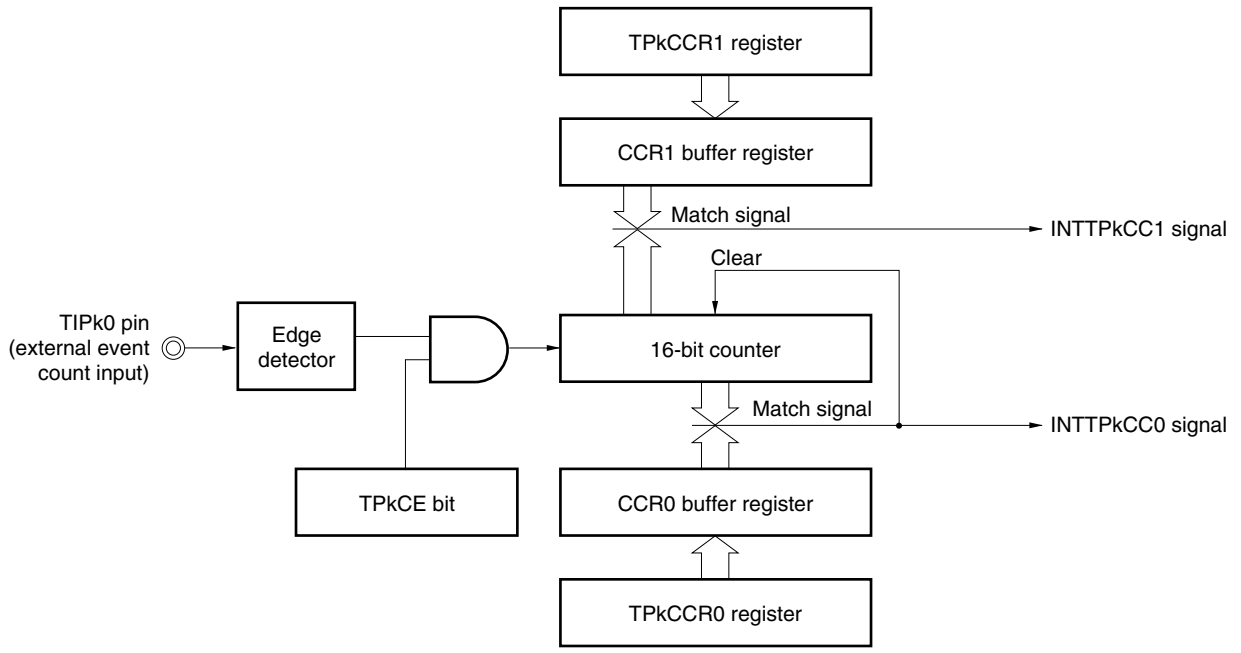
If the value of the TPkCCR0 register is changed from D_1 to D_2 while the count value is greater than D_2 but less than D_1, the count value is transferred to the CCR0 buffer register as soon as the TPkCCR0 register has been rewritten. Consequently, the value that is compared with the 16-bit counter is D_2.

Because the count value has already exceeded D_2, however, the 16-bit counter counts up to FFFFH, overflows, and then counts up again from 0000H. When the count value matches D_2, the INTTPkCC0 signal is generated.

Therefore, the INTTPkCC0 signal may not be generated at the valid edge count of "(D_1 + 1) times" or "(D_2 + 1) times" originally expected, but may be generated at the valid edge count of "(10000H + D_2 + 1) times".

(c) Operation of TPkCCR1 register

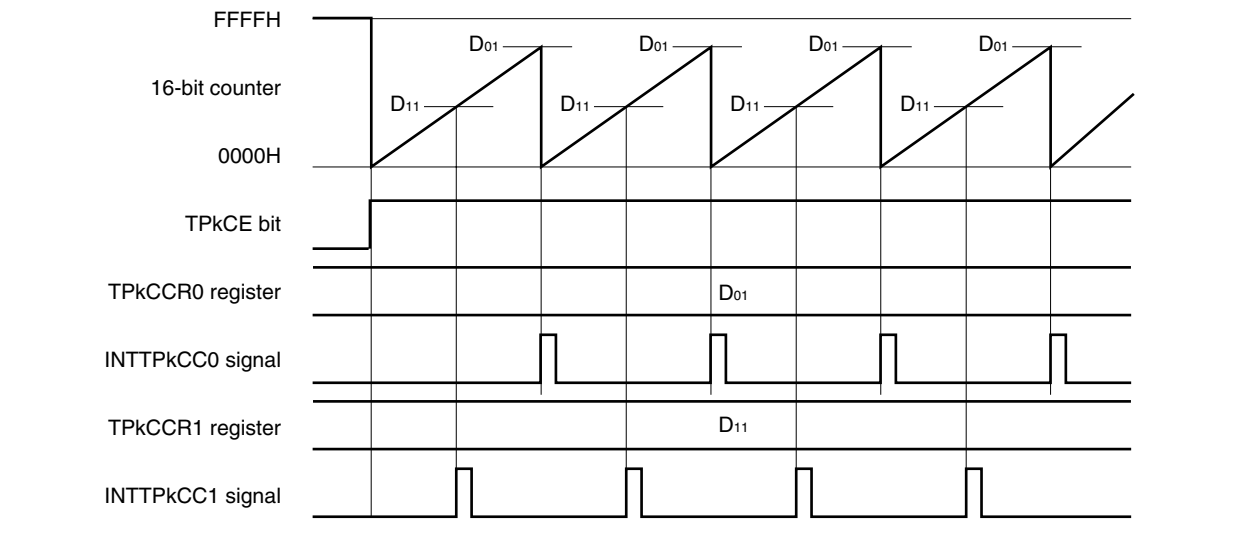
Figure 6-20. Configuration of TPkCCR1 Register



Remark k = 0, 2

If the set value of the TPkCCR1 register is smaller than the set value of the TPkCCR0 register, the INTTPkCC1 signal is generated once per cycle.

Figure 6-21. Timing Chart When $D_{01} \geq D_{11}$

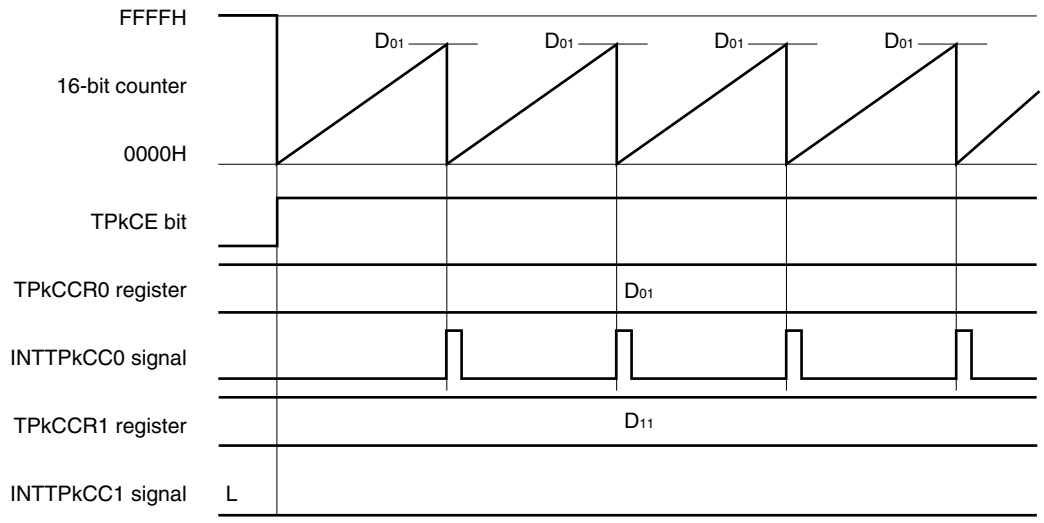


Remark k = 0, 2

If the set value of the TPkCCR1 register is greater than the set value of the TPkCCR0 register, the INTTPkCC1 signal is not generated because the count value of the 16-bit counter and the value of the TPkCCR1 register do not match.

When the TPkCCR1 register is not used, it is recommended to set its value to FFFFH.

Figure 6-22. Timing Chart When $D_{01} < D_{11}$



Remark k = 0, 2

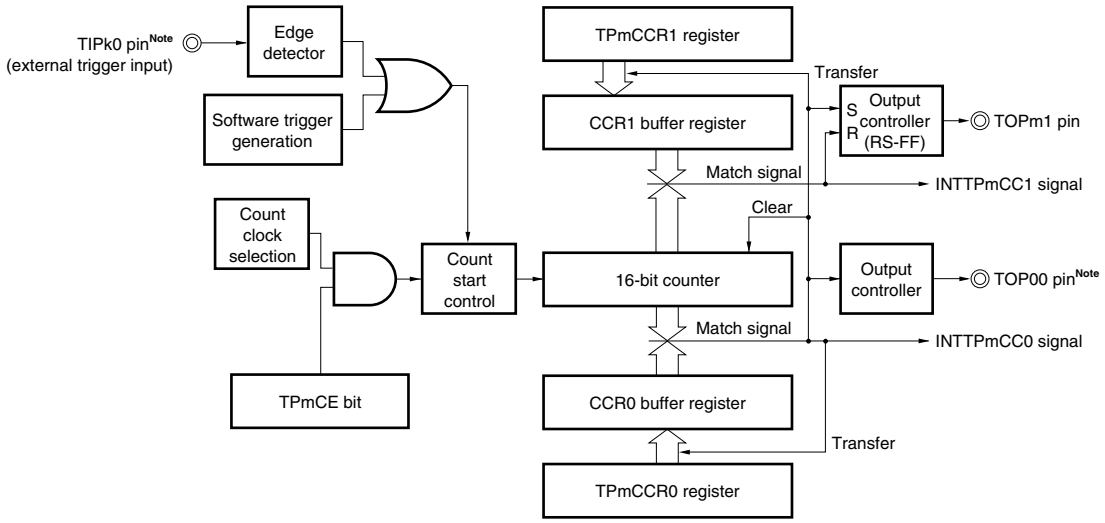
6.6.3 External trigger pulse output mode (TPmMD2 to TPmMD0 bits = 010)

This mode is valid only in TMP0, TMP2, and TMP3 (V850E/IA4 only) (software trigger only for TMP3).

In the external trigger pulse output mode, 16-bit timer/event counter P waits for a trigger when the TPmCTL0.TPmCE bit is set to 1. When the valid edge of an external trigger input (TIPk0) is detected, 16-bit timer/event counter P starts counting, and outputs a PWM waveform from the TOPm1 pin.

Pulses can also be output by generating a software trigger instead of using the external trigger input. When using a software trigger, a PWM waveform with a duty factor of 50% that has the set value of the TPmCCR0 register + 1 as half its cycle can also be output from the TOP00 pin.

<R> **Figure 6-23. Configuration in External Trigger Pulse Output Mode**

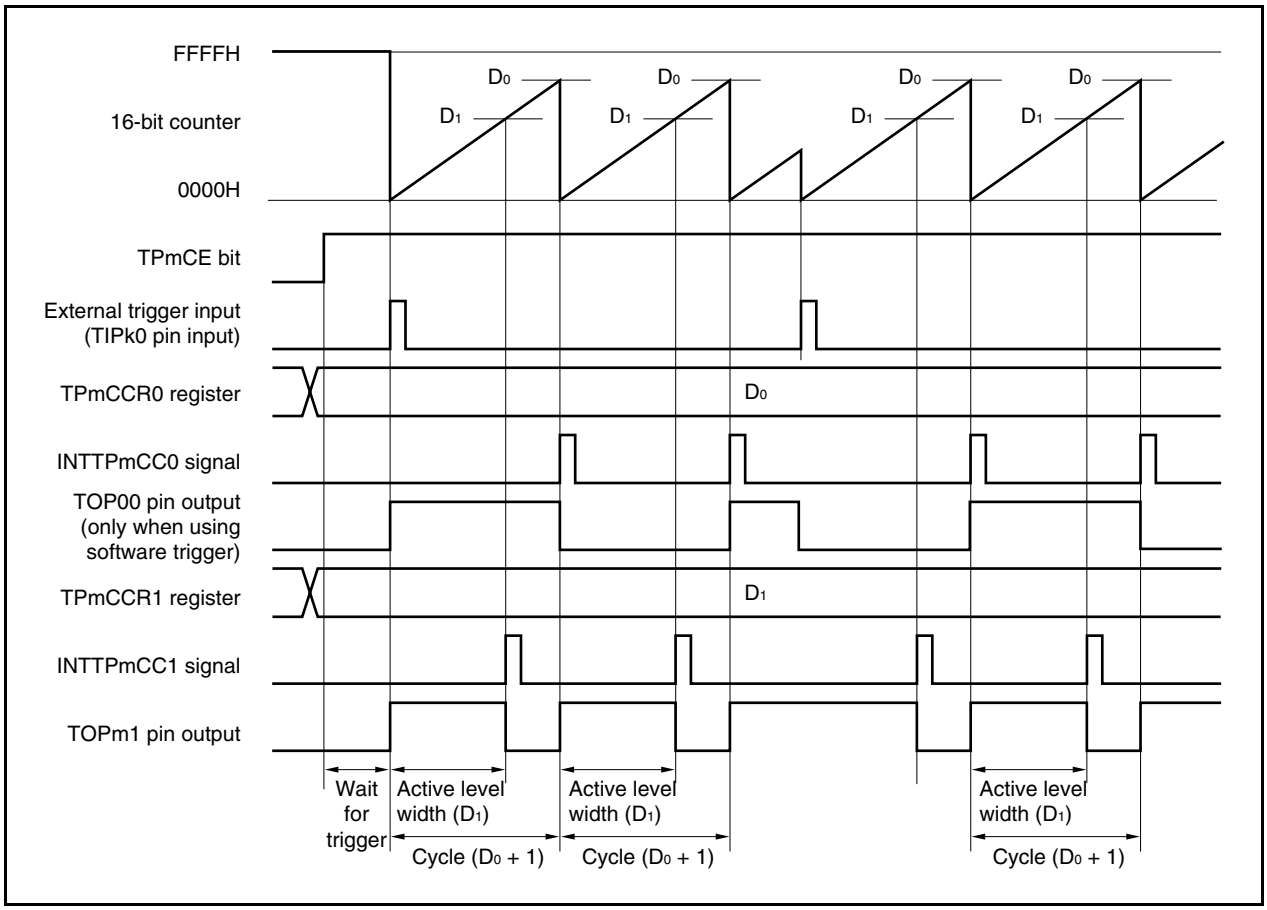


Note Because the external trigger input pin (TIP00) and timer output pin (TOP00) function alternately, two functions cannot be used at the same time.

Caution In the external trigger pulse output mode, select the internal clock as the count clock (by clearing the TPkCTL1.TPkEEE bit to 0).

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

Figure 6-24. Basic Timing in External Trigger Pulse Output Mode



16-bit timer/event counter P waits for a trigger when the TPmCE bit is set to 1. When the trigger is generated, the 16-bit counter is cleared from FFFFH to 0000H, starts counting at the same time, and outputs a PWM waveform from the TOPm1 pin. If the trigger is generated again while the counter is operating, the counter is cleared to 0000H and <R> restarted. (The output of the TOP00 pin is inverted. The TOPm1 pin outputs high level regardless of the status (high/low) when a trigger occurs.)

The active level width, cycle, and duty factor of the PWM waveform can be calculated as follows.

Active level width = (Set value of TPmCCR1 register) × Count clock cycle
Cycle = (Set value of TPmCCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TPmCCR1 register)/(Set value of TPmCCR0 register + 1)

The compare match interrupt request signal INTTPmCC0 is generated when the 16-bit counter counts next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The compare match interrupt request signal INTTPmCC1 is generated when the count value of the 16-bit counter matches the value of the CCR1 buffer register.

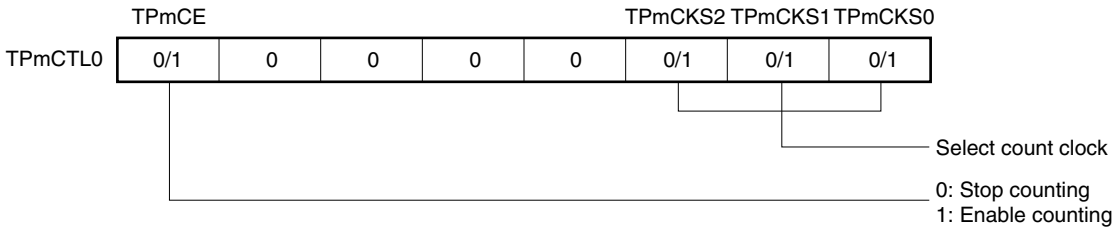
The value set to the TPmCCRa register is transferred to the CCRa buffer register when the count value of the 16-bit counter matches the value of the CCRa buffer register and the 16-bit counter is cleared to 0000H.

The valid edge of an external trigger input (TIPk0) or setting the software trigger (TPmCTL1.TPmEST bit) to 1 is used as the trigger.

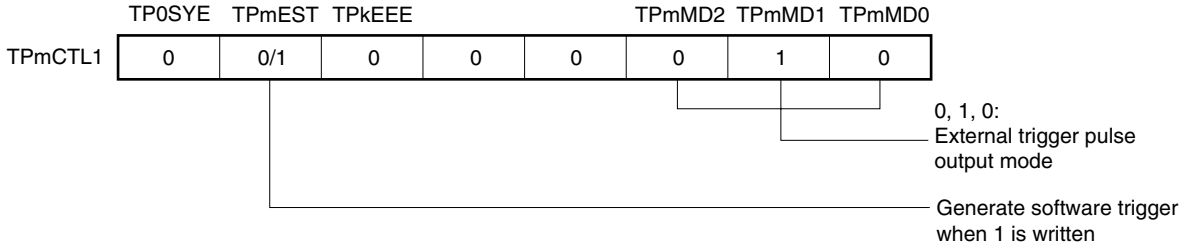
Remark V850E/IA3: m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, k = 0, 2, a = 0, 1

Figure 6-25. Setting of Registers in External Trigger Pulse Output Mode (1/2)

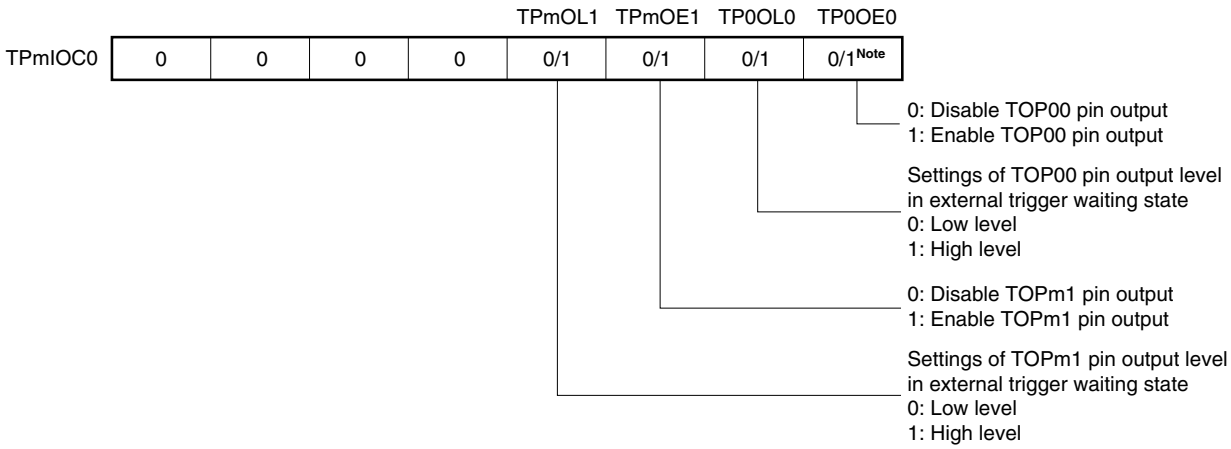
(a) TMPm control register 0 (TPmCTL0)



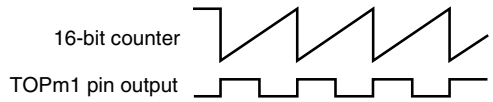
(b) TMPm control register 1 (TPmCTL1)



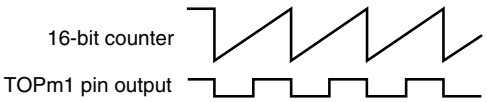
(c) TMPm I/O control register 0 (TPmIOC0)



- When TPmOL1 bit = 0



- When TPmOL1 bit = 1



Note Clear this bit to 0 when the TOP00 pin is not used in the external trigger pulse output mode.

Figure 6-25. Setting of Registers in External Trigger Pulse Output Mode (2/2)

(d) TMPk I/O control register 2 (TPkIOC2)

<R>

							TPkEES1	TPkEES0	TPkETS1	TPkETS0
TPkIOC2	0	0	0	0	0	0	0/1	0/1		

Select valid edge of external trigger input (TIPk0 pin)

(e) TMPm counter read buffer register (TPmCNT)

The value of the 16-bit counter can be read by reading the TPmCNT register.

(f) TMPm capture/compare registers 0 and 1 (TPmCCR0 and TPmCCR1)

If D_0 is set to the TPmCCR0 register and D_1 to the TPmCCR1 register, the cycle and active level of the PWM waveform are as follows.

Cycle = $(D_0 + 1) \times$ Count clock cycle

Active level width = $D_1 \times$ Count clock cycle

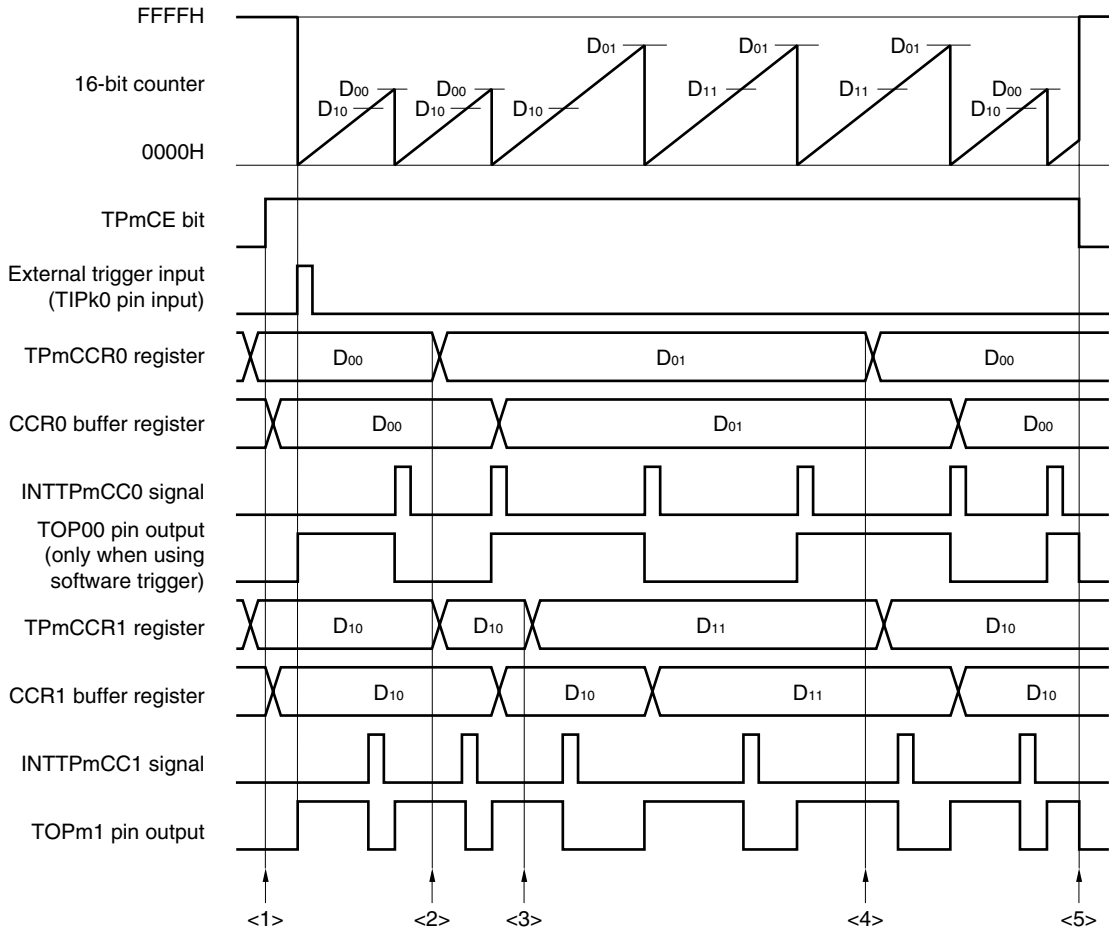
Remarks 1. TMPk I/O control register 1 (TPkIOC1) and TMPm option register 0 (TPmOPT0) are not used in the external trigger pulse output mode.

2. V850E/IA3: m = 0, 2, k = 0, 2

V850E/IA4: m = 0, 2, 3, k = 0, 2

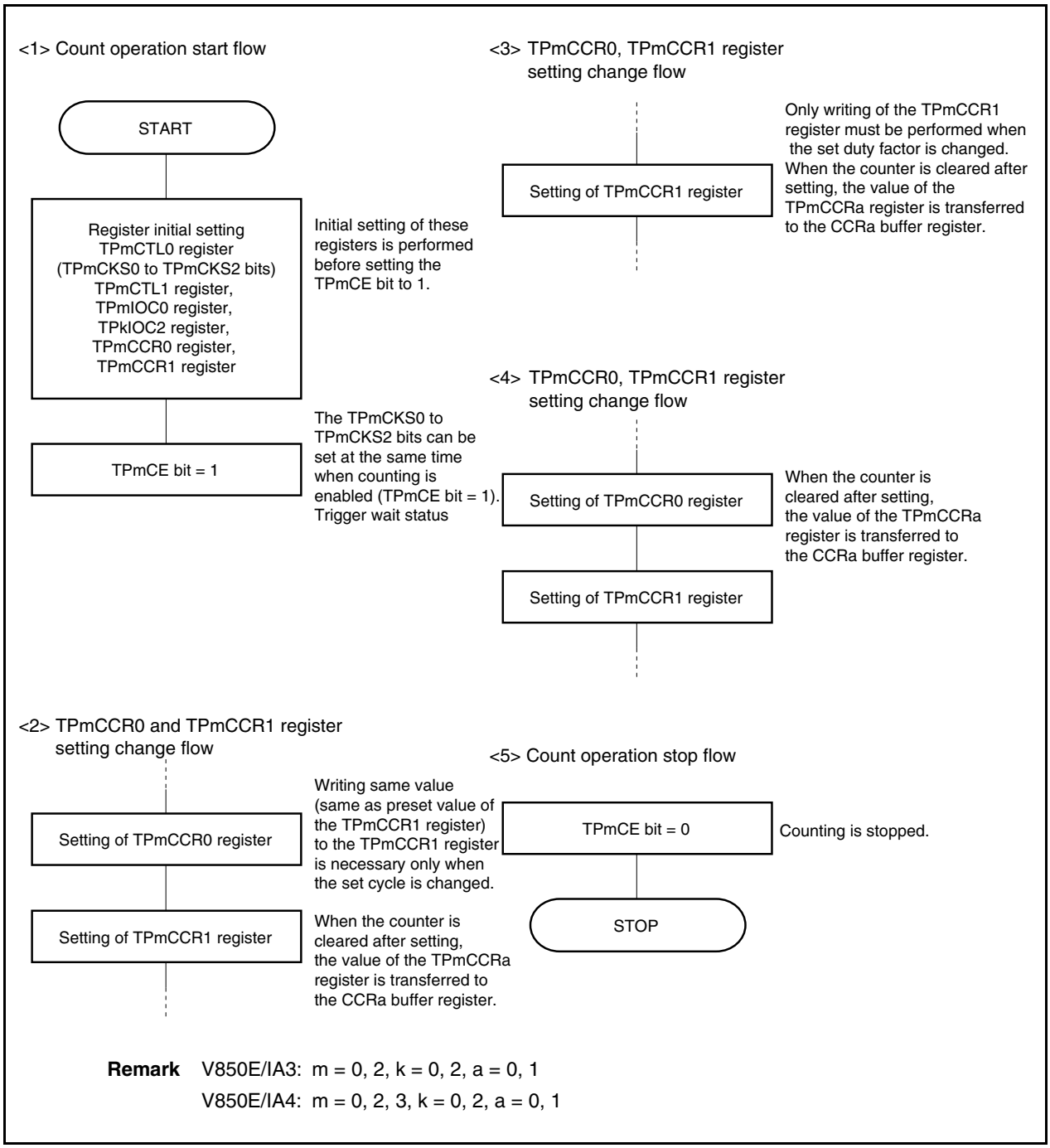
(1) Operation flow in external trigger pulse output mode

Figure 6-26. Software Processing Flow in External Trigger Pulse Output Mode (1/2)



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

Figure 6-26. Software Processing Flow in External Trigger Pulse Output Mode (2/2)



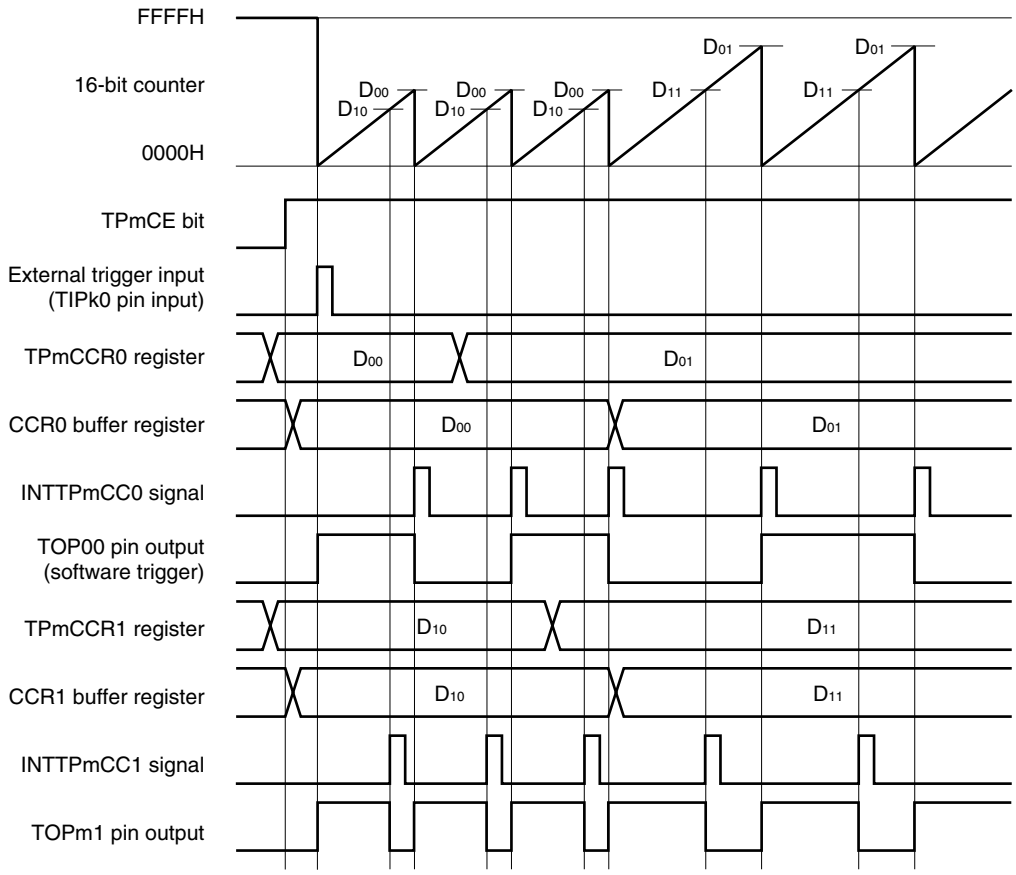
Remark V850E/IA3: m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, k = 0, 2, a = 0, 1

(2) External trigger pulse output mode operation timing

(a) Note on changing pulse width during operation

To change the PWM waveform while the counter is operating, write the TPmCCR1 register last.

Rewrite the TPmCCRa register after writing the TPmCCR1 register after the INTTPmCC0 signal is detected.



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

In order to transfer data from the TPmCCRa register to the CCRa buffer register, the TPmCCR1 register must be written.

To change both the cycle and active level width of the PWM waveform at this time, first set the cycle to the TPmCCR0 register and then set the active level width to the TPmCCR1 register.

To change only the cycle of the PWM waveform, first set the cycle to the TPmCCR0 register, and then write the same value (same as preset value of the TPmCCR1 register) to the TPmCCR1 register.

To change only the active level width (duty factor) of the PWM waveform, only the TPmCCR1 register has to be set.

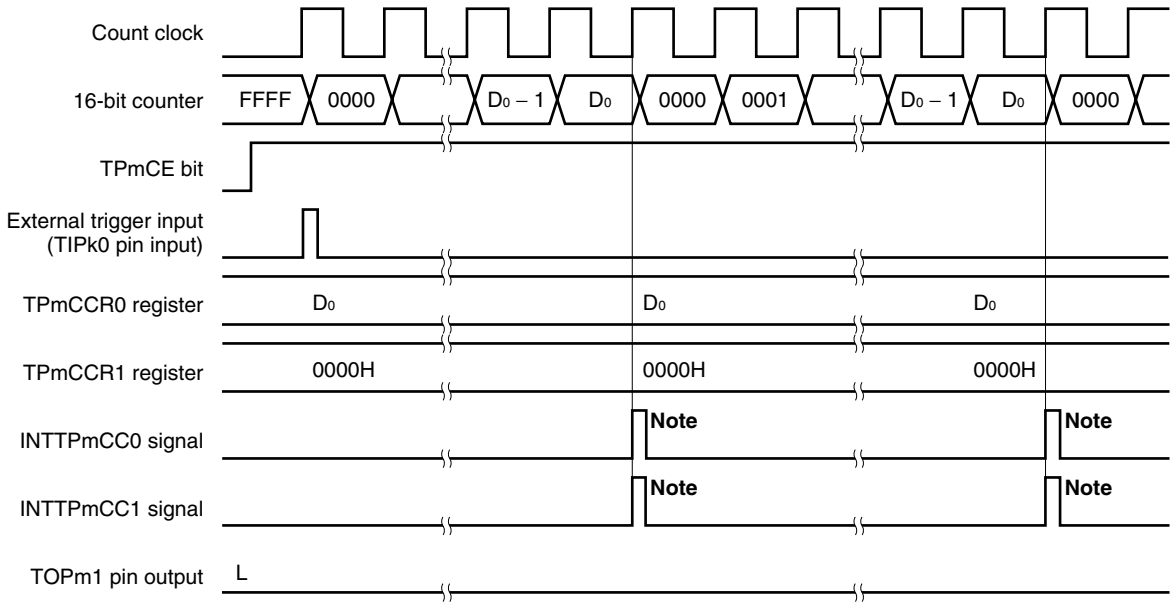
After data is written to the TPmCCR1 register, the value written to the TPmCCRa register is transferred to the CCRa buffer register in synchronization with clearing of the 16-bit counter, and is used as the value compared with the 16-bit counter.

To write the TPmCCR0 or TPmCCR1 register again after writing the TPmCCR1 register once, do so after the INTTPmCC0 signal is generated. Otherwise, the value of the CCRa buffer register may become undefined because the timing of transferring data from the TPmCCRa register to the CCRa buffer register conflicts with writing the TPmCCRa register.

Remark V850E/IA3: m = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, a = 0, 1

(b) 0%/100% output of PWM waveform

To output a 0% waveform, set the TPmCCR1 register to 0000H. The 16-bit counter is cleared to 0000H and the INTTPmCC0 and INTTPmCC1 signals are generated at the next timing after a match between the count value of the 16-bit counter and the value of the CCR0 buffer register.

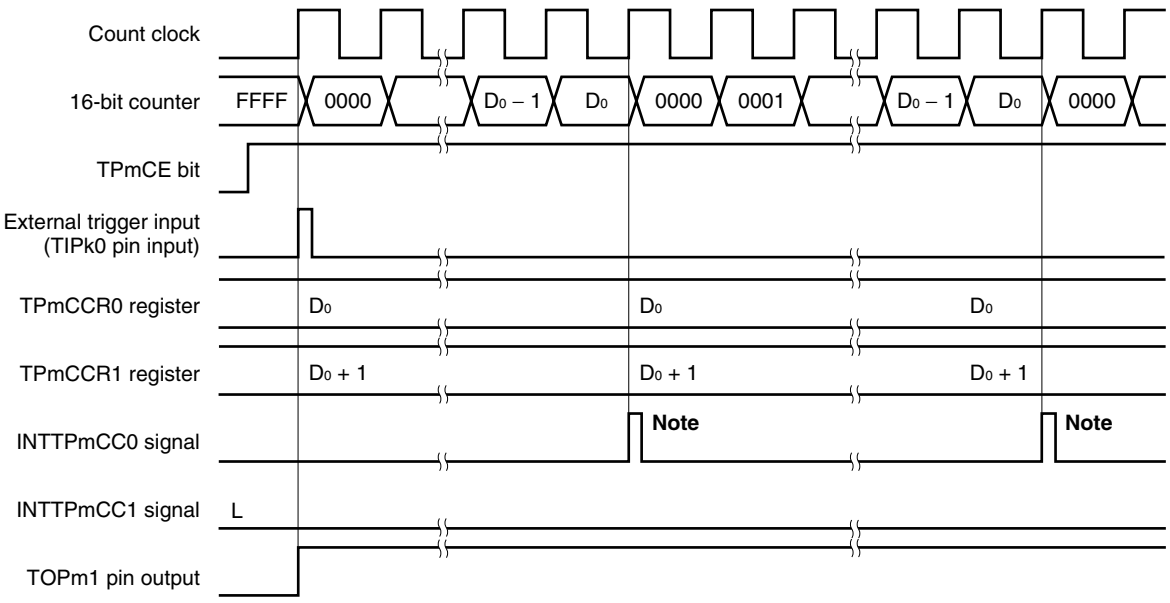


Note Actually, the timing is delayed by one operating clock (f_{XX}).

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

To output a 100% waveform, set a value of (set value of TPmCCR0 register + 1) to the TPmCCR1 register. If the set value of the TPmCCR0 register is FFFFH, 100% output cannot be produced.

<R>

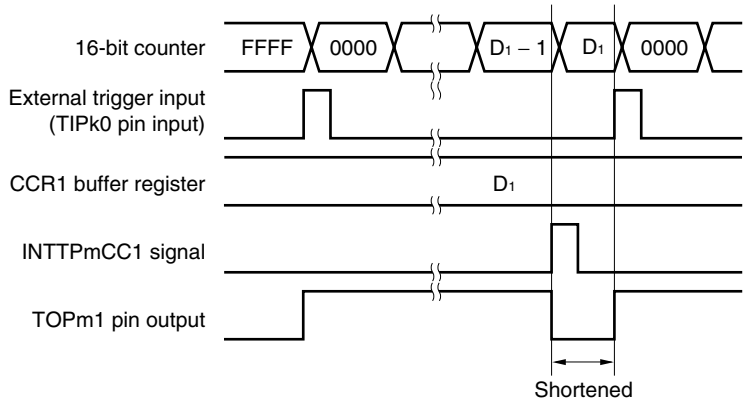


Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

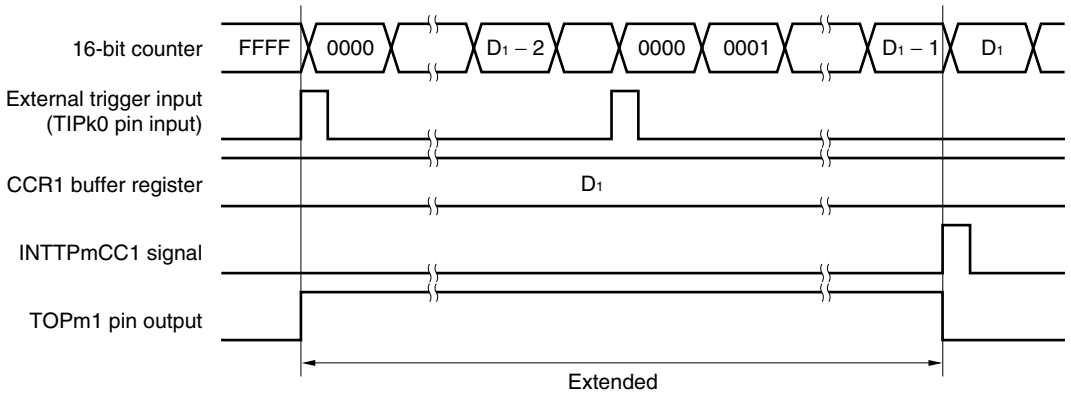
(c) Conflict between trigger detection and match with CCR1 buffer register

If the trigger is detected immediately after the INTTPmCC1 signal is generated, the 16-bit counter is immediately cleared to 0000H, the output signal of the TOPm1 pin is asserted, and the counter continues counting. Consequently, the inactive period of the PWM waveform is shortened.



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

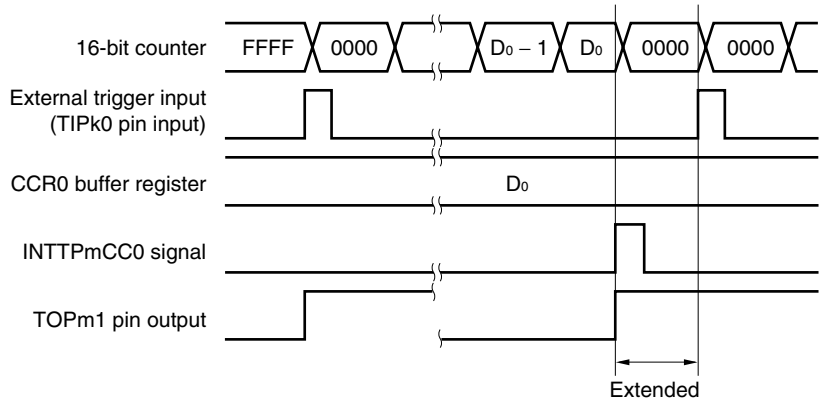
If the trigger is detected immediately before the INTTPmCC1 signal is generated, the INTTPmCC1 signal is not generated, and the 16-bit counter is cleared to 0000H and continues counting. The output signal of the TOPm1 pin remains active. Consequently, the active period of the PWM waveform is extended.



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

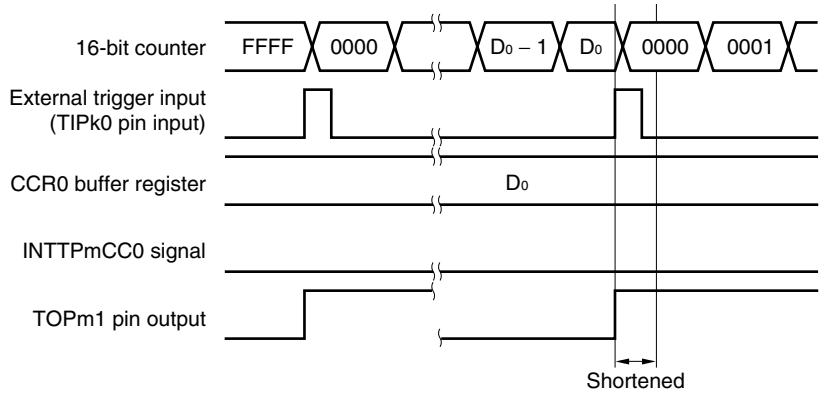
(d) Conflict between trigger detection and match with CCR0 buffer register

If the trigger is detected immediately after the INTTPmCC0 signal is generated, the 16-bit counter is cleared to 0000H and continues counting up. Therefore, the active period of the TOPm1 pin is extended by time from generation of the INTTPmCC0 signal to trigger detection.



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

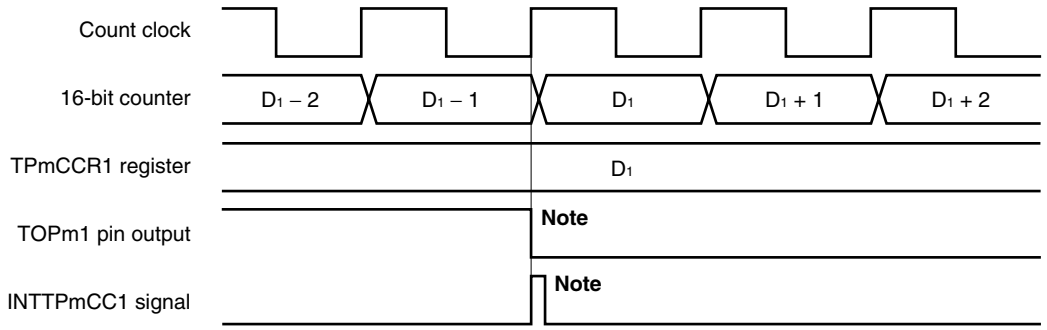
If the trigger is detected immediately before the INTTPmCC0 signal is generated, the INTTPmCC0 signal is not generated. The 16-bit counter is cleared to 0000H, the TOPm1 pin is asserted, and the counter continues counting. Consequently, the inactive period of the PWM waveform is shortened.



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

(e) Generation timing of compare match interrupt request signal (INTTPmCC1)

The timing of generation of the INTTPmCC1 signal in the external trigger pulse output mode differs from the timing of INTTPmCC1 signals in other modes; the INTTPmCC1 signal is generated when the count value of the 16-bit counter matches the value of the TPmCCR1 register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2,
V850E/IA4: m = 0, 2, 3

Usually, the INTTPmCC1 signal is generated in synchronization with the next count up, after the count value of the 16-bit counter matches the value of the TPmCCR1 register.

In the external trigger pulse output mode, however, it is generated one clock earlier. This is because the timing is changed to match the timing of changing the output signal of the TOPm1 pin.

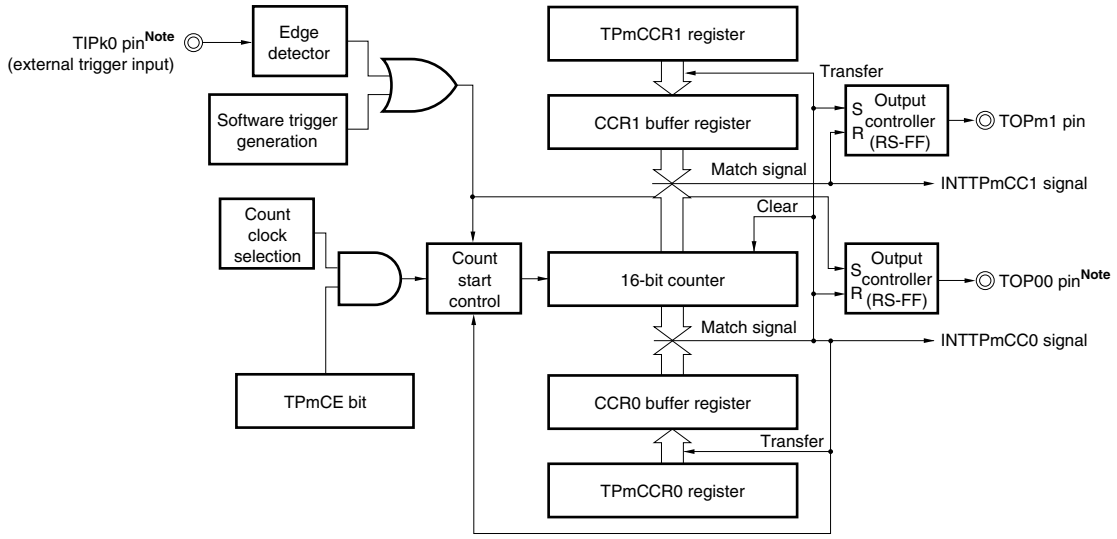
6.6.4 One-shot pulse output mode (TPmMD2 to TPmMD0 bits = 011)

This mode is valid only in TMP0, TMP2, and TMP3 (V850E/IA4 only) (software trigger only for TMP3).

In the one-shot pulse output mode, 16-bit timer/event counter P waits for a trigger when the TPmCTL0.TPmCE bit is set to 1. When the valid edge of an external trigger input (TIPk0) is detected, 16-bit timer/event counter P starts counting, and outputs a one-shot pulse from the TOPm1 pin.

Instead of the external trigger, a software trigger can also be generated to output the pulse. When the software trigger is used, the TOP00 pin outputs the active level while the 16-bit counter is counting, and the inactive level when the counter is stopped (waiting for a trigger).

<R> **Figure 6-27. Configuration in One-Shot Pulse Output Mode**

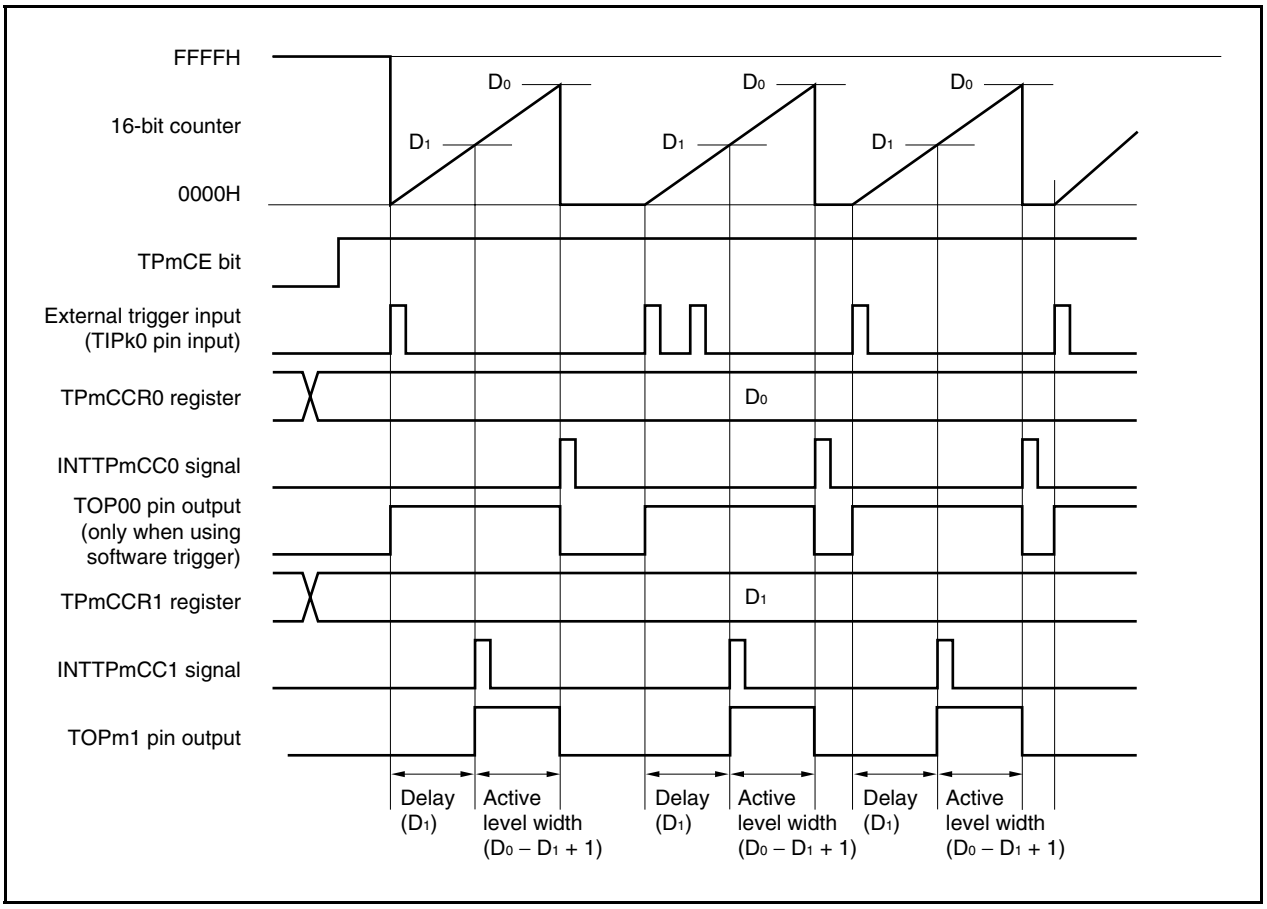


Note Because the external trigger input pin (TIP00) and timer output pin (TOP00) function alternately, two functions cannot be used at the same time.

Caution In the one-shot pulse output mode, select the internal clock as the count clock (by clearing the TPkCTL1.TPkEEE bit to 0).

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

Figure 6-28. Basic Timing in One-Shot Pulse Output Mode



When the TPmCE bit is set to 1, 16-bit timer/event counter P waits for a trigger. When the trigger is generated, the 16-bit counter is cleared from FFFFH to 0000H, starts counting, and outputs a one-shot pulse from the TOPm1 pin. After the one-shot pulse is output, the 16-bit counter is cleared to 0000H, stops counting, and waits for a trigger. When the trigger is generated again, the 16-bit counter starts counting from 0000H. If a trigger is generated again while the one-shot pulse is being output, it is ignored.

The output delay period and active level width of the one-shot pulse can be calculated as follows.

Output delay period = (Set value of TPmCCR1 register) × Count clock cycle

Active level width = (Set value of TPmCCR0 register − Set value of TPmCCR1 register + 1) × Count clock cycle

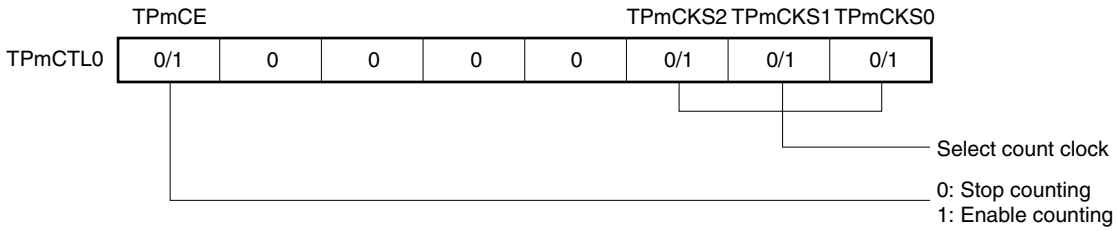
The compare match interrupt request signal INTTPmCC0 is generated when the 16-bit counter counts after its count value matches the value of the CCR0 buffer register. The compare match interrupt request signal INTTPmCC1 is generated when the count value of the 16-bit counter matches the value of the CCR1 buffer register.

The valid edge of an external trigger input (TIPk0 pin) or setting the software trigger (TPmCTL1.TPmEST bit) to 1 is used as the trigger.

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

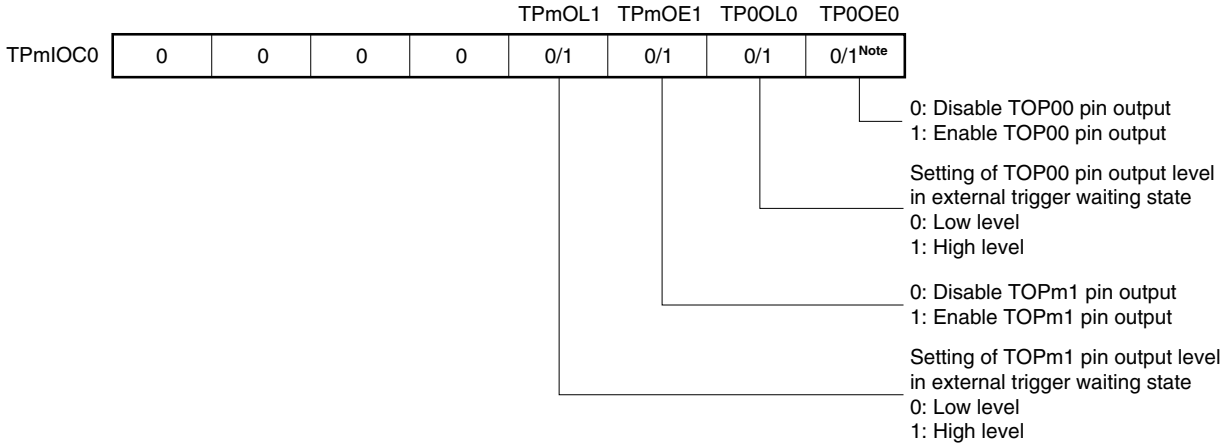
Figure 6-29. Setting of Registers in One-Shot Pulse Output Mode (1/2)

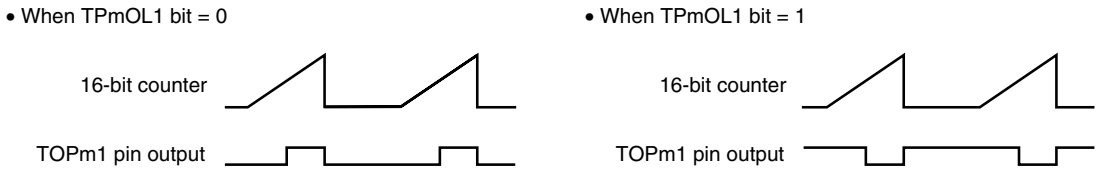
(a) TMPm control register 0 (TPmCTL0)



(b) TMPm control register 1 (TPmCTL1)

(c) TMPm I/O control register 0 (TPmIOC0)





Note Clear this bit to 0 when the TOP00 pin is not used in the one-shot pulse output mode.

Figure 6-29. Setting of Registers in One-Shot Pulse Output Mode (2/2)

(d) TMPk I/O control register 2 (TPkIOC2)

<R> TPkIOC2

						TPkEES1	TPkEES0	TPkETS1	TPkETS0
0	0	0	0	0	0	0/1	0/1		

Select valid edge of external trigger input (TIPk0 pin)

(e) TMPm counter read buffer register (TPmCNT)

The value of the 16-bit counter can be read by reading the TPmCNT register.

(f) TMPm capture/compare registers 0 and 1 (TPmCCR0 and TPmCCR1)

If D_0 is set to the TPmCCR0 register and D_1 to the TPmCCR1 register, the active level width and output delay period of the one-shot pulse are as follows.

<R> Active level width = $(D_0 - D_1 + 1) \times$ Count clock cycle

Output delay period = $D_1 \times$ Count clock cycle

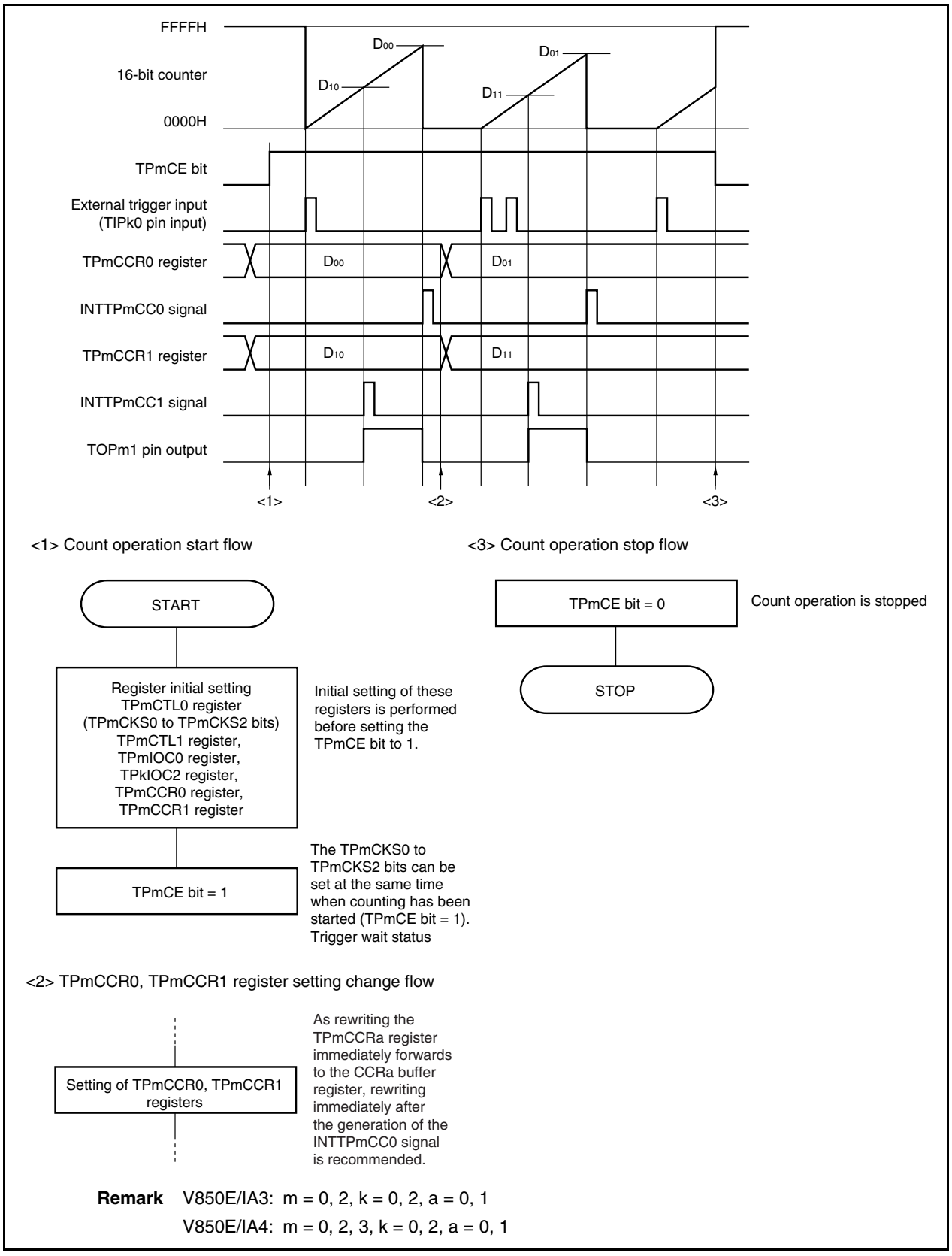
<R> **Caution One-shot pulses are not output even in the one-shot pulse output mode, if the value set in the TPmCCR1 register is greater than that set in the TPmCCR0 register.**

Remarks 1. TMPk I/O control register 1 (TPkIOC1) and TMPm option register 0 (TPmOPT0) are not used in the one-shot pulse output mode.

2. V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

(1) Operation flow in one-shot pulse output mode

<R> **Figure 6-30. Software Processing Flow in One-Shot Pulse Output Mode**

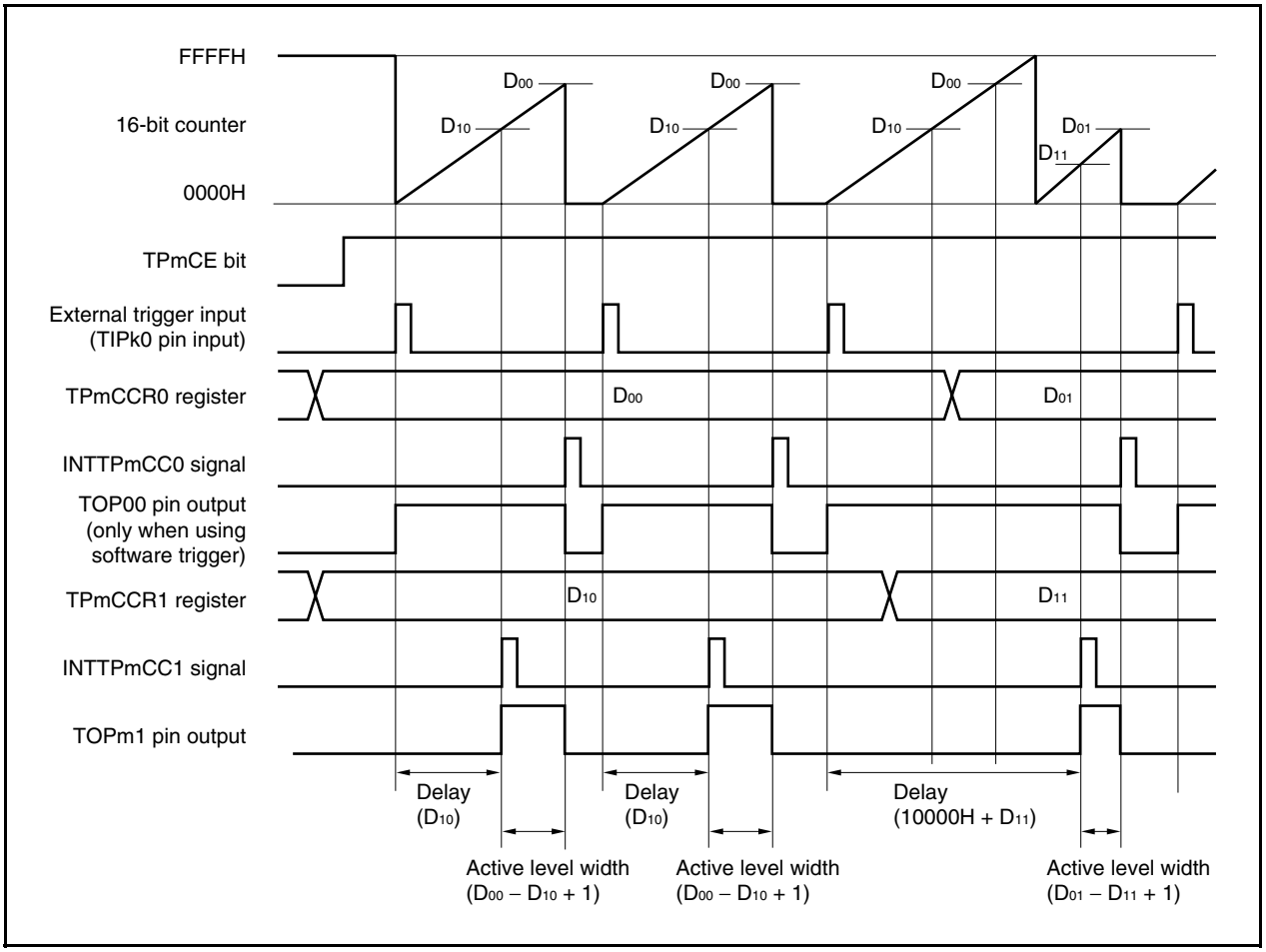


Remark V850E/IA3: m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, k = 0, 2, a = 0, 1

(2) Operation timing in one-shot pulse output mode

(a) Note on rewriting TPmCCRa register

If the value of the TPmCCRa register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When an overflow may occur, stop counting and then change the set value.



<R>

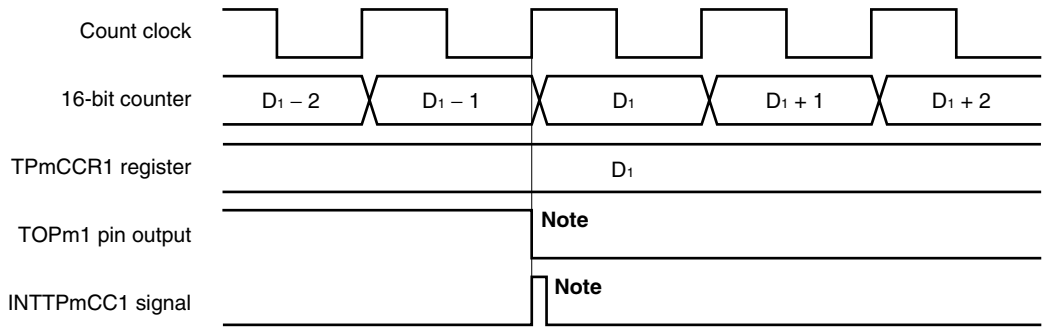
When the TPmCCR0 register is rewritten from D_{00} to D_{01} and the TPmCCR1 register from D_{10} to D_{11} where $D_{00} > D_{01}$ and $D_{10} > D_{11}$, if the TPmCCR1 register is rewritten when the count value of the 16-bit counter is greater than D_{11} and less than D_{10} and if the TPmCCR0 register is rewritten when the count value is greater than D_{01} and less than D_{00}, each set value is reflected as soon as the register has been rewritten and compared with the count value. The counter counts up to FFFFH and then counts up again from 0000H. When the count value matches D_{11}, the counter generates the INTTPmCC1 signal and asserts the TOPm1 pin. When the count value matches D_{01}, the counter generates the INTTPmCC0 signal, deasserts the TOPm1 pin, and stops counting.

Therefore, the counter may output a pulse with a delay period or active period different from that of the one-shot pulse that is originally expected.

Remark V850E/IA3: m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, k = 0, 2, a = 0, 1

(b) Generation timing of compare match interrupt request signal (INTTPmCC1)

The generation timing of the INTTPmCC1 signal in the one-shot pulse output mode is different from INTTPmCC1 signals in other modes; the INTTPmCC1 signal is generated when the count value of the 16-bit counter matches the value of the TPmCCR1 register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

Usually, the INTTPmCC1 signal is generated when the 16-bit counter counts up next time after its count value matches the value of the TPmCCR1 register.

In the one-shot pulse output mode, however, it is generated one clock earlier. This is because the timing is changed to match the change timing of the TOPm1 pin.

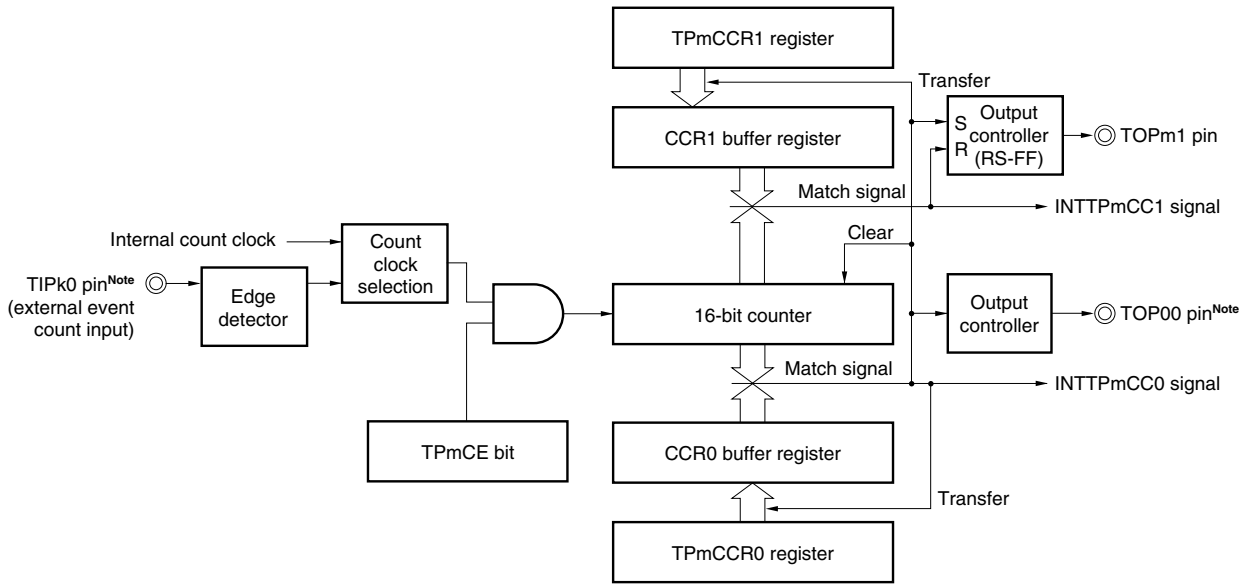
6.6.5 PWM output mode (TPmMD2 to TPmMD0 bits = 100)

This mode is valid only in TMP0, TMP2, and TMP3 (V850E/IA4 only).

In the PWM output mode, a PWM waveform is output from the TOPm1 pin when the TPmCTL0.TPmCE bit is set to 1.

In addition, a PWM waveform with a duty factor of 50% with the set value of the TPmCCR0 register + 1 as half its cycle is output from the TOP00 pin.

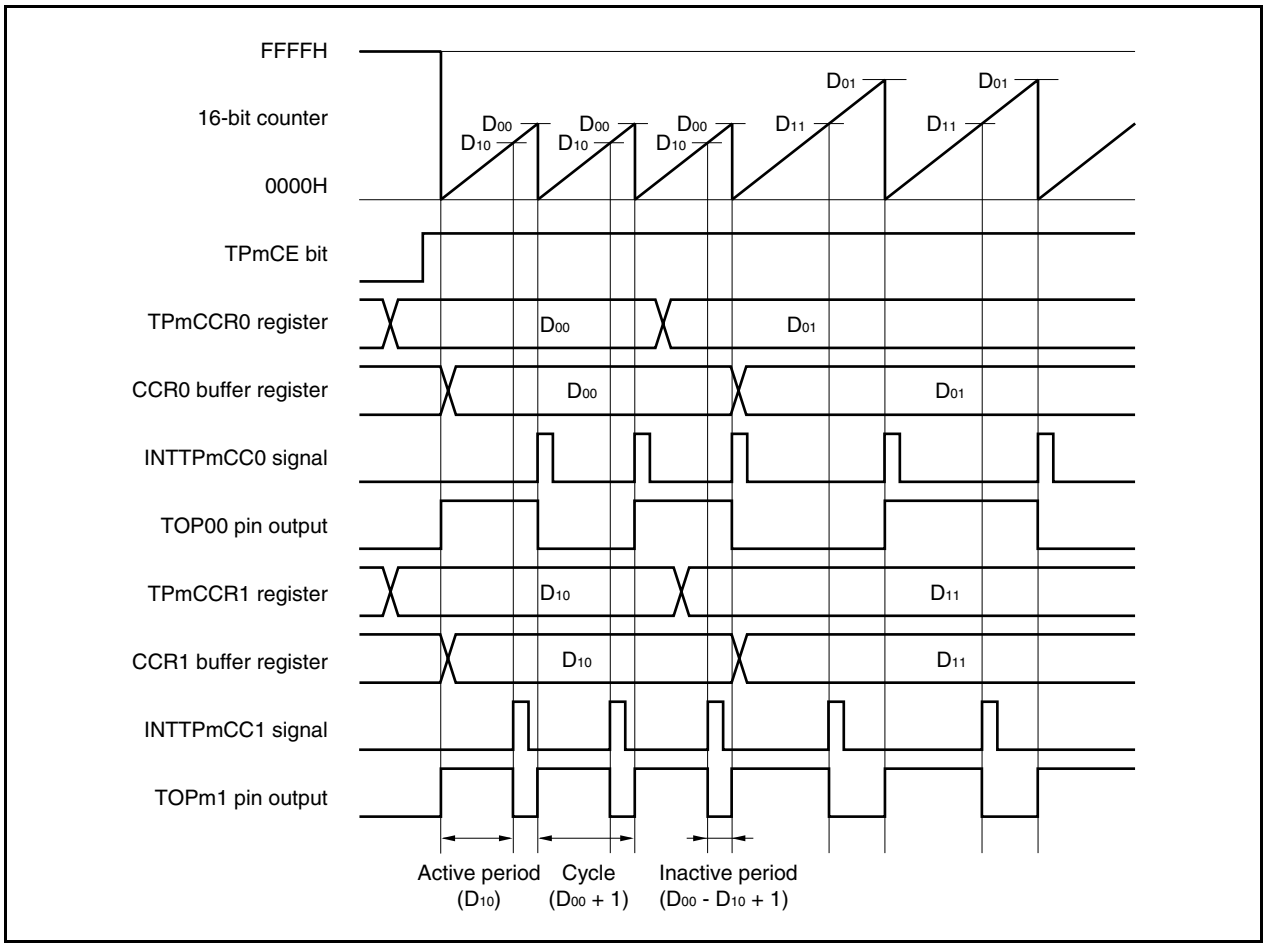
Figure 6-31. Configuration in PWM Output Mode



Note Because the external event count input pin (TIP00) and timer output pin (TOP00) function alternately, two or more functions cannot be used at the same time.

Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

Figure 6-32. Basic Timing in PWM Output Mode



When the TPmCE bit is set to 1, the 16-bit counter is cleared from FFFFH to 0000H, starts counting, and outputs a PWM waveform from the TOPm1 pin.

The active level width, cycle, and duty factor of the PWM waveform can be calculated as follows.

Active level width = (Set value of TPmCCR1 register) × Count clock cycle
Cycle = (Set value of TPmCCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TPmCCR1 register)/(Set value of TPmCCR0 register + 1)

The PWM waveform can be changed by rewriting the TPmCCRa register while the counter is operating. The newly written value is reflected when the count value of the 16-bit counter matches the value of the CCR0 buffer register and the 16-bit counter is cleared to 0000H.

The compare match interrupt request signal INTTPmCC0 is generated when the 16-bit counter counts next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The compare match interrupt request signal INTTPmCC1 is generated when the count value of the 16-bit counter matches the value of the CCR1 buffer register.

The value set to the TPmCCRa register is transferred to the CCRa buffer register when the count value of the 16-bit counter matches the value of the CCRa buffer register and the 16-bit counter is cleared to 0000H.

Remark V850E/IA3: m = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, a = 0, 1

Figure 6-33. Register Setting in PWM Output Mode (1/2)

(a) TMPm control register 0 (TPmCTL0)

	TPmCE					TPmCKS2	TPmCKS1	TPmCKS0
TPmCTL0	0/1	0	0	0	0	0/1	0/1	0/1

Select count clock[Note]

0: Stop counting
1: Enable counting

Note The setting is invalid when the TPmCTL1.TPkEEE bit = 1.

(b) TMPm control register 1 (TPmCTL1)

	TP0SYE	TPmEST	TPkEEE			TPmMD2	TPmMD1	TPmMD0
TPmCTL1	0	0	0/1	0	0	1	0	0

1, 0, 0:
PWM output mode

0: Operate on count clock selected by TPkCKS0 to TPkCKS2 bits
1: Count with external event count input signal

(c) TMPm I/O control register 0 (TPmIOC0)

<R>

					TPmOL1	TPmOE1	TP0OL0	TP0OE0
TPmIOC0	0	0	0	0	0/1	0/1	0/1	0/1[Note]

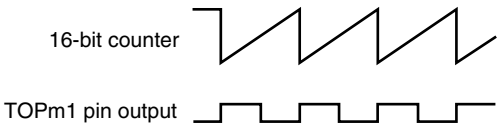
0: Disable TOP00 pin output
1: Enable TOP00 pin output

Setting of TOP00 pin output level before count operation
0: Low level
1: High level

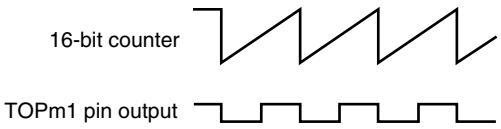
0: Disable TOPm1 pin output
1: Enable TOPm1 pin output

Setting of TOPm1 pin output level before count operation
0: Low level
1: High level

- When TPmOL1 bit = 0



- When TPmOL1 bit = 1

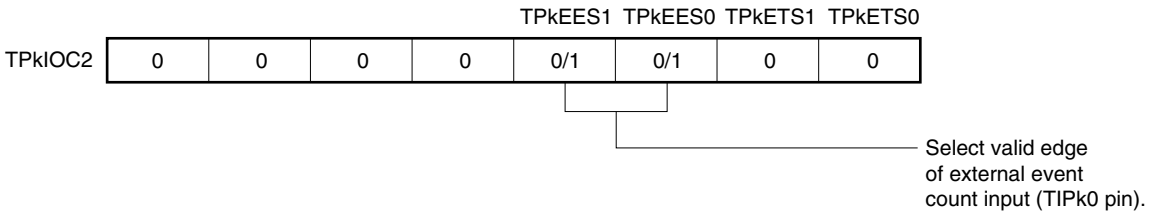


<R>

Note Clear this bit to 0 when the TOP00 pin is not used in the PWM output mode.

Figure 6-33. Register Setting in PWM Output Mode (2/2)

(d) TMPk I/O control register 2 (TPkIOC2)



(e) TMPm counter read buffer register (TPmCNT)

The value of the 16-bit counter can be read by reading the TPmCNT register.

(f) TMPm capture/compare registers 0 and 1 (TPmCCR0 and TPmCCR1)

If D_0 is set to the TPmCCR0 register and D_1 to the TPmCCR1 register, the cycle and active level of the PWM waveform are as follows.

Cycle = $(D_0 + 1) \times$ Count clock cycle

Active level width = $D_1 \times$ Count clock cycle

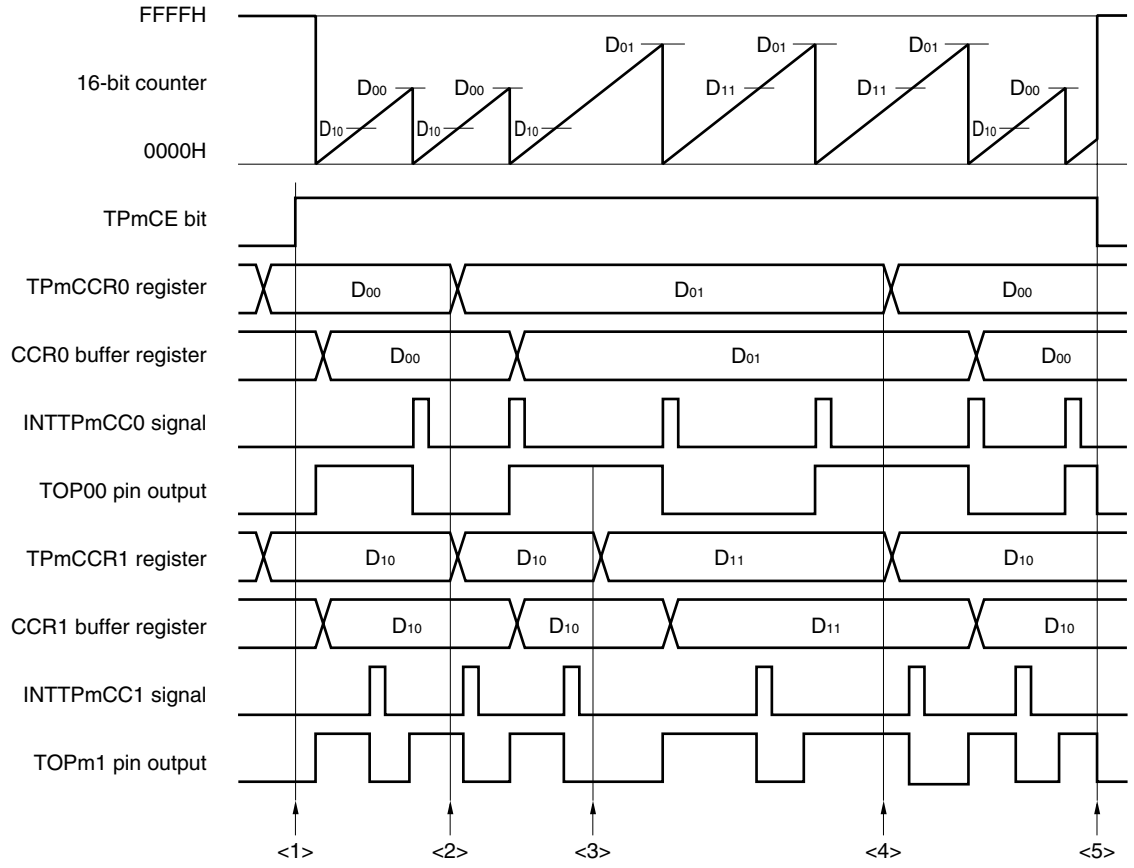
Remarks 1. TMPk I/O control register 1 (TPkIOC1) and TMPk option register 0 (TPkOPT0) are not used in the PWM output mode.

2. V850E/IA3: m = 0, 2, k = 0, 2

V850E/IA4: m = 0, 2, 3, k = 0, 2

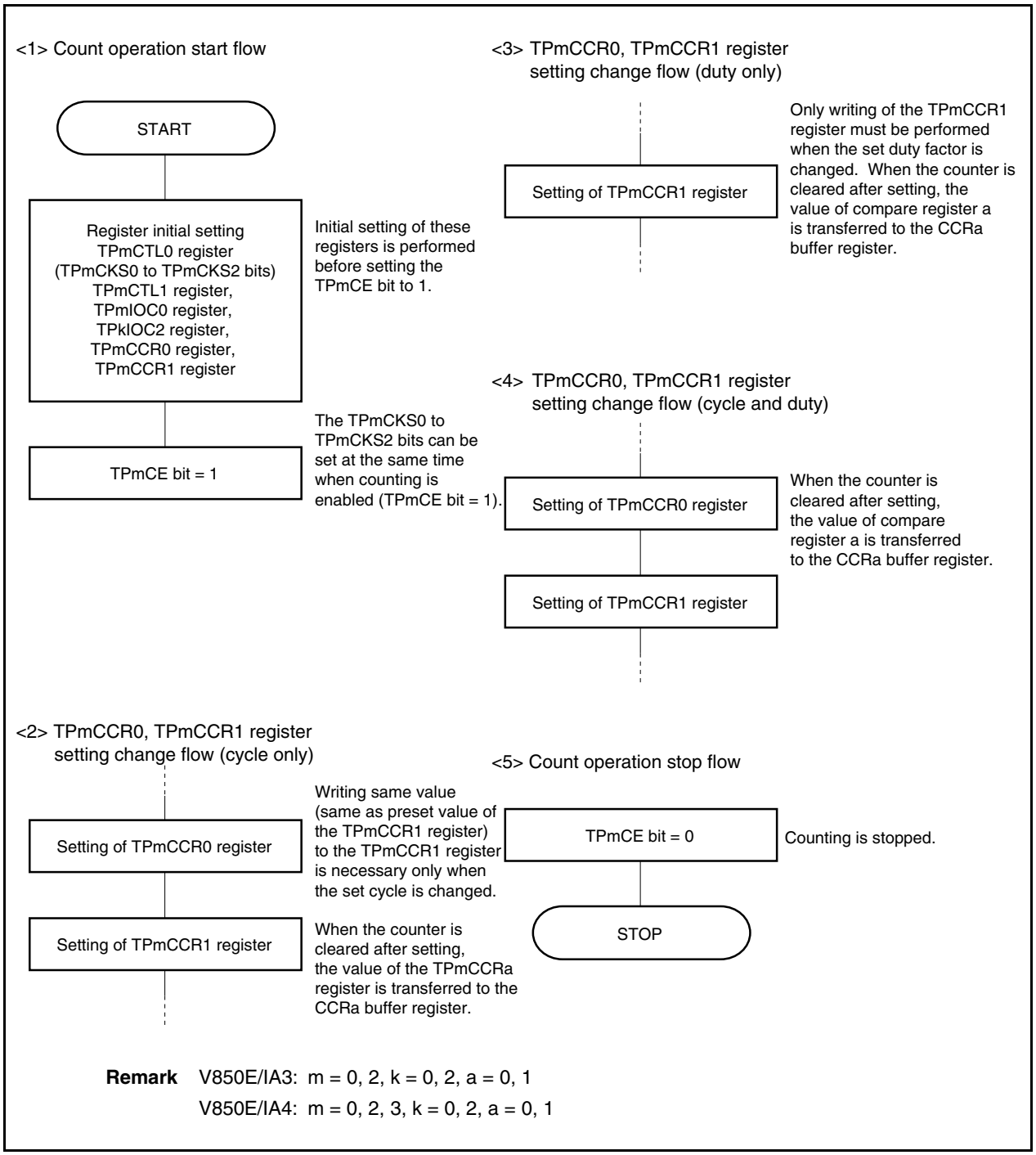
(1) Operation flow in PWM output mode

Figure 6-34. Software Processing Flow in PWM Output Mode (1/2)



Remark V850E/IA3: m = 0, 2, k = 0, 2
V850E/IA4: m = 0, 2, 3, k = 0, 2

Figure 6-34. Software Processing Flow in PWM Output Mode (2/2)



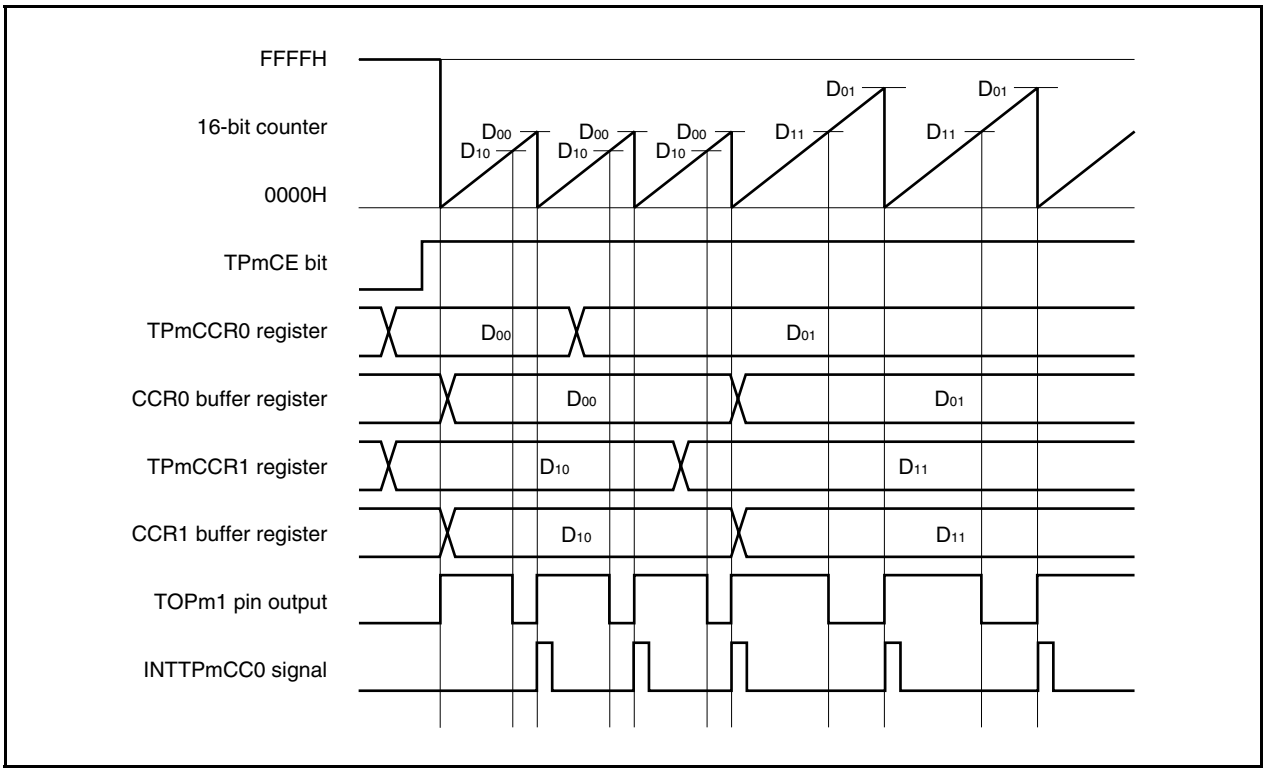
Remark V850E/IA3: m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, k = 0, 2, a = 0, 1

(2) PWM output mode operation timing

(a) Changing pulse width during operation

To change the PWM waveform while the counter is operating, write the TPmCCR1 register last.

<R> Rewrite the TPmCCRa register after writing the TPmCCR1 register after the INTTPmCC0 signal is detected.



To transfer data from the TPmCCRa register to the CCRa buffer register, the TPmCCR1 register must be written.

To change both the cycle and active level of the PWM waveform at this time, first set the cycle to the TPmCCR0 register and then set the active level to the TPmCCR1 register.

To change only the cycle of the PWM waveform, first set the cycle to the TPmCCR0 register, and then write the same value (same as preset value of the TPmCCR1 register) to the TPmCCR1 register.

To change only the active level width (duty factor) of the PWM waveform, only the TPmCCR1 register has to be set.

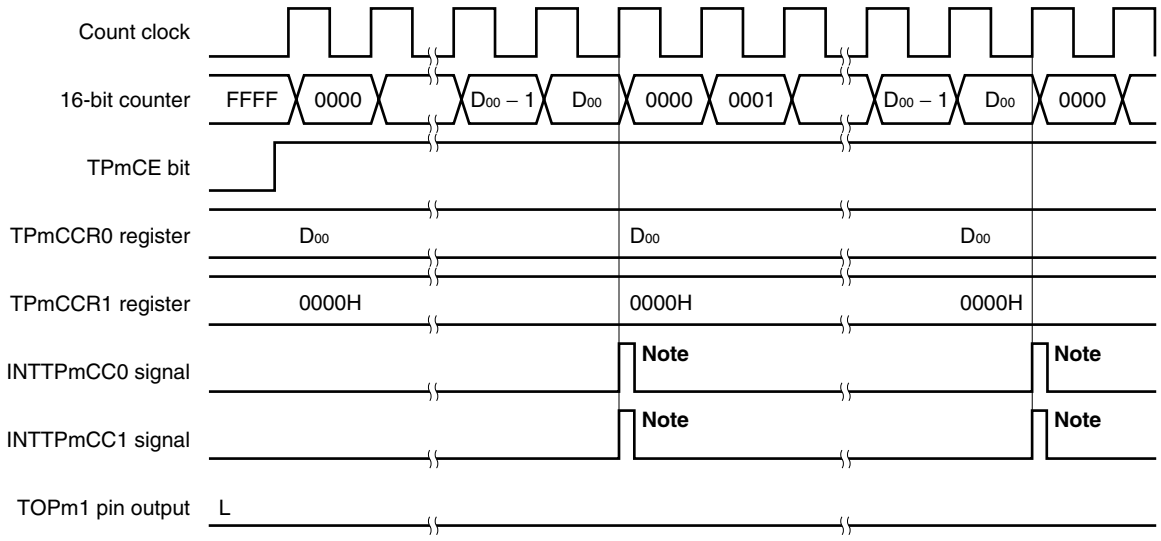
After data is written to the TPmCCR1 register, the value written to the TPmCCRa register is transferred to the CCRa buffer register in synchronization with clearing of the 16-bit counter, and is used as the value compared with the 16-bit counter.

To write the TPmCCR0 or TPmCCR1 register again after writing the TPmCCR1 register once, do so after the INTTPmCC0 signal is generated. Otherwise, the value of the CCRa buffer register may become undefined because the timing of transferring data from the TPmCCRa register to the CCRa buffer register conflicts with writing the TPmCCRa register.

Remark V850E/IA3: m = 0, 2, a = 0, 1
V850E/IA4: m = 0, 2, 3, a = 0, 1

(b) 0%/100% output of PWM waveform

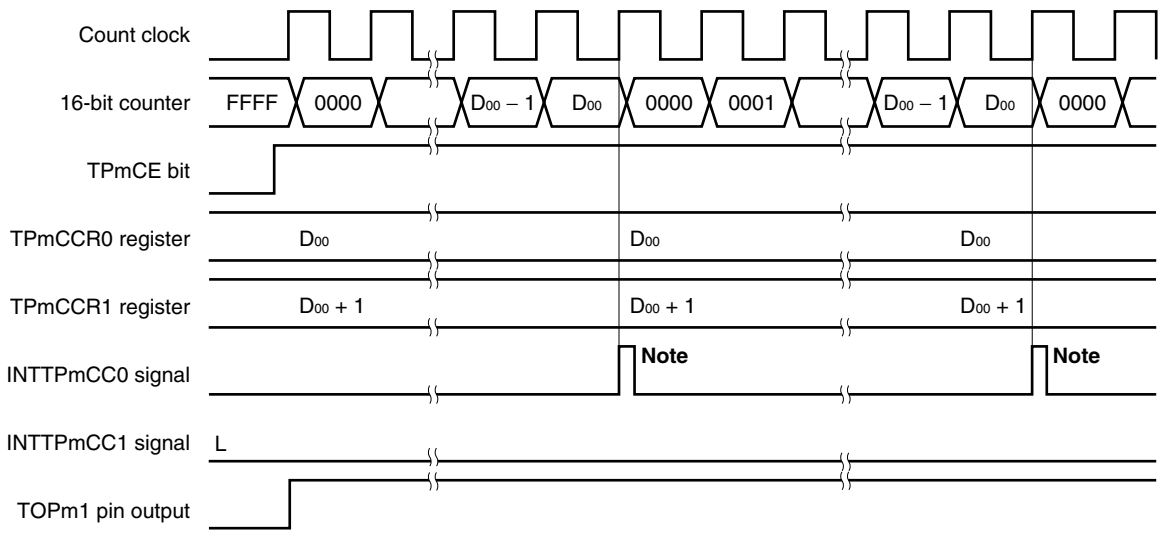
<R> To output a 0% waveform, set the TPmCCR1 register to 0000H. The 16-bit counter is cleared to 0000H and the INTTPmCC0 and INTTPmCC1 signals are generated at the next timing after a match between the count value of the 16-bit counter and the value of the CCR0 buffer register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

To output a 100% waveform, set a value of (set value of TPmCCR0 register + 1) to the TPmCCR1 register. If the set value of the TPmCCR0 register is FFFFH, 100% output cannot be produced.

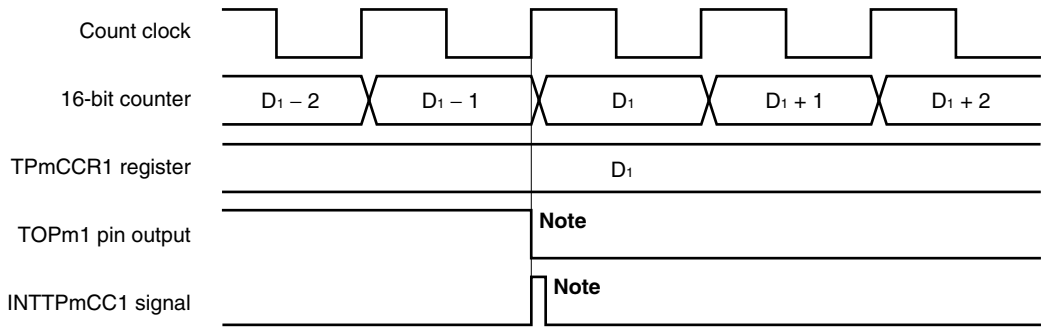


Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

(c) Generation timing of compare match interrupt request signal (INTTPmCC1)

The timing of generation of the INTTPmCC1 signal in the PWM output mode differs from the timing of INTTPmCC1 signals in other modes; the INTTPmCC1 signal is generated when the count value of the 16-bit counter matches the value of the TPmCCR1 register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark V850E/IA3: m = 0, 2
V850E/IA4: m = 0, 2, 3

Usually, the INTTPmCC1 signal is generated in synchronization with the next counting up after the count value of the 16-bit counter matches the value of the TPmCCR1 register.

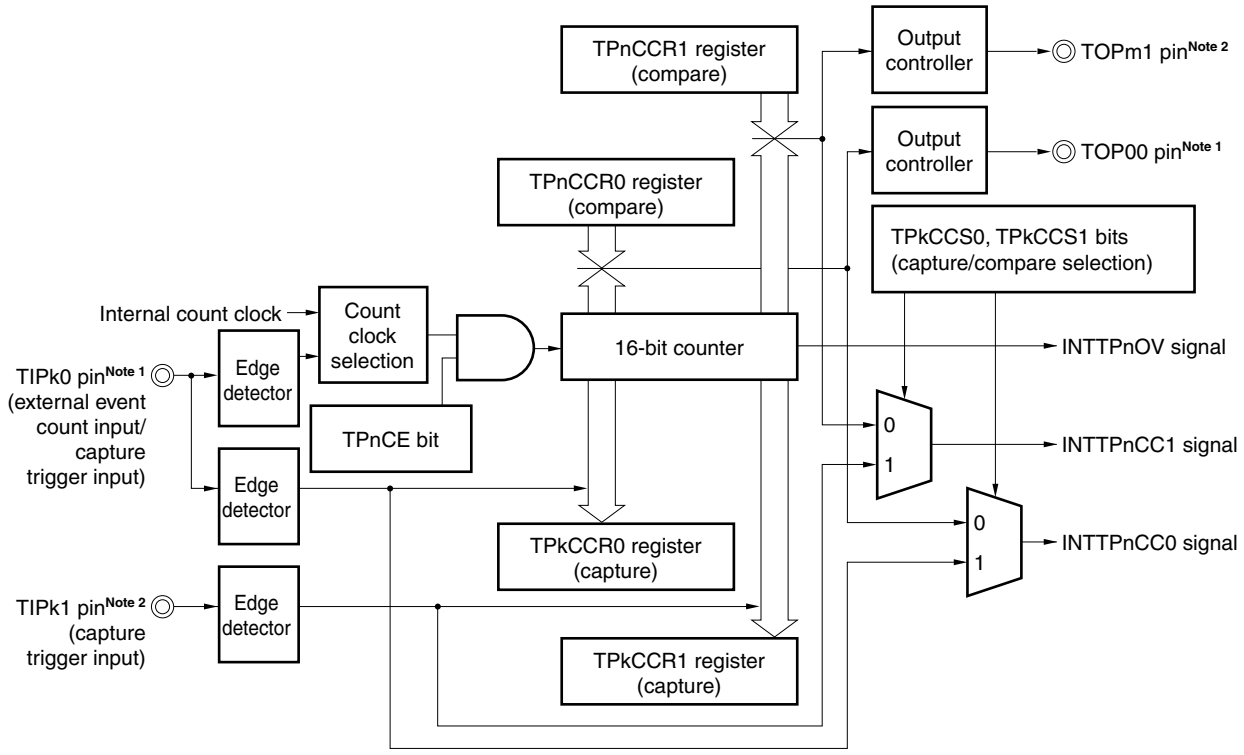
In the PWM output mode, however, it is generated one clock earlier. This is because the timing is changed to match the change timing of the output signal of the TOPm1 pin.

6.6.6 Free-running timer mode (TPnMD2 to TPnMD0 bits = 101)

The compare function is valid for all of TMP0 to TMP3. The capture function is valid only for TMP0 and TMP2.

In the free-running timer mode, 16-bit timer/event counter P starts counting when the TPnCTL0.TPnCE bit is set to 1. At this time, the TPkCCR0 and TPkCCR1 registers can be used as compare registers or capture registers, depending on the setting of the TPkOPT0.TPkCCS0 and TPkOPT0.TPkCCS1 bits.

Figure 6-35. Configuration in Free-Running Timer Mode



Notes 1. Because the external event count input pin (TIP00), capture trigger input pin (TIP00) and timer output pin (TOP00) function alternately, two or more functions cannot be used at the same time.

2. Because the capture trigger input pin (TIPk1) and timer output pin (TOPk1) function alternately, two functions cannot be used at the same time.

Remark V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2

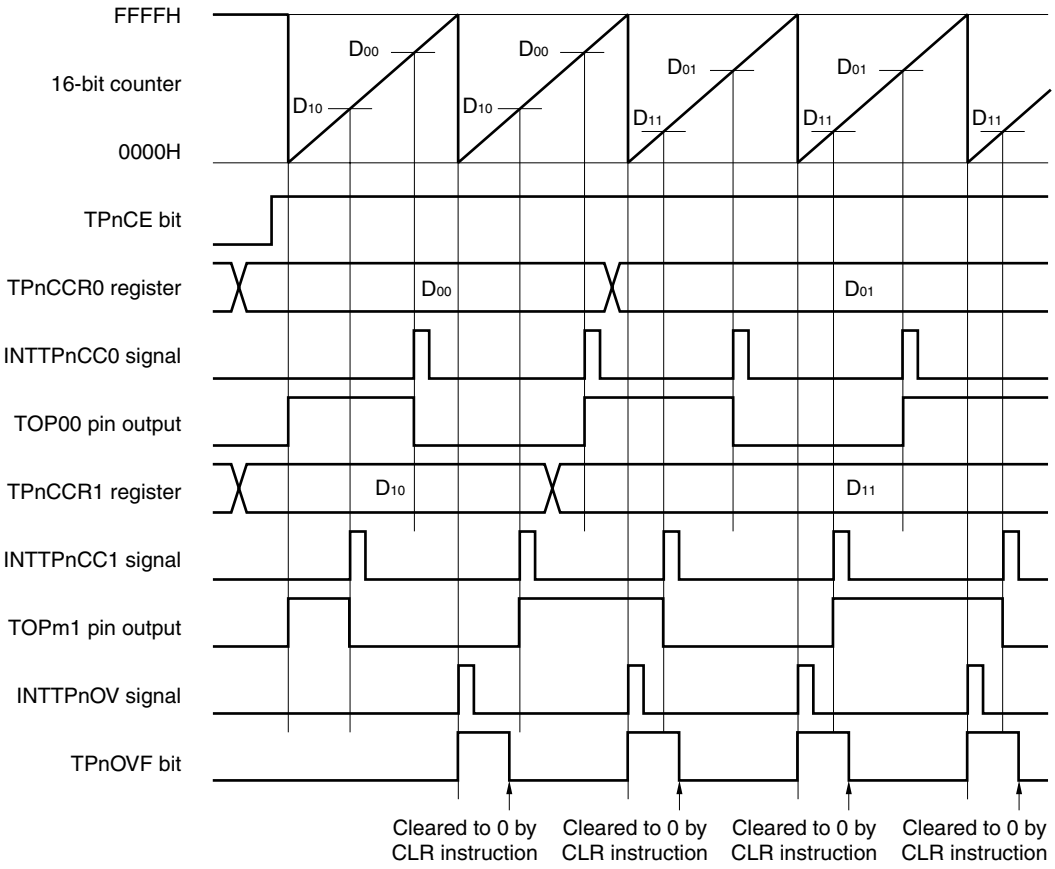
- Compare operation

  When the TPnCE bit is set to 1, 16-bit timer/event counter P starts counting, and the output signals of the TOP00 and TOPm1 pins are inverted. When the count value of the 16-bit counter later matches the set value of the TPnCCRa register, a compare match interrupt request signal (INTTPnCCa) is generated, and the output signals of the TOP00 and TOPm1 pins are inverted.

  The 16-bit counter continues counting in synchronization with the count clock. When it counts up to FFFFH, it generates an overflow interrupt request signal (INTTPnOV) at the next clock, is cleared to 0000H, and continues counting. At this time, the overflow flag (TPnOPT0.TPnOVF bit) is also set to 1. Confirm that the overflow flag is set to 1 and then clear it to 0 by executing the CLR instruction via software.

  The TPnCCRa register can be rewritten while the counter is operating. If it is rewritten, the new value is reflected at that time by anytime write, and compared with the count value.

Figure 6-36. Basic Timing in Free-Running Timer Mode (Compare Function)



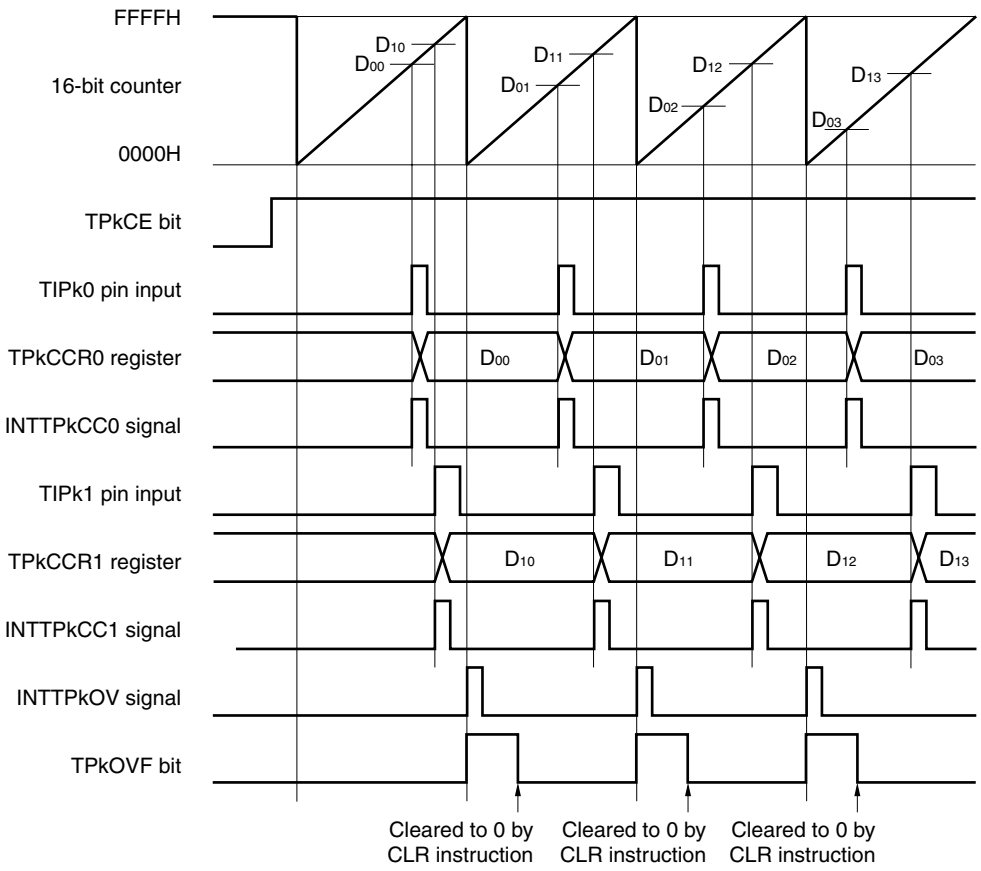
Remark V850E/IA3: n = 0 to 3, m = 0, 2, a = 0, 1
V850E/IA4: n = 0 to 3, m = 0, 2, 3, a = 0, 1

- Capture operation

  When the TPkCE bit is set to 1, the 16-bit counter starts counting. When the valid edge input to the TIPka pin is detected, the count value of the 16-bit counter is stored in the TPkCCRa register, and a capture interrupt request signal (INTTPkCCa) is generated.

  The 16-bit counter continues counting in synchronization with the count clock. When it counts up to FFFFH, it generates an overflow interrupt request signal (INTTPkOV) at the next clock, is cleared to 0000H, and continues counting. At this time, the overflow flag (TPkOPT0.TPkOVF bit) is also set to 1. Confirm that the overflow flag is set to 1 and then clear it to 0 by executing the CLR instruction via software.

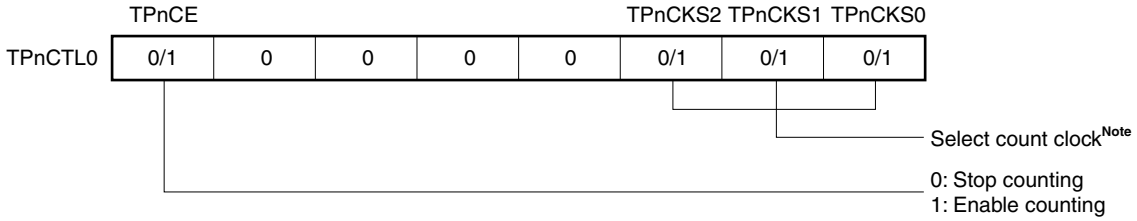
Figure 6-37. Basic Timing in Free-Running Timer Mode (Capture Function)



Remark k = 0, 2

Figure 6-38. Register Setting in Free-Running Timer Mode (1/2)

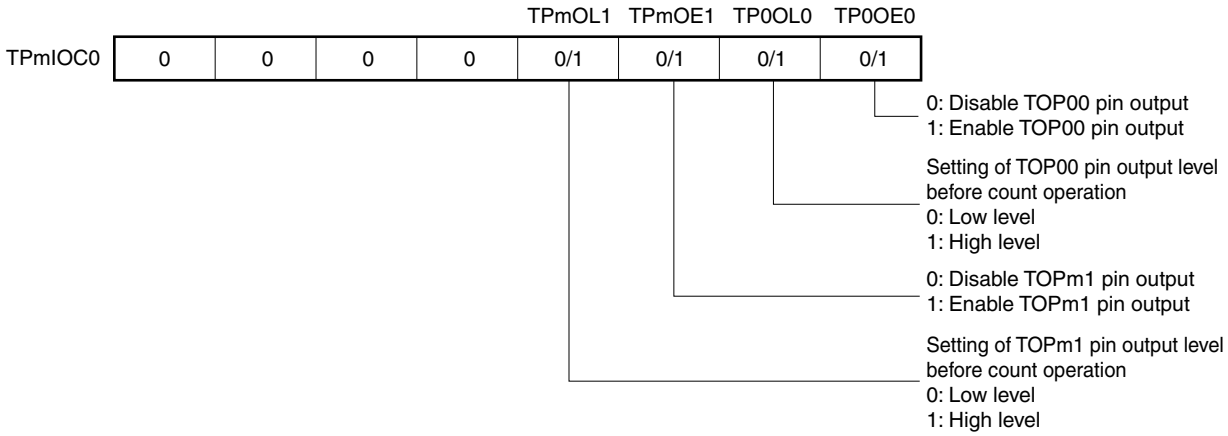
(a) TMPn control register 0 (TPnCTL0)



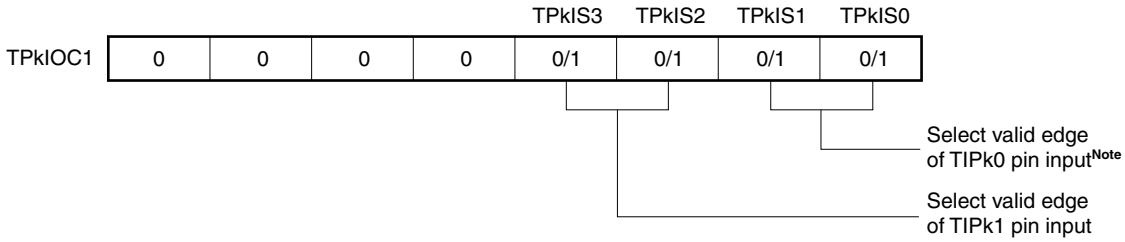
Note The setting is invalid when the TPkCTL1.TPkEEE bit = 1.

(b) TMPn control register 1 (TPnCTL1)

(c) TMPm I/O control register 0 (TPmIOC0)



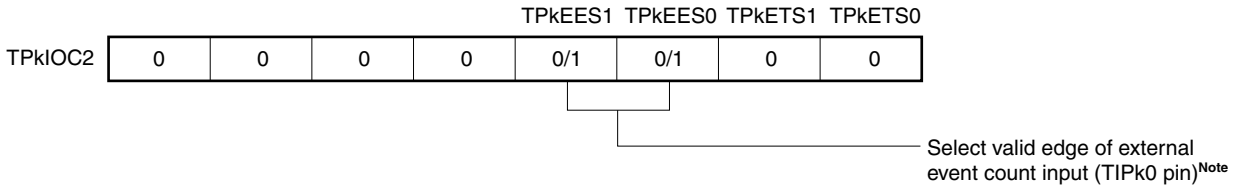
(d) TMPk I/O control register 1 (TPkIOC1)



Note Set the valid edge selection of the unused alternate external input signals to “No edge detection”.

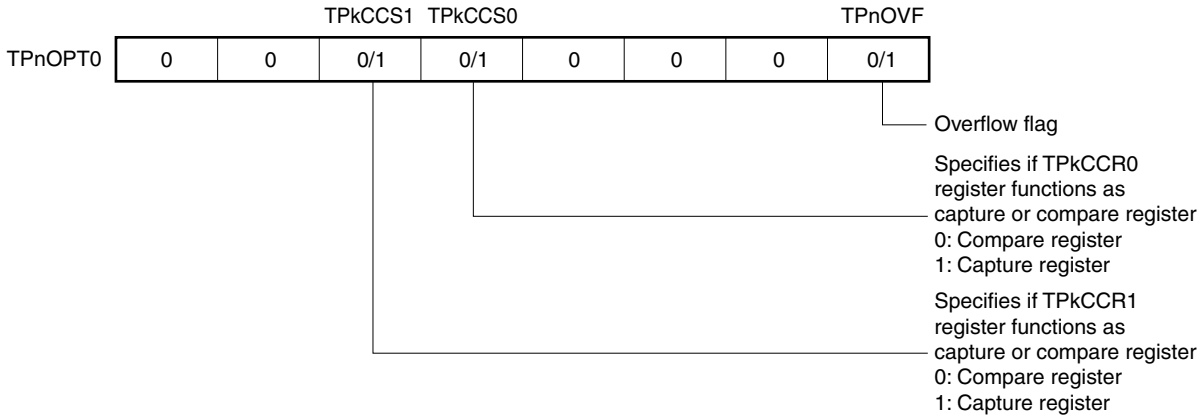
Figure 6-38. Register Setting in Free-Running Timer Mode (2/2)

(e) TMPk I/O control register 2 (TPkIOC2)



Note Set the valid edge selection of the unused alternate external input signals to “No edge detection”.

(f) TMPn option register 0 (TPnOPT0)



(g) TMPn counter read buffer register (TPnCNT)

The value of the 16-bit counter can be read by reading the TPnCNT register.

(h) TMPn capture/compare registers 0 and 1 (TPnCCR0 and TPnCCR1)

These registers function as capture registers or compare registers depending on the setting of the TPkOPT0.TPkCCSa bit.

When the registers function as capture registers, they store the count value of the 16-bit counter when the valid edge input to the TIPka pin is detected.

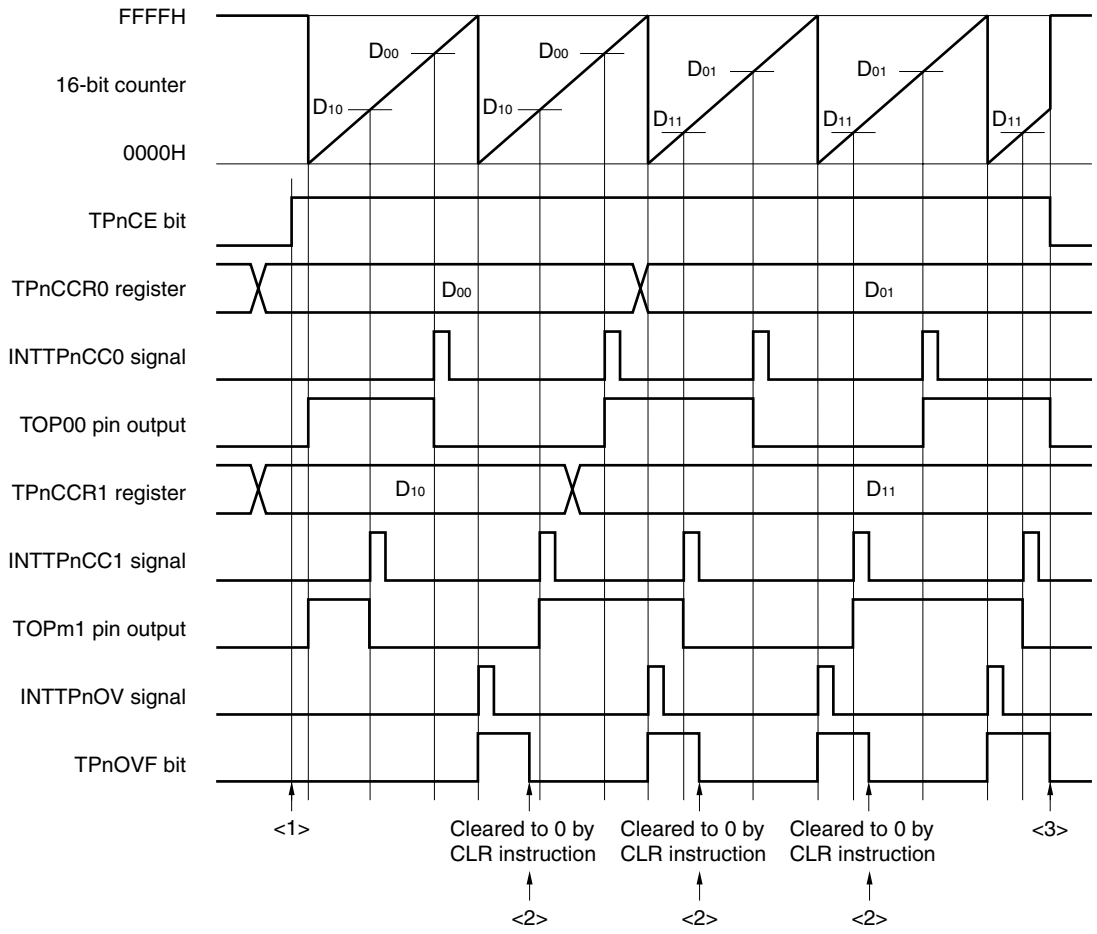
When the registers function as compare registers and when D_a is set to the TPnCCRa register, the INTTPnCCa signal is generated when the counter reaches (D_a + 1), and the output signals of the TOP00 and TOPm1 pins are inverted.

Remark V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2, a = 0, 1
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2, a = 0, 1

(1) Operation flow in free-running timer mode

(a) When using capture/compare register as compare register

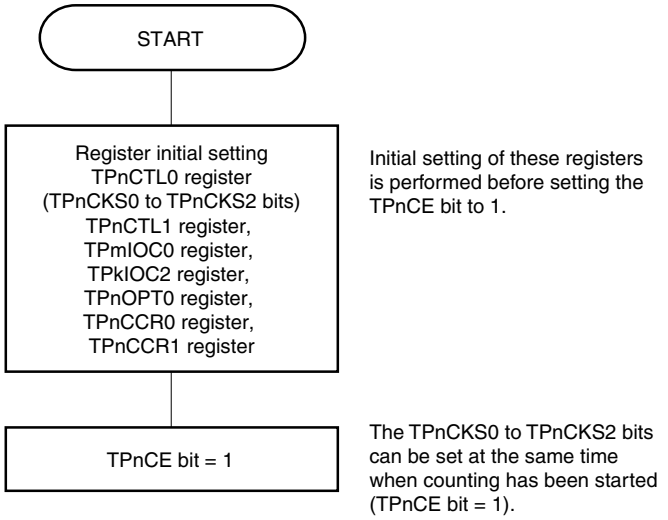
Figure 6-39. Software Processing Flow in Free-Running Timer Mode (Compare Function) (1/2)



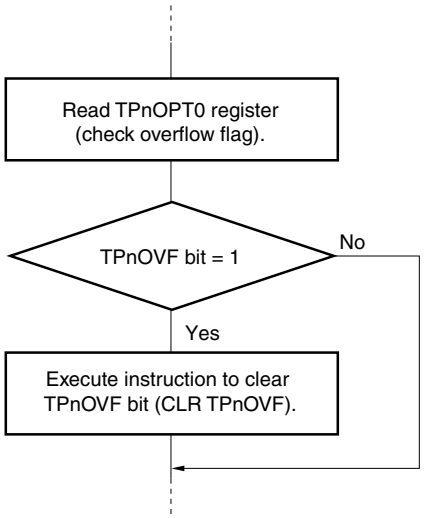
Remark V850E/IA3: n = 0 to 3, m = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3

Figure 6-39. Software Processing Flow in Free-Running Timer Mode (Compare Function) (2/2)

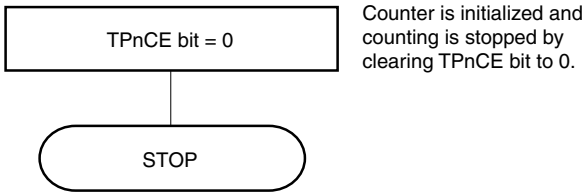
<1> Count operation start flow



<2> Overflow flag clear flow



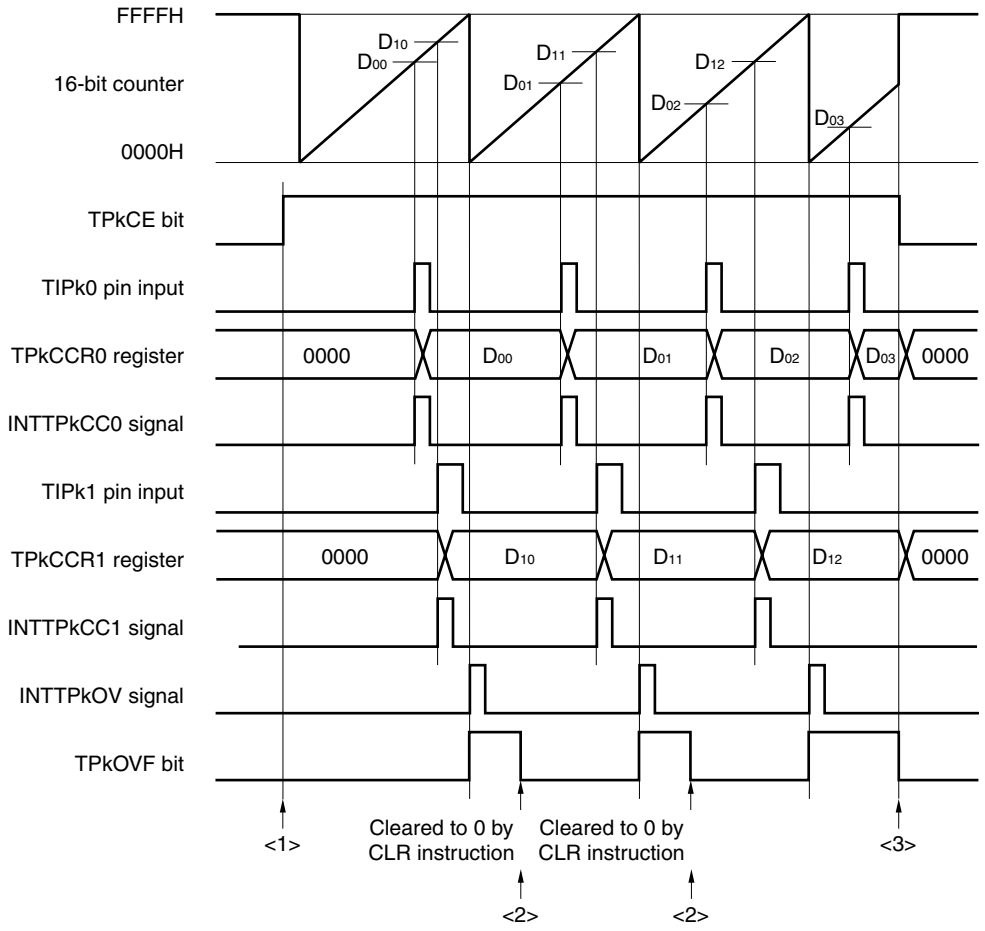
<3> Count operation stop flow



Remark V850E/IA3: n = 0 to 3, m = 0, 2, k = 0, 2
V850E/IA4: n = 0 to 3, m = 0, 2, 3, k = 0, 2

(b) When using capture/compare register as capture register

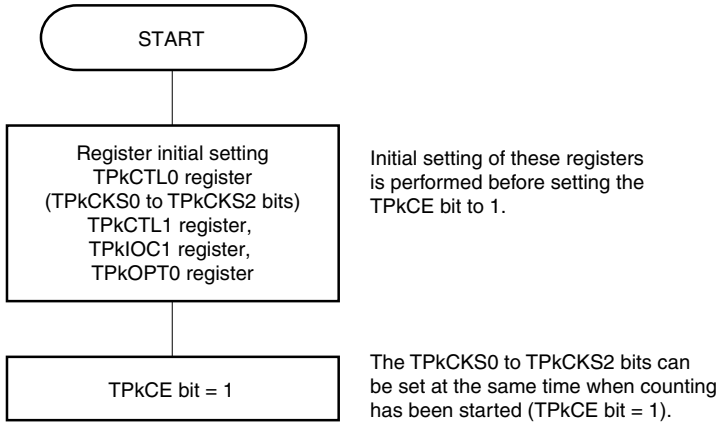
Figure 6-40. Software Processing Flow in Free-Running Timer Mode (Capture Function) (1/2)



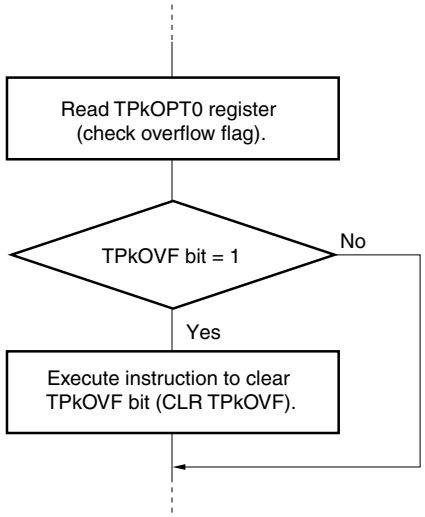
Remark k = 0, 2

Figure 6-40. Software Processing Flow in Free-Running Timer Mode (Capture Function) (2/2)

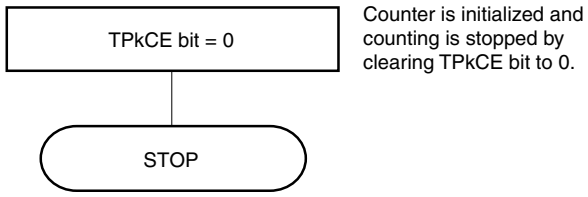
<1> Count operation start flow



<2> Overflow flag clear flow



<3> Count operation stop flow

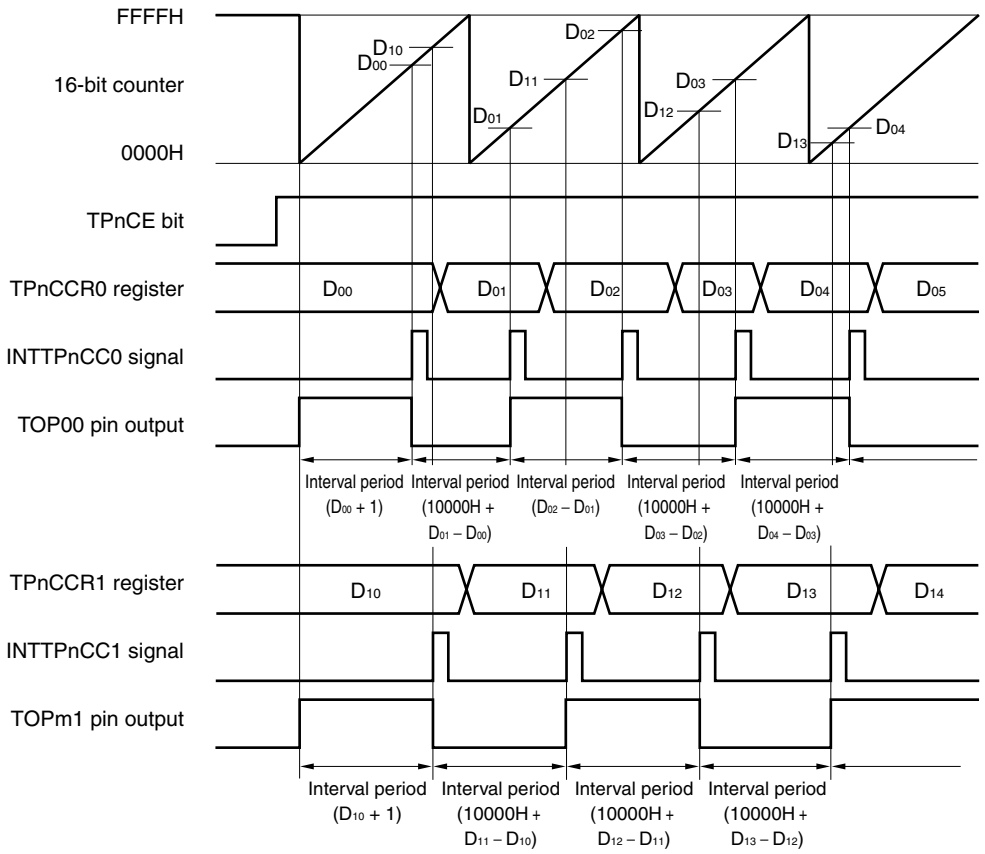


Remark k = 0, 2

(2) Operation timing in free-running timer mode

(a) Interval operation with compare register

When 16-bit timer/event counter P is used as an interval timer with the TPnCCRa register used as a compare register, software processing is necessary for setting a comparison value to generate the next interrupt request signal each time the INTTPnCCa signal has been detected.



When performing an interval operation in the free-running timer mode, two intervals can be set with one channel.

To perform the interval operation, the value of the corresponding TPnCCRa register must be re-set in the interrupt servicing that is executed when the INTTPnCCa signal is detected.

The set value for re-setting the TPnCCRa register can be calculated by the following expression, where "D_a" is the interval period.

Compare register default value: $D_a - 1$

Value set to compare register second and subsequent time: Previous set value + D_a

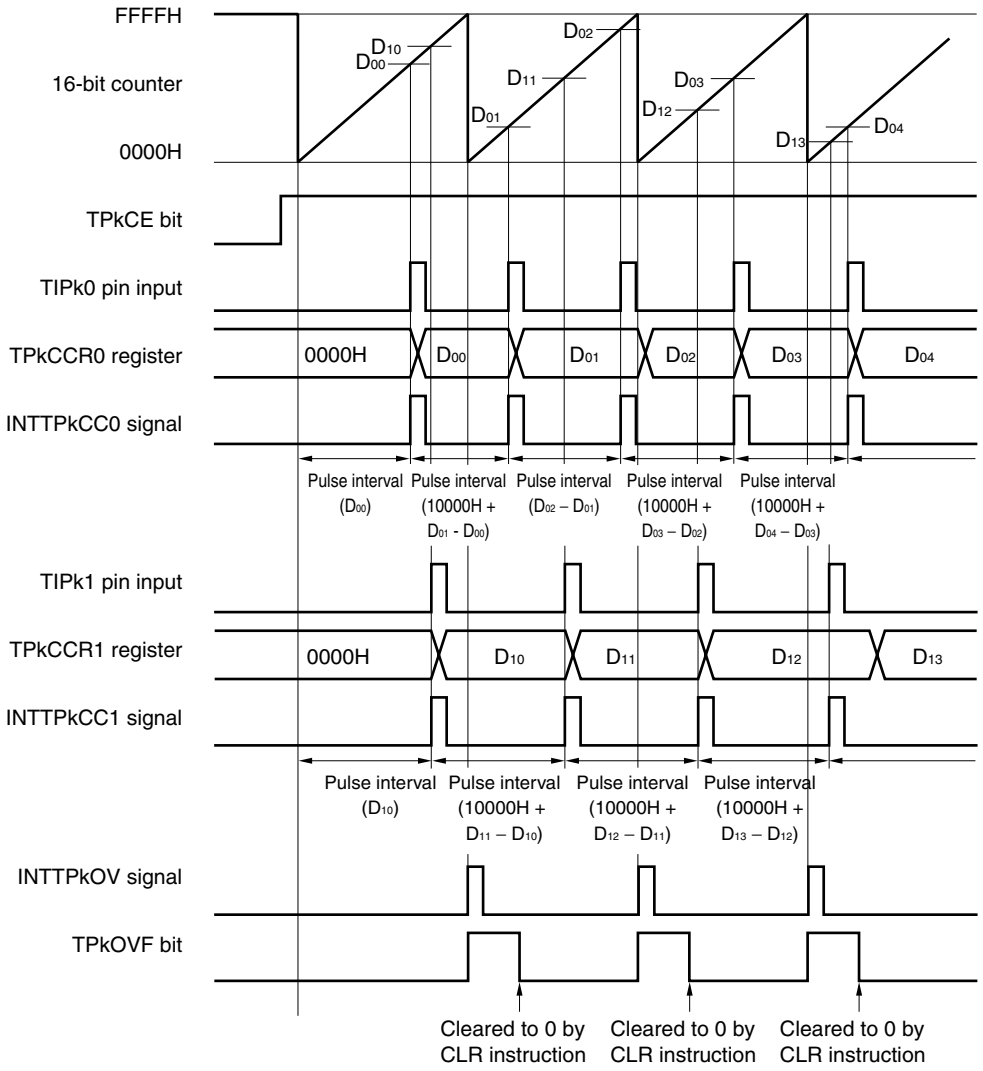
(If the calculation result is greater than FFFFH, subtract 10000H from the result and set this value to the register.)

Remark V850E/IA3: n = 0 to 3, m = 0, 2, a = 0, 1

V850E/IA4: n = 0 to 3, m = 0, 2, 3, a = 0, 1

(b) Pulse width measurement with capture register

When pulse width measurement is performed with the TPkCCRa register used as a capture register, software processing is necessary for reading the capture register each time the INTTPkCCa signal has been detected and for calculating an interval.



When executing pulse width measurement in the free-running timer mode, two pulse widths can be measured with one channel.

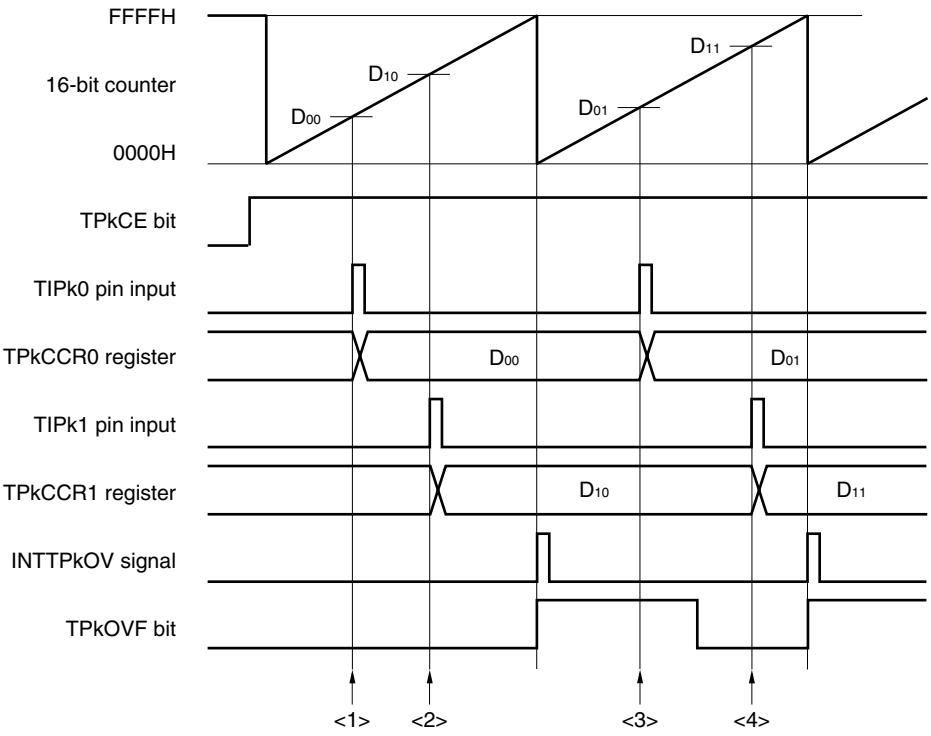
To measure a pulse width, the pulse width can be calculated by reading the value of the TPkCCRa register in synchronization with the INTTPkCCa signal, and calculating the difference between the read value and the previously read value.

Remark k = 0, 2
a = 0, 1

(c) Processing of overflow when two capture registers are used

Care must be exercised in processing the overflow flag when two capture registers are used. First, an example of incorrect processing is shown below.

Example of incorrect processing when two capture registers are used



The following problem may occur when two pulse widths are measured in the free-running timer mode.

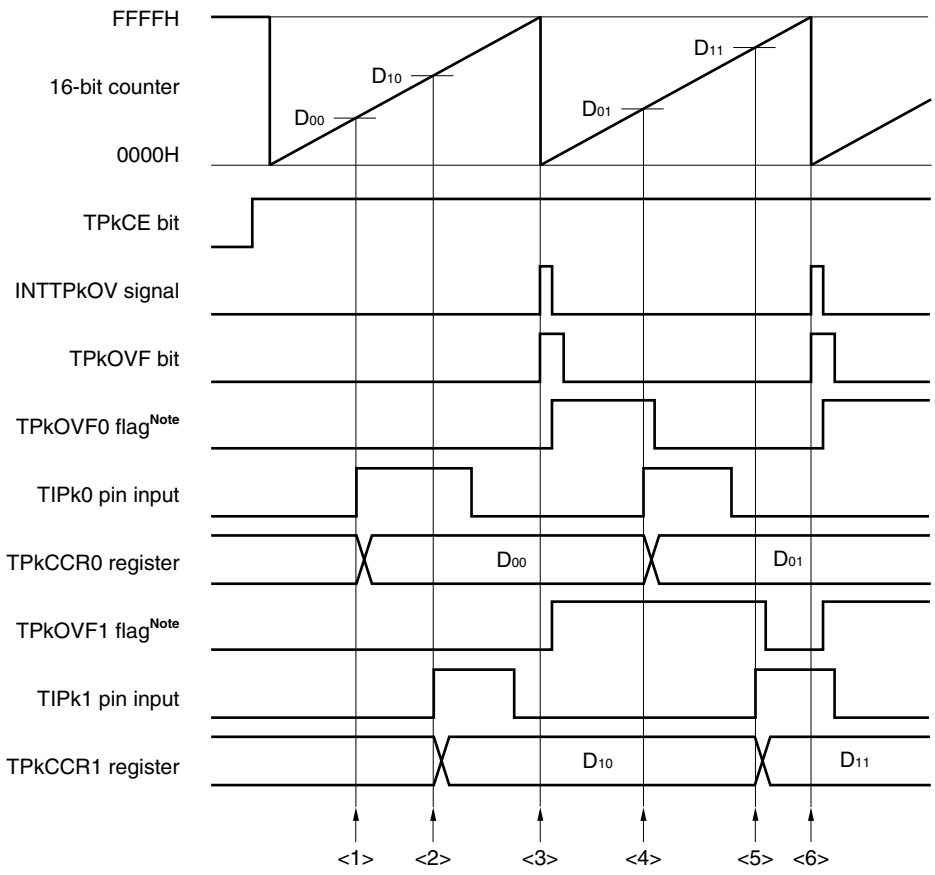
<1> Read the TPkCCR0 register (setting of the default value of the TIPk0 pin input).
<2> Read the TPkCCR1 register (setting of the default value of the TIPk1 pin input).
<3> Read the TPkCCR0 register.
Read the overflow flag. If the overflow flag is 1, clear it to 0.
Because the overflow flag is 1, the pulse width can be calculated by (10000H + D_{01} − D_{00}).
<4> Read the TPkCCR1 register.
Read the overflow flag. Because the flag is cleared in <3>, 0 is read.
Because the overflow flag is 0, the pulse width can be calculated by (D_{11} − D_{10}) (incorrect).

Remark k = 0, 2

When two capture registers are used, and if the overflow flag is cleared to 0 by one capture register, the other capture register may not obtain the correct pulse width.
Use software when using two capture registers. An example of how to use software is shown below.

(1/2)

Example when two capture registers are used (using overflow interrupt)



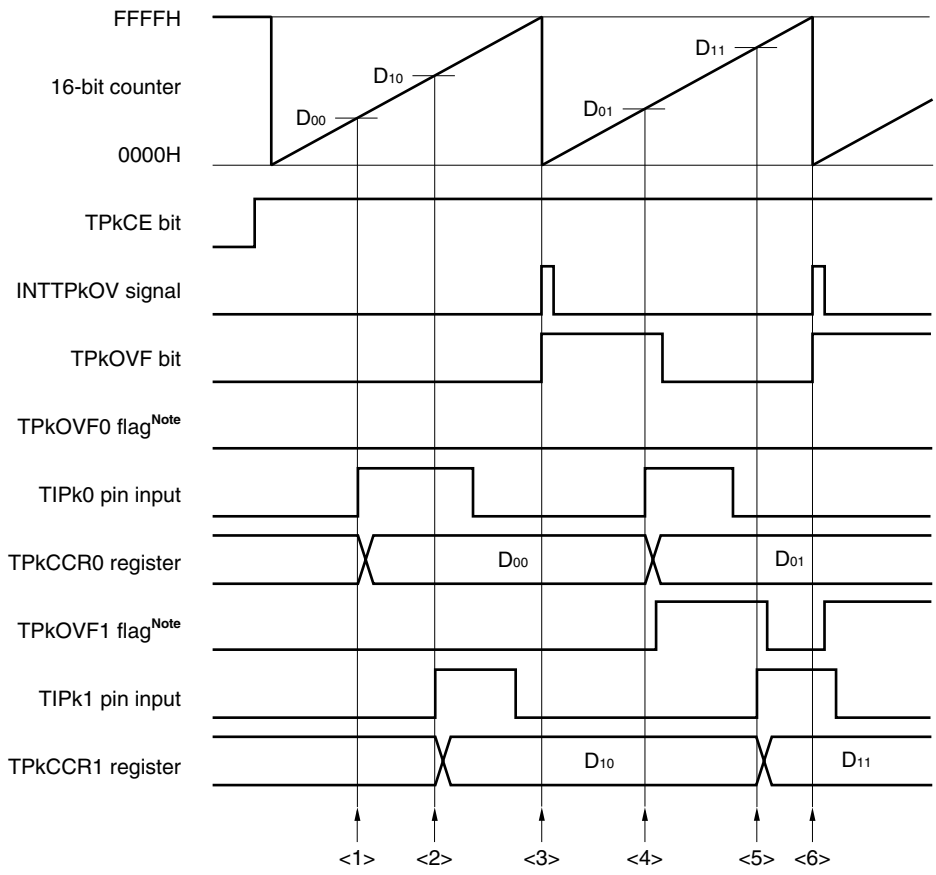
Note The TPkOVF0 and TPkOVF1 flags are set on the internal RAM by software.

<1> Read the TPkCCR0 register (setting of the default value of the TIk0 pin input).
<2> Read the TPkCCR1 register (setting of the default value of the TIPk1 pin input).
<3> An overflow occurs. Set the TPkOVF0 and TPkOVF1 flags to 1 in the overflow interrupt servicing, and clear the overflow flag to 0.
<4> Read the TPkCCR0 register.
Read the TPkOVF0 flag. If the TPkOVF0 flag is 1, clear it to 0.
Because the TPkOVF0 flag is 1, the pulse width can be calculated by (10000H + $D_{01} - D_{00}$).
<5> Read the TPkCCR1 register.
Read the TPkOVF1 flag. If the TPkOVF1 flag is 1, clear it to 0 (the TPkOVF0 flag is cleared in <4>, and the TPkOVF1 flag remains 1).
Because the TPkOVF1 flag is 1, the pulse width can be calculated by (10000H + $D_{11} - D_{10}$) (correct).
<6> Same as <3>

Remark k = 0, 2

(2/2)

Example when two capture registers are used (without using overflow interrupt)



Note The TPkOVF0 and TPkOVF1 flags are set on the internal RAM by software.

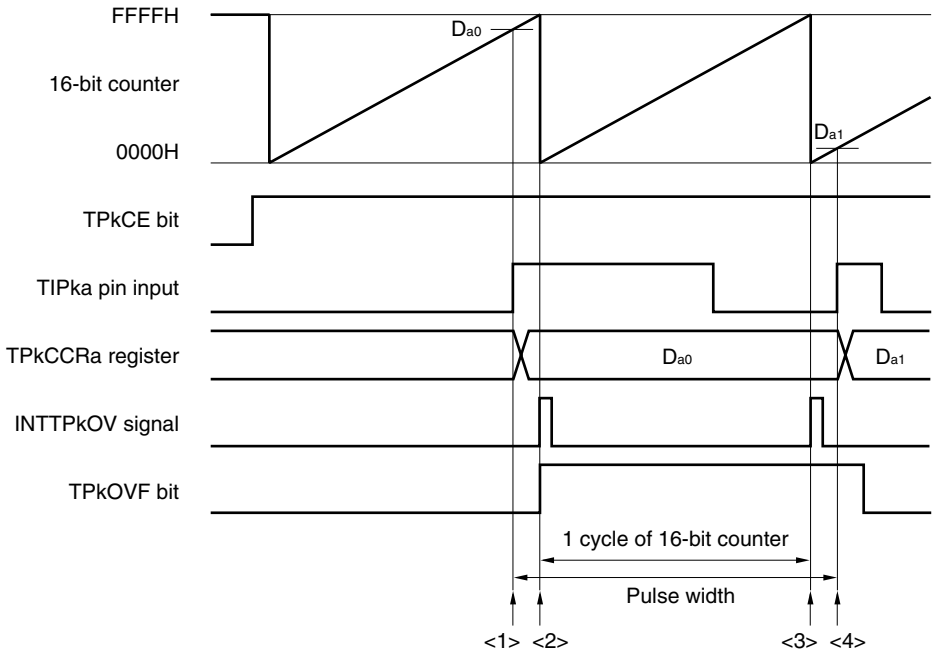
<1> Read the TPkCCR0 register (setting of the default value of the TIPk0 pin input).
<2> Read the TPkCCR1 register (setting of the default value of the TIPk1 pin input).
<3> An overflow occurs. Nothing is done by software.
<4> Read the TPkCCR0 register.
Read the overflow flag. If the overflow flag is 1, set only the TPkOVF1 flag to 1, and clear the overflow flag to 0.
Because the overflow flag is 1, the pulse width can be calculated by (10000H + $D_{01} - D_{00}$).
<5> Read the TPkCCR1 register.
Read the overflow flag. Because the overflow flag is cleared in <4>, 0 is read.
Read the TPkOVF1 flag. If the TPkOVF1 flag is 1, clear it to 0.
Because the TPkOVF1 flag is 1, the pulse width can be calculated by (10000H + $D_{11} - D_{10}$) (correct).
<6> Same as <3>

Remark k = 0, 2

(d) Processing of overflow if capture trigger interval is long

If the pulse width is greater than one cycle of the 16-bit counter, care must be exercised because an overflow may occur more than once from the first capture trigger to the next. First, an example of incorrect processing is shown below.

Example of incorrect processing when capture trigger interval is long



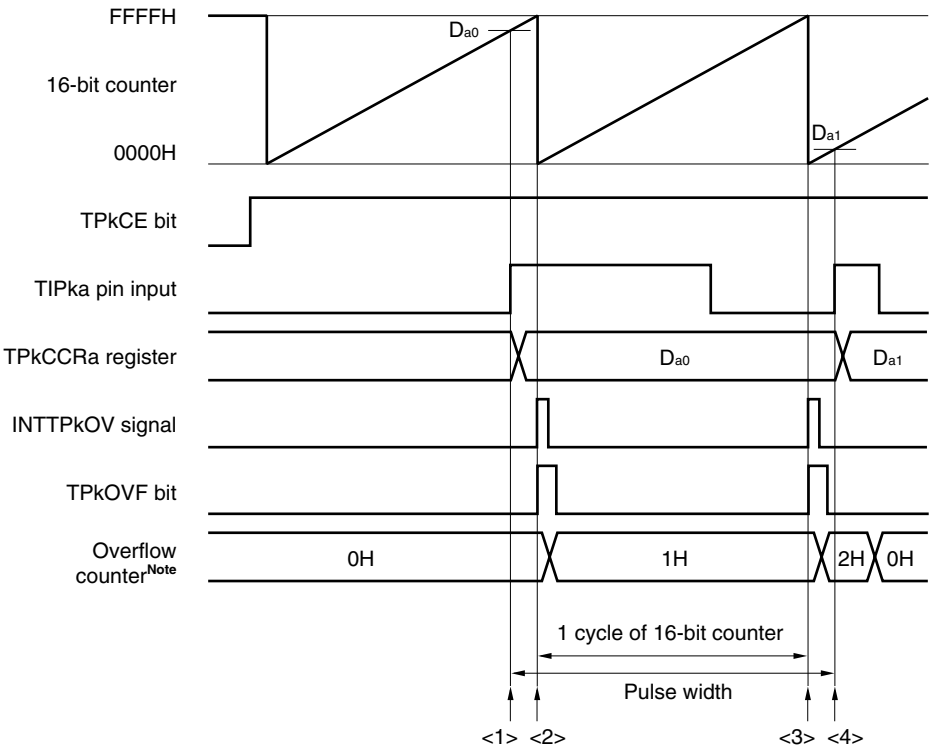
The following problem may occur when long pulse width is measured in the free-running timer mode.

<1> Read the TPkCCRa register (setting of the default value of the TIPka pin input).
<2> An overflow occurs. Nothing is done by software.
<3> An overflow occurs a second time. Nothing is done by software.
<4> Read the TPkCCRa register.
Read the overflow flag. If the overflow flag is 1, clear it to 0.
Because the overflow flag is 1, the pulse width can be calculated by (10000H + $D_{a1} - D_{a0}$) (incorrect).
Actually, the pulse width must be (20000H + $D_{a1} - D_{a0}$) because an overflow occurs twice.

Remark k = 0, 2
a = 0, 1

If an overflow occurs twice or more when the capture trigger interval is long, the correct pulse width may not be obtained.
If the capture trigger interval is long, slow the count clock to lengthen one cycle of the 16-bit counter, or use software. An example of how to use software is shown next.

Example when capture trigger interval is long



Note The overflow counter is set arbitrarily by software on the internal RAM.

<1> Read the TPkCCRa register (setting of the default value of the TIPka pin input).

<2> An overflow occurs. Increment the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<3> An overflow occurs a second time. Increment (+1) the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<4> Read the TPkCCRa register.

Read the overflow counter.

→ When the overflow counter is “N”, the pulse width can be calculated by (N × 10000H + D_{a1} – D_{a0}).

In this example, the pulse width is (20000H + D_{a1} – D_{a0}) because an overflow occurs twice.

Clear the overflow counter (0H).

Remark k = 0, 2

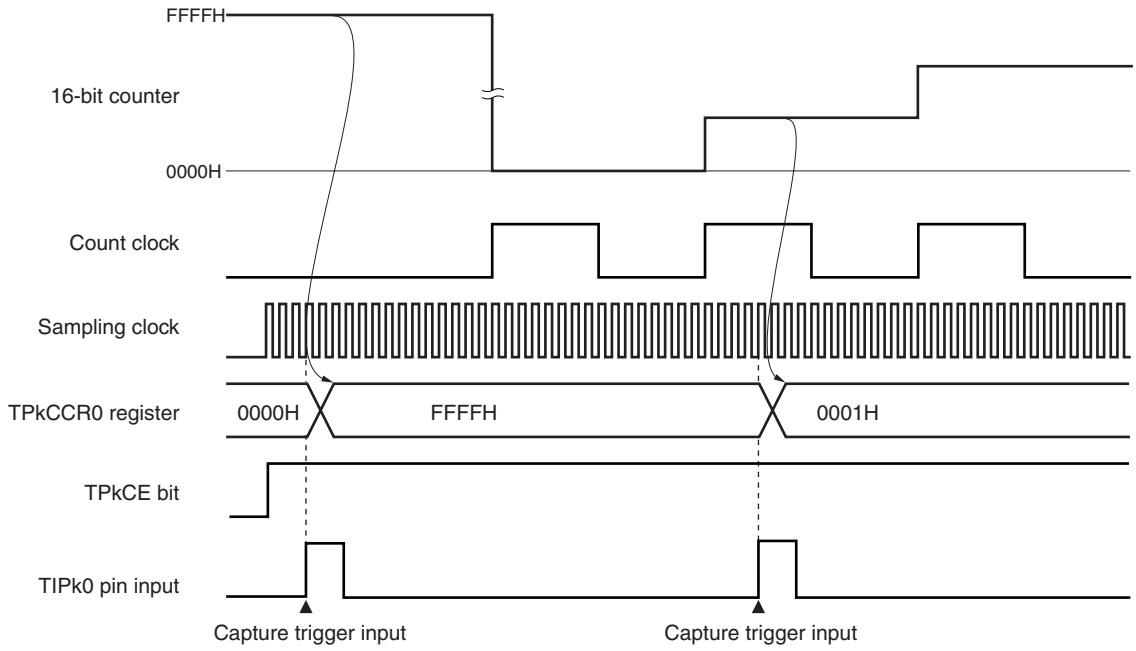
a = 0, 1

(e) Clearing overflow flag

The overflow flag can be cleared to 0 by clearing the TPnOVF bit to 0 with the CLR instruction after reading the TPnOVF bit when it is 1 and by writing 8-bit data (bit 0 is 0) to the TPnOPT0 register after reading the TPnOVF bit when it is 1.

(3) Note on capture operation

If the capture operation is used and if a slow clock is selected as the count clock, FFFFH, not 0000H, may be captured to the TPkCCRa register if the capture trigger is input immediately after the TPkCTL0.TPkCE bit is set to 1.



Remark k = 0, 2
a = 0, 1

6.6.7 Pulse width measurement mode (TPkMD2 to TPkMD0 bits = 110)

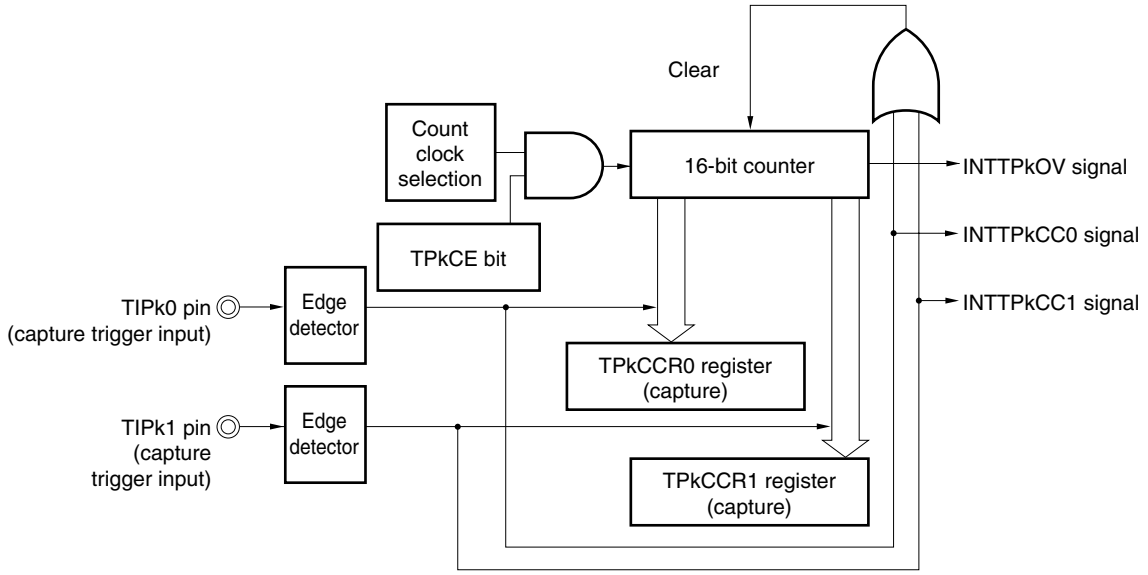
The mode is valid only in TMP0 and TMP2.

In the pulse width measurement mode, 16-bit timer/event counter P starts counting when the TPkCTL0.TPkCE bit is set to 1. Each time the valid edge input to the TIPka pin has been detected, the count value of the 16-bit counter is stored in the TPkCCRa register, and the 16-bit counter is cleared to 0000H.

The interval of the valid edge can be measured by reading the TPkCCRa register after a capture interrupt request signal (INTTPkCCa) occurs.

As shown in Figure 6-42, select either the TIPk0 or TIPk1 pin as the capture trigger input pin and set the unused pins to “No edge detection” by using the TPkIOC1 register.

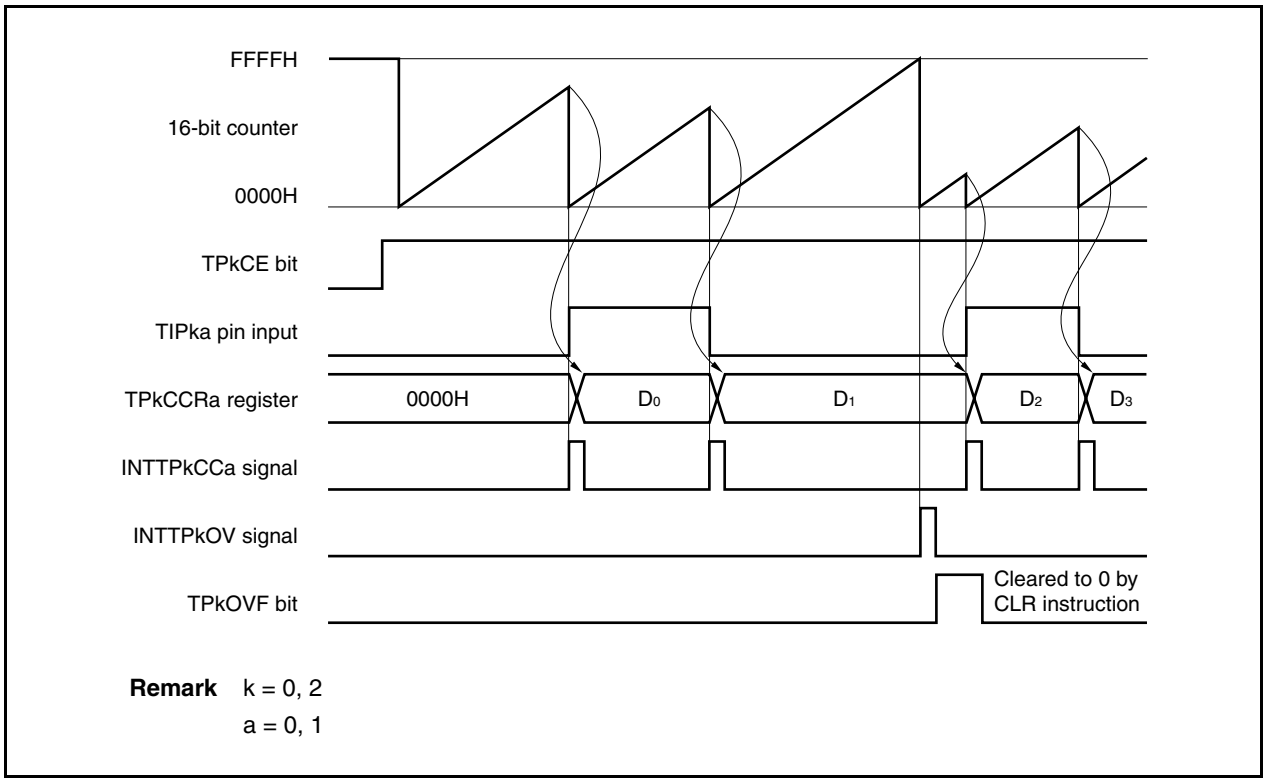
<R> **Figure 6-41. Configuration in Pulse Width Measurement Mode**



Caution In the pulse width measurement mode, select the internal clock as the count clock (by clearing the TPkCTL1.TPkEEE bit to 0).

Remark k = 0, 2
a = 0, 1

Figure 6-42. Basic Timing in Pulse Width Measurement Mode



Remark k = 0, 2
a = 0, 1

When the TPkCE bit is set to 1, the 16-bit counter starts counting. When the valid edge input to the TIPka pin is later detected, the count value of the 16-bit counter is stored in the TPkCCRa register, the 16-bit counter is cleared to 0000H, and a capture interrupt request signal (INTTPkCCa) is generated.

The pulse width is calculated as follows.

<R> Pulse width = Captured value × Count clock cycle

If the valid edge is not input to the TIPnm pin even when the 16-bit counter counted up to FFFFH, an overflow interrupt request signal (INTTPkOV) is generated at the next count clock, and the counter is cleared to 0000H and continues counting. At this time, the overflow flag (TPkOPT0.TPkOVF bit) is also set to 1. Clear the overflow flag to 0 by executing the CLR instruction via software.

If the overflow flag is set to 1, the pulse width can be calculated as follows.

<R> Pulse width = (10000H × Number of times for which TPkOVF bit is set to 1 + Captured value) × Count clock cycle

Remark k = 0, 2
a = 0, 1

Figure 6-43. Register Setting in Pulse Width Measurement Mode (1/2)

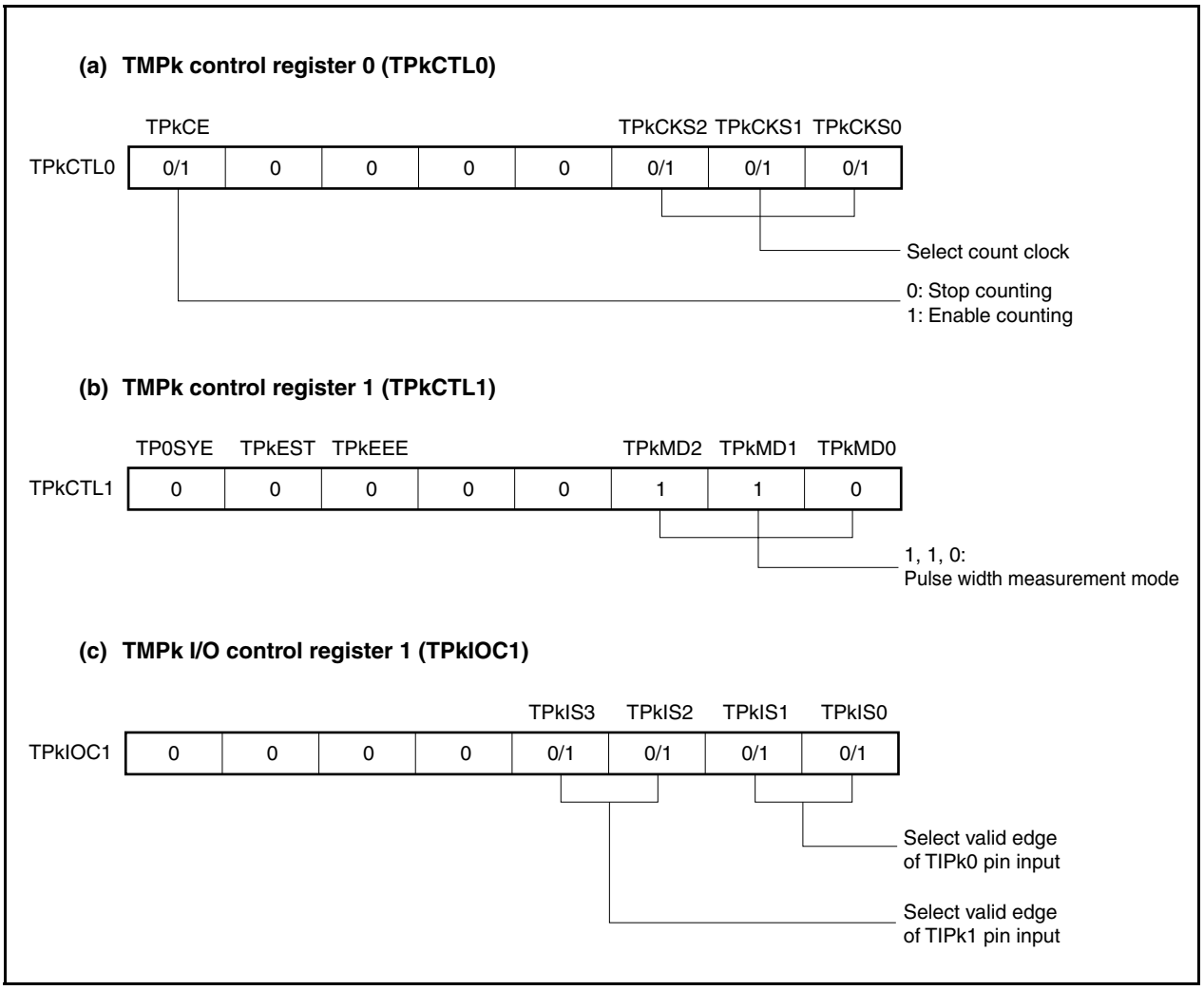


Figure 6-43. Register Setting in Pulse Width Measurement Mode (2/2)

(d) TMPk option register 0 (TPkOPT0)

			TPkCCS1	TPkCCS0				TPkOVF
TPkOPT0	0	0	0	0	0	0	0	0/1

Overflow flag

(e) TMPk counter read buffer register (TPkCNT)

The value of the 16-bit counter can be read by reading the TPkCNT register.

(f) TMPk capture/compare registers 0 and 1 (TPkCCR0 and TPkCCR1)

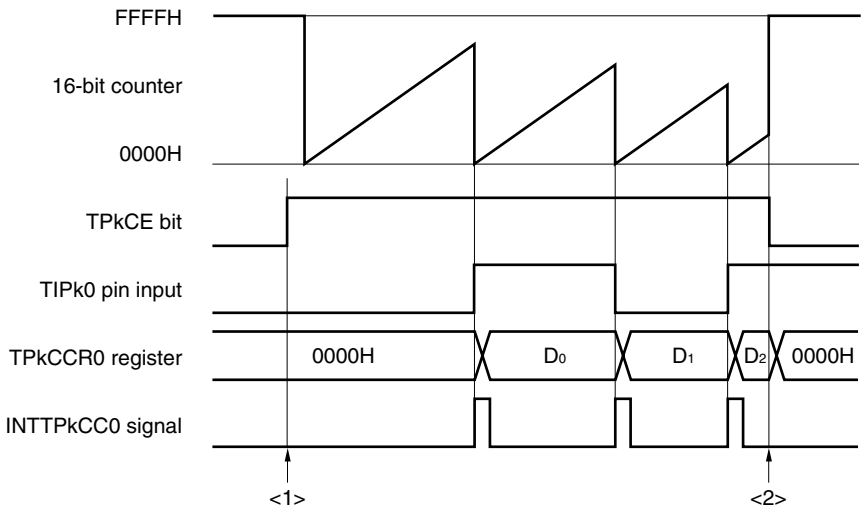
These registers store the count value of the 16-bit counter when the valid edge input to the TIPk0 and TIPk1 pins is detected.

<R> **Remarks 1.** TMPk I/O control register 0 (TPkIOC0) and TMPk I/O control register 2 (TPkIOC2) are not used in the pulse width measurement mode.

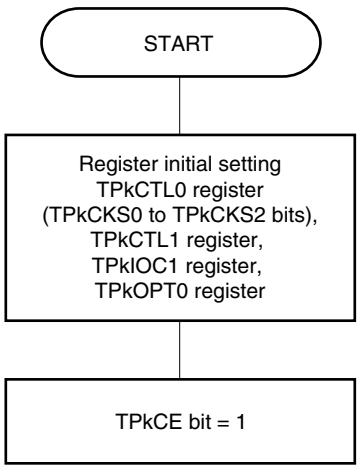
2. k = 0, 2

(1) Operation flow in pulse width measurement mode

Figure 6-44. Software Processing Flow in Pulse Width Measurement Mode



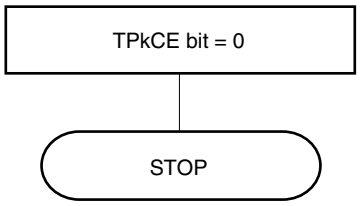
<R> <1> Count operation start flow



Initial setting of these registers is performed before setting the TPkCE bit to 1.

The TPkCKS0 to TPkCKS2 bits can be set at the same time when counting has been started (TPkCE bit = 1).

<2> Count operation stop flow



The counter is initialized and counting is stopped by clearing the TPkCE bit to 0.

Remark k = 0, 2

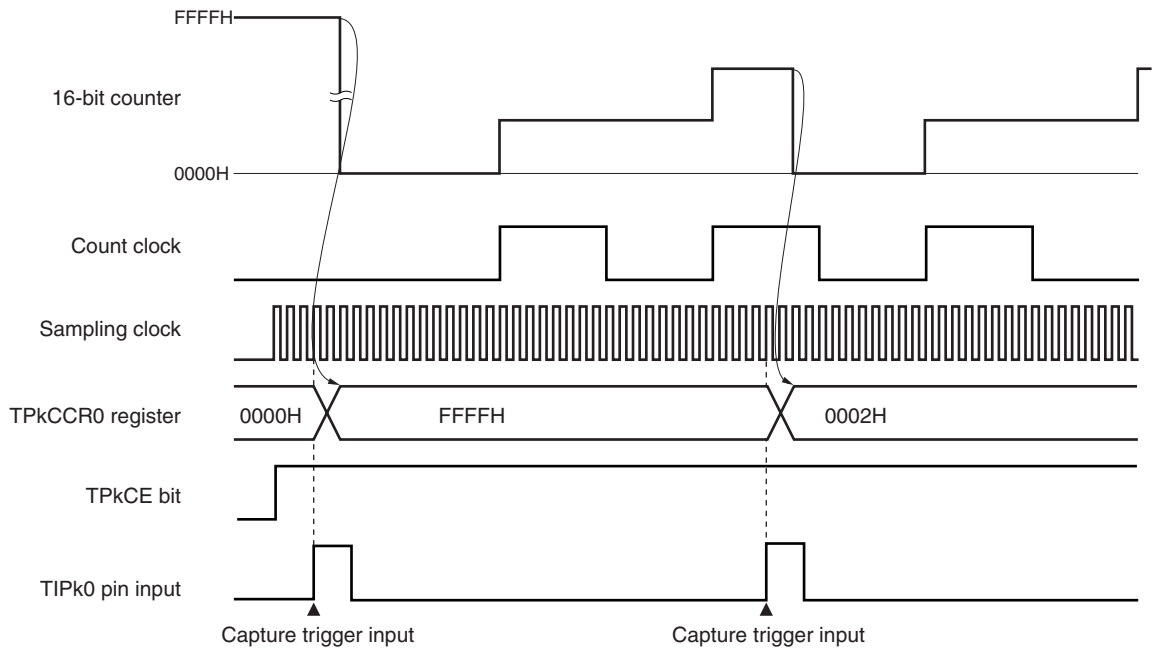
(2) Operation timing in pulse width measurement mode

(a) Clearing overflow flag

The overflow flag can be cleared to 0 by clearing the TPkOVF bit to 0 with the CLR instruction after reading the TPkOVF bit when it is 1 and by writing 8-bit data (bit 0 is 0) to the TPkOPT0 register after reading the TPkOVF bit when it is 1.

(3) Note

If a slow clock is selected as the count clock, FFFFH, not 0000H, may be captured to the TPkCCRa register if the capture trigger is input immediately after the TPkCTL0.TPkCE bit has been set to 1.



Remark k = 0, 2
a = 0, 1

CHAPTER 7 16-BIT TIMER/EVENT COUNTER Q (TMQ)

Timer Q (TMQ) is a 16-bit timer/event counter.
The V850E/IA3 and V850E/IA4 incorporate TMQ1 and TMQ0.

7.1 Overview

The TMQn channels are outlined below (n = 0, 1).

Table 7-1. TMQn Overview

Item	TMQ0	TMQ1
Clock selection	8 ways	8 ways
Capture trigger input pin	4	None
External event count input pin	1	None
External trigger input pin	1	None
Timer counter	1	1
Capture/compare register	4	4 Note 1
Capture/compare match interrupt request signal	4	4 Note 1
<R> Overflow interrupt request signal	1	1
Timer output pin Note 2	4	1

Notes 1. Compare function only

2. This is the number of output pins of TMQn; it does not include the output pins of TMQOPn. For details of the output pins of TMQOPn, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

7.2 Functions

The TMQn functions that can be realized differ from one channel to another, as shown in the table below (n = 0, 1).

Table 7-2. TMQn Functions

Function	TMQ0	TMQ1
6-phase PWM output[Note 1]	√	**Note 2**
Interval timer	√	√
External event counter	√	×
External trigger pulse output	√	×
One-shot pulse output	√	×
PWM output	√	×
Free-running timer	√	√
Pulse width measurement	√	×

Notes 1. This is connected to TMQOPn. For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

2. V850E/IA3: ×
V850E/IA4: √

7.3 Configuration

TMQn includes the following hardware.

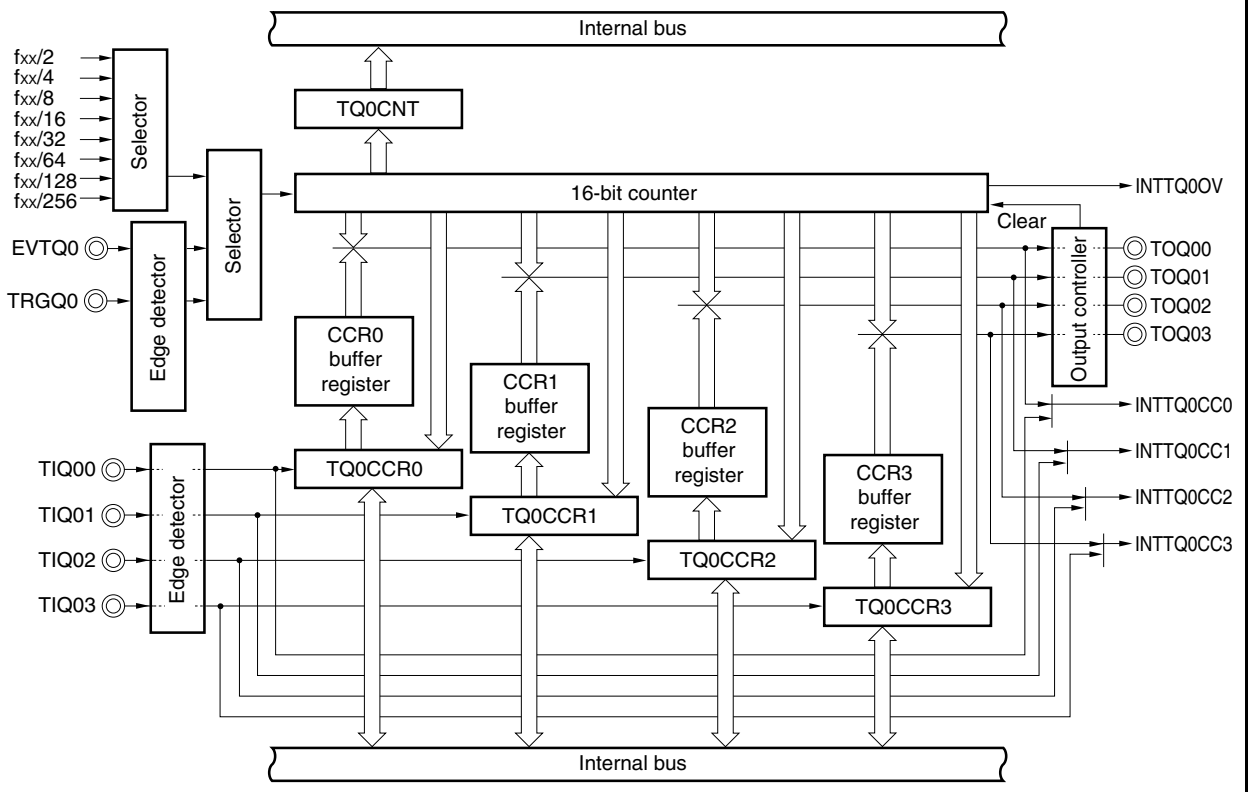
Table 7-3. TMQn Configuration

Item	Configuration
Timer register	16-bit counter × 1
Registers	TMQn counter read buffer register (TQnCNT): Total of 2 TMQn capture/compare registers 0 to 3 (TQnCCR0 to TQnCCR3): Total of 8 CCR0 to CCR3 buffer registers: Total of 8
Timer input	Total of 6 (TIQ00 to TIQ03, EVTQ0, TRGQ0 pins)
Timer output	Total of 5 (TOQ00 to TOQ03, TOQ10[Note] pins)
Control registers	TMQn control registers 0, 1 (TQnCTL0, TQnCTL1) TMQn I/O control register 0 (TQnIOC0)[Note] TMQ0 I/O control registers 1, 2 (TQ0IOC1, TQ0IOC2) TMQn option register 0 (TQnOPT0)

Note The TOQ10 pin and TQ1IOC0 register are provided in the V850E/IA4 only.

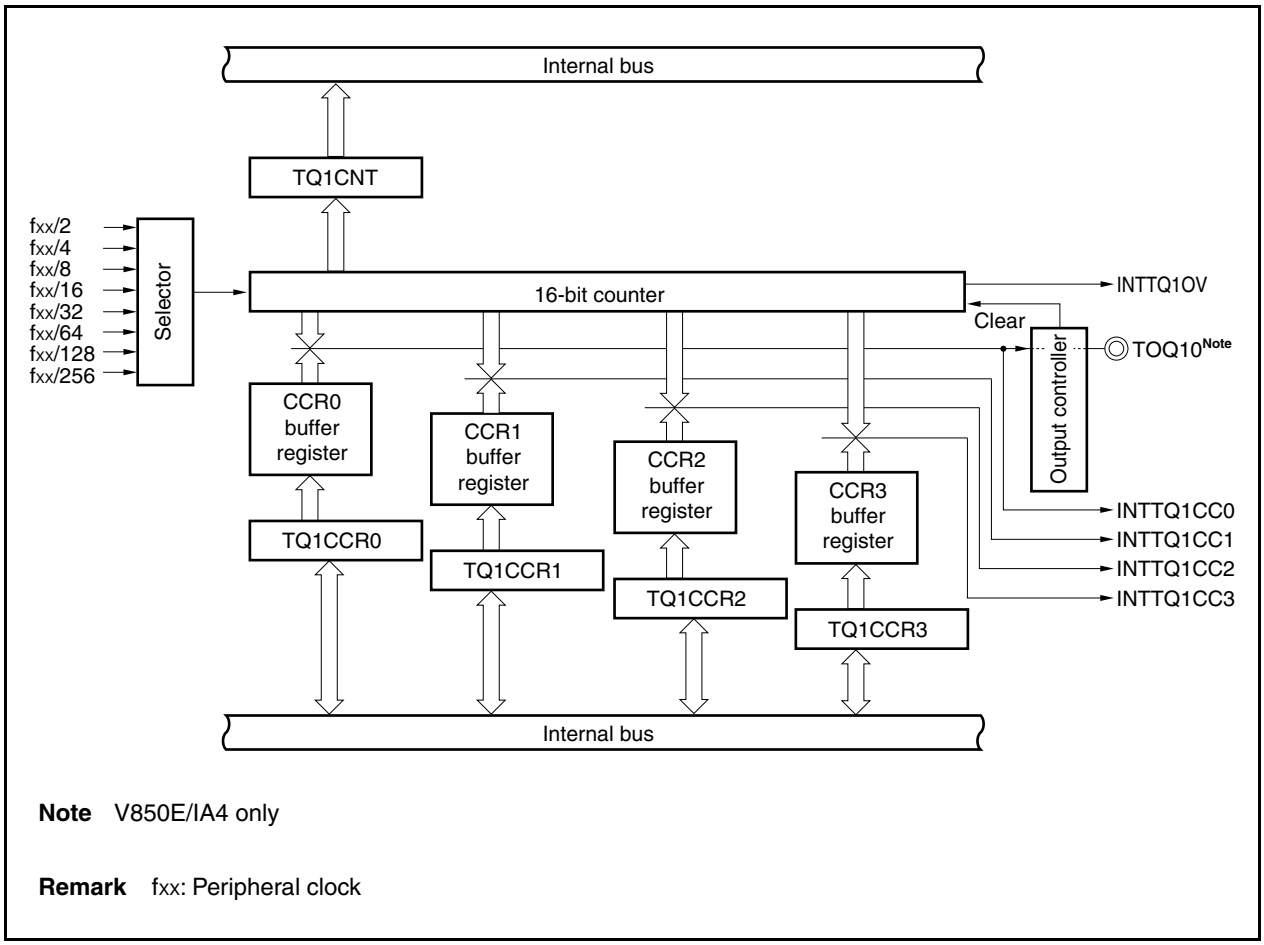
Remark n = 0, 1

Figure 7-1. TMQ0 Block Diagram



Remark fxx: Peripheral clock

Figure 7-2. TMQ1 Block Diagram



Note V850E/IA4 only

Remark fxx: Peripheral clock

(1) 16-bit counter

This 16-bit counter can count internal clocks or external events.

The count value of this counter can be read by using the TQnCNT register.

When the TQnCTL0.TQnCE bit = 0, the value of the 16-bit counter is FFFFH. If the TQnCNT register is read at this time, 0000H is read.

The TQnCE bit is cleared to 0 after reset.

(2) CCR0 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TQnCCR0 register is used as a compare register, the value written to the TQnCCR0 register is transferred to the CCR0 buffer register. When the count value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTQnCC0) is generated.

The CCR0 buffer register cannot be read or written directly.

The TQnCCR0 register is cleared to 0000H after reset, and the CCR0 buffer register is cleared to 0000H.

(3) CCR1 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TQnCCR1 register is used as a compare register, the value written to the TQnCCR1 register is transferred to the CCR1 buffer register. When the count value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTQnCC1) is generated.

The CCR1 buffer register cannot be read or written directly.

The TQnCCR1 register is cleared to 0000H after reset, and the CCR1 buffer register is cleared to 0000H.

(4) CCR2 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TQnCCR2 register is used as a compare register, the value written to the TQnCCR2 register is transferred to the CCR2 buffer register. When the count value of the 16-bit counter matches the value of the CCR2 buffer register, a compare match interrupt request signal (INTTQnCC2) is generated.

The CCR2 buffer register cannot be read or written directly.

The TQnCCR2 register is cleared to 0000H after reset, and the CCR2 buffer register is cleared to 0000H.

(5) CCR3 buffer register

This is a 16-bit compare register that compares the count value of the 16-bit counter.

When the TQnCCR3 register is used as a compare register, the value written to the TQnCCR3 register is transferred to the CCR3 buffer register. When the count value of the 16-bit counter matches the value of the CCR3 buffer register, a compare match interrupt request signal (INTTQnCC3) is generated.

The CCR3 buffer register cannot be read or written directly.

The TQnCCR3 register is cleared to 0000H after reset, and the CCR3 buffer register is cleared to 0000H.

(6) Edge detector

This circuit detects the valid edges input to the TIQ00 to TIQ03, EVTQ0, and TRGQ0 pins. No edge, rising edge, falling edge, or both the rising and falling edges can be selected as the valid edge by using the TQ0IOC1 and TQ0IOC2 registers.

(7) Output controller

This circuit controls the output of the TOQ00 to TOQ03 and TOQ10 (V850E/IA4 only) pins. The output of the TOQ00 to TOQ03 pins is controlled by the TQ0IOC0 register. The output of the TOQ10 (V850E/IA4 only) pin is controlled by the TQ1IOC0 register.

(8) Selector

This selector selects the count clock for the 16-bit counter. Eight types of internal clocks or an external event can be selected as the count clock.

7.4 Registers

(1) TMQn control register 0 (TQnCTL0)

The TQnCTL0 register is an 8-bit register that controls the TMQn operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

The same value can always be written to the TQnCTL0 register by software.

After reset: 00H R/W Address: TQ0CTL0 FFFFF5C0H, TQ1CTL0 FFFFF600H

	<7>	6	5	4	3	2	1	0
TQnCTL0 (n = 0, 1)	TQnCE	0	0	0	0	TQnCKS2	TQnCKS1	TQnCKS0

TQnCE	TMQn operation control
0	TMQn operation disabled (TMQn reset asynchronously[Note])
1	TMQn operation enabled. Start TMQn operation

TQnCKS2	TQnCKS1	TQnCKS0	Internal count clock selection
0	0	0	$f_{XX}/2$
0	0	1	$f_{XX}/4$
0	1	0	$f_{XX}/8$
0	1	1	$f_{XX}/16$
1	0	0	$f_{XX}/32$
1	0	1	$f_{XX}/64$
1	1	0	$f_{XX}/128$
1	1	1	$f_{XX}/256$

Note The TQnOPT0.TQnOVF bit and 16-bit counter are reset simultaneously. Moreover, timer outputs (TOQ00 to TOQ03 and TOQ10 (V850E/IA4 only) pins) are reset to the TQnIOC0 register set status at the same time as the 16-bit counter.

Cautions 1. Set the TQnCKS2 to TQnCKS0 bits when the TQnCE bit = 0.
When the value of the TQnCE bit is changed from 0 to 1, the TQnCKS2 to TQnCKS0 bits can be set simultaneously.
2. Be sure to clear bits 3 to 6 to "0".

Remark f_{XX}: Peripheral clock

(2) TMQn control register 1 (TQnCTL1)

The TQnCTL1 register is an 8-bit register that controls the TMQn operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: TQ0CTL1 FFFFF5C1H, TQ1CTL1 FFFFF601H

TQnCTL1 (n = 0, 1)

7	6	5	4	3	2	1	0
0	TQ0EST Note 1	TQ0EEE Note 1	0	0	TQnMD2	TQnMD1	TQnMD0

TQ0EST Note 1	Software trigger control
0	–
1	Generate a valid signal for external trigger input. • In one-shot pulse output mode: A one-shot pulse is output with writing 1 to the TQ0EST bit as the trigger. • In external trigger pulse output mode: A PWM waveform is output with writing 1 to the TQ0EST bit as the trigger.
Read value of the TQ0EST bit is always 0.	

TQ0EEE Note 1	Count clock selection
0	Disable operation with external event count input (EVTQ0 pin). (Perform counting with the count clock selected by the TQ0CTL0.TQ0CKS0 to TQ0CTL0.TQ0CKS2 bits.)
1	Enable operation with external event count input (EVTQ0 pin). (Perform counting at the valid edge of the external event count input signal.)
The TQ0EEE bit selects whether counting is performed with the internal count clock or the valid edge of the external event count input.	

TQnMD2	TQnMD1	TQnMD0	Timer mode selection
0	0	0	Interval timer mode
0	0	1	External event count mode Note 2
0	1	0	External trigger pulse output mode Note 2
0	1	1	One-shot pulse output mode Note 2
1	0	0	PWM output mode Note 2
1	0	1	Free-running timer mode
1	1	0	Pulse width measurement mode Note 2
1	1	1	6-phase PWM output mode Note 3

Notes 1. These bits can be set only in TMQ0. Be sure to clear bits 5 and 6 of TMQ1 to 0.

2. These modes can be set only in TMQ0. Do not set them in TMQ1.

3. This mode cannot be used when only TMQn is used. For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

Cautions 1. The TQ0EST bit is valid only in the external trigger pulse output mode or one-shot pulse output mode. In any other mode, writing 1 to this bit is ignored.

2. External event count input is selected in the external event count mode regardless of the value of the TQ0EEE bit.

3. Set the TQ0EEE and TQnMD2 to TQnMD0 bits when the TQnCTL0.TQnCE bit = 0. (The same value can be written when the TQnCE bit = 1.) The operation is not guaranteed when rewriting is performed with the TQnCE bit = 1. If rewriting was mistakenly performed, clear the TQnCE bit to 0 and then set the bits again.

4. Be sure to clear bits 3, 4, and 7 to "0".

(3) TMQn I/O control register 0 (TQnIOC0)

The TQnIOC0 register is an 8-bit register that controls the timer output (TOQn0, TOQ01 to TOQ03, and TOQnT1 to TOQnT3 pins).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

(1/2)

After reset: 00H R/W Address: TQ0IOC0 FFFFF5C2H, TQ1IOC0 FFFFF602H[Note 1]

	7	<6>	5	<4>	3	<2>	1	<0>
TQnIOC0	TQ00L3[Note 2]	TQ00E3[Note 2]	TQ00L2[Note 2]	TQ00E2[Note 2]	TQ00L1[Note 2]	TQ00E1[Note 2]	TQnOL0	TQnOE0

V850E/IA3
n = 0
b = 1 to 3

V850E/IA4
n = 0, 1
b = 1 to 3

TQnOLm	Output level setting of TOQnm and TOQnTb pins[Note 3] (TMQ0: m = 0 to 3, TMQ1: m = 0)
0	TOQnm and TOQnTb pins start output at high level.
1	TOQnm and TOQnTb pins start output at low level.

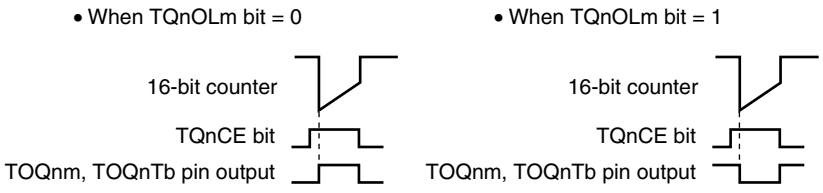
TQnOEm	Output setting of TOQnm and TOQnTb pins (TMQ0: m = 0 to 3, TMQ1: m = 0)
0	Timer output prohibited • Low level is output from the TOQnm and TOQnTb pins when the TQnOLm bit = 0. • High level is output from the TOQnm and TOQnTb pins when the TQnOLm bit = 1.
1	Timer output enabled (A pulse is output from the TOQnm and TOQnTb pins.)

Notes 1. V850E/IA4 only

2. Be sure to clear bits 2 to 7 of the TQ1IOC0 register to 0 when using TMQ1 as an interval timer or a free-running timer. In addition, set bits 2, 4, and 6 of the TQ1IOC0 register to 1, and bits 3, 5, and 7 to 0 or 1 when using the functions of TOQ1T1 to TOQ1T3 and TOQ1B1 to TOQ1B3 with TMQ1 as a 6-phase PWM output.

3. The output level of the timer output pins (TOQnm and TOQnTb) specified by the TQnOLm bit is shown below.

<R>



<R>

Cautions 1. If the setting of the TQnIOC0 register is changed when TOQnm and TOQnTb are set to the output mode, the output of the pins change. Set the port in the input mode and make the pin output go into a high-impedance state, noting changes in the pin status.

(2/2)

Cautions 2. Rewrite the TQnOLm and TQnOEm bits when the TQnCTL0.TQnCE bit = 0. (The same value can be written when the TQnCE bit = 1.) If rewriting was mistakenly performed, clear (0) the TQnCE bit and then set the bits again.

3. If the TQnOLm bit is manipulated when the TQnCE and TQnOEm bits are 0, the output level of the TOQnm and TOQnTb pins changes.

4. To generate the TOQnTb pin output and the A/D conversion start trigger signal of A/D converters 0 and 1 in the 6-phase PWM output mode, be sure to set the TOQnTb pin output mode using the TQnIOC0 register. At this time, be sure to clear the TQnOL0 bit to 0 and set the TQnOE0 bit to 1 (b = 1 to 3).

(4) TMQ0 I/O control register 1 (TQ0IOC1)

The TQ0IOC1 register is an 8-bit register that controls the valid edge for the capture trigger input signals (TIQ00 to TIQ03 pins).

This register can be read or written in 8-bit or 1-bit units.

Rest sets this register to 00H.

Remark TMQ1 does not have the TQ1IOC1 register.

After reset: 00H R/W Address: FFFFF5C3H

	7	6	5	4	3	2	1	0
TQ0IOC1	TQ0IS7	TQ0IS6	TQ0IS5	TQ0IS4	TQ0IS3	TQ0IS2	TQ0IS1	TQ0IS0

TQ0IS7	TQ0IS6	Capture trigger input signal (TIQ03 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TQ0IS5	TQ0IS4	Capture trigger input signal (TIQ02 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TQ0IS3	TQ0IS2	Capture trigger input signal (TIQ01 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TQ0IS1	TQ0IS0	Capture trigger input signal (TIQ00 pin) valid edge setting
0	0	No edge detection (capture operation invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

Cautions 1. Rewrite the TQ0IS7 to TQ0IS0 bits when the TQ0CTL0.TQ0CE bit = 0. (The same value can be written when the TQ0CE bit = 1.) If rewriting was mistakenly performed, clear (0) the TQ0CE bit and then set the bits again.

2. The TQ0IS7 to TQ0IS0 bits are valid only in the free-running timer mode (only when TQ0OPT0.TQ0CCS3 to TQ0OPT0.TQ0CCS0 bits = 1111) and pulse width measurement mode. In all other modes, a capture operation is not performed.

(5) TMQ0 I/O control register 2 (TQ0IOC2)

The TQ0IOC2 register is an 8-bit register that controls the valid edge for the external event count input signal (EVTQ0 pin) and external trigger input signal (TRGQ0 pin).

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Remark TMQ1 does not have the TQ1IOC2 register.

After reset: 00H R/W Address: FFFFF5C4H

	7	6	5	4	3	2	1	0
TQ0IOC2	0	0	0	0	TQ0EES1	TQ0EES0	TQ0ETS1	TQ0ETS0

TQ0EES1	TQ0EES0	External event count input signal (EVTQ0 pin) valid edge setting
0	0	No edge detection (external event count invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

TQ0ETS1	TQ0ETS0	External trigger input signal (TRGQ0 pin) valid edge setting
0	0	No edge detection (external trigger invalid)
0	1	Detection of rising edge
1	0	Detection of falling edge
1	1	Detection of both edges

Cautions 1. Rewrite the TQ0EES1, TQ0EES0, TQ0ETS1, and TQ0ETS0 bits when the TQ0CTL0.TQ0CE bit = 0. (The same value can be written when the TQ0CE bit = 1.) If rewriting was mistakenly performed, clear (0) the TQ0CE bit and then set the bits again.

2. The TQ0EES1 and TQ0EES0 bits are valid only when the TQ0CTL1.TQ0EEE bit = 1 or when the external event count mode is set (TQ0CTL1.TQ0MD2 to TQ0CTL1.TQ0MD0 bits = 001).

3. The TQ0ETS1 and TQ0ETS0 bits are valid only in the external trigger pulse output mode or one-shot pulse output mode.

(6) TMQn option register 0 (TQnOPT0)

The TQnOPT0 register is an 8-bit register used to set the capture/compare operation and detect overflow.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: TQ0OPT0 FFFFF5C5H, TQ1OPT0 FFFFF605H

TQnOPT0 (n = 0, 1)

<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
TQ0CCS3[Note 1]	TQ0CCS2[Note 1]	TQ0CCS1[Note 1]	TQ0CCS0[Note 1]	0	TQnCMS[Note 2]	TQnCUF[Note 2]	TQnOVF

TQ0CCSm	TQ0CCRm register capture/compare selection (m = 0 to 3)
0	Compare register selected
1	Capture register selected (Cleared by TQ0CTL0.TQ0CE bit = 0)
The TQ0CCSm bit setting is valid only in the free-running timer mode.	

TQnOVF	TMQn overflow flag
Set (1)	Overflow occurred
Reset (0)	0 written to TQnOVF bit or TQnCTL0.TQnCE bit = 0

- The TQnOVF bit is set (1) when the 16-bit counter value overflows from FFFFH to 0000H in the free-running timer mode or the pulse width measurement mode.
- An overflow interrupt request signal (INTTQnOV) is generated at the same time that the TQnOVF bit is set (1). The INTTQnOV signal is not generated in modes other than the free-running timer mode and the pulse width measurement mode.
- The TQnOVF bit is not cleared to 0 even when the TQnOVF bit or the TQnOPT0 register is read when the TQnOVF bit = 1.
- Before clearing the TQnOVF bit to 0 after generation of the INTTQnOV signal, be sure to confirm (by reading) that the TQnOVF bit is set to 1.
- The TQnOVF bit can be read or written, but the TQnOVF bit cannot be set (1) by software. Writing 1 has no effect on the TMQn operation.

Notes 1. Valid only in TMQ0. Be sure to clear bits 7 to 4 in TMQ1 to 0.

2. In the V850E/IA3, be sure to clear bits 2 and 1 of TMQ1 to 0. For details of the TQnCMS and TQnCUF bits, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

Cautions 1. Rewrite the TQ0CCS3 to TQ0CCS0 bits when the TQ0CE bit = 0. (The same value can be written when the TQ0CE bit = 1.) If rewriting was mistakenly performed, clear (0) the TQ0CE bit = 0 and then set the bits again.

2. Be sure to clear bit 3 to "0".

(7) TMQn capture/compare register 0 (TQnCCR0)

The TQ0CCR0 register is a 16-bit register that can be used as a capture register or a compare register depending on the mode. The TQ1CCR0 register is a 16-bit register that can only be used as a compare register.

The TQ0CCR0 register can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TQ0OPT0.TQ0CCS0 bit. In the pulse width measurement mode, the TQ0CCR0 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TQnCCR0 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TQ0CCR0 FFFFF5C6H, TQ1CCR0 FFFFF606H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQnCCR0 (n = 0, 1)																

(a) Function as compare register

The TQnCCR0 register can be rewritten even when the TQnCTL0.TQnCE bit = 1.

The set value of the TQnCCR0 register is transferred to the CCR0 buffer register. When the value of the 16-bit counter matches the value of the CCR0 buffer register, a compare match interrupt request signal (INTTQnCC0) is generated. If TOQn0 pin output is enabled at this time, the output of the TOQn0 pin is inverted.

When the TQnCCR0 register is used as a cycle register in the interval timer mode, external event count mode[Note], external trigger pulse output mode[Note], one-shot pulse output mode[Note], or PWM output mode[Note], the value of the 16-bit counter is cleared (0000H) if its count value matches the value of the CCR0 buffer register.

The compare register is not cleared by the TQnCTL0.TQnCE bit = 0.

Note These modes can be set only in TMQ0. They cannot be set in TMQ1.

(b) Function as capture register (TQ0CCR0 register only)

When the TQ0CCR0 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TQ0CCR0 register if the valid edge of the capture trigger input pin (TIQ00 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TQ0CCR0 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIQ00 pin) is detected.

Even if the capture operation and reading the TQ0CCR0 register conflict, the correct value of the TQ0CCR0 register can be read.

The capture register is cleared by the TQ0CTL0.TQ0CE bit = 0.

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 7-4. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 1]	Compare register	Batch write[Note 2]
One-shot pulse output[Note 1]	Compare register	Anytime write
PWM output[Note 1]	Compare register	Batch write[Note 2]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMQ0 only

2. Writing to the TQ0CCR1 register is the trigger.

Remark For anytime write and batch write, see **7.6 (2) Anytime write and batch write**.

(8) TMQn capture/compare register 1 (TQnCCR1)

The TQ0CCR1 register is a 16-bit register that can be used as a capture register or a compare register depending on the mode. The TQ1CCR1 register is a 16-bit register that can only be used as a compare register.

The TQ0CCR1 register can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TQ0OPT0.TQ0CCS1 bit. In the pulse width measurement mode, the TQ0CCR1 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TQnCCR1 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TQ0CCR1 FFFFF5C8H, TQ1CCR1 FFFFF608H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQnCCR1 (n = 0, 1)																

(a) Function as compare register

The TQnCCR1 register can be rewritten even when the TQnCTL0.TQnCE bit = 1.

The set value of the TQnCCR1 register is transferred to the CCR1 buffer register. When the value of the 16-bit counter matches the value of the CCR1 buffer register, a compare match interrupt request signal (INTTQnCC1) is generated. If TOQ01 pin output is enabled at this time, the output of the TOQ01 pin is inverted (the TOQ11 pin is not provided).

The compare register is not cleared by the TQnCTL0.TQnCE bit = 0.

(b) Function as capture register (TQ0CCR1 register only)

When the TQ0CCR1 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TQ0CCR1 register if the valid edge of the capture trigger input pin (TIQ01 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TQ0CCR1 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIQ01 pin) is detected.

Even if the capture operation and reading the TQ0CCR1 register conflict, the correct value of the TQ0CCR1 register can be read.

The capture register is cleared by the TQ0CTL0.TQ0CE bit = 0.

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 7-5. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 1]	Compare register	Batch write[Note 2]
One-shot pulse output[Note 1]	Compare register	Anytime write
PWM output[Note 1]	Compare register	Batch write[Note 2]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMQ0 only

2. Writing to the TQ0CCR1 register is the trigger.

Remark For anytime write and batch write, see **7.6 (2) Anytime write and batch write**.

(9) TMQn capture/compare register 2 (TQnCCR2)

The TQ0CCR2 register is a 16-bit register that can be used as a capture register or a compare register depending on the mode. The TQ1CCR2 register is a 16-bit register that can only be used as a compare register.

The TQ0CCR2 register can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TQ0OPT0.TQ0CCS2 bit. In the pulse width measurement mode, the TQ0CCR2 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TQnCCR2 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TQ0CCR2 FFFFF5CAH, TQ1CCR2 FFFFF60AH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQnCCR2 (n = 0, 1)																

(a) Function as compare register

The TQnCCR2 register can be rewritten even when the TQnCTL0.TQnCE bit = 1.

The set value of the TQnCCR2 register is transferred to the CCR2 buffer register. When the value of the 16-bit counter matches the value of the CCR2 buffer register, a compare match interrupt request signal (INTTQnCC2) is generated. If TOQ02 pin output is enabled at this time, the output of the TOQ02 pin is inverted (the TOQ12 pin is not provided).

The compare register is not cleared by the TQnCTL0.TQnCE bit = 0.

(b) Function as capture register (TQ0CCR2 register only)

When the TQ0CCR2 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TQ0CCR2 register if the valid edge of the capture trigger input pin (TIQ02 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TQ0CCR2 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIQ02 pin) is detected.

Even if the capture operation and reading the TQ0CCR2 register conflict, the correct value of the TQ0CCR2 register can be read.

The capture register is cleared by the TQ0CTL0.TQ0CE bit = 0.

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 7-6. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 1]	Compare register	Batch write[Note 2]
One-shot pulse output[Note 1]	Compare register	Anytime write
PWM output[Note 1]	Compare register	Batch write[Note 2]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMQ0 only

2. Writing to the TQ0CCR1 register is the trigger.

Remark For anytime write and batch write, see **7.6 (2) Anytime write and batch write**.

(10) TMQn capture/compare register 3 (TQnCCR3)

The TQ0CCR3 register is a 16-bit register that can be used as a capture register or a compare register depending on the mode. The TQ1CCR3 register is a 16-bit register that can only be used as a compare register.

The TQ0CCR3 register can be used as a capture register or a compare register only in the free-running timer mode, depending on the setting of the TQ0OPT0.TQ0CCS3 bit. In the pulse width measurement mode, the TQ0CCR3 register can be used only as a capture register. In any other mode, this register can be used only as a compare register.

The TQnCCR3 register can be read or written during operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: TQ0CCR3 FFFFF5CCH, TQ1CCR3 FFFFF60CH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQnCCR3 (n = 0, 1)																

(a) Function as compare register

The TQnCCR3 register can be rewritten even when the TQnCTL0.TQnCE bit = 1.

The set value of the TQnCCR3 register is transferred to the CCR3 buffer register. When the value of the 16-bit counter matches the value of the CCR3 buffer register, a compare match interrupt request signal (INTTQnCC3) is generated. If TOQ03 pin output is enabled at this time, the output of the TOQ03 pin is inverted (the TOQ13 pin is not provided).

The compare register is not cleared by the TQnCTL0.TQnCE bit = 0.

(b) Function as capture register (TQ0CCR3 register only)

When the TQ0CCR3 register is used as a capture register in the free-running timer mode, the count value of the 16-bit counter is stored in the TQ0CCR3 register if the valid edge of the capture trigger input pin (TIQ03 pin) is detected. In the pulse width measurement mode, the count value of the 16-bit counter is stored in the TQ0CCR3 register and the 16-bit counter is cleared (0000H) if the valid edge of the capture trigger input pin (TIQ03 pin) is detected.

Even if the capture operation and reading the TQ0CCR3 register conflict, the correct value of the TQ0CCR3 register can be read.

The capture register is cleared by the TQ0CTL0.TQ0CE bit = 0.

The following table shows the functions of the capture/compare register in each mode, and how to write data to the compare register.

Table 7-7. Function of Capture/Compare Register in Each Mode and How to Write Compare Register

Operation Mode	Capture/Compare Register	How to Write Compare Register
Interval timer	Compare register	Anytime write
External event counter[Note 1]	Compare register	Anytime write
External trigger pulse output[Note 1]	Compare register	Batch write[Note 2]
One-shot pulse output[Note 1]	Compare register	Anytime write
PWM output[Note 1]	Compare register	Batch write[Note 2]
Free-running timer	Capture/compare register	Anytime write
Pulse width measurement[Note 1]	Capture register	None

Notes 1. TMQ0 only

2. Writing to the TQ0CCR1 register is the trigger.

Remark For anytime write and batch write, see **7.6 (2) Anytime write and batch write**.

(11) TMQn counter read buffer register (TQnCNT)

The TQnCNT register is a read buffer register that can read the count value of the 16-bit counter.

If this register is read when the TQnCTL0.TQnCE bit = 1, the count value of the 16-bit counter can be read.

This register is read-only, in 16-bit units.

The value of the TQnCNT register is cleared to 0000H when the TQnCE bit = 0. If the TQnCNT register is read at this time, the value of the 16-bit counter (FFFFH) is not read, but 0000H is read.

The value of the TQnCE bit is cleared to 0 after reset, and the TQnCNT register is cleared to 0000H.

After reset: 0000H R Address: TQ0CNT FFFFF5CEH, TQ1CNT FFFFF60EH

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TQnCNT (n = 0, 1)																

7.5 Timer Output Operations

The following table shows the operations and output levels of the TOQ00 to TOQ03 and TOQ10 pins (the TOQ10 pin is provided only in the V850E/IA4).

Table 7-8. Timer Output Control in Each Mode

Operation Mode	TOQn0 Pin[Note 1]	TOQ01 Pin	TOQ02 Pin	TOQ03 Pin
Interval timer mode	PWM output			
External event count mode	None			
External trigger pulse output mode	PWM output[Note 2]	External trigger pulse output	External trigger pulse output	External trigger pulse output
One-shot pulse output mode		One-shot pulse output	One-shot pulse output	One-shot pulse output
PWM output mode		PWM output	PWM output	PWM output
Free-running timer mode	PWM output (only when compare function is used)			
Pulse width measurement mode	None			

Notes 1. The TOQ10 pin is provided only in the V850E/IA4.

2. TOQ00 pin only

Remark n = 0, 1

Table 7-9. Truth Table of TOQ00 to TOQ03 and TOQ10[Note] Pins Under Control of Timer Output Control Bits

TQnIOC0.TQnOLa Bit	TQnIOC0.TQnOEa Bit	TQnCTL0.TQnCE Bit	Level of TOQna[Note] Pins
0	0	×	Low-level output
	1	0	Low-level output
		1	Low level immediately before counting, high level after counting is started
1	0	×	High-level output
	1	0	High-level output
		1	High level immediately before counting, low level after counting is started

Note The TOQ10 pin is provided only in the V850E/IA4.

Remark a = 0 to 3 when n = 0
a = 0 when n = 1

7.6 Operation

The functions that can be realized differ between TMQ0 and TMQ1. The functions of each channel are shown below.

Table 7-10. TMQ0 Specifications in Each Mode

Operation	TQ0CTL1.TQ0EST Bit (Software Trigger Bit)	TRGQ0 Pin (External Trigger Input)	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode	Invalid	Invalid	Compare only	Anytime write
External trigger pulse output mode	Valid	Valid	Compare only	Batch write
One-shot pulse output mode	Valid	Valid	Compare only	Anytime write
PWM output mode	Invalid	Invalid	Compare only	Batch write
Free-running timer mode	Invalid	Invalid	Switchable	Anytime write
Pulse width measurement mode	Invalid	Invalid	Capture only	Not applicable

Remark TMQ0 has a function to execute tuning with TMP0. For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

Table 7-11. TMQ1 Specifications in Each Mode

Operation	Software Trigger Bit	External Trigger Input	Capture/Compare Register Setting	Compare Register Write Method
Interval timer mode	Invalid	Invalid	Compare only	Anytime write
External event count mode	None			
External trigger pulse output mode	None			
One-shot pulse output mode	None			
PWM output mode	None			
Free-running timer mode	Invalid	Invalid	Compare only	Anytime write
Pulse width measurement mode	None			

Remark TMQ1 has a function to execute tuning with TMP1 (V850E/IA4 only). For details, see **CHAPTER 10 MOTOR CONTROL FUNCTION**.

(1) Counter basic operation

This section explains the basic operation of the 16-bit counter. For details, refer to the description of the operation in each mode.

Remark n = 0, 1
a = 0 to 3

<R>

(a) Counter start operation

- In external event count mode
  When the TQ0CTL0.TQ0CE bit is set from 0 to 1, the 16-bit counter is set to 0000H.
  After that, it counts up from 0001H to 0002H, 0003H, and so on, each time the valid edge of an external event count input (EVTQ0) is detected.
- In modes other than the above
  The 16-bit counter of TMQn starts counting from the default value FFFFH in all modes.
  It counts up from FFFFH to 0000H, 0001H, 0002H, 0003H, and so on.

(b) Clear operation

The 16-bit counter is cleared to 0000H when its value matches the value of the compare register and when its value is captured. The counting operation from FFFFH to 0000H that takes place immediately after the counter has started counting or when the counter overflows is not a clearing operation. Therefore, the INTTQnCCa interrupt signal is not generated.

(c) Overflow operation

The 16-bit counter overflows when the counter counts up from FFFFH to 0000H in the free-running timer mode or pulse width measurement mode. If the counter overflows, the TQnOPT0.TQnOVF bit is set to 1 and an interrupt request signal (INTTQnOV) is generated. Note that the INTTQnOV signal is not generated under the following conditions.

- Immediately after a counting operation has been started
- If the counter value matches the compare value FFFFH and is cleared
- When FFFFH is captured in the pulse width measurement mode and the counter counts up from FFFFH to 0000H

Caution After the overflow interrupt request signal (INTTQnOV) has been generated, be sure to check that the overflow flag (TQnOVF bit) is set to 1.

(d) Counter read operation during counting operation

The value of the 16-bit counter of TMQn can be read by using the TQnCNT register during the count operation. When the TQnCTL0.TQnCE bit = 1, the value of the 16-bit counter can be read by reading the TQ0CNT register. When the TQnCE bit = 0, the 16-bit counter is FFFFH and the TQnCNT register is 0000H.

(e) Interrupt operation

TMQn generates the following five interrupt request signals.

- INTTQnCC0 interrupt: This signal functions as a match interrupt request signal of the CCR0 buffer register and as a capture interrupt request signal to the TQnCCR0 register.
- INTTQnCC1 interrupt: This signal functions as a match interrupt request signal of the CCR1 buffer register and as a capture interrupt request signal to the TQnCCR1 register.
- INTTQnCC2 interrupt: This signal functions as a match interrupt request signal of the CCR2 buffer register and as a capture interrupt request signal to the TQnCCR2 register.
- INTTQnCC3 interrupt: This signal functions as a match interrupt request signal of the CCR3 buffer register and as a capture interrupt request signal to the TQnCCR3 register.
- INTTQnOV interrupt: This signal functions as an overflow interrupt request signal.

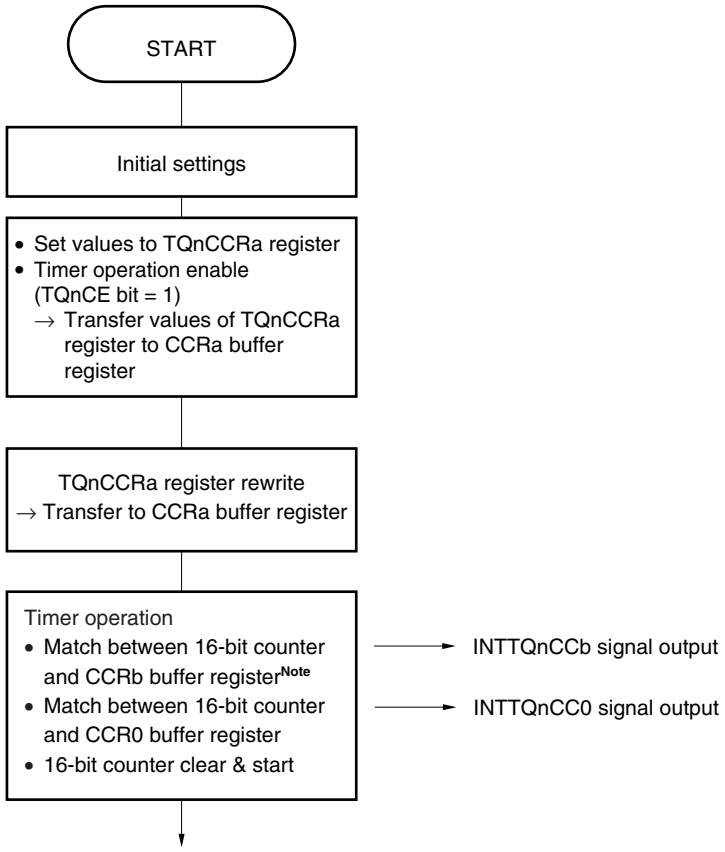
(2) Anytime write and batch write

The TQnCCR0 to TQnCCR3 registers can be rewritten in the TMQn during timer operation (TQnCTL0.TQnCE bit = 1), but the write method (anytime write, batch write) of the CCR0 to CCR3 buffer registers differs depending on the mode.

(a) Anytime write

In this mode, data is transferred at any time from the TQnCCR0 to TQnCCR3 registers to the CCR0 to CCR3 buffer registers during the timer operation.

Figure 7-3. Flowchart of Basic Operation for Anytime Write

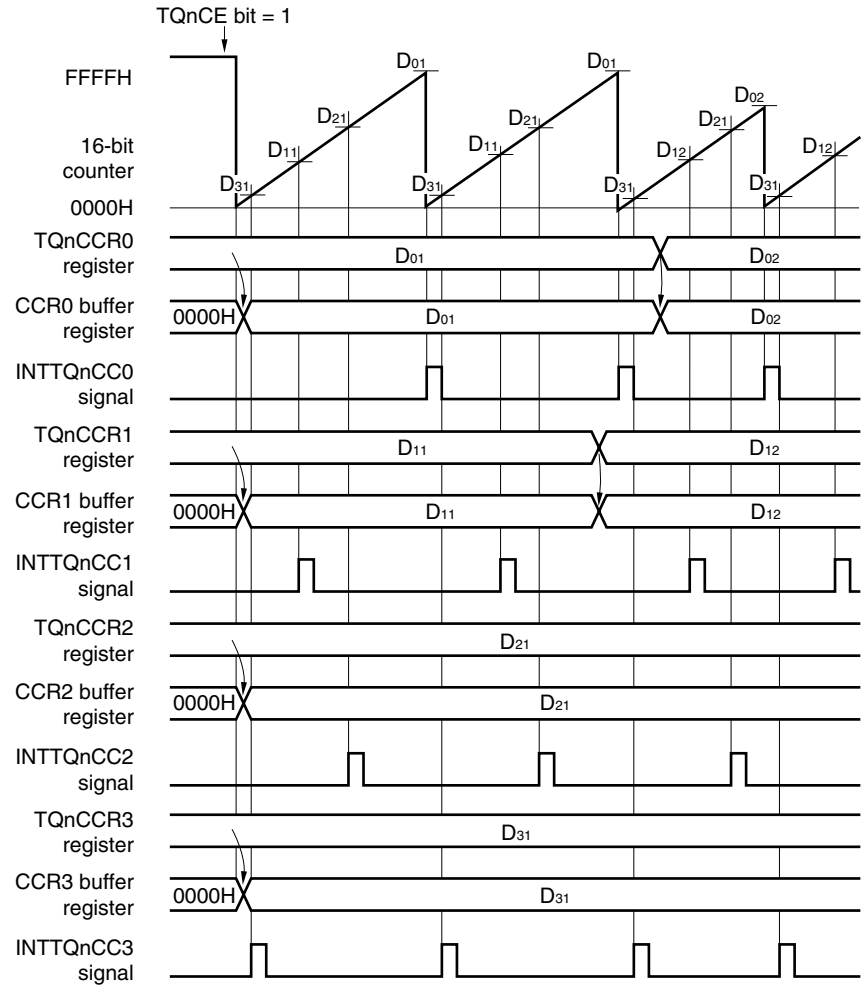


Note The 16-bit counter is not cleared upon a match between the value of the 16-bit counter and the value of the CCRb buffer register. It is cleared upon a match between the value of the 16-bit counter and the value of the CCR0 buffer register.

Remarks 1. The above flowchart illustrates an example of the operation in the interval timer mode.

2. n = 0, 1
a = 0 to 3
b = 1 to 3

Figure 7-4. Timing of Anytime Write



Remarks 1. D_{01}, D_{02}: Setting values of TQnCCR0 register

D_{11}, D_{12}: Setting values of TQnCCR1 register

D_{21}: Setting value of TQnCCR2 register

D_{31}: Setting value of TQnCCR3 register

2. The above timing chart illustrates an example of the operation in the interval timer mode.

3. n = 0, 1

(b) Batch write

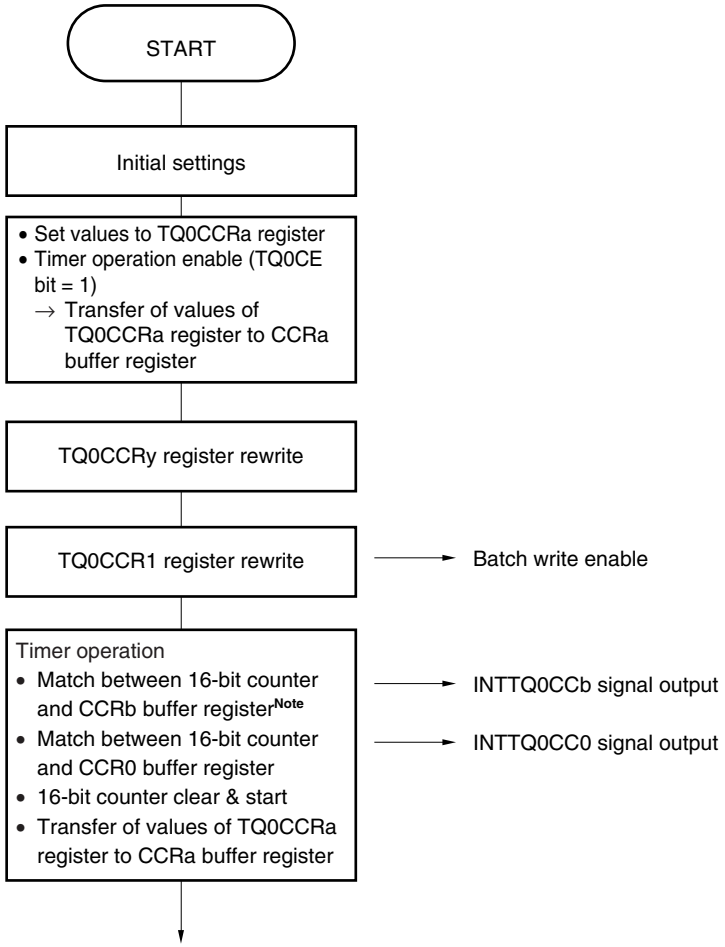
In this mode, data is transferred all at once from the TQ0CCR0 to TQ0CCR3 registers to the CCR0 to CCR3 buffer registers during timer operation. This data is transferred upon a match between the value of the CCR0 buffer register and the value of the 16-bit counter. Transfer is enabled by writing to the TQ0CCR1 register.

Whether to enable or disable the next transfer timing is controlled by writing or not writing to the TQ0CCR1 register.

In order for the setting value when the TQ0CCR0 to TQ0CCR3 registers are rewritten to become the 16-bit counter comparison value (in other words, in order for this value to be transferred to the CCR0 to CCR3 buffer registers), it is necessary to rewrite TQ0CCR0 and finally write to the TQ0CCR1 register before the 16-bit counter value and the CCR0 buffer register value match. The values of the TQ0CCR0 to TQ0CCR3 registers are transferred to the CCR0 to CCR3 buffer registers upon a match between the count value of the 16-bit counter and the value of the CCR0 buffer register. Thus, even when wishing only to rewrite the value of the TQ0CCR0, TQ0CCR2, or TQ0CCR3 register, also write the same value (same as preset value of the TQ0CCR1 register) to the TQ0CCR1 register.

Remark TMQ1 cannot be set in a mode in which it can be rewritten by batch write.

Figure 7-5. Flowchart of Basic Operation for Batch Write



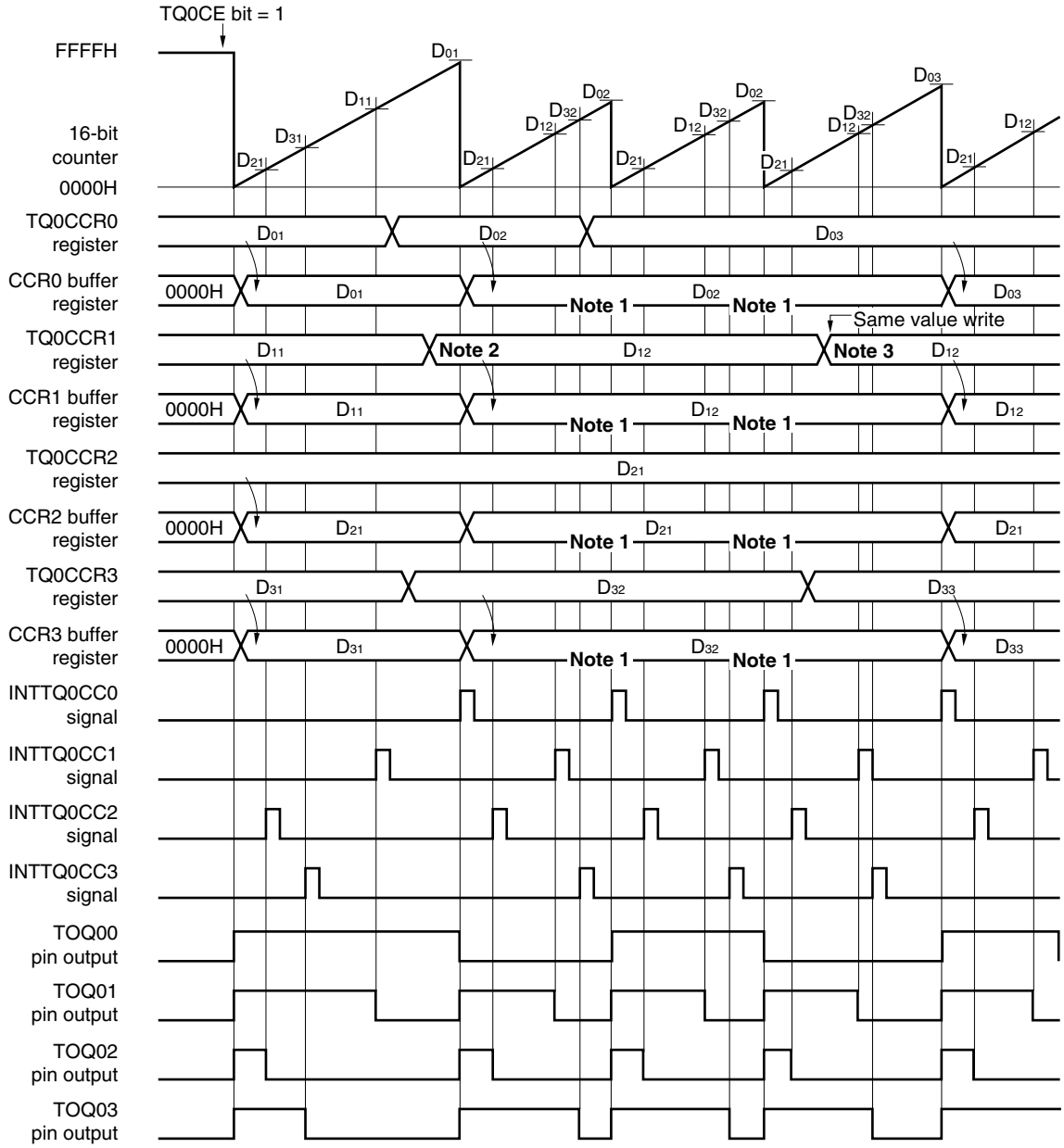
Note The 16-bit counter is not cleared upon a match between the value of the 16-bit counter and the value of the CCRb buffer register. It is cleared upon a match between the value of the 16-bit counter and the value of the CCR0 buffer register.

Caution Writing to the TQ0CCR1 register includes enabling of batch write. Thus, rewrite the TQ0CCR1 register after rewriting the TQ0CCR0, TQ0CCR2, and TQ0CCR3 registers.

Remarks 1. The above flowchart illustrates an example of the operation in the PWM output mode.

2. a = 0 to 3
b = 1 to 3
y = 0, 2, 3

Figure 7-6. Timing of Batch Write



Notes 1. Because the TQ0CCR1 register was not rewritten, D_{02} is not transferred.

2. Because TQ0CCR1 register has been written (D_{12}), data is transferred to the CCR1 buffer register upon a match between the value of the 16-bit timer and the value of the TQ0CCR0 register (D_{01}).

3. Because TQ0CCR1 register has been written (D_{12}), data is transferred to the CCR1 buffer register upon a match between the value of the 16-bit timer and the value of the TQ0CCR0 register (D_{12}).

Remarks 1. D_{01}, D_{02}, D_{03}: Setting values of TQ0CCR0 register
D_{11}, D_{12}: Setting values of TQ0CCR1 register
D_{21}: Setting value of TQ0CCR2 register
D_{31}, D_{32}, D_{33}: Setting values of TQ0CCR3 register

2. The above timing chart illustrates an example of the operation in the PWM output mode.

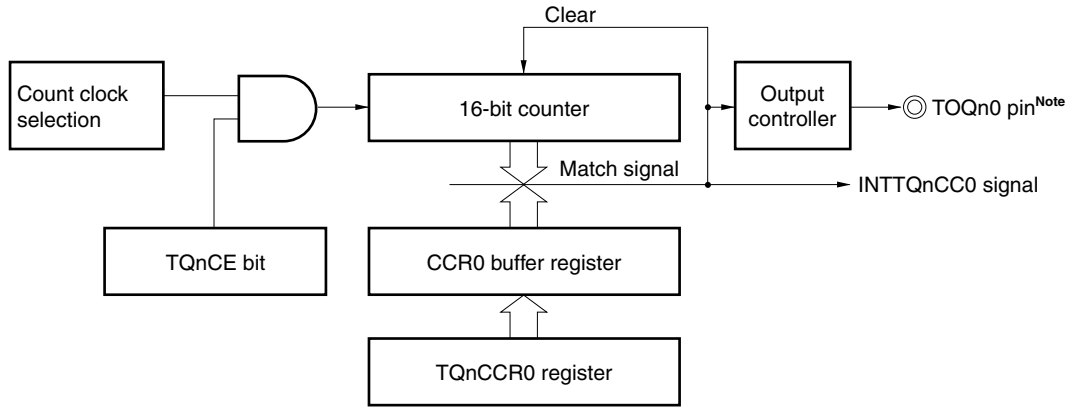
7.6.1 Interval timer mode (TQnMD2 to TQnMD0 = 000)

In the interval timer mode, an interrupt request signal (INTTQnCC0) is generated at the interval set by the TQnCCR0 register if the TQnCTL0.TQnCE bit is set to 1. A PWM waveform with a duty factor of 50% whose half cycle is equal to the interval can be output from the TOQn0 pin (the TOQ10 pin is provided in the V850E/IA4 only).

The TQnCCR1 to TQnCCR3 registers are not used in the interval timer mode. However, the set value of the TQnCCR1 to TQnCCR3 registers is transferred to the CCR1 to CCR3 buffer registers and, when the count value of the 16-bit counter matches the value of the CCR1 to CCR3 buffer registers, compare match interrupt request signals (INTTQnCC1 to INTTQnCC3) are generated. In addition, a PWM waveform with a duty factor of 50%, which is inverted when the INTTQ0CC1 to INTTQ0CC3 signals are generated, can be output from the TOQ01 to TOQ03 pins.

The value of the TQnCCR1 to TQnCCR3 registers can be rewritten even while the timer is operating.

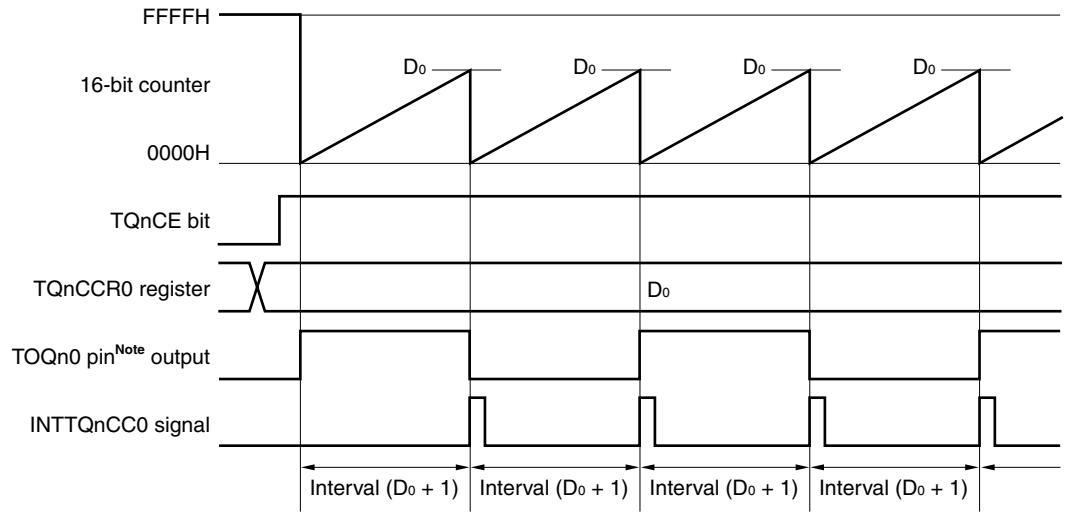
Figure 7-7. Interval Timer Configuration



Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

Figure 7-8. Basic Timing of Operation in Interval Timer Mode



Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

When the TQnCE bit is set to 1, the value of the 16-bit counter is cleared from FFFFH to 0000H in synchronization with the count clock, and the counter starts counting. At this time, the output of the TOQn0 pin[Note] is inverted. Additionally, the set value of the TQnCCR0 register is transferred to the CCR0 buffer register.

When the count value of the 16-bit counter matches the value of the CCR0 buffer register, the 16-bit counter is cleared to 0000H, the output of the TOQn0 pin[Note] is inverted, and a compare match interrupt request signal (INTTQnCC0) is generated.

The interval can be calculated by the following expression.

Interval = (Set value of TQnCCR0 register + 1) × Count clock cycle

Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

Figure 7-9. Register Setting for Interval Timer Mode Operation (1/3)

(a) TMQn control register 0 (TQnCTL0)

	TQnCE					TQnCKS2	TQnCKS1	TQnCKS0
TQnCTL0	0/1	0	0	0	0	0/1	0/1	0/1

Select count clock

0: Stop counting
1: Enable counting

(b) TMQn control register 1 (TQnCTL1)

		TQ0EST	TQ0EEE			TQnMD2	TQnMD1	TQnMD0
<R>	TQnCTL1	0	0	0/1[Note]	0	0	0	0	0

0, 0, 0:
Interval timer mode

0: Operate on count clock selected by TQ0CKS0 to TQ0CKS2 bits
1: Count with external event count input signal

<R> **Note** Enable setting of the TQ0EEE bit to 1 only when timer output (TOQ00, TOQ0b) is used. In this case, set the TQ0CCR0 and TQ0CCRb registers to the same value (b = 1 to 3).

Figure 7-9. Register Setting for Interval Timer Mode Operation (2/3)

(c) TMQn I/O control register 0 (TQnIOC0)

	TQ0OL3	TQ0OE3	TQ0OL2	TQ0OE2	TQ0OL1	TQ0OE1	TQnOL0	TQnOE0
TQnIOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

- TQnOE0: 0: Disable TOQn0 pin output / 1: Enable TOQn0 pin output
- TQnOL0: Setting of TOQn0 pin output level before count operation / 0: Low level / 1: High level
- TQ0OE1: 0: Disable TOQ01 pin output / 1: Enable TOQ01 pin output
- TQ0OL1: Setting of TOQ01 pin output level before count operation / 0: Low level / 1: High level
- TQ0OE2: 0: Disable TOQ02 pin output / 1: Enable TOQ02 pin output
- TQ0OL2: Setting of TOQ02 pin output level before count operation / 0: Low level / 1: High level
- TQ0OE3: 0: Disable TOQ03 pin output / 1: Enable TOQ03 pin output
- TQ0OL3: Setting of TOQ03 pin output level before count operation / 0: Low level / 1: High level

<R> **(d) TMQ0 I/O control register 2 (TQ0IOC2)**

					TQ0EES1	TQ0EES0	TQ0ETS1	TQ0ETS0
TQ0IOC2	0	0	0	0	0/1 Note	0/1 Note	0	0

- TQ0EES1, TQ0EES0: Selection of valid edge of external event count input (EVTQ0 pin)

Note Enable setting of the TQ0EES1 and TQ0EES0 bits only when timer output (TOQ00 to TOQ03) is used. In this case, set the TQ0CCR0 to TQ0CCR3 registers to the same value.

(e) TMQn counter read buffer register (TQnCNT)

By reading the TQnCNT register, the count value of the 16-bit counter can be read.

(f) TMQn capture/compare register 0 (TQnCCR0)

If the TQnCCR0 register is set to D_0, the interval is as follows.

Interval = $(D_0 + 1) \times$ Count clock cycle

Figure 7-9. Register Setting for Interval Timer Mode Operation (3/3)

(g) TMQn capture/compare registers 1 to 3 (TQnCCR1 to TQnCCR3)

The TQnCCR1 to TQnCCR3 registers are not used in the interval timer mode. However, the set values of the TQnCCR1 to TQnCCR3 registers are transferred to the CCR1 to CCR3 buffer registers. When the count value of the 16-bit counter matches the value of the CCR1 to CCR3 buffer registers, the TOQ01 to TOQ03 pin outputs are inverted and the compare match interrupt request signals (INTTQnCC1 to INTTQnCC3) are generated.

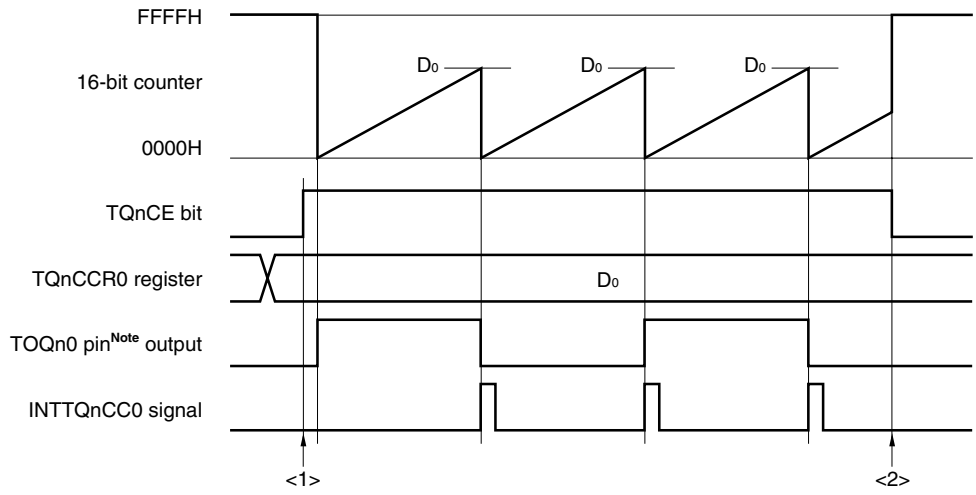
When the TQnCCR1 to TQnCCR3 registers are not used, it is recommended to set their values to FFFFH. Also mask the registers by the interrupt mask flags (TQnCCIC1.TQnCCMK1 to TQnCCIC3.TQnCCMK3).

<R> **Remarks 1.** TMQ0 I/O control register 1 (TQ0IOC1) and TMQn option register 0 (TQnOPT0) are not used in the interval timer mode.

2. n = 0, 1

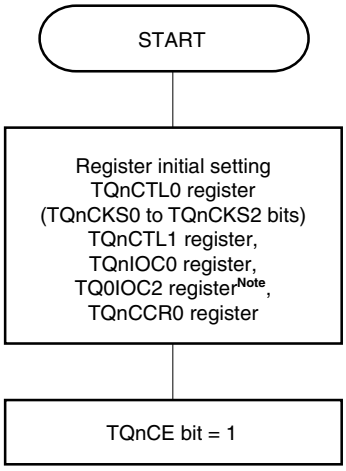
(1) Interval timer mode operation flow

Figure 7-10. Software Processing Flow in Interval Timer Mode



Note The TOQ10 pin is provided only in the V850E/IA4.

<1> Count operation start flow

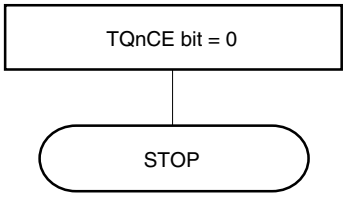


Initial setting of these registers is performed before setting the TQnCE bit to 1.

The TQnCKS0 to TQnCKS2 bits can be set at the same time when counting has been started (TQnCE bit = 1).

<R> **Note** Enable setting of the TQ0EES1 and TQ0EES0 bits only when timer output (TOQ00 to TOQ03) is used. In this case, set the TQ0CCR0 to TQ0CCR3 registers to the same value.

<2> Count operation stop flow



The counter is initialized and counting is stopped by clearing the TQnCE bit to 0. The output level of the TOQn0 pinNote is as specified by the TQnIOC0 register.

Note The TOQ10 pin is provided only in the V850E/IA4.

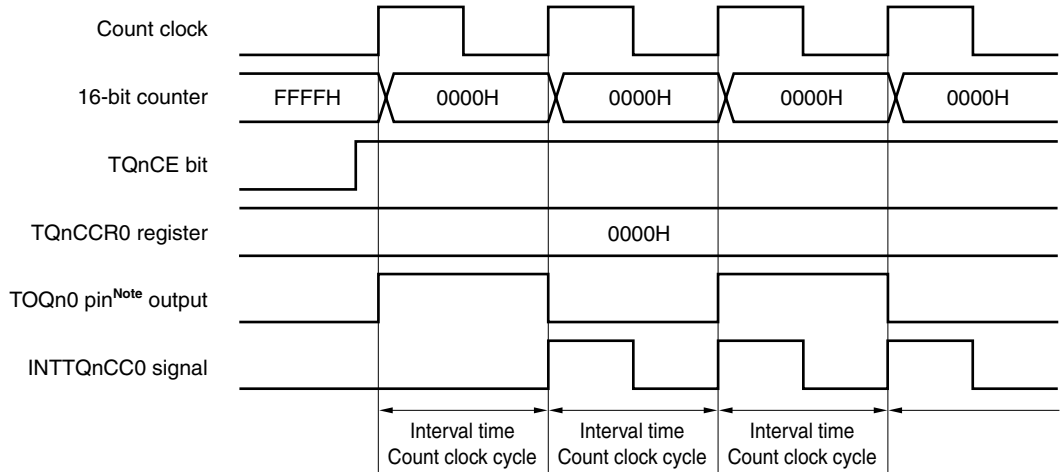
Remark n = 0, 1

(2) Interval timer mode operation timing

(a) Operation if TQnCCR0 register is set to 0000H

If the TQnCCR0 register is set to 0000H, the INTTQnCC0 signal is generated at each count clock, and the output of the TOQn0 pin is inverted (the TOQ10 pin is provided only in the V850E/IA4).

The value of the 16-bit counter is always 0000H.

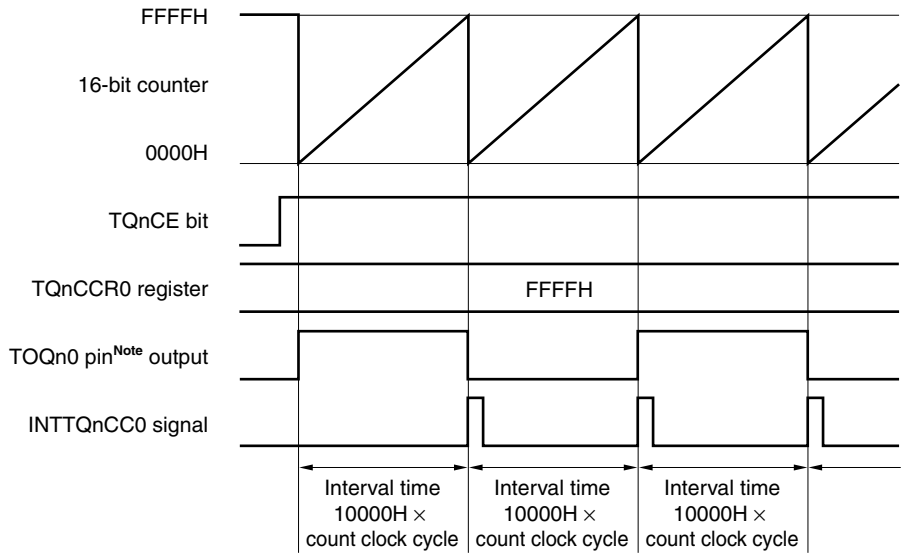


Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

(b) Operation if TQnCCR0 register is set to FFFFH

If the TQnCCR0 register is set to FFFFH, the 16-bit counter counts up to FFFFH. The counter is cleared to 0000H in synchronization with the next count-up timing. The INTTQnCC0 signal is generated and the output of the TOQn0 pin is inverted (the TOQ10 pin is provided only in the V850E/IA4). At this time, an overflow interrupt request signal (INTTQnOV) is not generated, nor is the overflow flag (TQnOPT0.TQnOVF bit) set to 1.

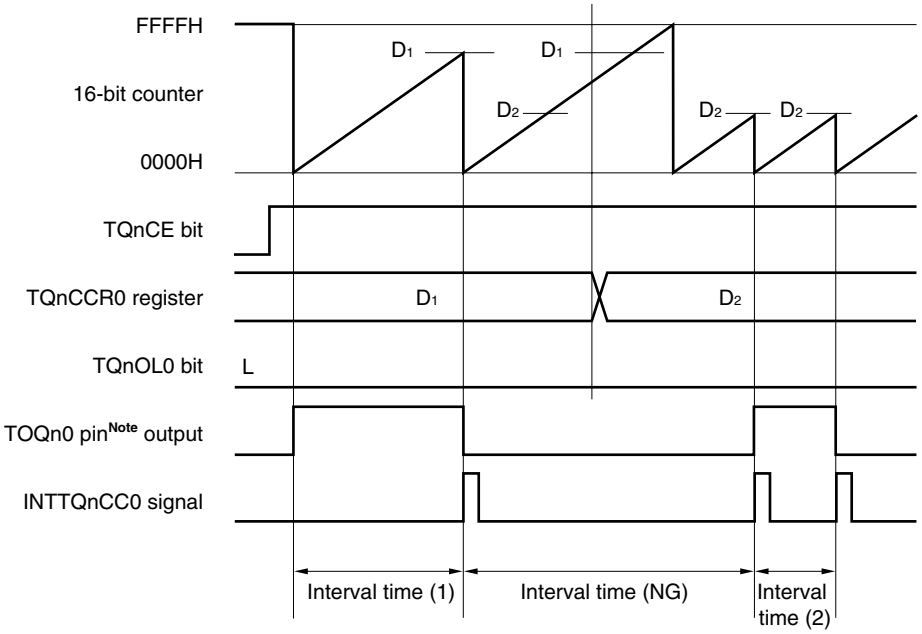


Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

(c) Notes on rewriting TQnCCR0 register

If the value of the TQnCCR0 register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When the overflow may occur, stop counting once and then change the set value.



Note The TOQ10 pin is provided only in the V850E/IA4.

Remarks 1. Interval time (1): $(D_1 + 1) \times$ Count clock cycle
Interval time (NG): $(10000H + D_2 + 1) \times$ Count clock cycle
Interval time (2): $(D_2 + 1) \times$ Count clock cycle

2. n = 0, 1

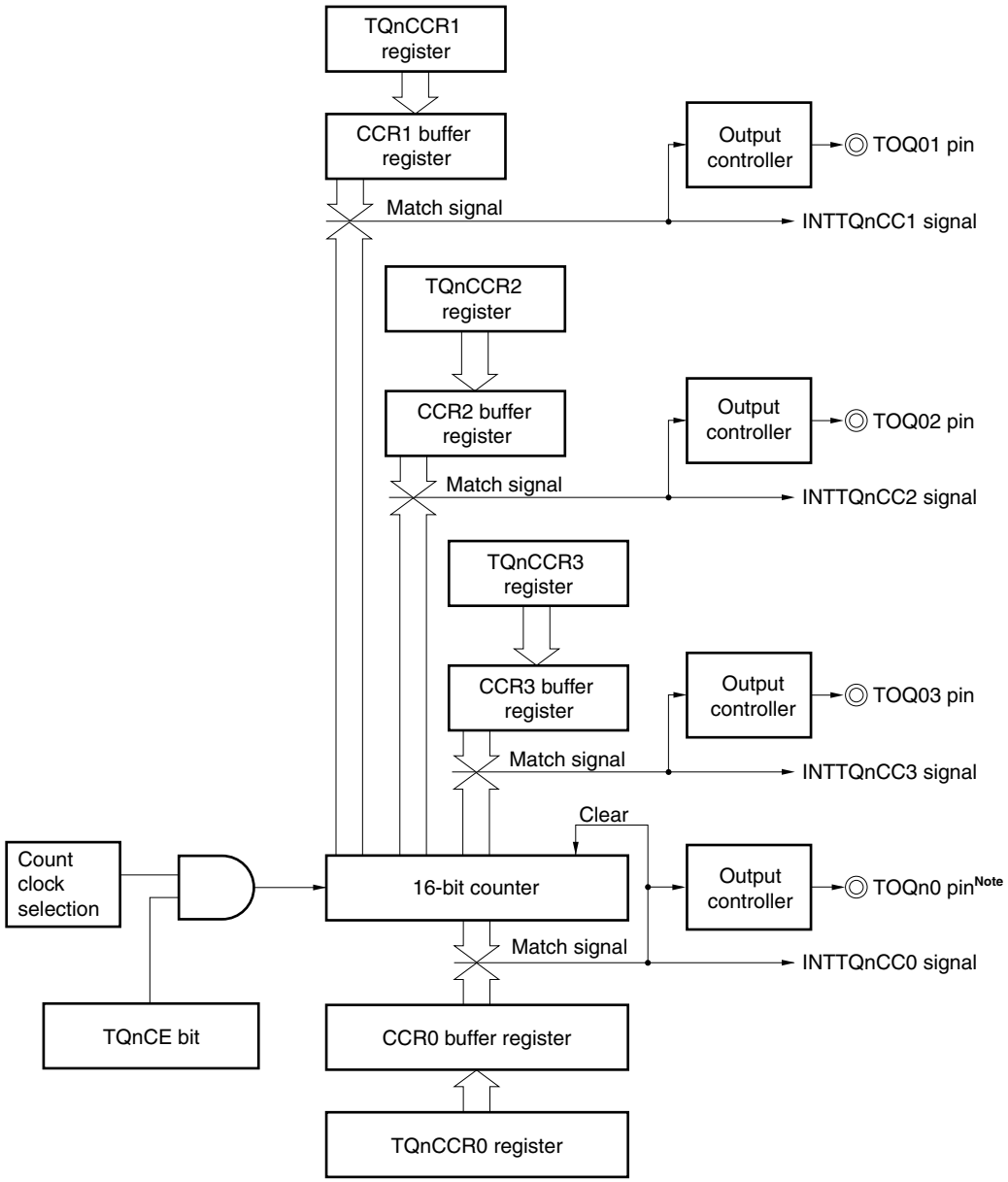
If the value of the TQnCCR0 register is changed from D_1 to D_2 while the count value is greater than D_2 but less than D_1, the count value is transferred to the CCR0 buffer register as soon as the TQnCCR0 register has been rewritten. Consequently, the value of the 16-bit counter that is compared is D_2.

Because the count value has already exceeded D_2, however, the 16-bit counter counts up to FFFFH, overflows, and then counts up again from 0000H. When the count value matches D_2, the INTTQnCC0 signal is generated and the output of the TOQn0 pin is inverted (the TOQ10 pin is provided only in the V850E/IA4).

Therefore, the INTTQnCC0 signal may not be generated at the interval time "$(D_1 + 1) \times$ Count clock cycle" or "$(D_2 + 1) \times$ Count clock cycle" originally expected, but may be generated at an interval of "$(10000H + D_2 + 1) \times$ Count clock cycle".

(d) Operation of TQnCCR1 to TQnCCR3 registers

Figure 7-11. Configuration of TQnCCR1 to TQnCCR3 Registers



Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1

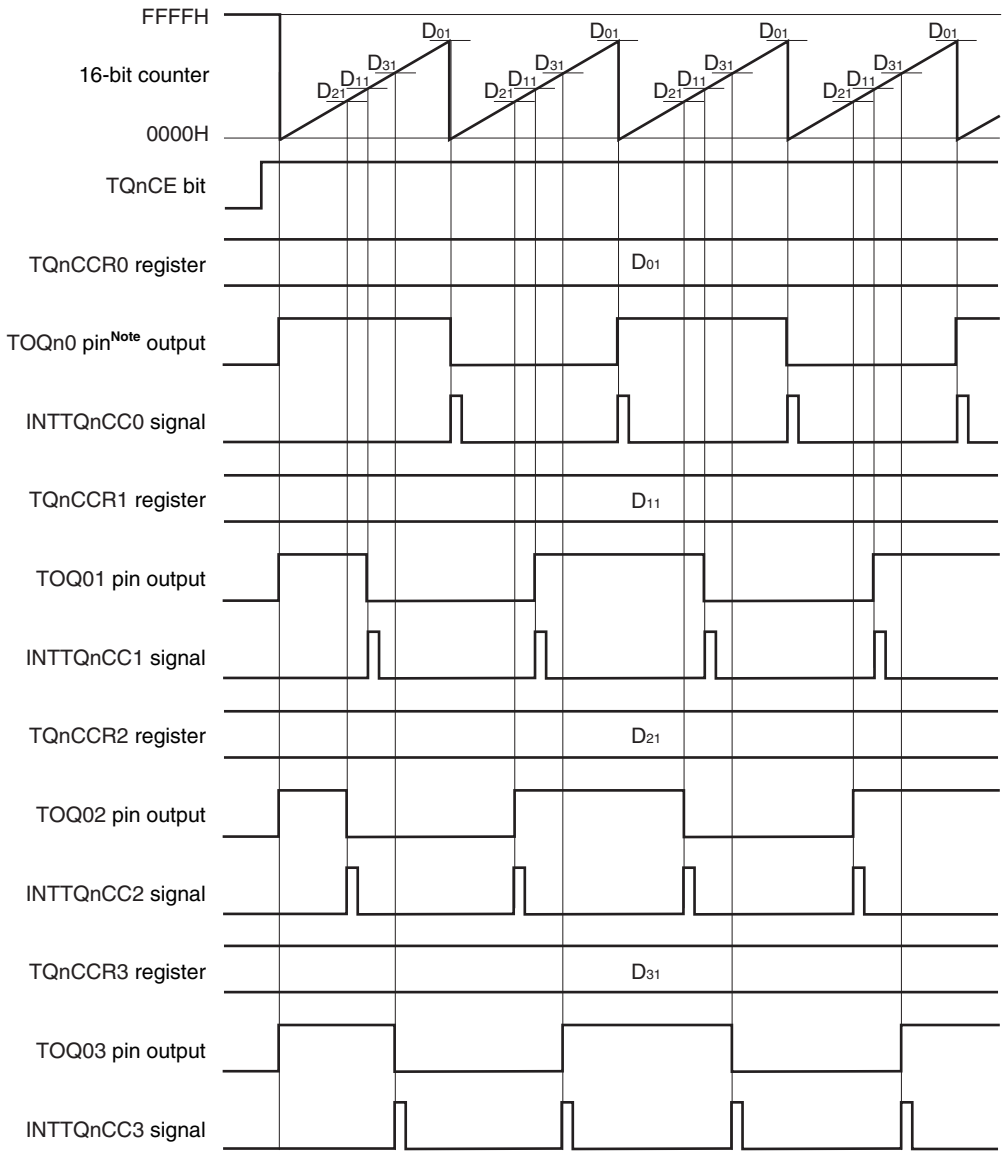
<R> When the TQnCCRb register is set to the same value as that of the TQnCCR0 register, the INTTQnCCb signal is generated at the same timing as the INTTQnCC0 signal is generated, and the TOQ0b pin output is inverted. In other words, a PWM waveform with a duty factor of 50% can be output from the TOQ0b pin.

The following shows the operation when the TQnCCRb register is set to other than the value set in the TQnCCR0 register.

If the set value of the TQnCCRb register is less than the set value of the TQnCCR0 register, the INTTQnCCb signal is generated once per cycle. At the same time, the output of the TOQ0b pin is inverted.

<R> After outputting the short-width pulse first, the TOQ0b pin outputs a PWM waveform with a duty factor of 50%.

Figure 7-12. Timing Chart When $D_{01} \geq D_{b1}$

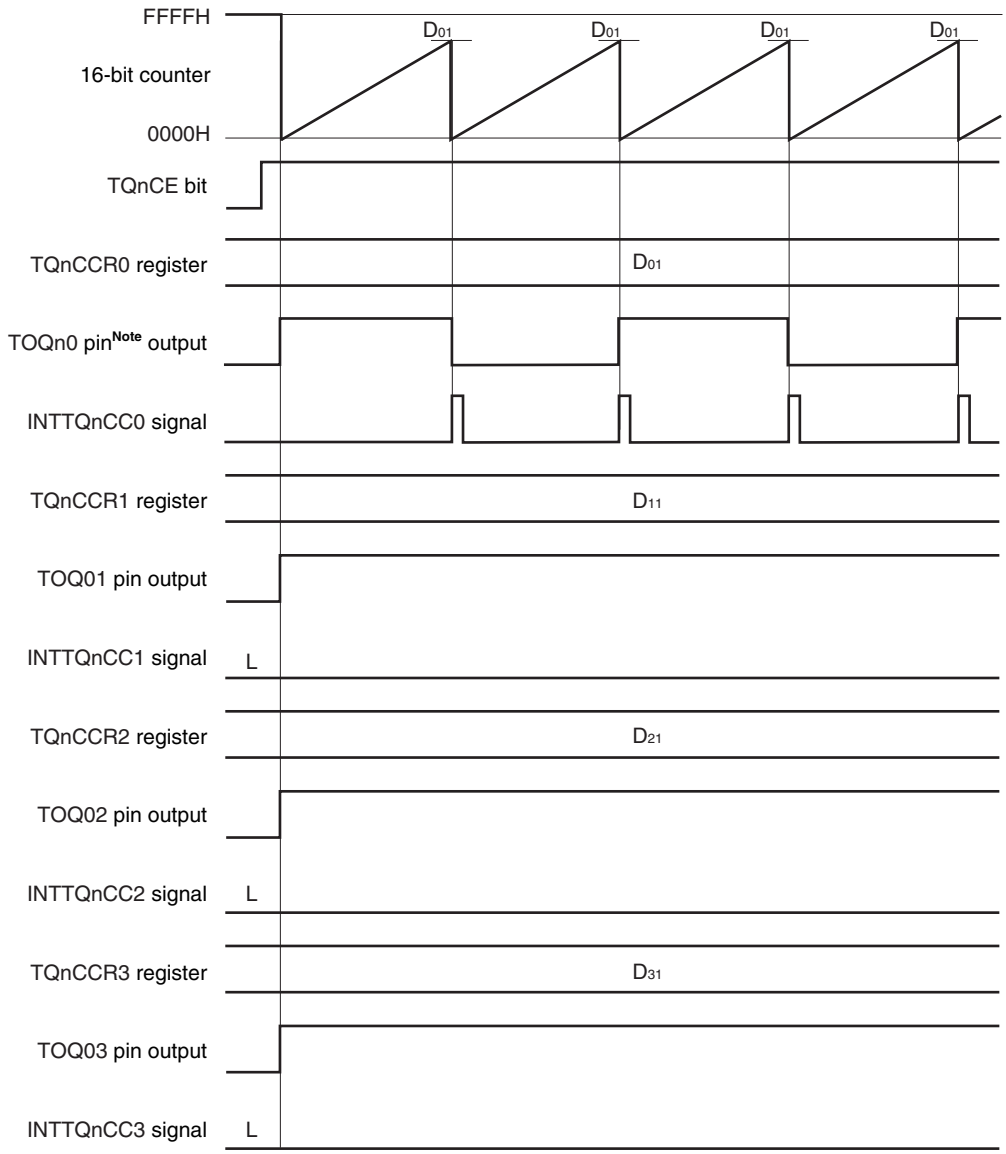


Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1
b = 1 to 3

If the set value of the TQnCCRb register is greater than the set value of the TQnCCR0 register, the count value of the 16-bit counter does not match the value of the TQnCCRb register. Consequently, the INTTQnCCb signal is not generated, nor is the output of the TOQ0b pin changed.
When the TQnCCRb register is not used, it is recommended to set its value to FFFFH.

Figure 7-13. Timing Chart When $D_{01} < D_{b1}$



Note The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1
b = 1 to 3

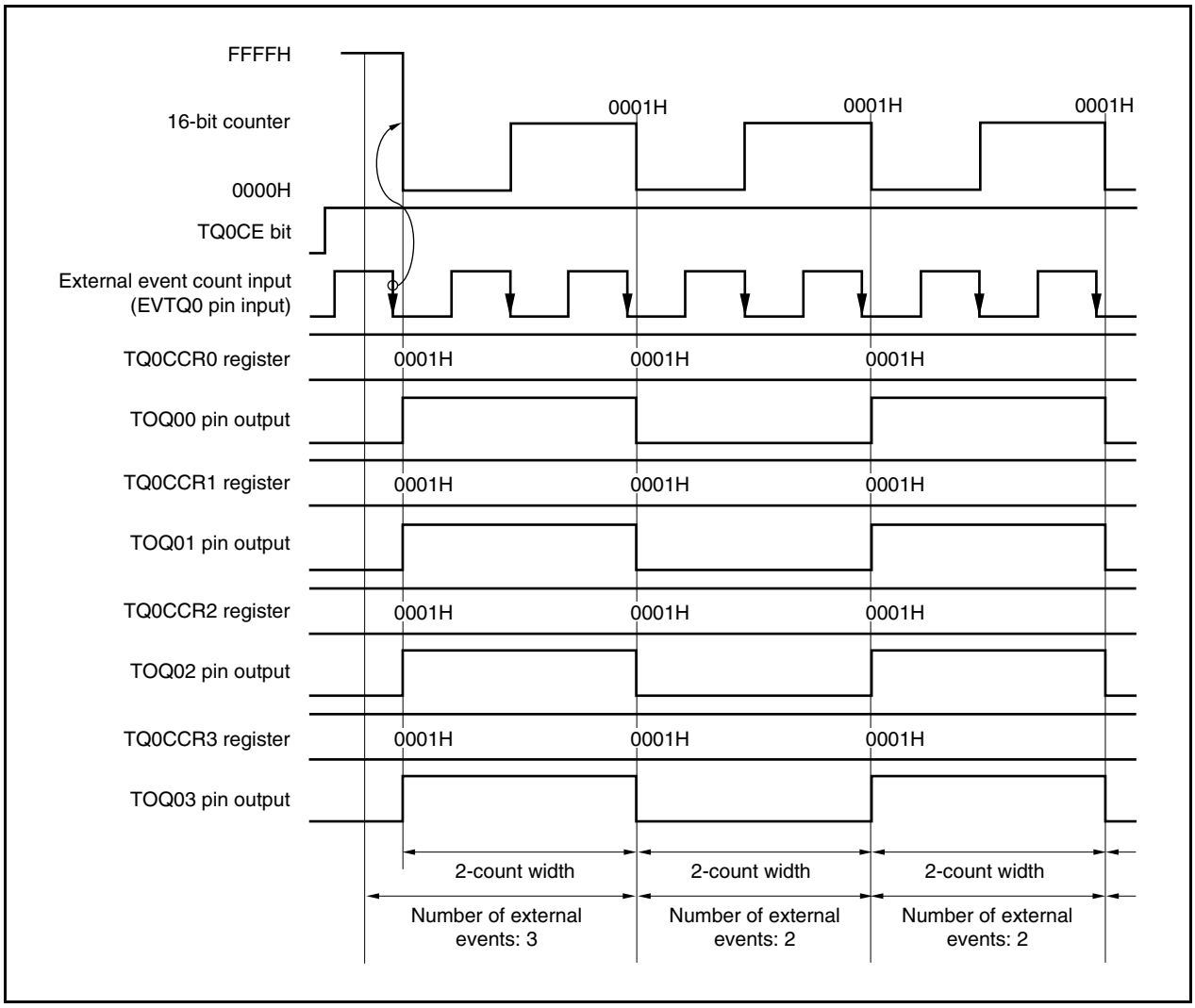
<R> **(3) Operation by external event count input (EVTQ0)**

(a) Operation

To count the 16-bit counter at the valid edge of external event count input (EVTQ0) in the interval timer mode, clear the 16-bit counter from FFFFH to 0000H at the valid edge of the first external event count input after the TQ0CE bit is set from 0 to 1.

When both the TQ0CCR0 and TQ0CCRb registers are set to 0001H, the output of the TOQ00 and TOQ0b pins is inverted each time the 16-bit counter counts twice (b = 1 to 3).

The TQ0CTL0.TQ0EEE bit can be set to 1 in the interval timer mode only when the timer output (TOQ00, TOQ0b) is used with the external event count input.



7.6.2 External event count mode (TQ0MD2 to TQ0MD0 bits = 001)

This mode is valid only in TMQ0.

In the external event count mode, the valid edge of the external event count input (EVTQ0) is counted when the TQ0CTL0.TQ0CE bit is set to 1, and an interrupt request signal (INTTQ0CC0) is generated each time the specified number of edges set by the TQ0CCR0 register have been counted. The TOQ00 to TOQ03 pins cannot be used.

<R> When using the TOQ00 to TOQ03 pins in the external event count input mode, set the TQ0CTL1.TQ0EEE bit to 1 in the interval timer mode (see **7.6.1 (3) Operation by external event count input (EVTQ0)**).

The TQ0CCR1 to TQ0CCR3 registers are not used in the external event count mode.

Caution In the external event count mode, the TQ0CCR0 to TQ0CCR3 registers must not be cleared to 0000H.

Figure 7-14. Configuration in External Event Count Mode

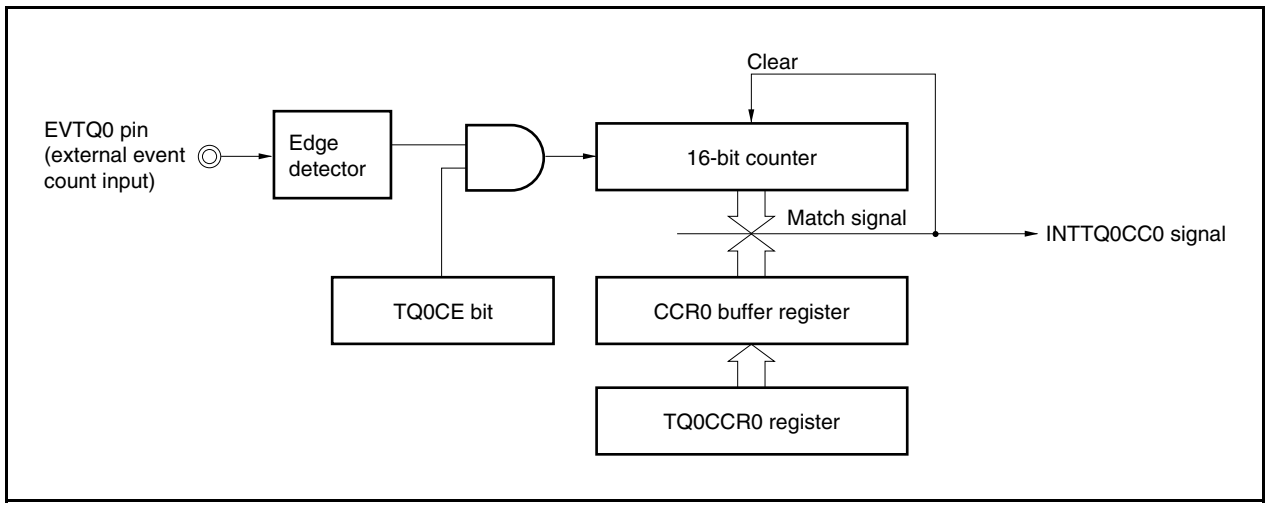
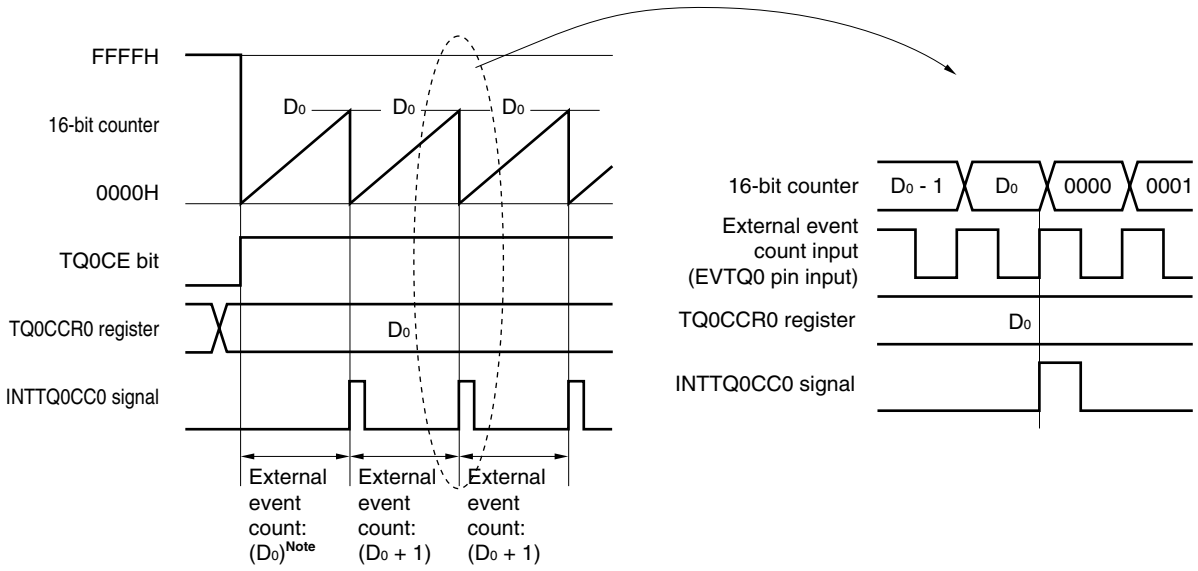


Figure 7-15. Basic Timing in External Event Count Mode



Note In the external event count mode, when the TQ0CTL0.TQ0CE bit is set to 1 (operation starts), the 16-bit counter is cleared from FFFFH to 0000H at the same time. The first count operation starts from 0001H every time the valid edge of the external event count input is detected. Therefore, the count of the first count operation is one number smaller than the count of second or subsequent count operation.

Remark This figure shows the basic timing when the rising edge is specified as the valid edge of the external event count input.

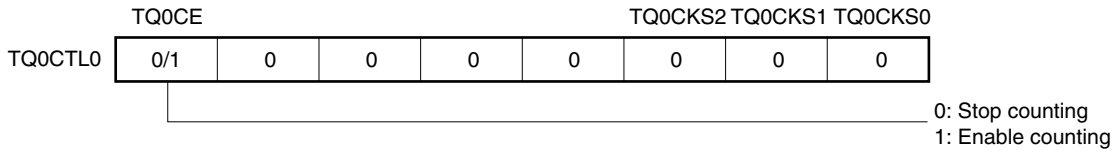
When the TQ0CE bit is set to 1, the value of the 16-bit counter is cleared from FFFFH to 0000H. The counter counts each time the valid edge of external event count input is detected. Additionally, the set value of the TQ0CCR0 register is transferred to the CCR0 buffer register.

When the count value of the 16-bit counter matches the value of the CCR0 buffer register, the 16-bit counter is cleared to 0000H, and a compare match interrupt request signal (INTTQ0CC0) is generated.

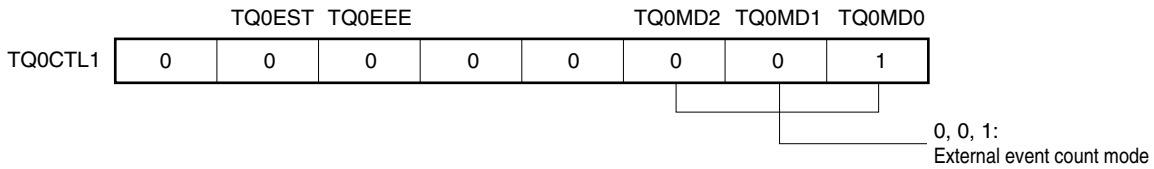
The INTTQ0CC0 signal is generated for the first time when the valid edge of the external event count input has been detected "value set to TQ0CCR0 register" times. After that, the INTTQ0CC0 signal is generated each time the valid edge of the external event count has been detected "value set to TQ0CCR0 register + 1" times.

Figure 7-16. Register Setting for Operation in External Event Count Mode (1/2)

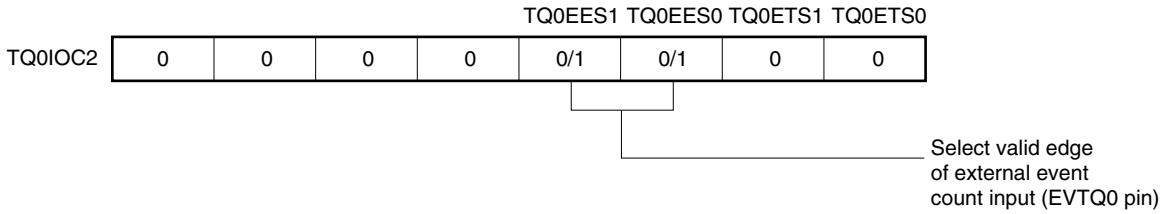
(a) TMQ0 control register 0 (TQ0CTL0)



(b) TMQ0 control register 1 (TQ0CTL1)



(c) TMQ0 I/O control register 2 (TQ0IOC2)



(d) TMQ0 counter read buffer register (TQ0CNT)

The count value of the 16-bit counter can be read by reading the TQ0CNT register.

(e) TMQ0 capture/compare register 0 (TQ0CCR0)

If the TQ0CCR0 register is set to D_0, the count is cleared when the number of external event counts has reached (D_0) and the first compare match interrupt request signal (INTTQ0CC0) is generated. The second compare match interrupt request signal (INTTQ0CC0) is generated when the number of external event counts has reached ($D_0 + 1$).

Figure 7-16. Register Setting for Operation in External Event Count Mode (2/2)

(f) TMQ0 capture/compare registers 1 to 3 (TQ0CCR1 to TQ0CCR3)

The TQ0CCR1 to TQ0CCR3 registers are not used in the external event count mode. However, the set values of the TQ0CCR1 to TQ0CCR3 registers are transferred to the CCR1 to CCR3 buffer registers. When the count value of the 16-bit counter matches the value of the CCR1 to CCR3 buffer registers, compare match interrupt request signals (INTTQ0CC1 to INTTQ0CC3) are generated.

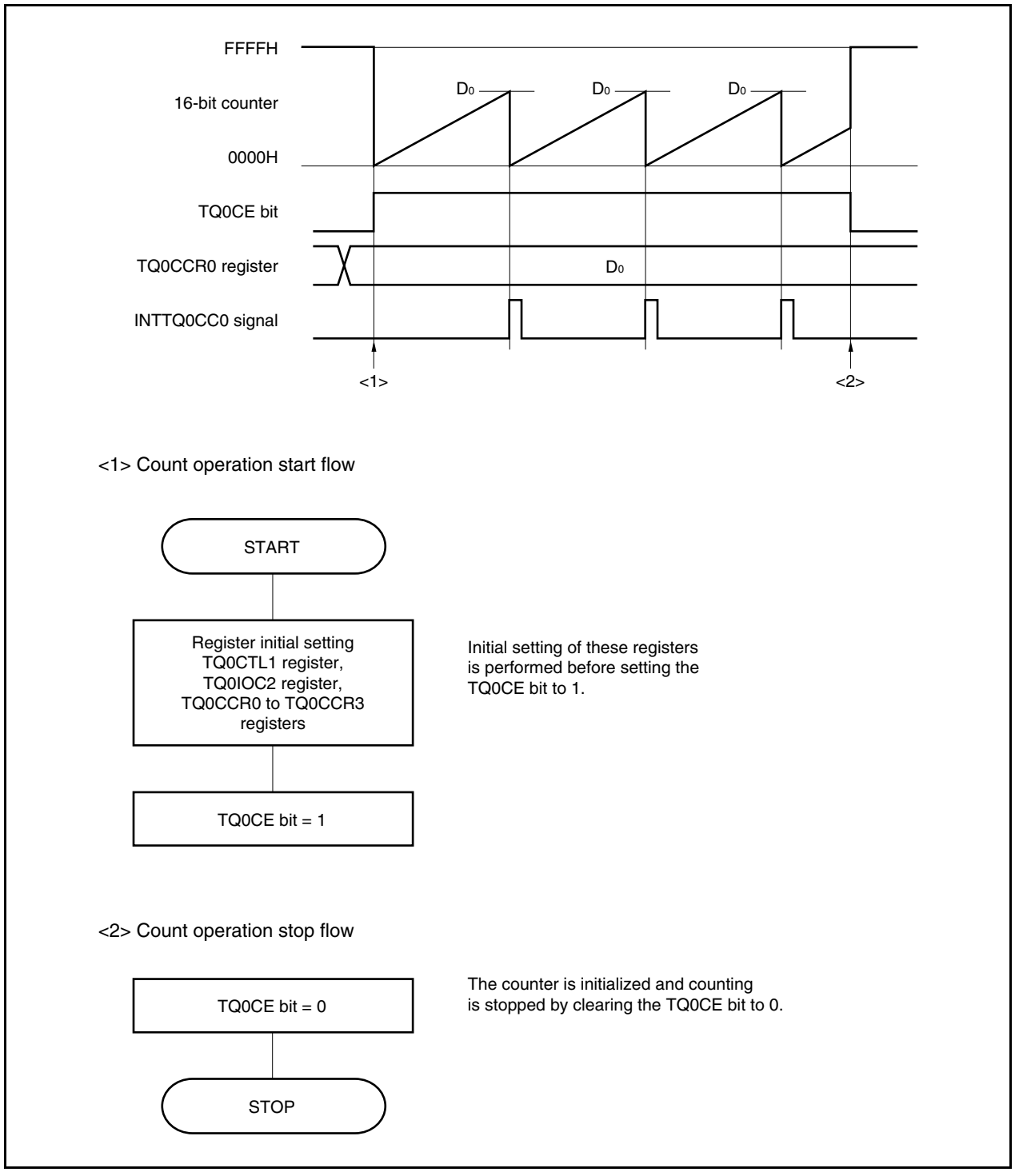
When the TQ0CCR1 to TQ0CCR3 registers are not used, it is recommended to set their values to FFFFH. Also mask the registers by the interrupt mask flags (TQ0CCIC1.TQ0CCMK1 to TQ0CCIC3.TQ0CCMK3).

<R> **Caution Set the TQ0IOC0 register to 00H.**

Remark TMQ0 I/O control register 1 (TQ0IOC1) and TMQ0 option register 0 (TQ0OPT0) are not used in the external event count mode.

(1) External event count mode operation flow

Figure 7-17. Flow of Software Processing in External Event Count Mode



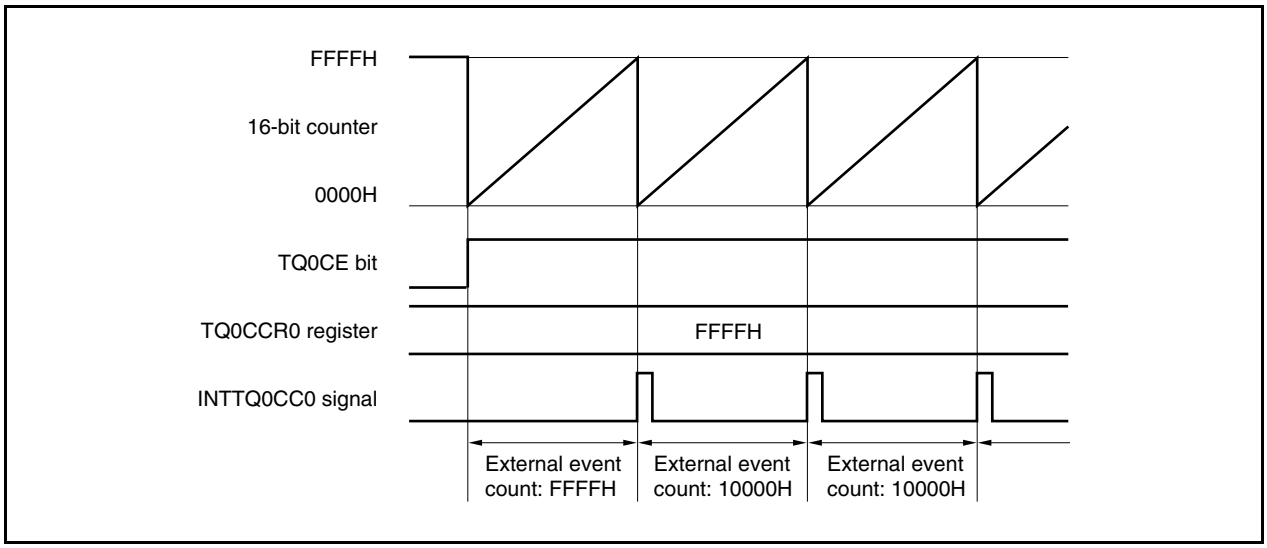
(2) Operation timing in external event count mode

Cautions 1. In the external event count mode, the TQ0CCR0 to TQ0CCR3 registers must not be cleared to 0000H.

<R> **2. In the external event count mode, use of the timer output (TOQ00 to TOQ03) is disabled. If using timer output (TOQ00 to TOQ03) with external event count input (EVTQ0), set the interval timer mode, and enable the count clock operation with the external event count input (TQ0CTL1.TQ0EEE bit = 1) (see 7.6.1 (3) Operation by external event count input (EVTQ0)).**

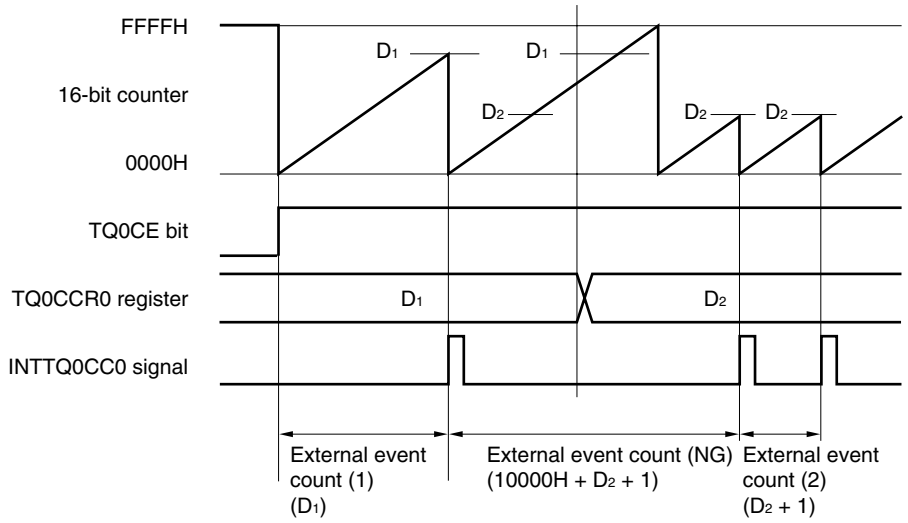
(a) Operation if TQ0CCR0 register is set to FFFFH

If the TQ0CCR0 register is set to FFFFH, the 16-bit counter counts to FFFFH each time the valid edge of the external event count signal has been detected. The 16-bit counter is cleared to 0000H in synchronization with the next count-up timing, and the INTTQ0CC0 signal is generated. At this time, the TQ0OPT0.TQ0OVF bit is not set.



(b) Notes on rewriting the TQ0CCR0 register

If the value of the TQ0CCR0 register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When the overflow may occur, stop counting once and then change the set value.



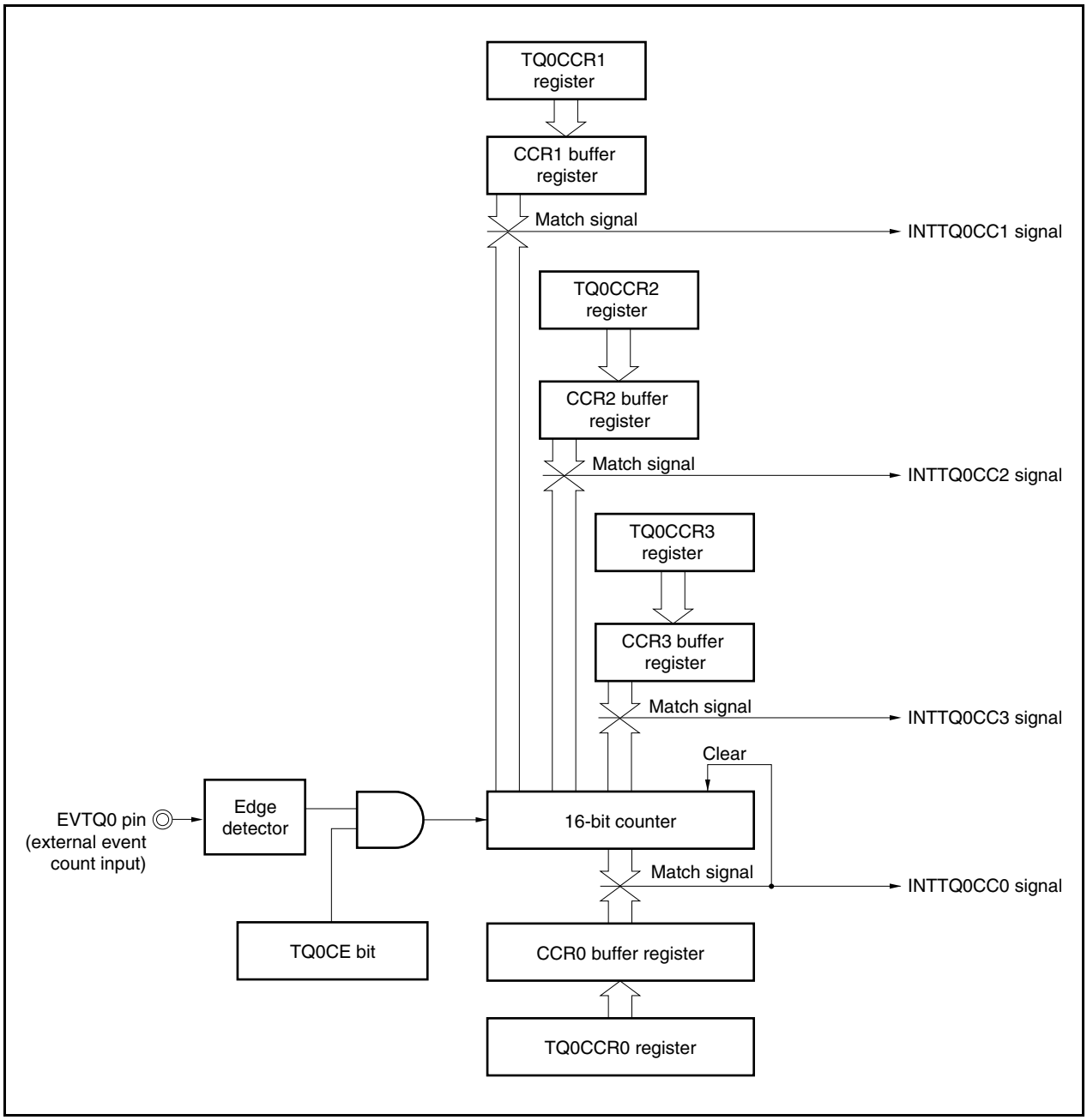
If the value of the TQ0CCR0 register is changed from D_1 to D_2 while the count value is greater than D_2 but less than D_1, the count value is transferred to the CCR0 buffer register as soon as the TQ0CCR0 register has been rewritten. Consequently, the value that is compared with the 16-bit counter is D_2.

Because the count value has already exceeded D_2, however, the 16-bit counter counts up to FFFFH, overflows, and then counts up again from 0000H. When the count value matches D_2, the INTTQ0CC0 signal is generated.

Therefore, the INTTQ0CC0 signal may not be generated at the valid edge count of "(D_1 + 1) times" or "(D_2 + 1) times" originally expected, but may be generated at the valid edge count of "(10000H + D_2 + 1) times".

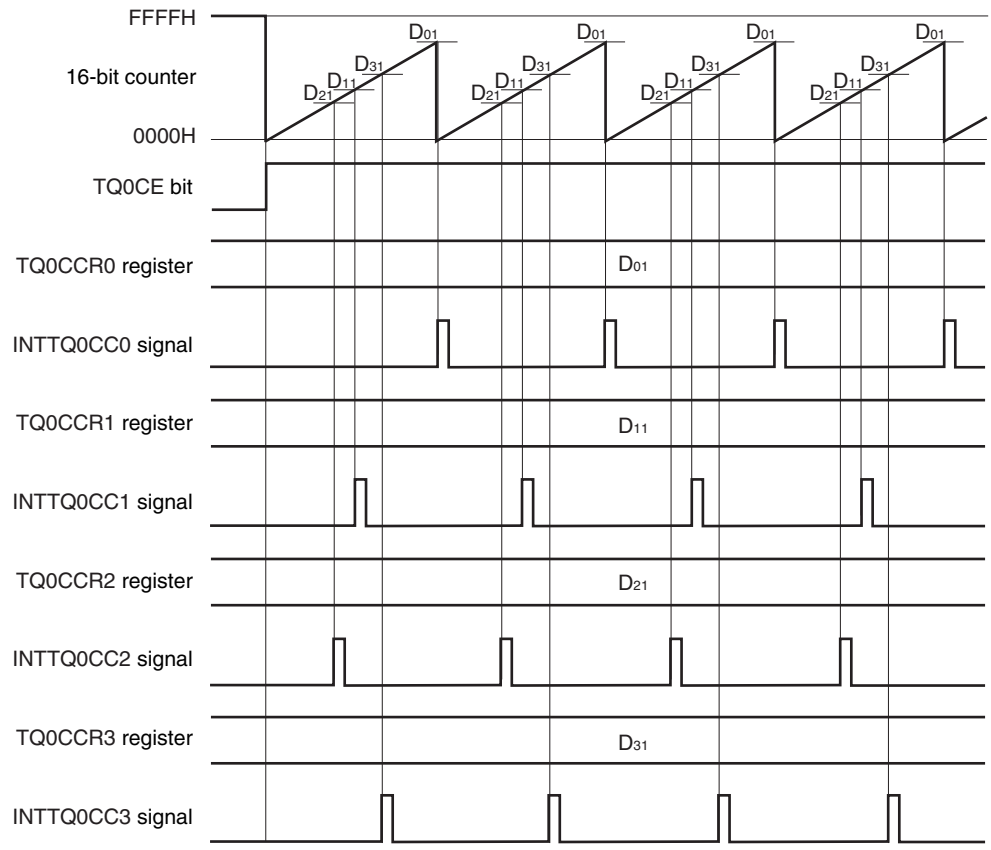
(c) Operation of TQ0CCR1 to TQ0CCR3 registers

Figure 7-18. Configuration of TQ0CCR1 to TQ0CCR3 Registers



If the set value of the TQ0CCRb register is smaller than the set value of the TQ0CCR0 register, the INTTQ0CCb signal is generated once per cycle.

Figure 7-19. Timing Chart When $D_{01} \geq D_{b1}$

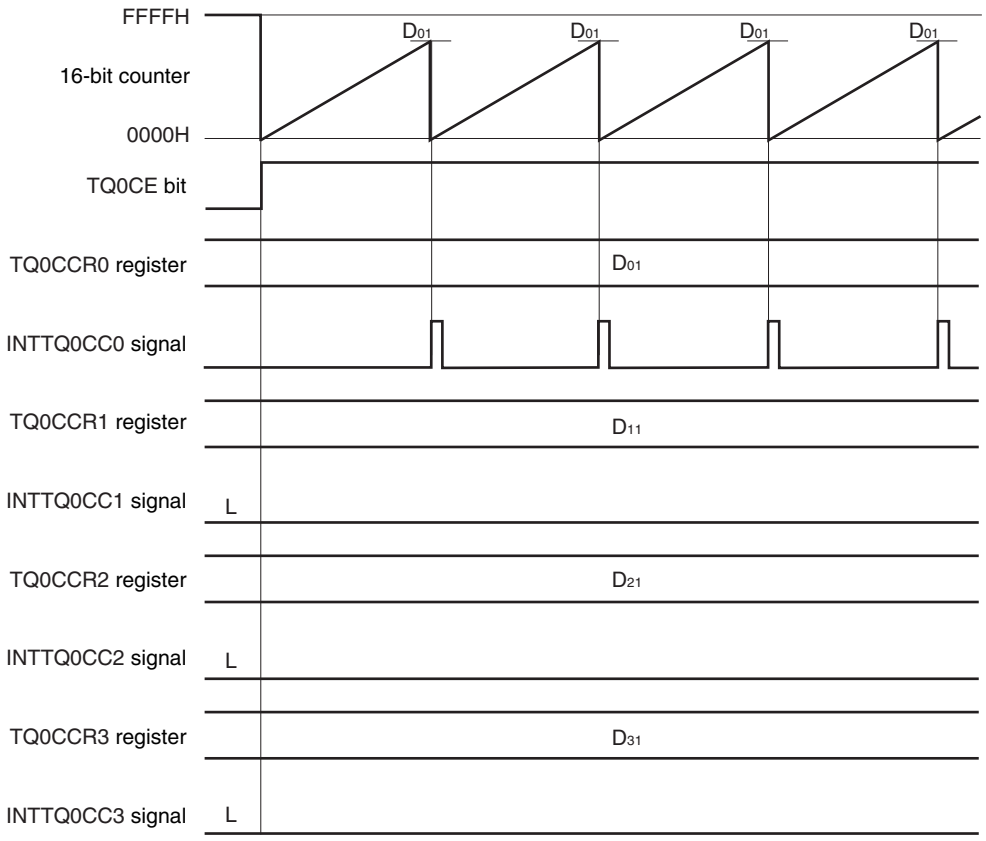


Remark b = 1 to 3

If the set value of the TQ0CCRb register is greater than the set value of the TQ0CCR0 register, the INTTQ0CCb signal is not generated because the count value of the 16-bit counter and the value of the TQ0CCRb register do not match.

When the TQ0CCRb register is not used, it is recommended to set its value to FFFFH.

Figure 7-20. Timing Chart When $D_{01} < D_{b1}$



Remark b = 1 to 3

7.6.3 External trigger pulse output mode (TQ0MD2 to TQ0MD0 bits = 010)

This mode is valid only in TMQ0.

In the external trigger pulse output mode, 16-bit timer/event counter Q waits for a trigger when the TQ0CTL0.TQ0CE bit is set to 1. When the valid edge of an external trigger input (TRGQ0) is detected, 16-bit timer/event counter Q starts counting, and outputs up to 3-phase PWM waveform from the TOQ01 to TOQ03 pins. A PWM waveform with a duty factor of 50% whose half cycle is the set value of the TQ0CCR0 register + 1 can also be output from the TOQ00 pin.

Pulses can also be output by generating a software trigger instead of using the external trigger input.

Figure 7-21. Configuration in External Trigger Pulse Output Mode

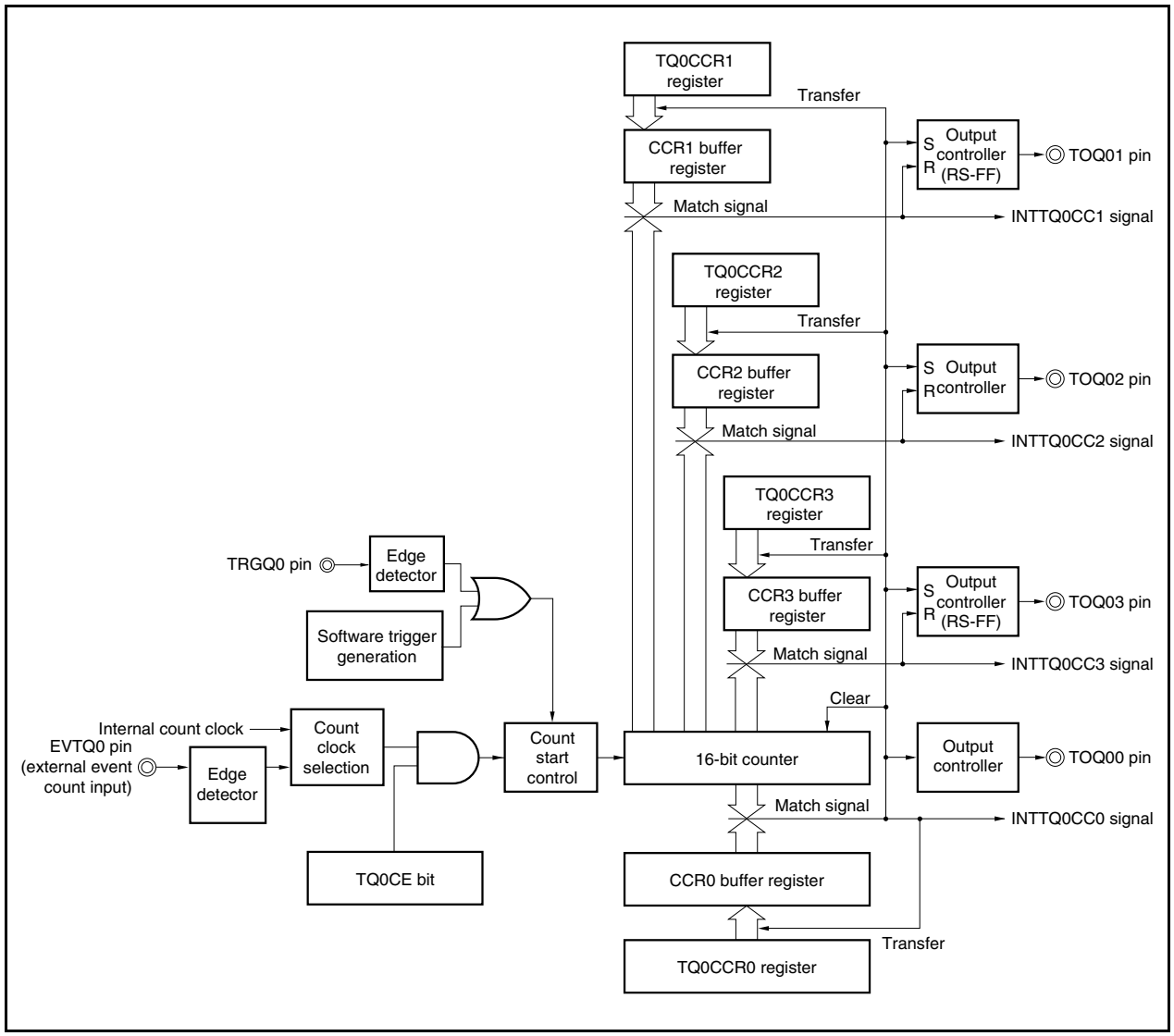
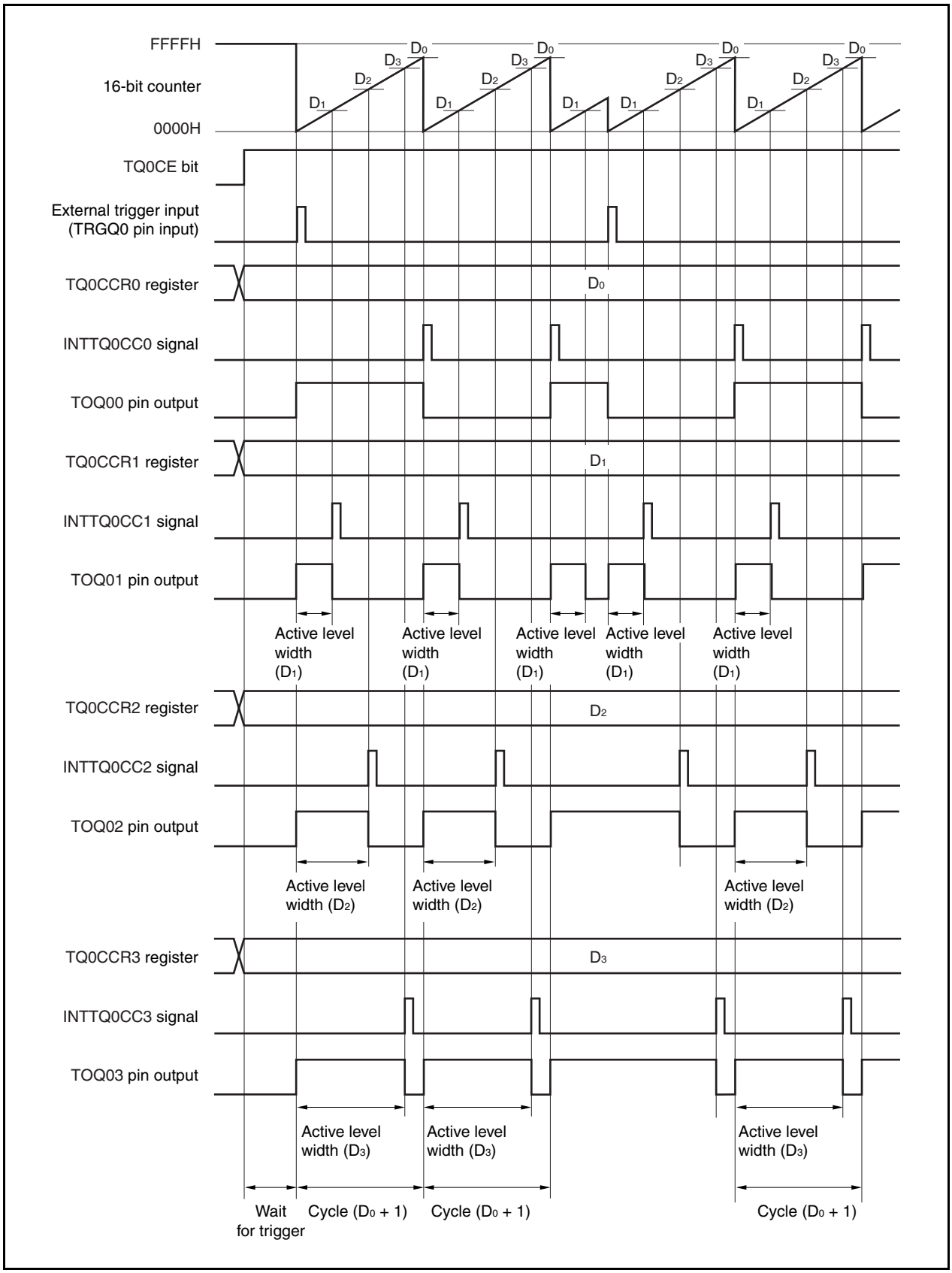


Figure 7-22. Basic Timing in External Trigger Pulse Output Mode

<R>



16-bit timer/event counter Q waits for a trigger when the TQ0CE bit is set to 1. When the trigger is generated, the 16-bit counter is cleared from FFFFH to 0000H, starts counting at the same time, and outputs a PWM waveform from the TOQ0b pin. If the trigger is generated again while the counter is operating, the counter is cleared to 0000H and
<R> restarted. (The output of the TOQ00 pin is inverted. The TOQ0b pin outputs high level regardless of the status (high/low) when a trigger occurs.)

The active level width, cycle, and duty factor of the PWM waveform can be calculated as follows.

Active level width = (Set value of TQ0CCRb register) × Count clock cycle
Cycle = (Set value of TQ0CCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TQ0CCRb register)/(Set value of TQ0CCR0 register + 1)

The compare match interrupt request signal INTTQ0CC0 is generated when the 16-bit counter counts next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The compare match interrupt request signal INTTQ0CCb is generated when the count value of the 16-bit counter matches the value of the CCRb buffer register.

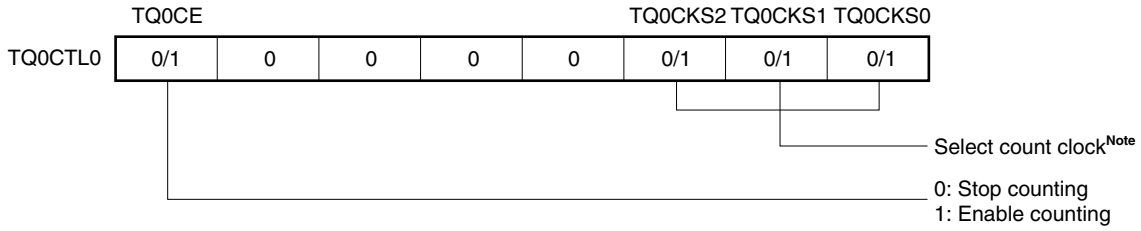
The value set to the TQ0CCRa register is transferred to the CCRa buffer register when the count value of the 16-bit counter matches the value of the CCRa buffer register and the 16-bit counter is cleared to 0000H.

The valid edge of the external trigger input (TRGQ0) and setting the software trigger (TQ0CTL1.TQ0EST bit) to 1 are used as the trigger.

Remark a = 0 to 3
b = 1 to 3

Figure 7-23. Setting of Registers in External Trigger Pulse Output Mode (1/3)

(a) TMQ0 control register 0 (TQ0CTL0)



Note The setting is invalid when the TQ0CTL1.TQ0EEE bit = 1.

Figure 7-23. Setting of Registers in External Trigger Pulse Output Mode (2/3)

(b) TMQ0 control register 1 (TQ0CTL1)

	TQ0EST	TQ0EEE				TQ0MD2	TQ0MD1	TQ0MD0
TQ0CTL1	0	0/1	0/1	0	0	0	1	0

0, 1, 0:
External trigger pulse output mode

0: Operate on count clock selected by TQ0CKS0 to TQ0CKS2 bits
1: Count with external event count input signal

Generate software trigger when 1 is written

(c) TMQ0 I/O control register 0 (TQ0IOC0)

	TQ0OL3	TQ0OE3	TQ0OL2	TQ0OE2	TQ0OL1	TQ0OE1	TQ0OL0	TQ0OE0
TQ0IOC0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

0: Disable TOQ00 pin output
1: Enable TOQ00 pin output

Setting of TOQ00 pin output level in external trigger waiting state
0: Low level
1: High level

0: Disable TOQ01 pin output
1: Enable TOQ01 pin output

Setting of TOQ01 pin output level in external trigger waiting state
0: Low level
1: High level

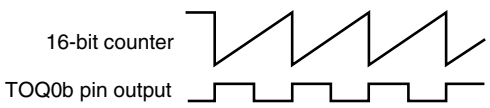
0: Disable TOQ02 pin output
1: Enable TOQ02 pin output

Setting of TOQ02 pin output level in external trigger waiting state
0: Low level
1: High level

0: Disable TOQ03 pin output
1: Enable TOQ03 pin output

Setting of TOQ03 pin output level in external trigger waiting state
0: Low level
1: High level

- When TQ0OLb bit = 0



- When TQ0OLb bit = 1

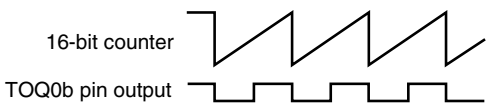
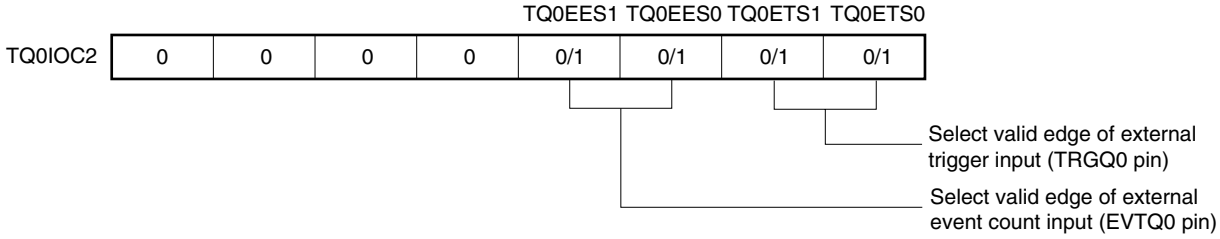


Figure 7-23. Setting of Registers in External Trigger Pulse Output Mode (3/3)

(d) TMQ0 I/O control register 2 (TQ0IOC2)



(e) TMQ0 counter read buffer register (TQ0CNT)

The value of the 16-bit counter can be read by reading the TQ0CNT register.

(f) TMQ0 capture/compare registers 0 to 3 (TQ0CCR0 to TQ0CCR3)

If D_0 is set to the TQ0CCR0 register, D_1 to the TQ0CCR1 register, D_2 to the TQ0CCR2 register, and D_3, to the TQ0CCR3 register, the cycle and active level of the PWM waveform are as follows.

PWM waveform cycle = $(D_0 + 1) \times$ Count clock cycle
TOQ01 pin PWM waveform active level width = $D_1 \times$ Count clock cycle
TOQ02 pin PWM waveform active level width = $D_2 \times$ Count clock cycle
TOQ03 pin PWM waveform active level width = $D_3 \times$ Count clock cycle

Remarks 1. TMQ0 I/O control register 1 (TQ0IOC1) and TMQ0 option register 0 (TQ0OPT0) are not used in the external trigger pulse output mode.

2. b = 1 to 3

(1) Operation flow in external trigger pulse output mode

Figure 7-24. Software Processing Flow in External Trigger Pulse Output Mode (1/2)

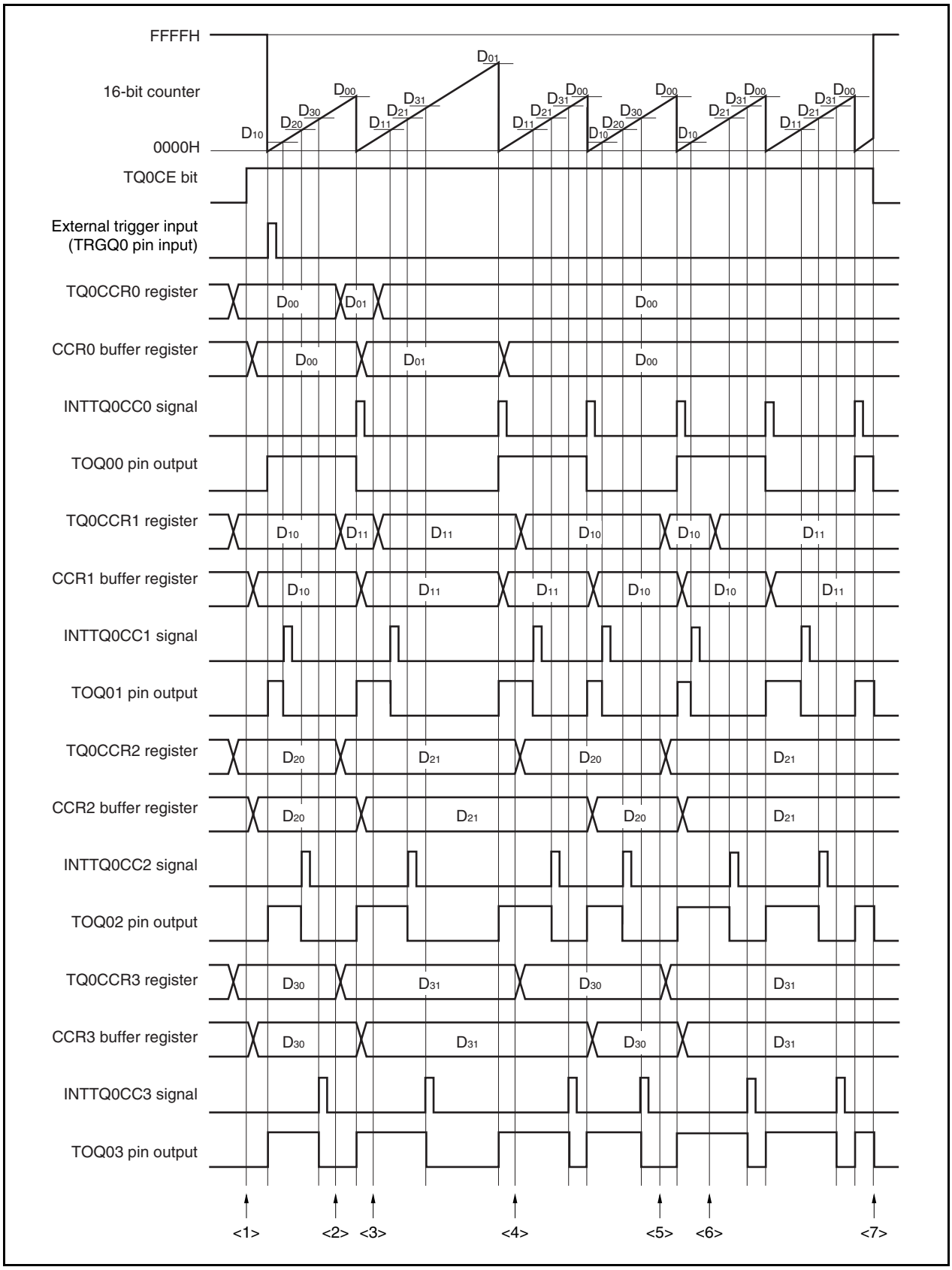
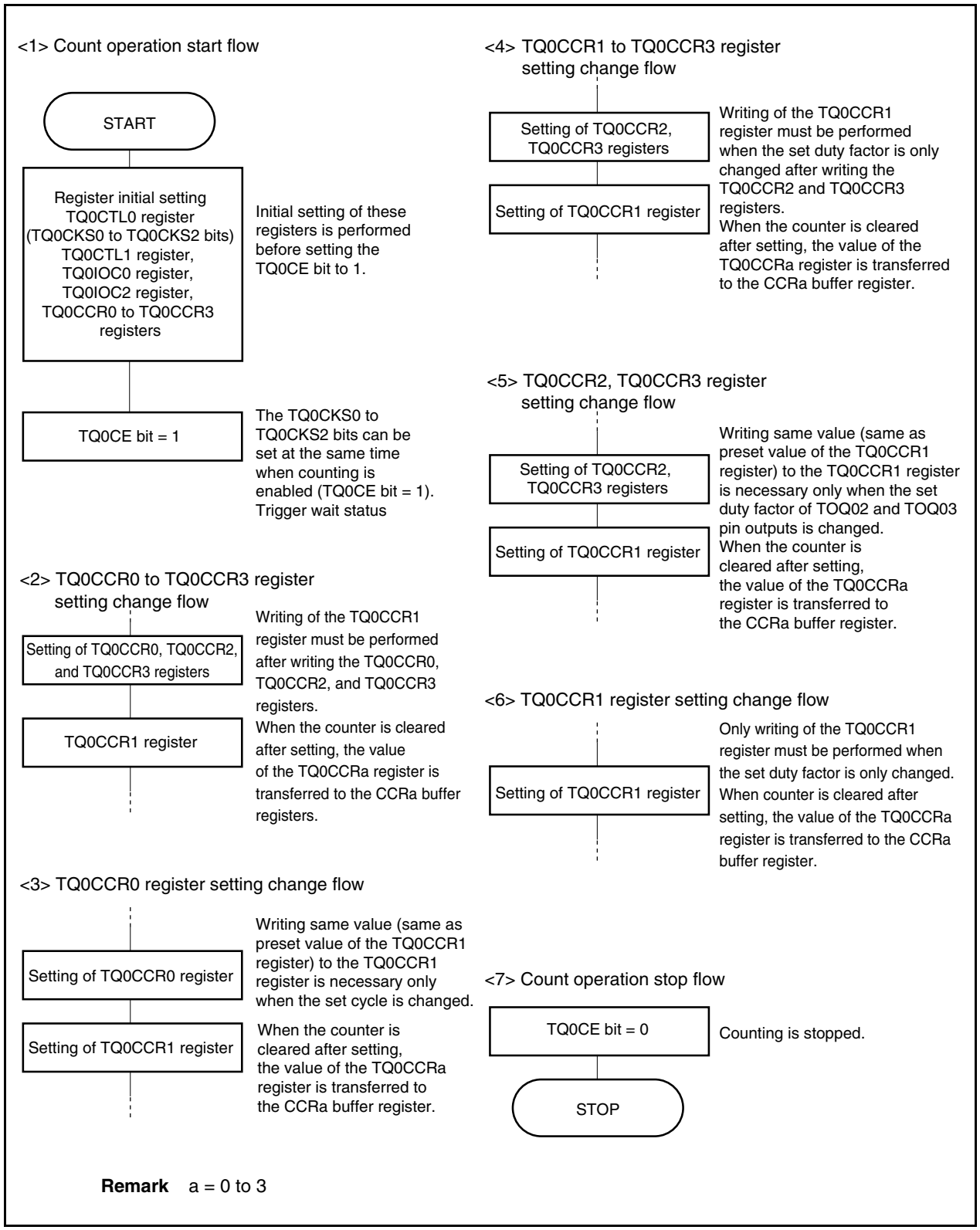


Figure 7-24. Software Processing Flow in External Trigger Pulse Output Mode (2/2)



Remark a = 0 to 3

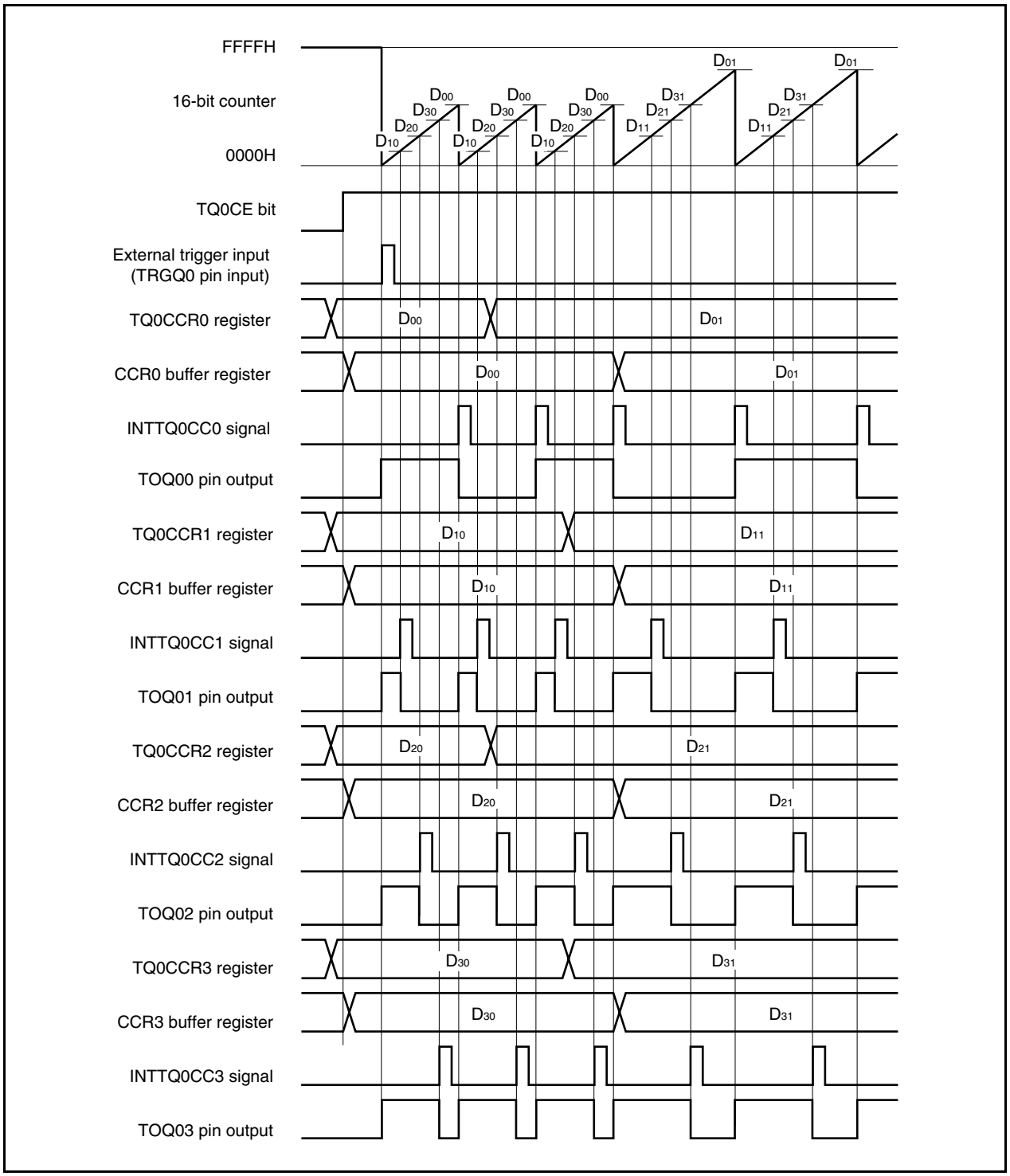
(2) External trigger pulse output mode operation timing

(a) Note on changing pulse width during operation

To change the PWM waveform while the counter is operating, write the TQ0CCR1 register last.

Rewrite the TQ0CCRb register after writing the TQ0CCR1 register after the INTTQ0CC0 signal is detected.

Remark b = 1 to 3



In order to transfer data from the TQ0CCRa register to the CCRa buffer register, the TQ0CCR1 register must be written.

To change both the cycle and active level width of the PWM waveform at this time, first set the cycle to the TQ0CCR0 register, set the active level width to the TQ0CCR2 and TQ0CCR3 registers, and then set an active level to the TQ0CCR1 register.

To change only the cycle of the PWM waveform, first set the cycle to the TQ0CCR0 register, and then write the same value (same as preset value of the TQ0CCR1 register) to the TQ0CCR1 register.

To change only the active level width (duty factor) of the PWM waveform, first set an active level to the TQ0CCR2 and TQ0CCR3 registers and then set an active level to the TQ0CCR1 register.

To change only the active level width (duty factor) of the PWM waveform output by the TOQ01 pin, only the TQ0CCR1 register has to be set.

To change only the active level width (duty factor) of the PWM waveform output by the TOQ02 and TOQ03 pins, first set an active level width to the TQ0CCR2 and TQ0CCR3 registers, and then write the same value (same as preset value of the TQ0CCR1 register) to the TQ0CCR1 register.

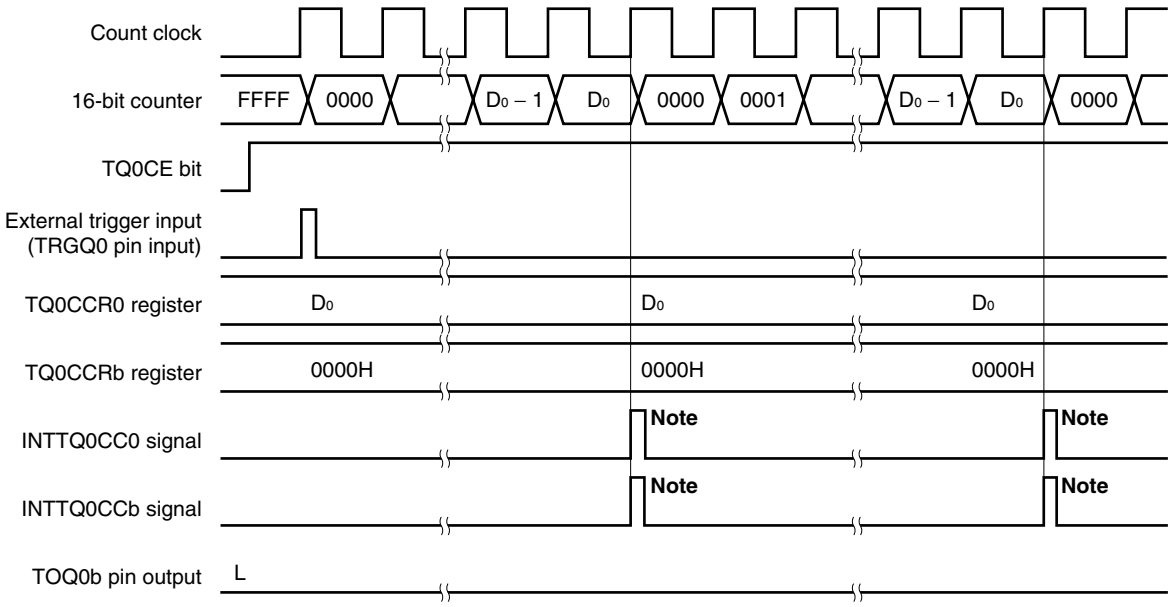
After data is written to the TQ0CCR1 register, the value written to the TQ0CCRa register is transferred to the CCRa buffer register in synchronization with clearing of the 16-bit counter, and is used as the value compared with the 16-bit counter.

To write the TQ0CCR0 to TQ0CCR3 registers again after writing the TQ0CCR1 register once, do so after the INTTQ0CC0 signal is generated. Otherwise, the value of the CCRa buffer register may become undefined because timing of transferring data from the TQ0CCRa register to the CCRa buffer register conflicts with writing the TQ0CCRa register.

Remark a = 0 to 3

(b) 0%/100% output of PWM waveform

<R> To output a 0% waveform, set the TQ0CCRb register to 0000H. The 16-bit counter is cleared to 0000H and the INTTQ0CC0 and INTTQ0CCb signals are generated at the next timing after a match between the count value of the 16-bit counter and the value of the CCR0 buffer register.

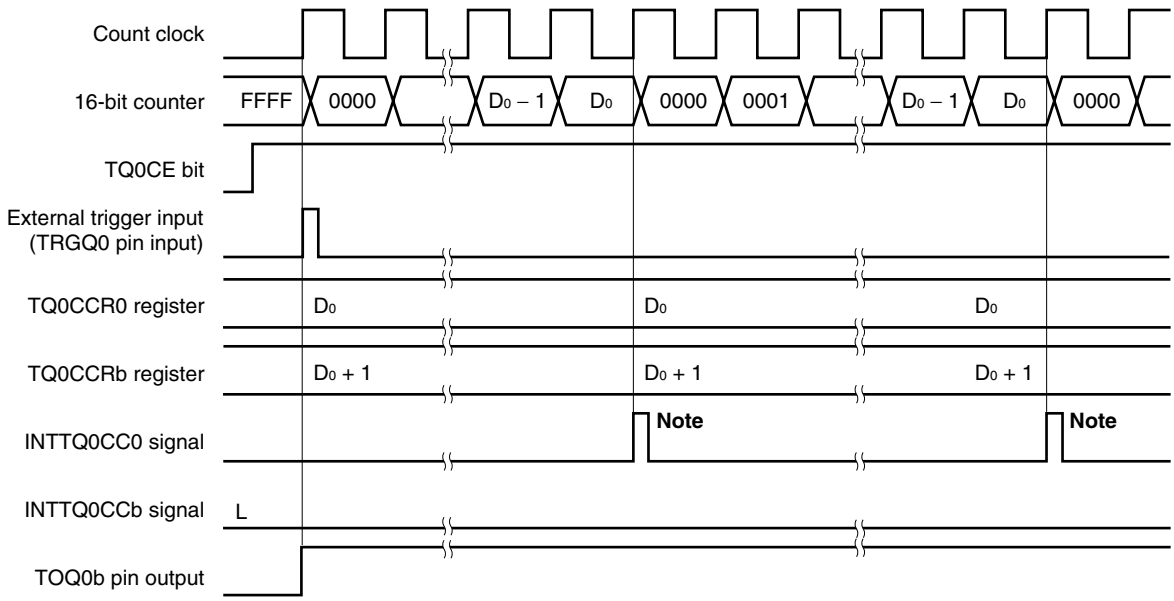


Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

To output a 100% waveform, set a value of (set value of TQ0CCR0 register + 1) to the TQ0CCRb register. If the set value of the TQ0CCR0 register is FFFFH, 100% output cannot be produced.

<R>

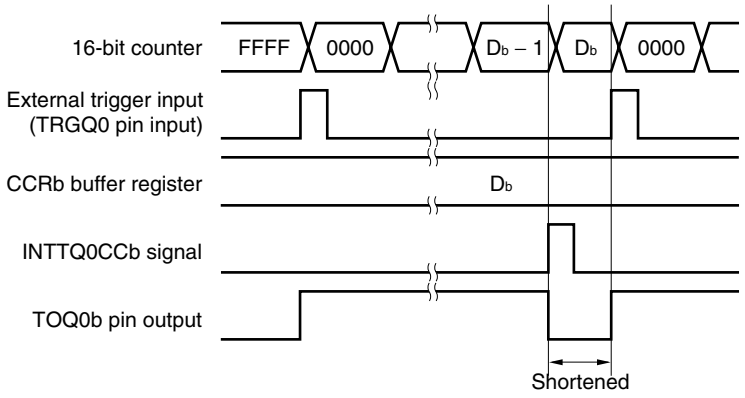


Note Actually, the timing is delayed by one operating clock (f_{XX}).

Remark b = 1 to 3

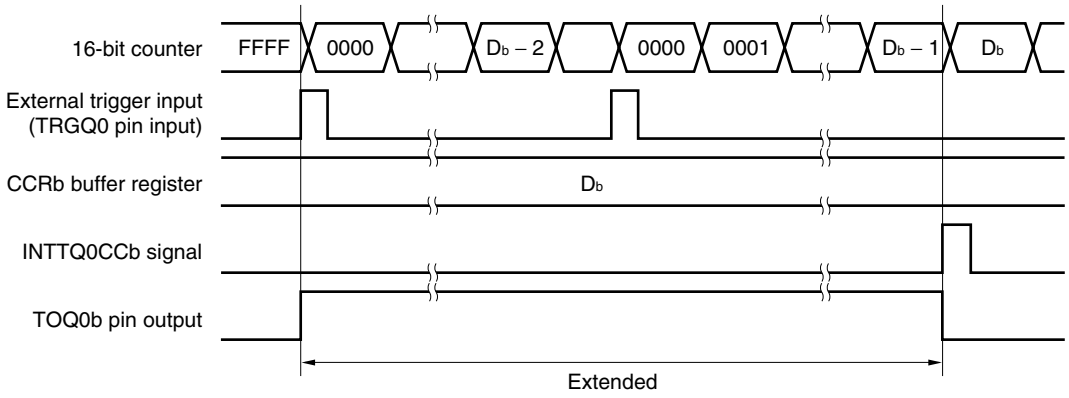
(c) Conflict between trigger detection and match with CCRb buffer register

If the trigger is detected immediately after the INTTQ0CCb signal is generated, the 16-bit counter is immediately cleared to 0000H, the output signal of the TOQ0b pin is asserted, and the counter continues counting. Consequently, the inactive period of the PWM waveform is shortened.



Remark b = 1 to 3

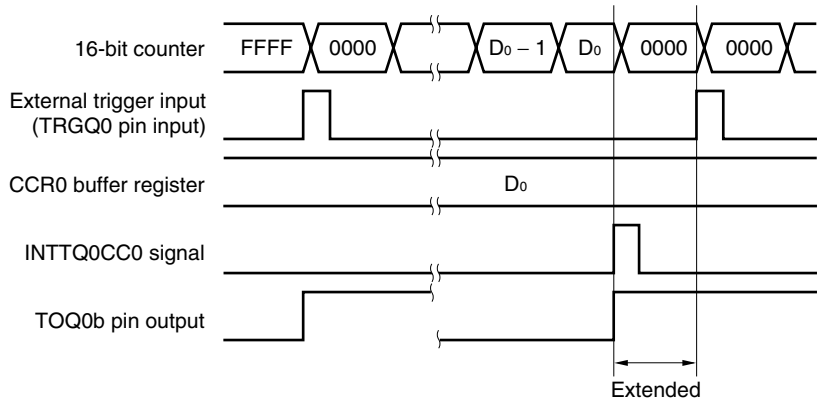
If the trigger is detected immediately before the INTTQ0CCb signal is generated, the INTTQ0CCb signal is not generated, and the 16-bit counter is cleared to 0000H and continues counting. The output signal of the TOQ0b pin remains active. Consequently, the active period of the PWM waveform is extended.



Remark b = 1 to 3

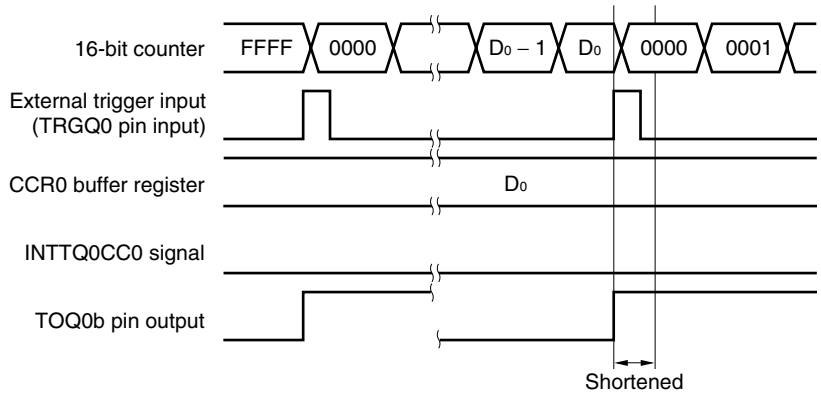
(d) Conflict between trigger detection and match with CCR0 buffer register

If the trigger is detected immediately after the INTTQ0CC0 signal is generated, the 16-bit counter is cleared to 0000H and continues counting up. Therefore, the active period of the TOQ0b pin is extended by time from generation of the INTTQ0CC0 signal to trigger detection.



Remark b = 1 to 3

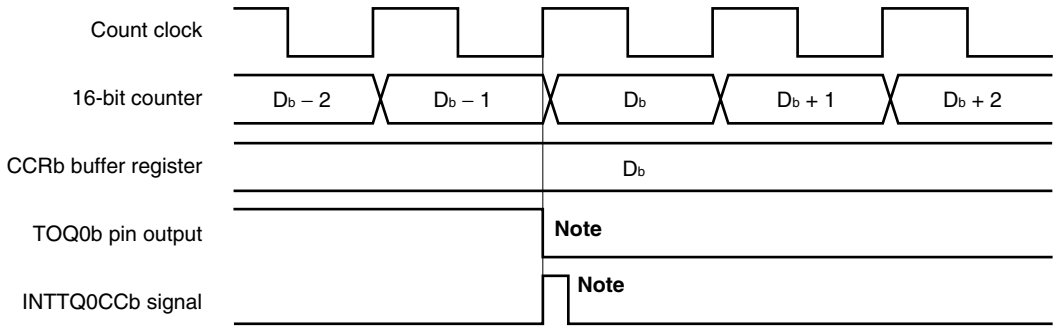
If the trigger is detected immediately before the INTTQ0CC0 signal is generated, the INTTQ0CC0 signal is not generated. The 16-bit counter is cleared to 0000H, the TOQ0b pin is asserted, and the counter continues counting. Consequently, the inactive period of the PWM waveform is shortened.



Remark b = 1 to 3

(e) Generation timing of compare match interrupt request signal (INTTQ0CCb)

The timing of generation of the INTTQ0CCb signal in the external trigger pulse output mode differs from the timing of INTTQ0CCb signals in other modes; the INTTQ0CCb signal is generated when the count value of the 16-bit counter matches the value of the CCRb buffer register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

Usually, the INTTQ0CCb signal is generated in synchronization with the next count up after the count value of the 16-bit counter matches the value of the CCRb buffer register.

In the external trigger pulse output mode, however, it is generated one clock earlier. This is because the timing is changed to match the timing of changing the output signal of the TOQ0b pin.

7.6.4 One-shot pulse output mode (TQ0MD2 to TQ0MD0 bits = 011)

This mode is valid only in TMQ0.

In the one-shot pulse output mode, 16-bit timer/event counter Q waits for a trigger when the TQ0CTL0.TQ0CE bit is set to 1. When a valid edge of the external trigger input (TRGQ0) is detected, 16-bit timer/event counter Q starts counting, and outputs a one-shot pulse from the TOQ01 to TOQ03 pins. The TOQ00 pin outputs the active level while the 16-bit counter is counting, and the inactive level when the counter is stopped (waiting for a trigger).

The 16-bit timer/event counter Q also outputs a pulse by generating a software trigger instead of an external trigger input.

Figure 7-25. Configuration in One-Shot Pulse Output Mode

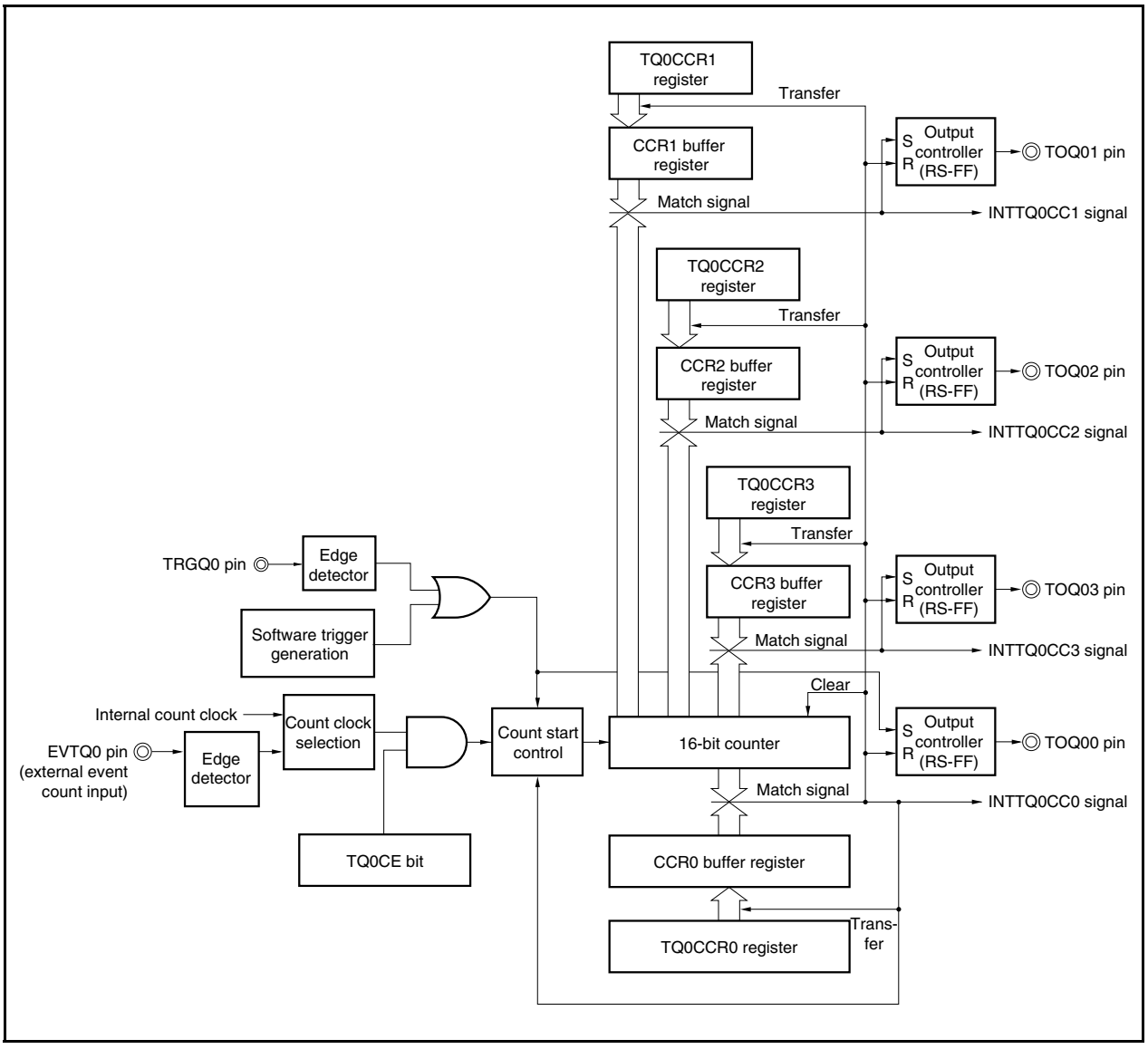
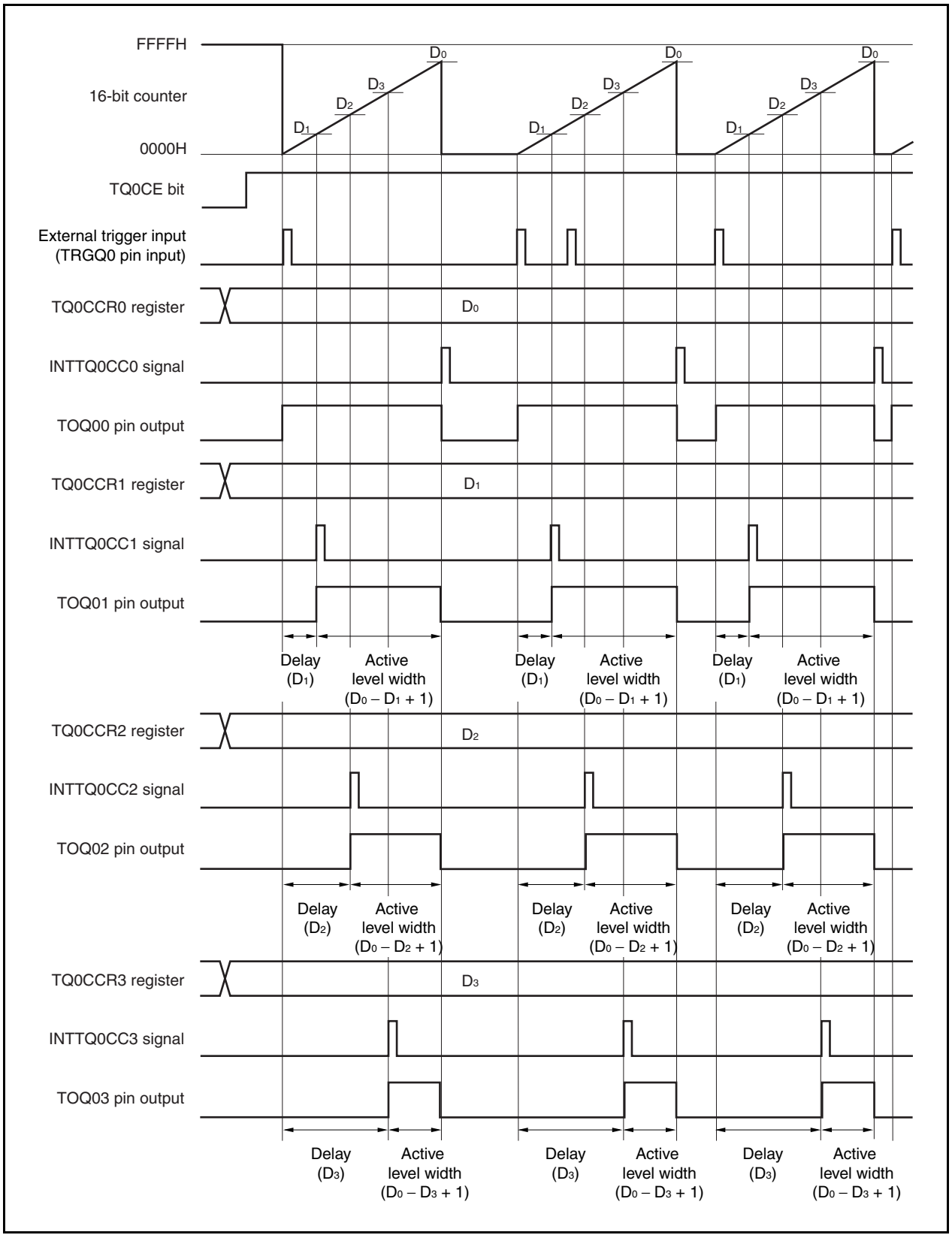


Figure 7-26. Basic Timing in One-Shot Pulse Output Mode



When the TQ0CE bit is set to 1, 16-bit timer/event counter Q waits for a trigger. When the trigger is generated, the 16-bit counter is cleared from FFFFH to 0000H, starts counting, and outputs a one-shot pulse from the TOQ0b pin. After the one-shot pulse is output, the 16-bit counter is cleared to 0000H, stops counting, and waits for a trigger. When the trigger is generated again, the 16-bit counter starts counting from 0000H. If a trigger is generated again while the one-shot pulse is being output, it is ignored.

The output delay period and active level width of the one-shot pulse can be calculated as follows.

Output delay period = (Set value of TQ0CCRb register) × Count clock cycle

Active level width = (Set value of TQ0CCR0 register − Set value of TQ0CCRb register + 1) × Count clock cycle

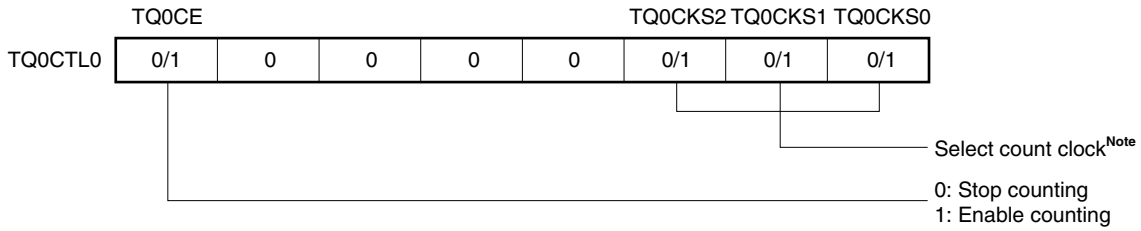
The compare match interrupt request signal INTTQ0CC0 is generated when the 16-bit counter counts after its count value matches the value of the CCR0 buffer register. The compare match interrupt request signal INTTQ0CCb is generated when the count value of the 16-bit counter matches the value of the CCRb buffer register.

The valid edge of the external trigger input (TRGQ0) and setting the software trigger (TQ0CTL1.TQ0EST bit) to 1 are used as the trigger.

Remark b = 1 to 3

Figure 7-27. Register Setting in One-Shot Pulse Output Mode (1/3)

(a) TMQ0 control register 0 (TQ0CTL0)



Note The setting is invalid when the TQ0CTL1.TQ0EEE bit = 1.

(b) TMQ0 control register 1 (TQ0CTL1)

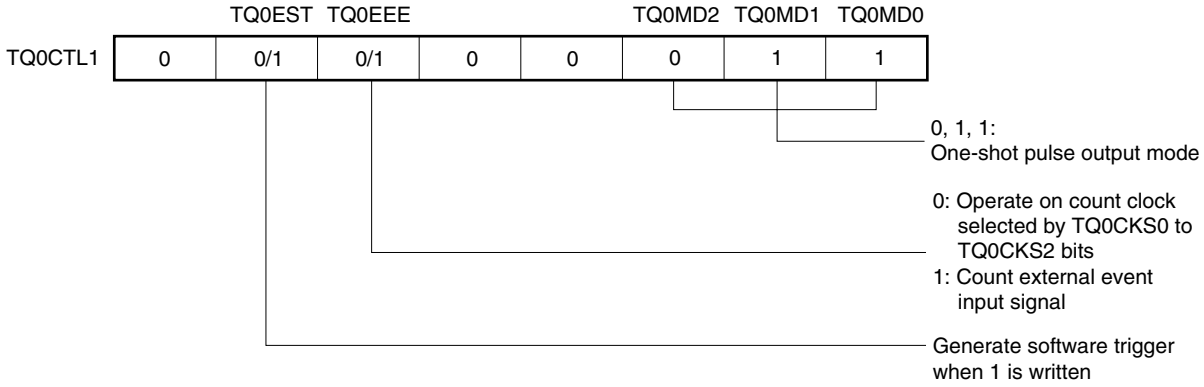
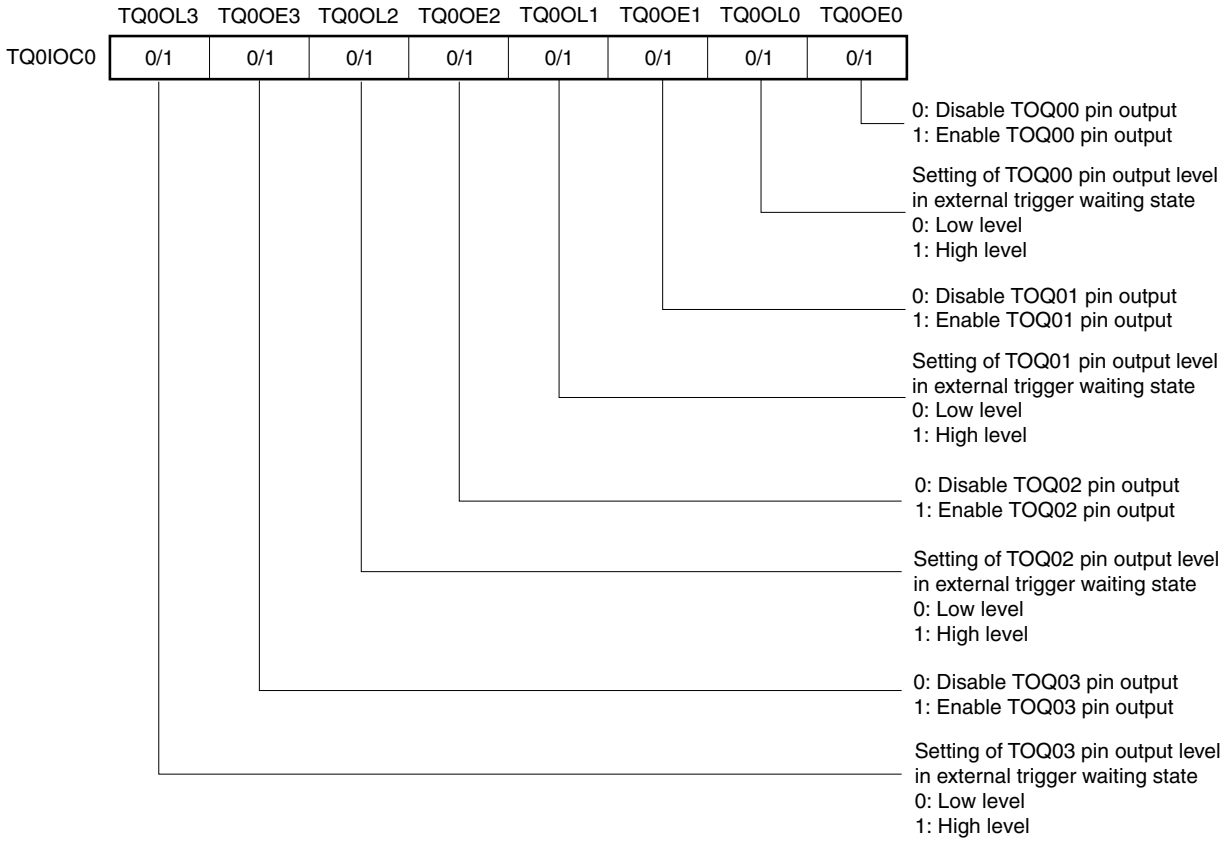
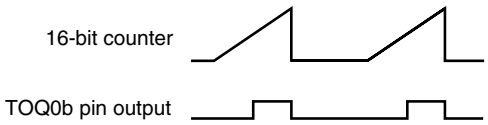


Figure 7-27. Register Setting in One-Shot Pulse Output Mode (2/3)

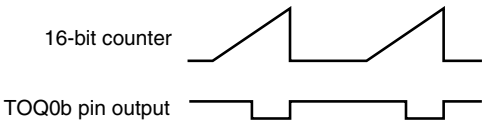
(c) TMQ0 I/O control register 0 (TQ0IOC0)



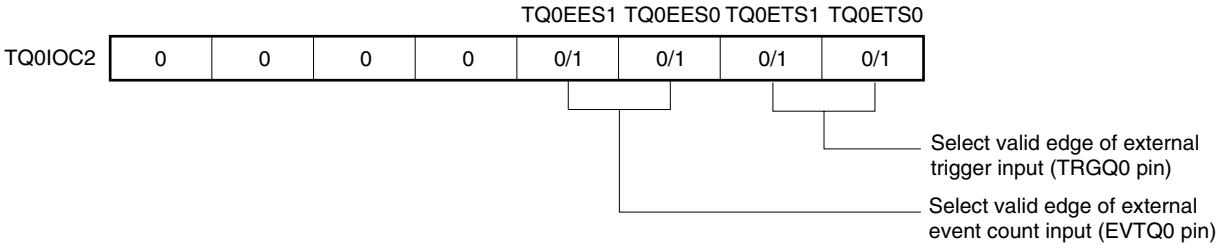
• When TQ0OLb bit = 0



• When TQ0OLb bit = 1



(d) TMQ0 I/O control register 2 (TQ0IOC2)



(e) TMQ0 counter read buffer register (TQ0CNT)

The value of the 16-bit counter can be read by reading the TQ0CNT register.

Figure 7-27. Register Setting in One-Shot Pulse Output Mode (3/3)

(f) TMQ0 capture/compare registers 0 to 3 (TQ0CCR0 to TQ0CCR3)

If D_0 is set to the TQ0CCR0 register and D_b to the TQ0CCRb register, the active level width and output delay period of the one-shot pulse are as follows.

<R> Active level width = $(D_0 - D_b + 1) \times$ Count clock cycle

Output delay period = $D_b \times$ Count clock cycle

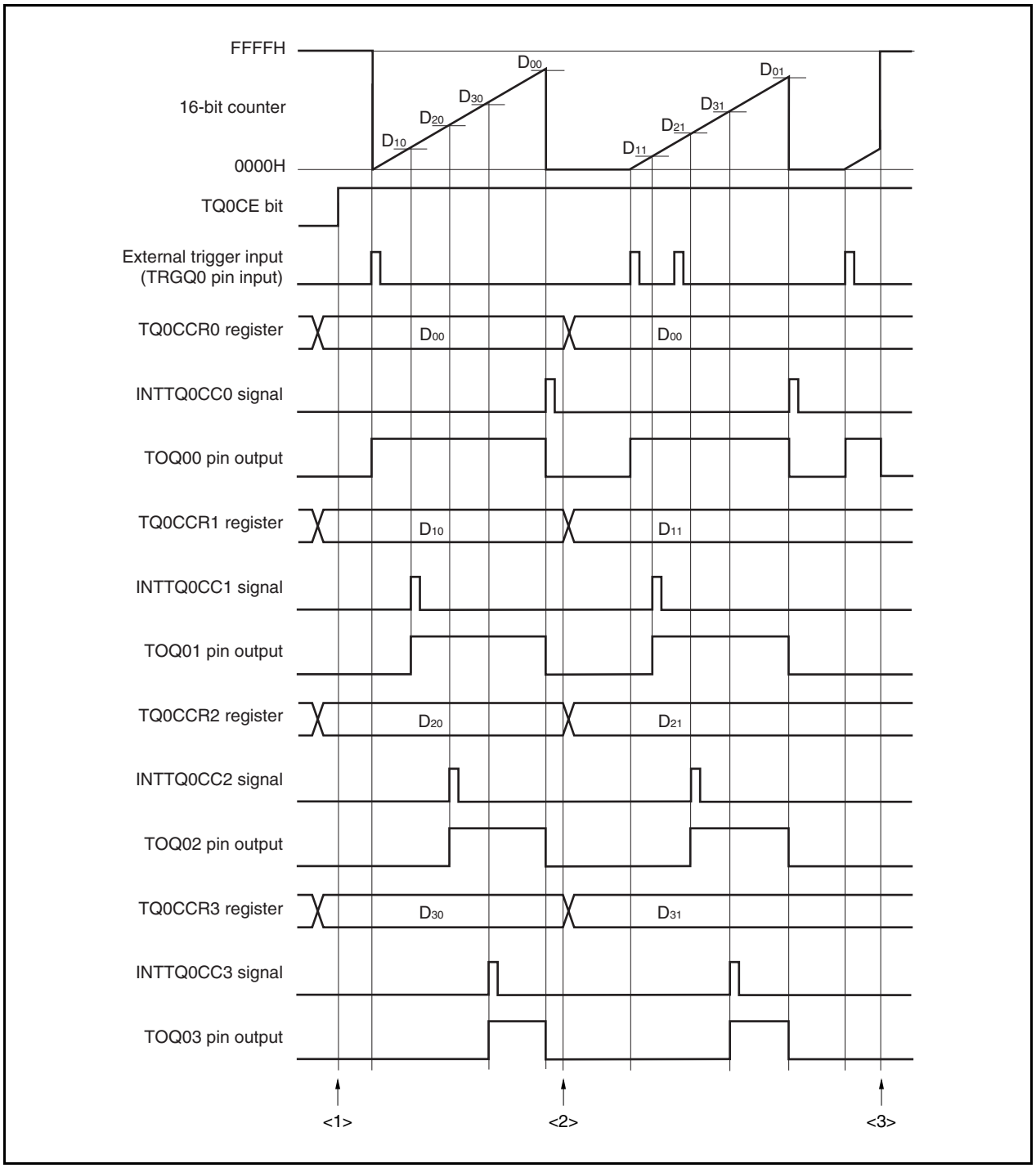
<R> **Caution One-shot pulses are not output even in the one-shot pulse output mode, if the value set in the TQ0CCRb register is greater than that set in the TQ0CCR0 register.**

Remarks 1. TMQ0 I/O control register 1 (TQ0IOC1) and TMQ0 option register 0 (TQ0OPT0) are not used in the one-shot pulse output mode.

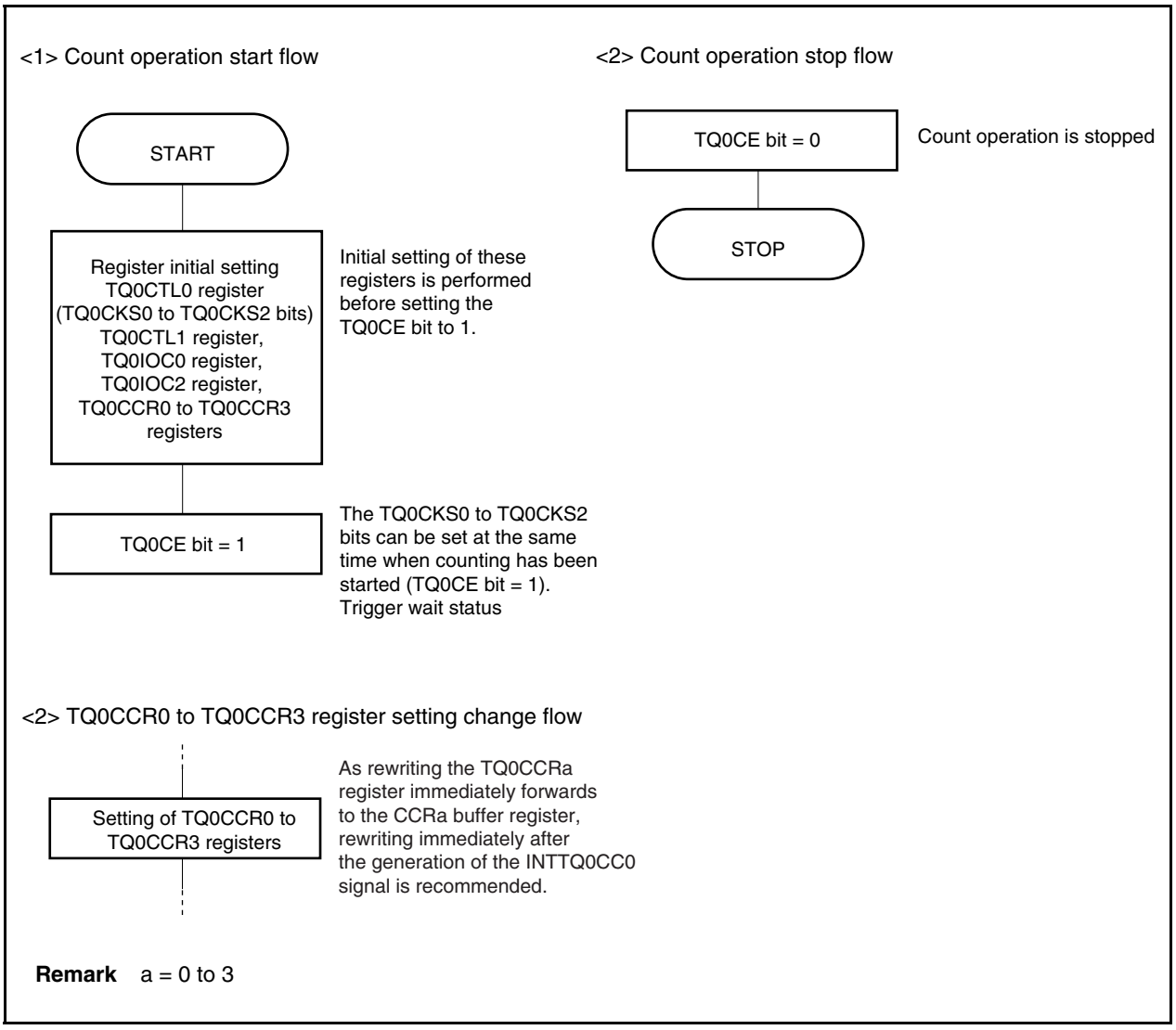
2. b = 1 to 3

(1) Operation flow in one-shot pulse output mode

<R> **Figure 7-28. Software Processing Flow in One-Shot Pulse Output Mode (1/2)**



<R> **Figure 7-28. Software Processing Flow in One-Shot Pulse Output Mode (2/2)**

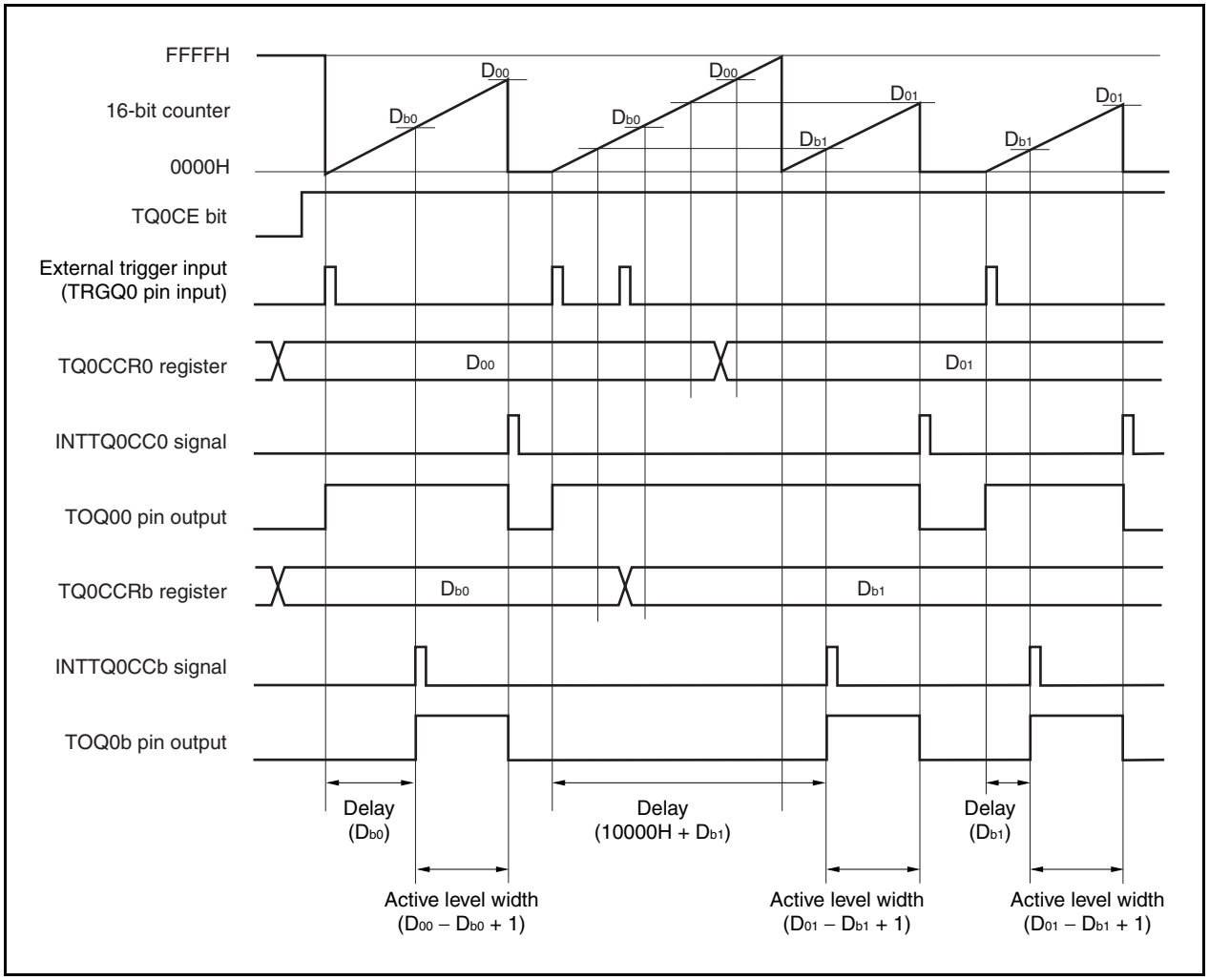


Remark a = 0 to 3

(2) Operation timing in one-shot pulse output mode

(a) Note on rewriting TQ0CCRa register

If the value of the TQ0CCRa register is rewritten to a smaller value during counting, the 16-bit counter may overflow. When the overflow may occur, stop counting once, and then change the set value.



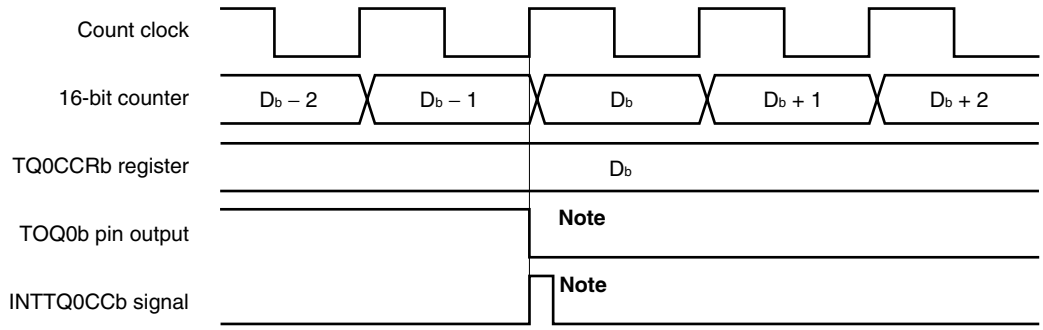
When the TQ0CCR0 register is rewritten from D_{00} to D_{01} and the TQ0CCRb register from D_{b0} to D_{b1} where $D_{00} > D_{01}$ and $D_{b0} > D_{b1}$, if the TQ0CCRb register is rewritten when the count value of the 16-bit counter is greater than D_{b1} and less than D_{b0} and if the TQ0CCR0 register is rewritten when the count value is greater than D_{01} and less than D_{00}, each set value is reflected as soon as the register has been rewritten and compared with the count value. The counter counts up to FFFFH and then counts up again from 0000H. When the count value matches D_{b1}, the counter generates the INTTQ0CCb signal and asserts the TOQ0b pin. When the count value matches D_{01}, the counter generates the INTTQ0CC0 signal, deasserts the TOQ0b pin, and stops counting.

Therefore, the counter may output a pulse with a delay period or active period different from that of the one-shot pulse that is originally expected.

Remark a = 0 to 3, b = 1 to 3

(b) Generation timing of compare match interrupt request signal (INTTQ0CCb)

The generation timing of the INTTQ0CCb signal in the one-shot pulse output mode is different from INTTQ0CCb signals in other modes; the INTTQ0CCb signal is generated when the count value of the 16-bit counter matches the value of the TQ0CCRb register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

Usually, the INTTQ0CCb signal is generated when the 16-bit counter counts up next time after its count value matches the value of the TQ0CCRb register.

In the one-shot pulse output mode, however, it is generated one clock earlier. This is because the timing is changed to match the change timing of the TOQ0b pin.

7.6.5 PWM output mode (TQ0MD2 to TQ0MD0 bits = 100)

This mode is valid only in TMQ0.

In the PWM output mode, a PWM waveform is output from the TOQ01 to TOQ03 pins when the TQ0CTL0.TQ0CE bit is set to 1.

In addition, a PWM waveform with a duty factor of 50% with the set value of the TQ0CCR0 register + 1 as half its cycle is output from the TOQ00 pin.

Figure 7-29. Configuration in PWM Output Mode

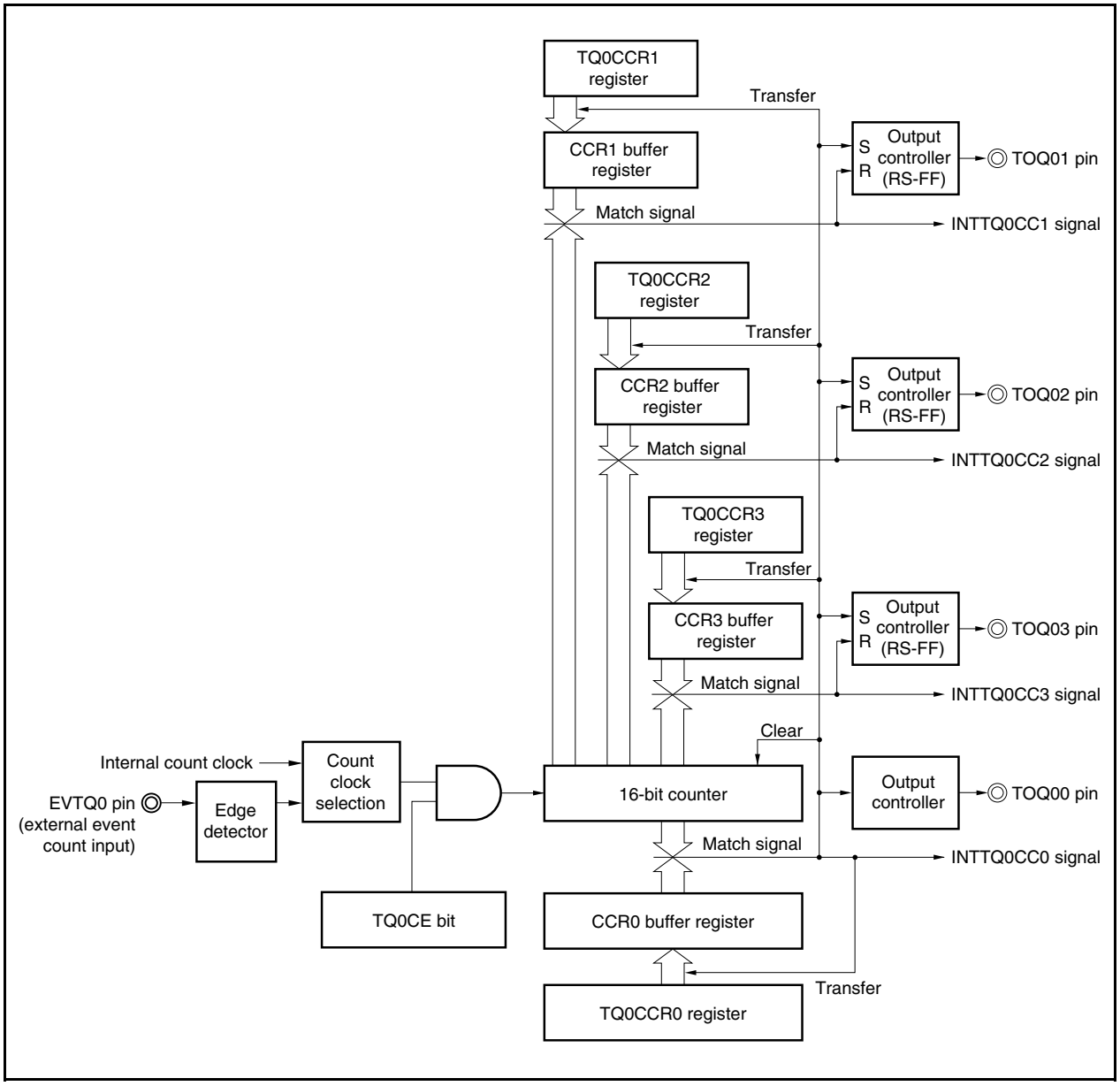
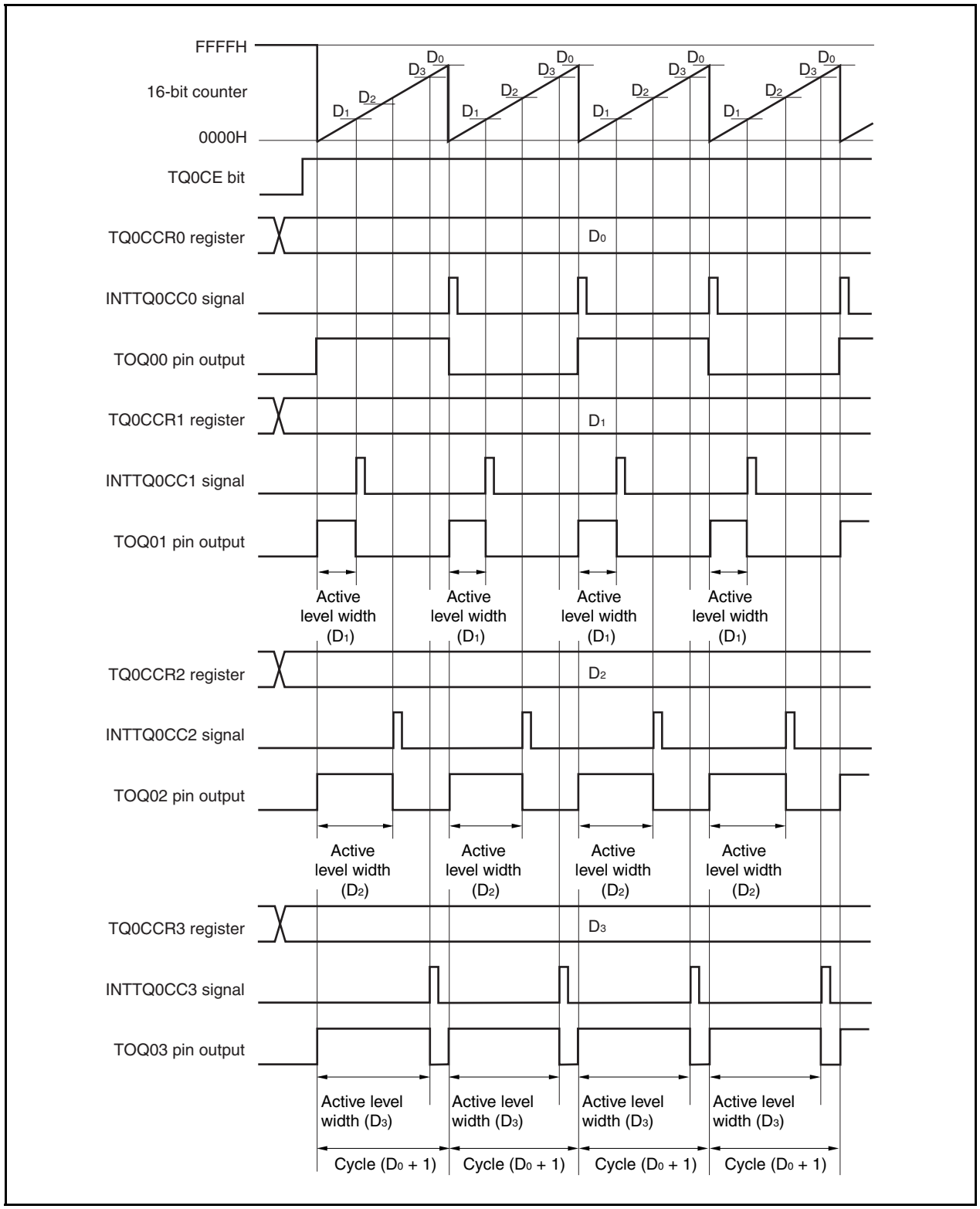


Figure 7-30. Basic Timing in PWM Output Mode



When the TQ0CE bit is set to 1, the 16-bit counter is cleared from FFFFH to 0000H, starts counting, and outputs PWM waveform from the TOQ0b pin.

The active level width, cycle, and duty factor of the PWM waveform can be calculated as follows.

Active level width = (Set value of TQ0CCRb register) × Count clock cycle
Cycle = (Set value of TQ0CCR0 register + 1) × Count clock cycle
Duty factor = (Set value of TQ0CCRb register)/(Set value of TQ0CCR0 register + 1)

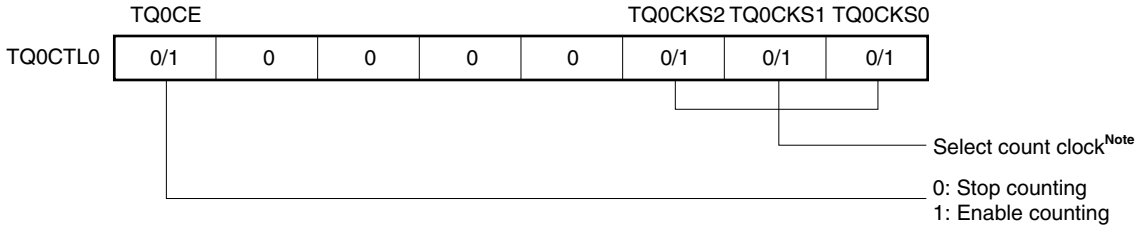
The PWM waveform can be changed by rewriting the TQ0CCRa register while the counter is operating. The newly written value is reflected when the count value of the 16-bit counter matches the value of the CCR0 buffer register and the 16-bit counter is cleared to 0000H.

The compare match interrupt request signal INTTQ0CC0 is generated when the 16-bit counter counts next time after its count value matches the value of the CCR0 buffer register, and the 16-bit counter is cleared to 0000H. The compare match interrupt request signal INTTQ0CCb is generated when the count value of the 16-bit counter matches the value of the CCRb buffer register.

Remark a = 0 to 3
b = 1 to 3

Figure 7-31. Register Setting in PWM Output Mode (1/3)

(a) TMQ0 control register 0 (TQ0CTL0)



Note The setting is invalid when the TQ0CTL1.TQ0EEE bit = 1.

(b) TMQ0 control register 1 (TQ0CTL1)

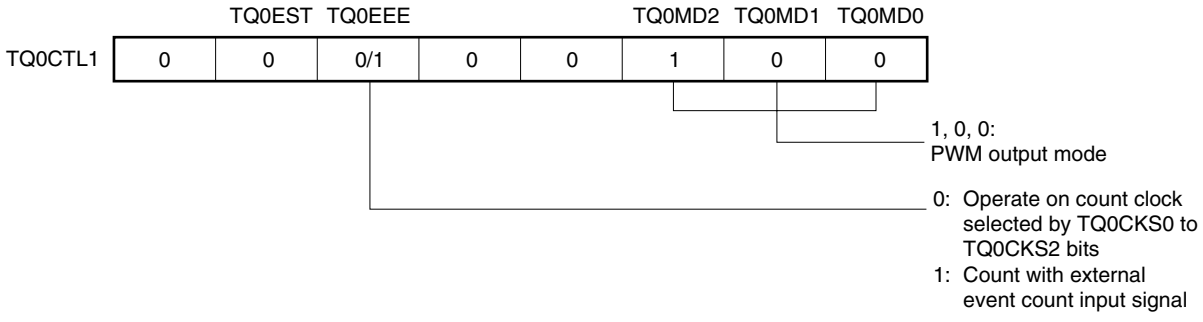
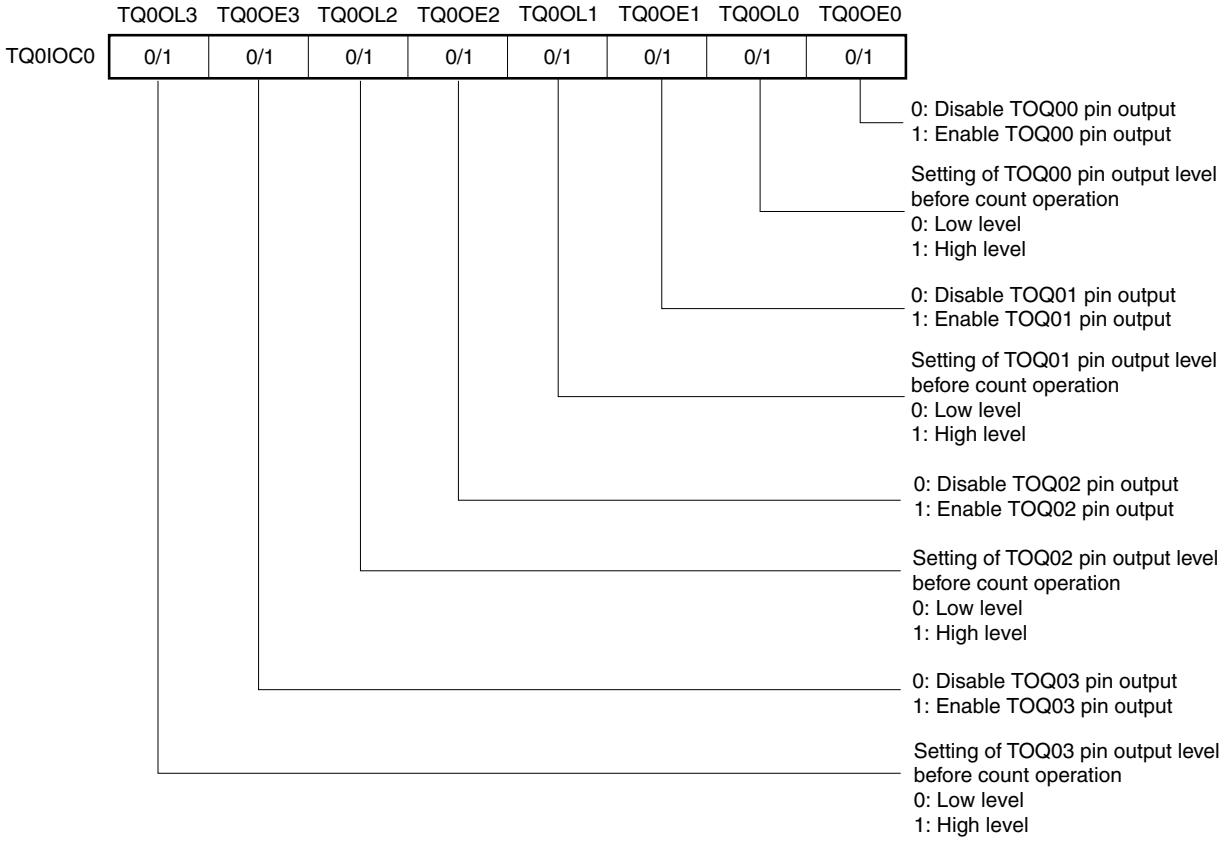
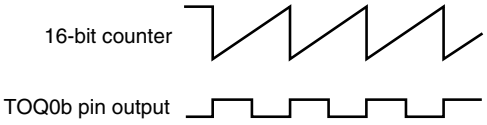


Figure 7-31. Register Setting in PWM Output Mode (2/3)

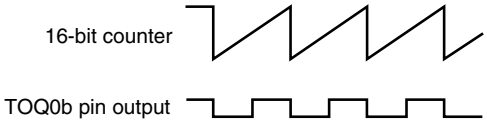
(c) TMQ0 I/O control register 0 (TQ0IOC0)



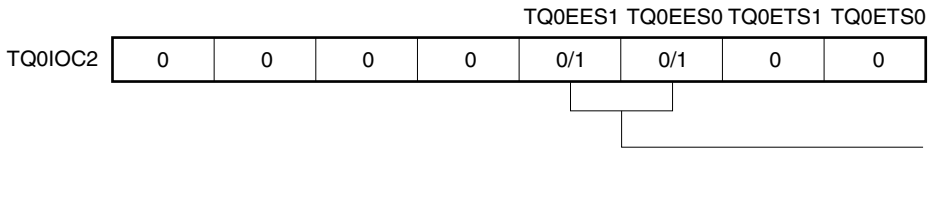
- When TQ0OLb bit = 0



- When TQ0OLb bit = 1



(d) TMQ0 I/O control register 2 (TQ0IOC2)



(e) TMQ0 counter read buffer register (TQ0CNT)

The value of the 16-bit counter can be read by reading the TQ0CNT register.

Figure 7-31. Register Setting in PWM Output Mode (3/3)

(f) TMQ0 capture/compare registers 0 to 3 (TQ0CCR0 to TQ0CCR3)

If D_0 is set to the TQ0CCR0 register and D_b to the TQ0CCRb register, the cycle and active level of the PWM waveform are as follows.

PWM waveform cycle = $(D_0 + 1) \times$ Count clock cycle

PWM waveform active level width = $D_b \times$ Count clock cycle

Remarks 1. TMQ0 I/O control register 1 (TQ0IOC1) and TMQ0 option register 0 (TQ0OPT0) are not used in the PWM output mode.

2. b = 1 to 3

(1) Operation flow in PWM output mode

Figure 7-32. Software Processing Flow in PWM Output Mode (1/2)

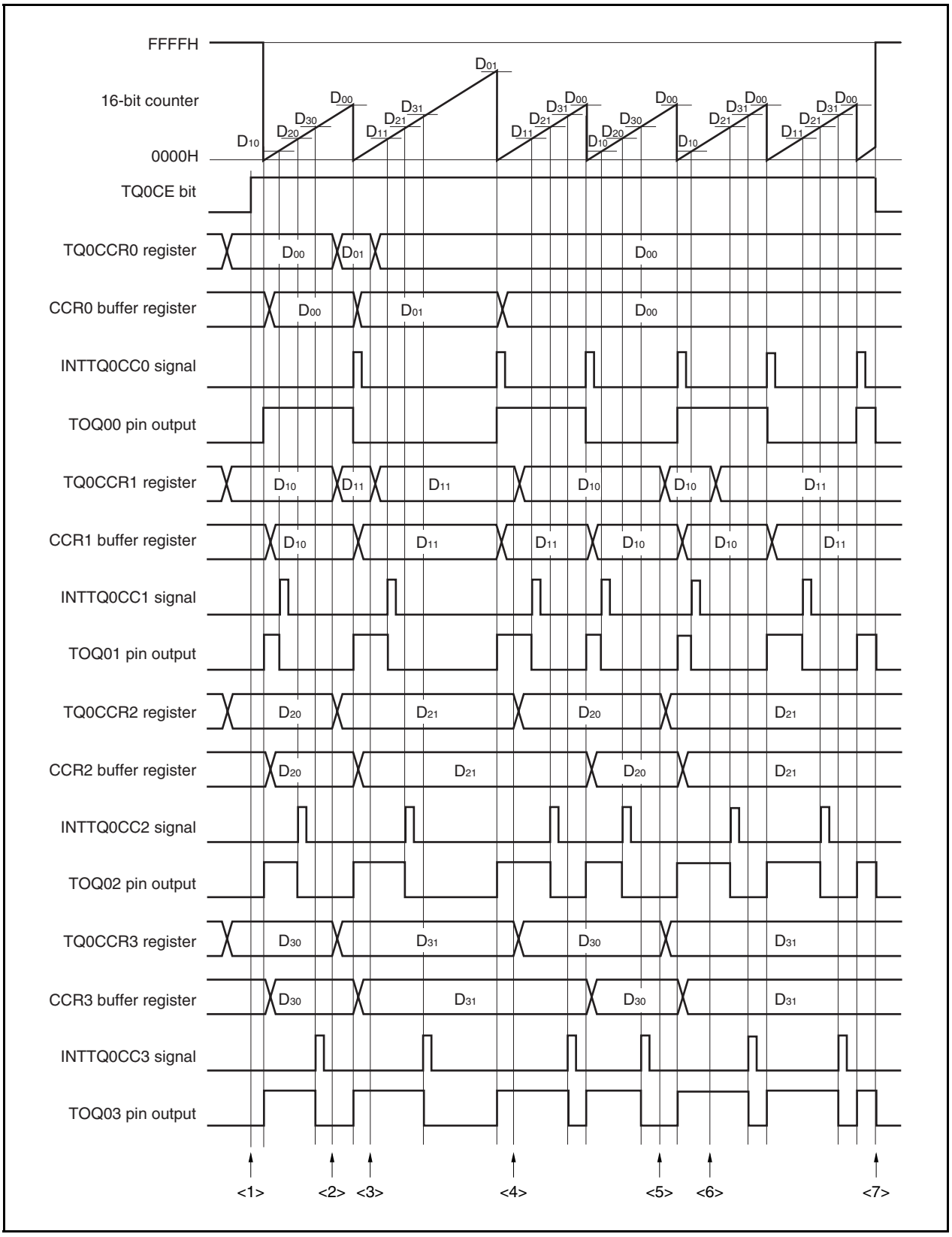
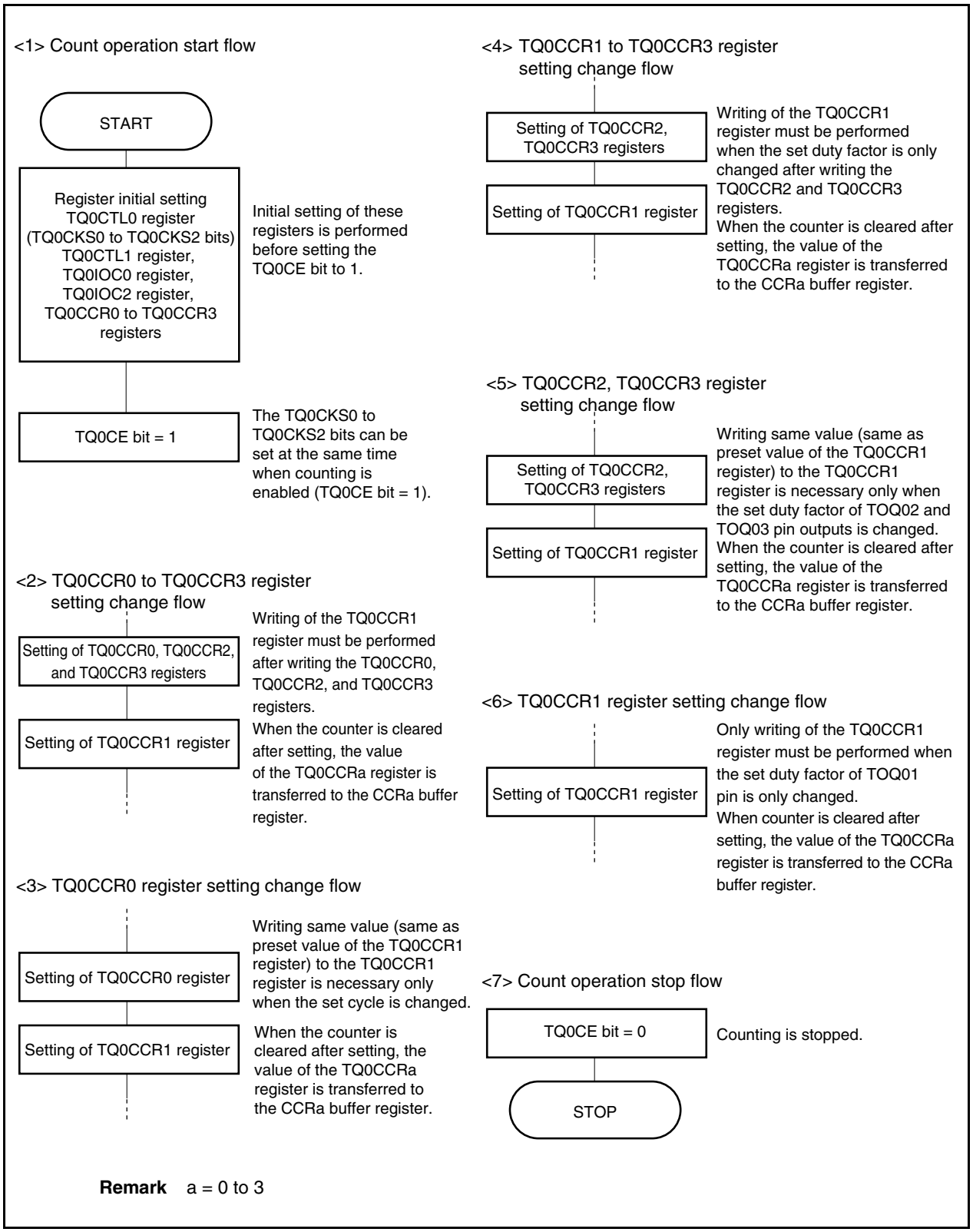


Figure 7-32. Software Processing Flow in PWM Output Mode (2/2)



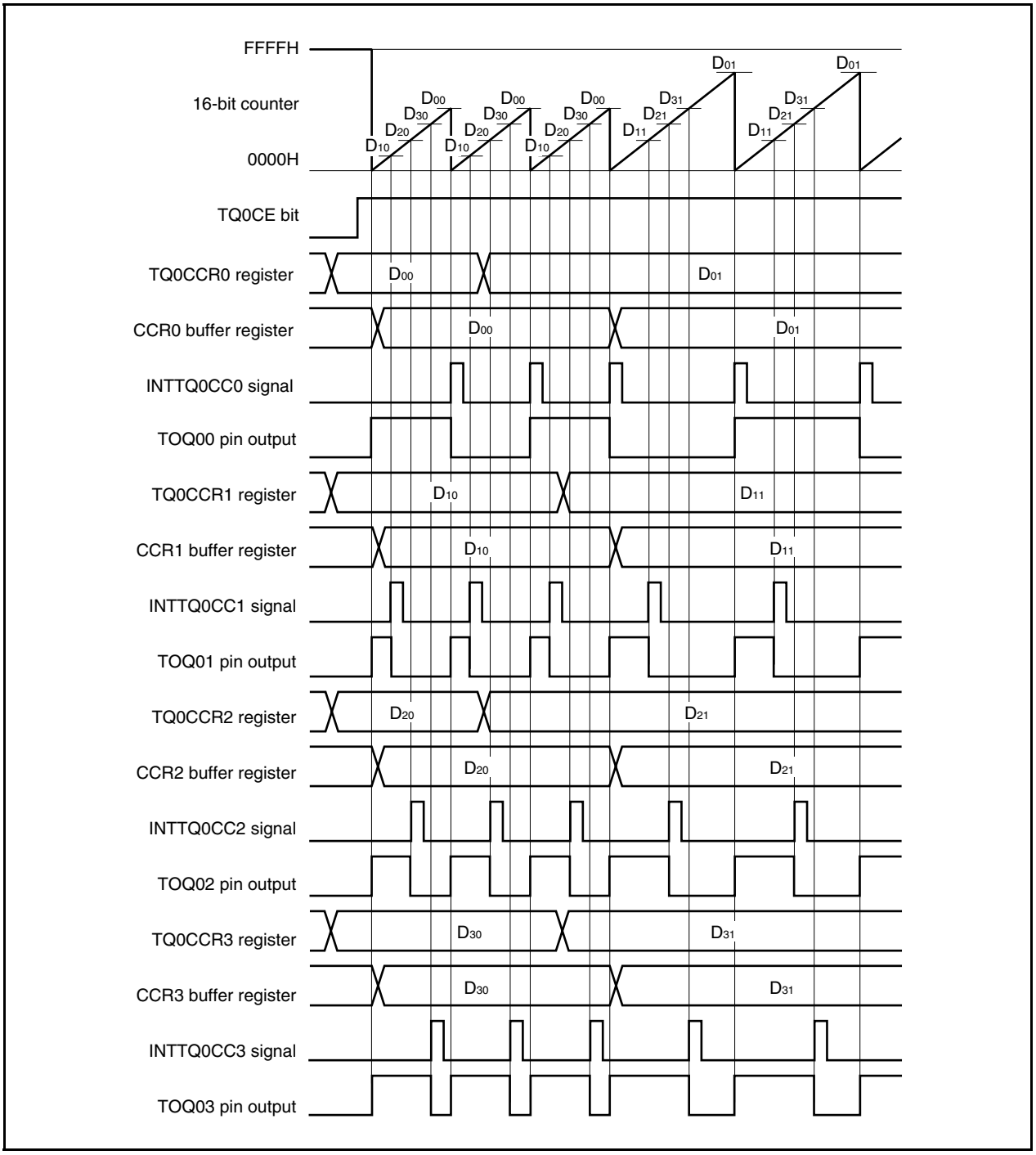
Remark a = 0 to 3

(2) PWM output mode operation timing

(a) Changing pulse width during operation

To change the PWM waveform while the counter is operating, write the TQ0CCR1 register last.

<R> Rewrite the TQ0CCRa register after writing the TQ0CCR1 register after the INTTQ0CC0 signal is detected.



To transfer data from the TQ0CCRa register to the CCRa buffer register, the TQ0CCR1 register must be written.

To change both the cycle and active level of the PWM waveform at this time, first set the cycle to the TQ0CCR0 register, set the active level width to the TQ0CCR2 and TQ0CCR3 registers, and then set an active level width to the TQ0CCR1 register.

To change only the cycle of the PWM waveform, first set a cycle to the TQ0CCR0 register, and then write the same value (same as preset value of the TQ0CCR1 register) to the TQ0CCR1 register.

To change only the active level width (duty factor) of PWM wave, first set the active level to the TQ0CCR2 and TQ0CCR3 registers, and then set an active level to the TQ0CCR1 register.

To change only the active level width (duty factor) of the PWM waveform output by the TOQ01 pin, only the TQ0CCR1 register has to be set.

To change only the active level width (duty factor) of the PWM waveform output by the TOQ02 and TOQ03 pins, first set an active level width to the TQ0CCR2 and TQ0CCR3 registers, and then write the same value (same as preset value of the TQ0CCR1 register) to the TQ0CCR1 register.

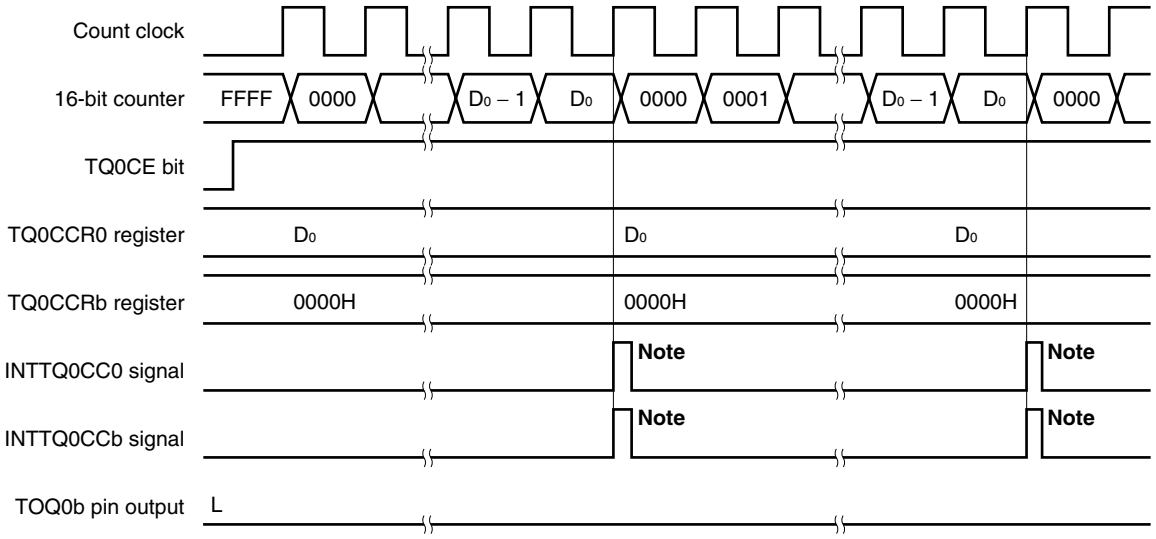
After the TQ0CCR1 register is written, the value written to the TQ0CCRa register is transferred to the CCRa buffer register in synchronization with the timing of clearing the 16-bit counter, and is used as a value to be compared with the value of the 16-bit counter.

To write the TQ0CCR0 to TQ0CCR3 registers again after writing the TQ0CCR1 register once, do so after the INTTQ0CC0 signal is generated. Otherwise, the value of the CCRa buffer register may become undefined because the timing of transferring data from the TQ0CCRa register to the CCRa buffer register conflicts with writing the TQ0CCRa register.

Remark a = 0 to 3

(b) 0%/100% output of PWM waveform

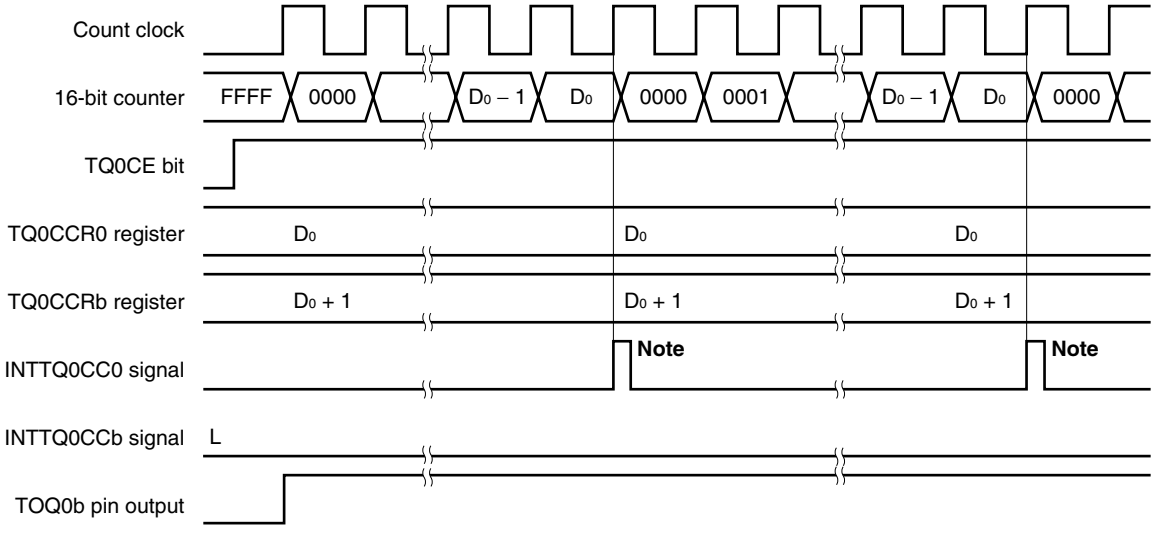
To output a 0% waveform, set the TQ0CCRb register to 0000H. The 16-bit counter is cleared to 0000H and the INTTQ0CC0 and INTTQ0CCb signals are generated at the next timing after a match between the count value of the 16-bit counter and the value of the CCR0 buffer register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

To output a 100% waveform, set a value of (set value of TQ0CCR0 register + 1) to the TQ0CCRb register. If the set value of the TQ0CCR0 register is FFFFH, 100% output cannot be produced.

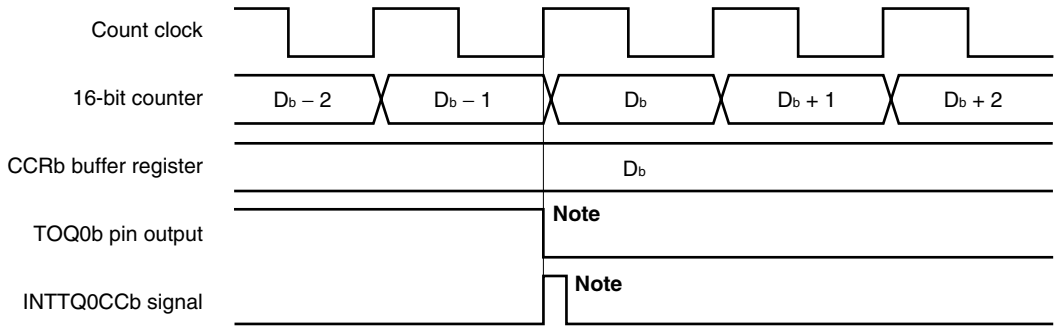


Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

(c) Generation timing of compare match interrupt request signal (INTTQ0CCb)

The timing of generation of the INTTQ0CCb signal in the PWM output mode differs from the timing of INTTQ0CCb signals in other modes; the INTTQ0CCb signal is generated when the count value of the 16-bit counter matches the value of the TQ0CCRb register.



Note Actually, the timing is delayed by one operating clock (fxx).

Remark b = 1 to 3

Usually, the INTTQ0CCb signal is generated in synchronization with the next counting up after the count value of the 16-bit counter matches the value of the TQ0CCRb register.

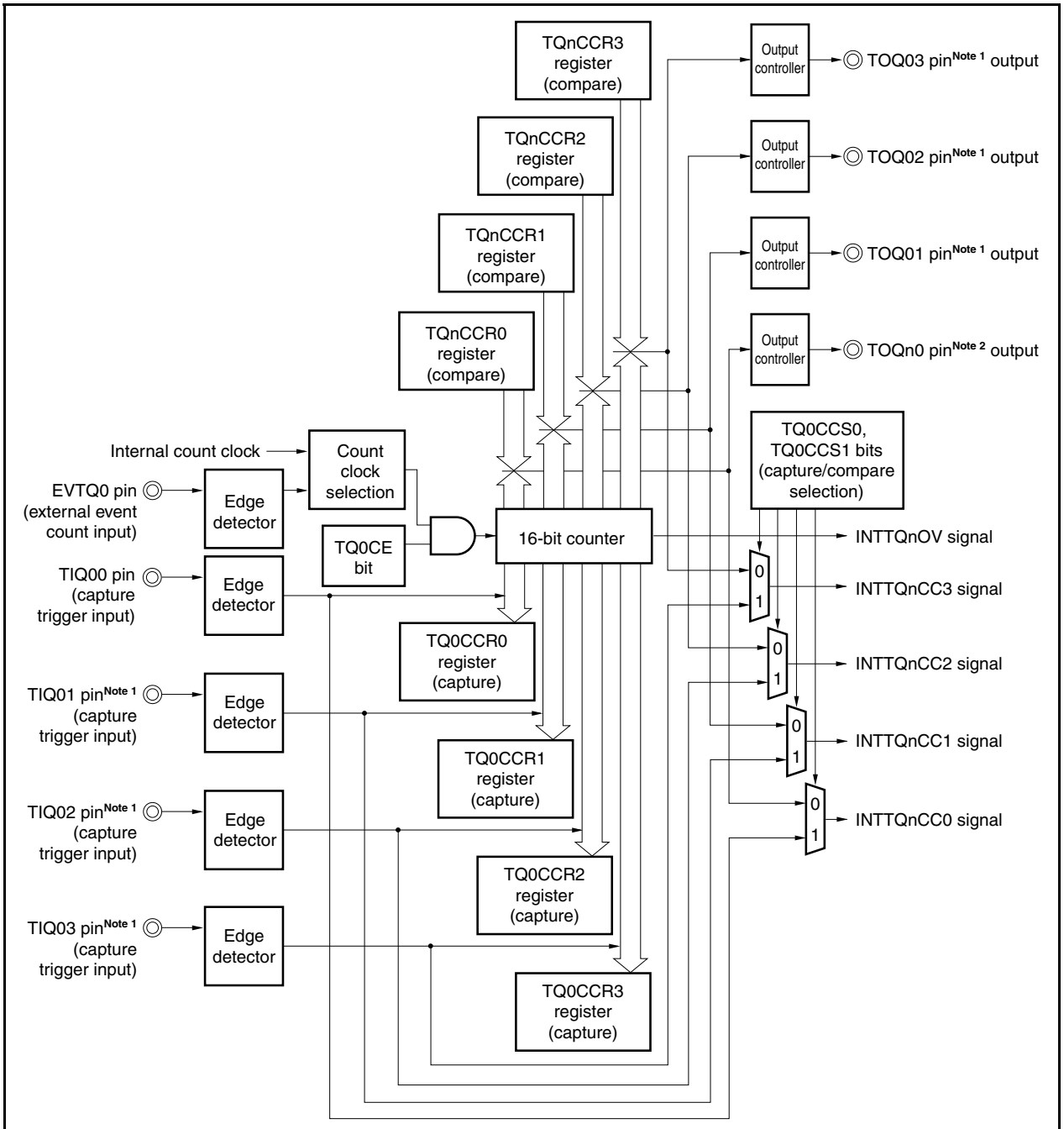
In the PWM output mode, however, it is generated one clock earlier. This is because the timing is changed to match the change timing of the output signal of the TOQ0b pin.

7.6.6 Free-running timer mode (TQnMD2 to TQnMD0 bits = 101)

The compare function is valid in both TMQ0 and TMQ1. The capture function is valid in TMQ0 only.

In the free-running timer mode, 16-bit timer/event counter Q starts counting when the TQnCTL0.TQnCE bit is set to 1. At this time, the TQ0CCRa register can be used as a compare register or a capture register, depending on the setting of the TQ0OPT0.TQ0CCSa bit.

Figure 7-33. Configuration in Free-Running Timer Mode



Notes 1. Because the capture trigger input pin (TIQ0b) and timer output pin (TOQ0b) are the same alternate function, these functions cannot be used at the same time.

2. The TOQ10 pin is provided only in the V850E/IA4.

Remark n = 0, 1, a = 0 to 3, b = 1 to 3

- Compare operation

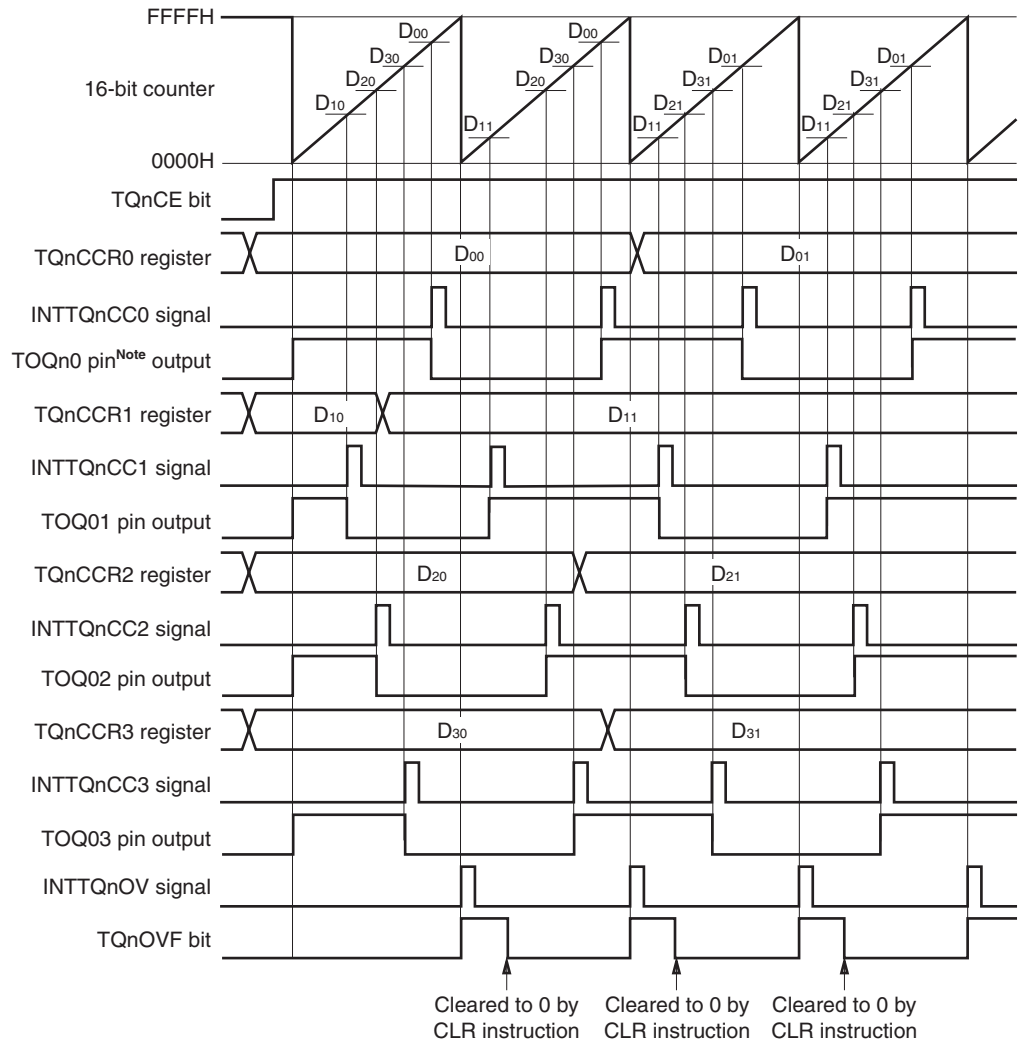
  When the TQnCE bit is set to 1, 16-bit timer/event counter Q starts counting, and the output signals of the TOQ00 to TOQ03 and TOQ10[Note] pins are inverted. When the count value of the 16-bit counter later matches the set value of the TQnCCRa register, a compare match interrupt request signal (INTTQnCCa) is generated, and the output signals of the TOQ00 to TOQ03 and TOQ10[Note] pins are inverted.

  The 16-bit counter continues counting in synchronization with the count clock. When it counts up to FFFFH, it generates an overflow interrupt request signal (INTTQnOV) at the next clock, is cleared to 0000H, and continues counting. At this time, the overflow flag (TQnOPT0.TQnOVF bit) is also set to 1. Confirm that the overflow flag is set to 1 and then clear it to 0 by executing the CLR instruction via software.

  The TQnCCRa register can be rewritten while the counter is operating. If it is rewritten, the new value is reflected at that time by anytime write, and compared with the count value.

Note V850E/IA4 only

Figure 7-34. Basic Timing in Free-Running Timer Mode (Compare Function)



Note The TOQ10 pin is provided only in the V850E/IA4.

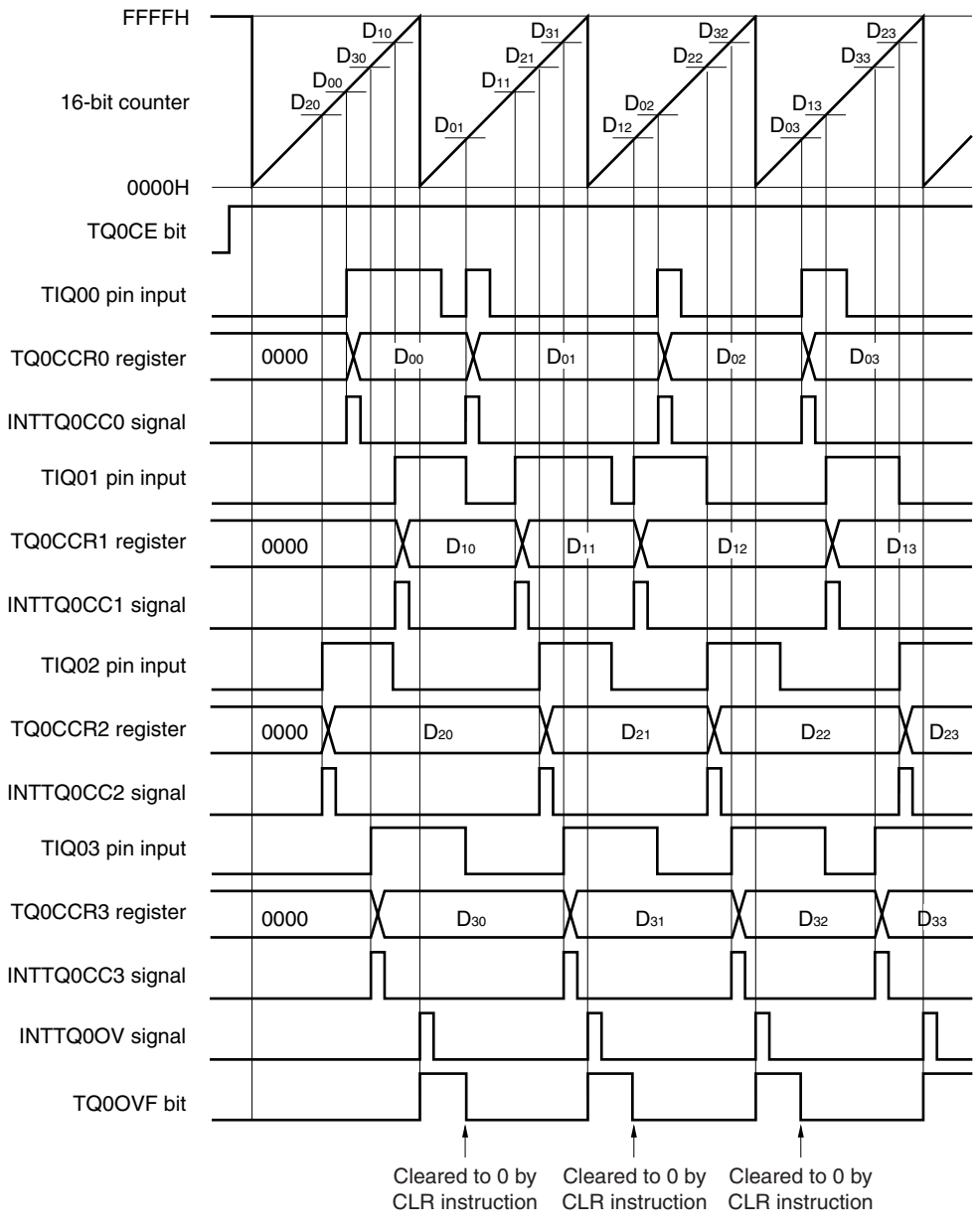
Remark n = 0, 1
a = 0 to 3

- Capture operation

  When the TQ0CE bit is set to 1, the 16-bit counter starts counting. When the valid edge input to the TIQ0a pin is detected, the count value of the 16-bit counter is stored in the TQ0CCRa register, and a capture interrupt request signal (INTTQ0CCa) is generated.

  The 16-bit counter continues counting in synchronization with the count clock. When it counts up to FFFFH, it generates an overflow interrupt request signal (INTTQ0OV) at the next clock, is cleared to 0000H, and continues counting. At this time, the overflow flag (TQ0OVF bit) is also set to 1. Confirm that the overflow flag is set to 1 and then clear it to 0 by executing the CLR instruction via software.

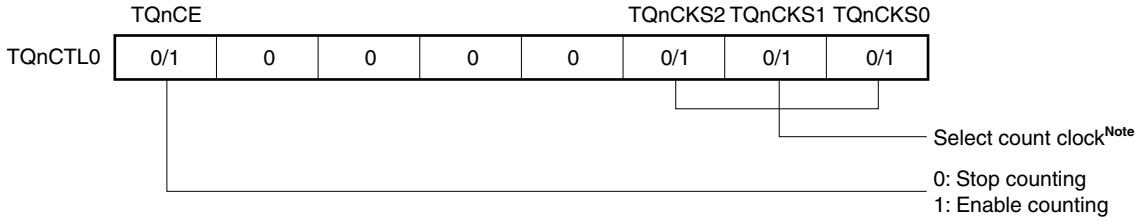
Figure 7-35. Basic Timing in Free-Running Timer Mode (Capture Function)



Remark a = 0 to 3

Figure 7-36. Register Setting in Free-Running Timer Mode (1/3)

(a) TMQn control register 0 (TQnCTL0)



Note The setting is invalid when the TQ0CTL1.TQ0EEE bit = 1.

(b) TMQn control register 1 (TQnCTL1)

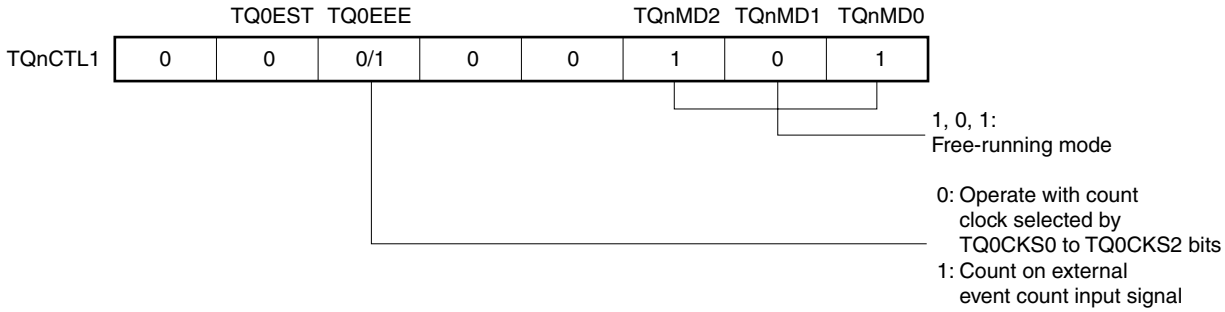
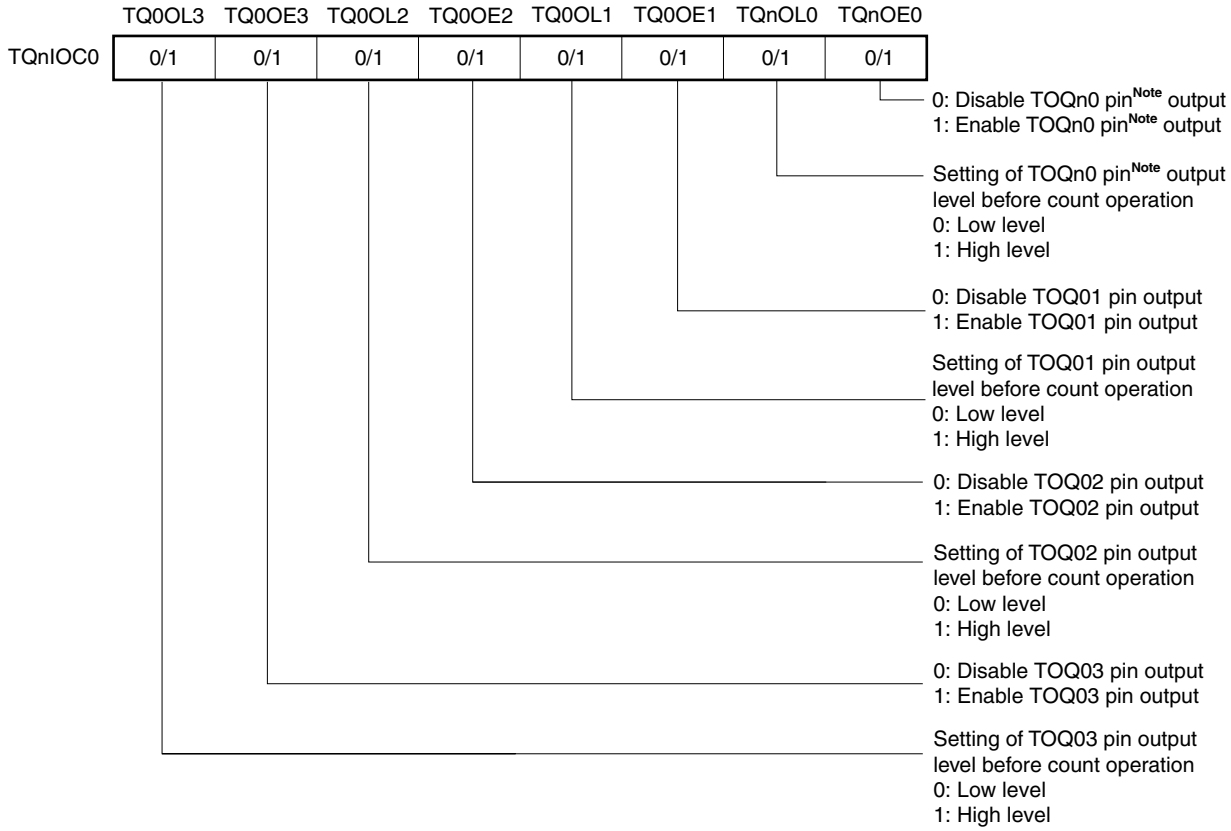


Figure 7-36. Register Setting in Free-Running Timer Mode (2/3)

(c) TMQn I/O control register 0 (TQnIOC0)



Note The TOQ10 pin is provided only in the V850E/IA4.

(d) TMQ0 I/O control register 1 (TQ0IOC1)

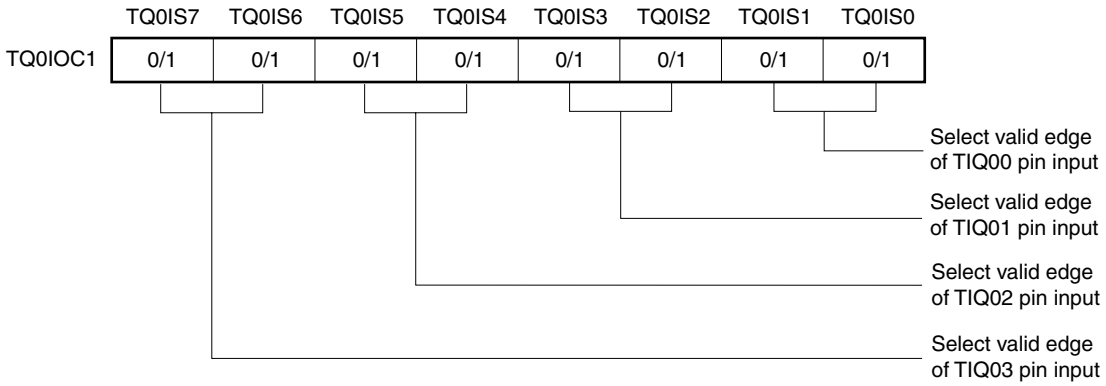
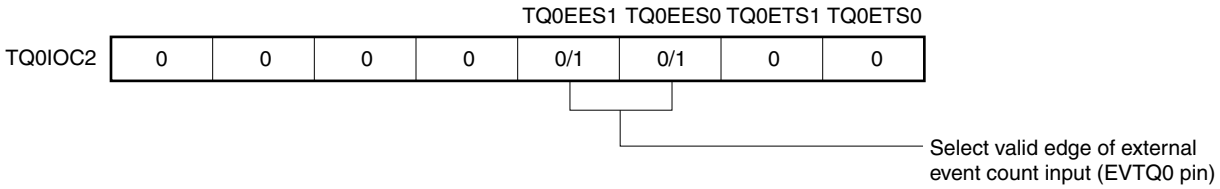
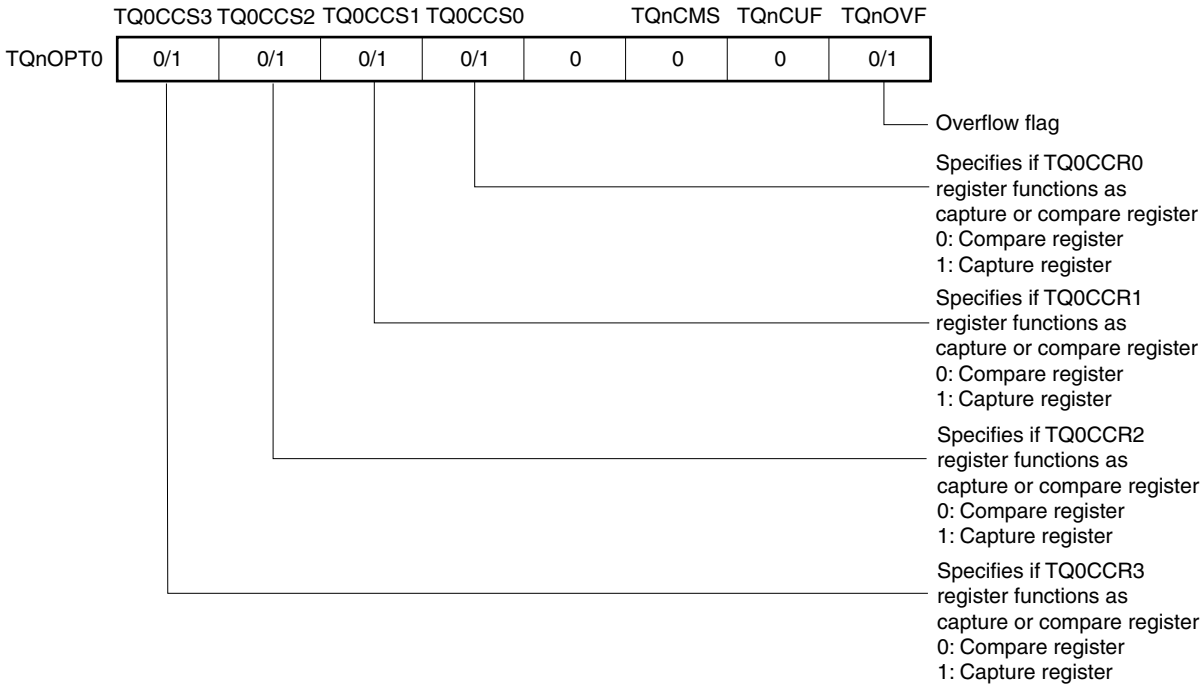


Figure 7-36. Register Setting in Free-Running Timer Mode (3/3)

(e) TMQ0 I/O control register 2 (TQ0IOC2)



(f) TMQn option register 0 (TQnOPT0)



(g) TMQn counter read buffer register (TQnCNT)

The value of the 16-bit counter can be read by reading the TQnCNT register.

(h) TMQn capture/compare registers 0 to 3 (TQnCCR0 to TQnCCR3)

These registers function as capture registers or compare registers depending on the setting of the TQ0OPT0.TQ0CCSa bit.

When the registers function as capture registers, they store the count value of the 16-bit counter when the valid edge input to the TIQ0a pin is detected.

When the registers function as compare registers and when D_a is set to the TQnCCRa register, the INTTQnCCa signal is generated when the counter reaches (D_a + 1), and the output signals of the TOQ00 to TOQ03 and TOQ10[Note] pins are inverted.

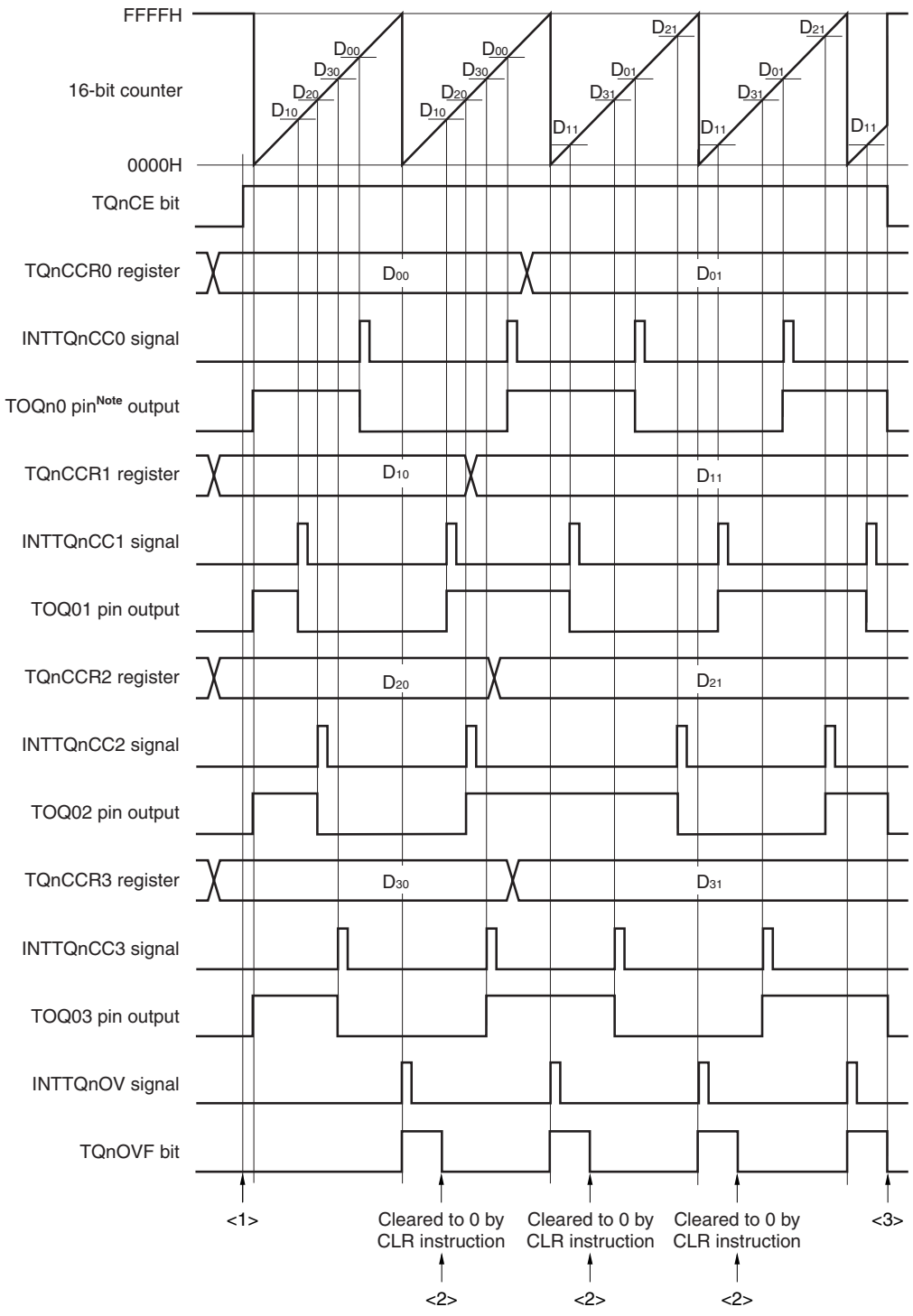
Note The TOQ10 pin is provided only in the V850E/IA4.

Remark a = 0 to 3

(1) Operation flow in free-running timer mode

(a) When using capture/compare register as compare register

Figure 7-37. Software Processing Flow in Free-Running Timer Mode (Compare Function) (1/2)

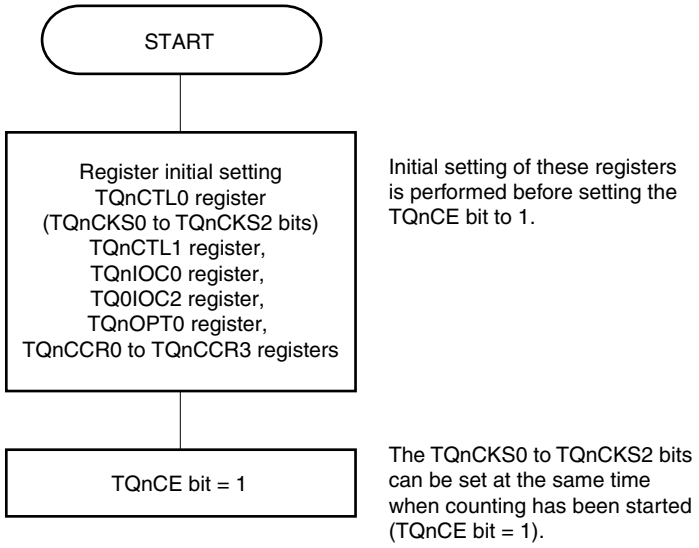


Note The TOQ10 pin is provided only in the V850E/IA4.

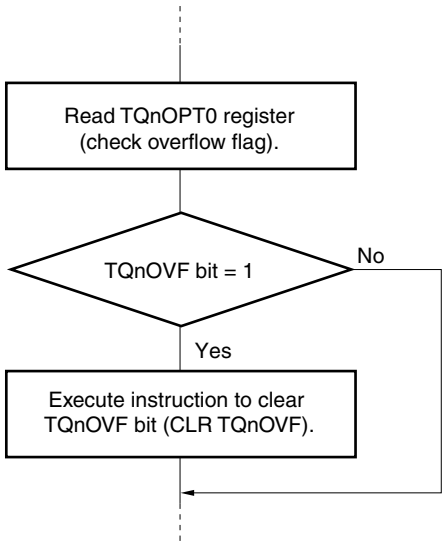
Remark n = 0, 1

Figure 7-37. Software Processing Flow in Free-Running Timer Mode (Compare Function) (2/2)

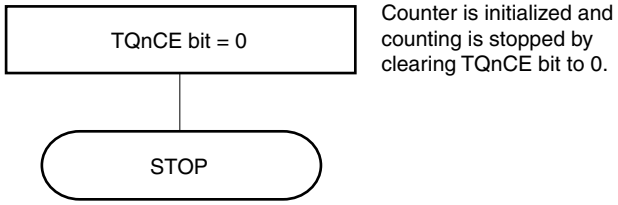
<1> Count operation start flow



<2> Overflow flag clear flow



<3> Count operation stop flow



Remark n = 0, 1

(b) When using capture/compare register as capture register

Figure 7-38. Software Processing Flow in Free-Running Timer Mode (Capture Function) (1/2)

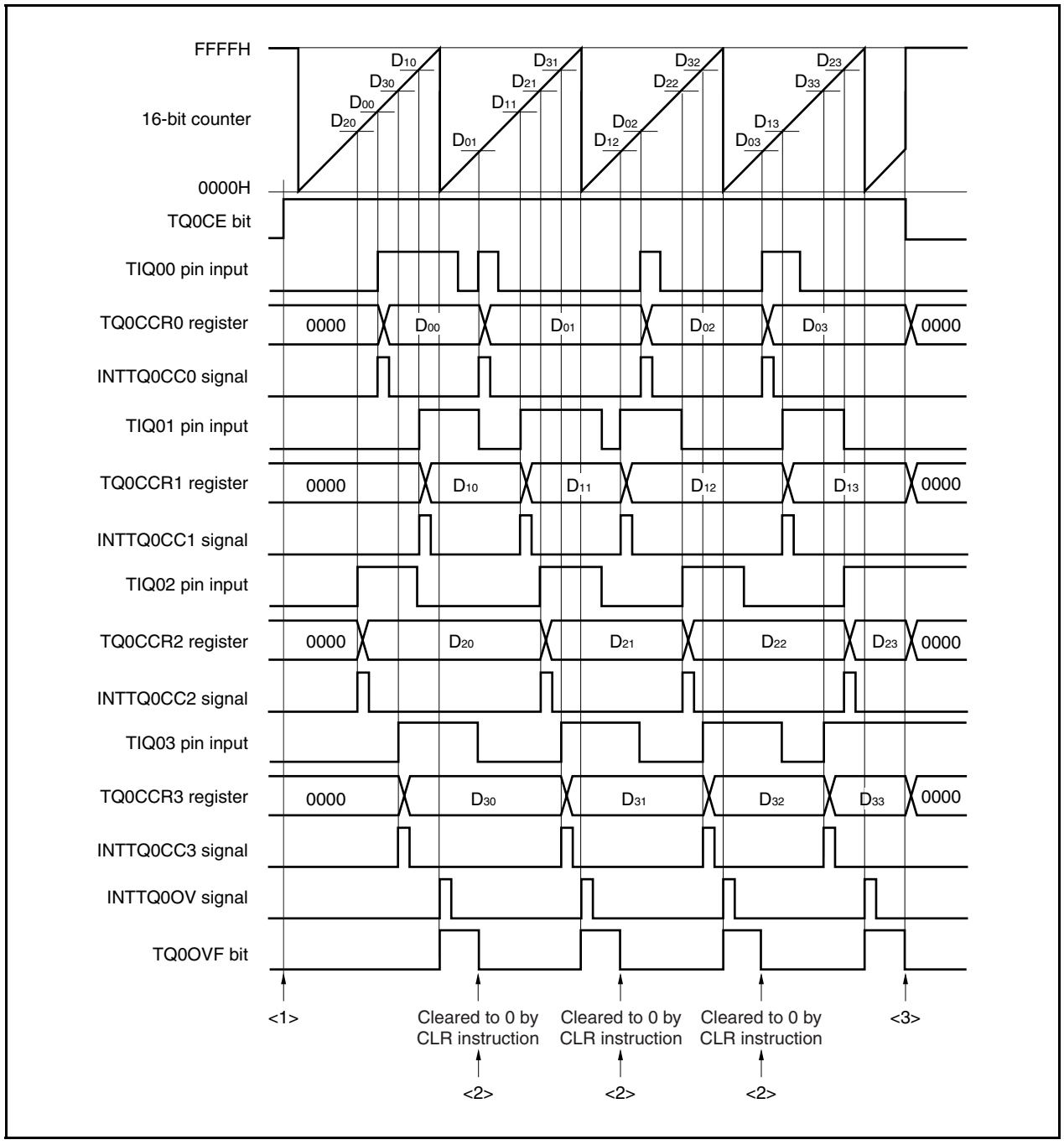
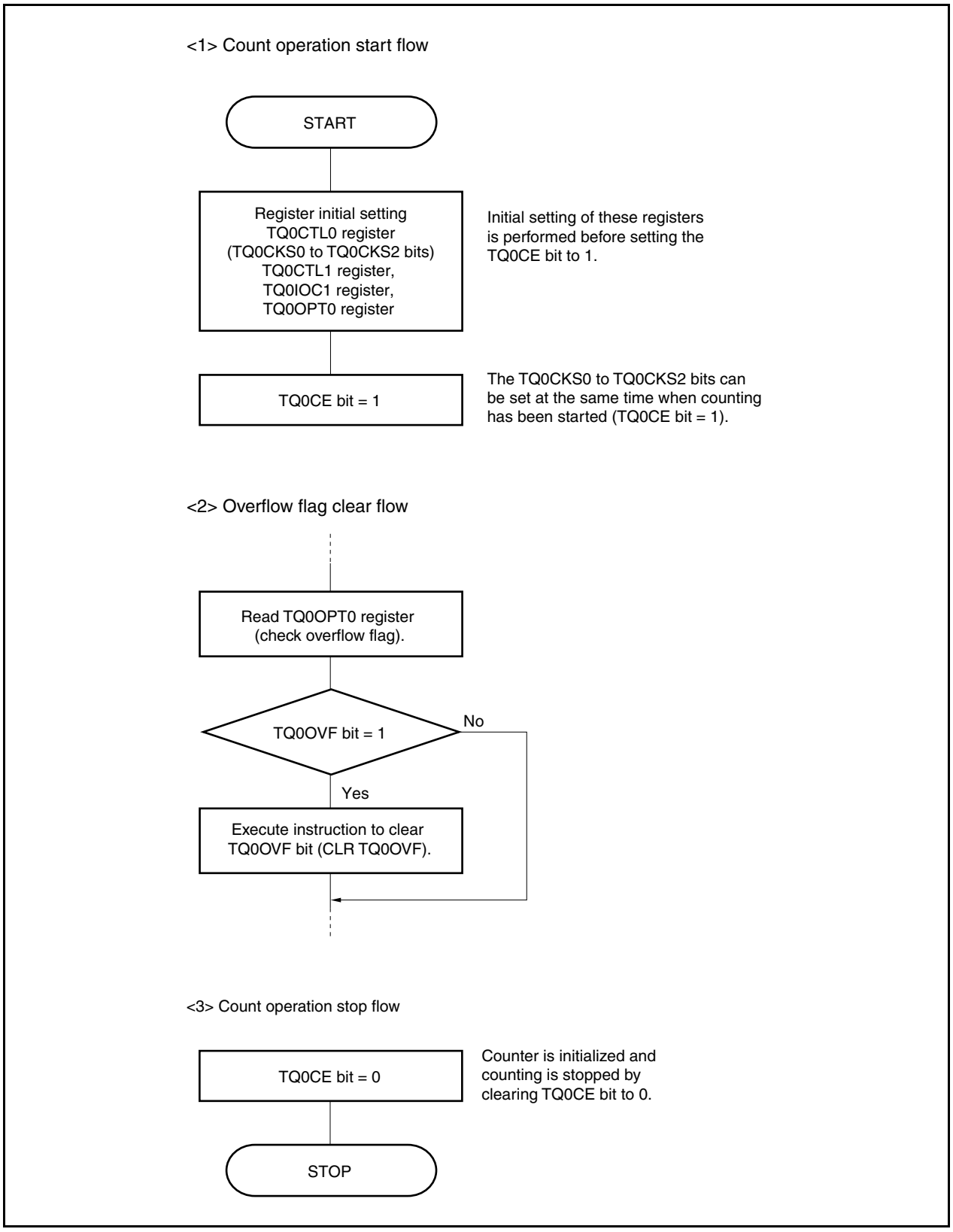


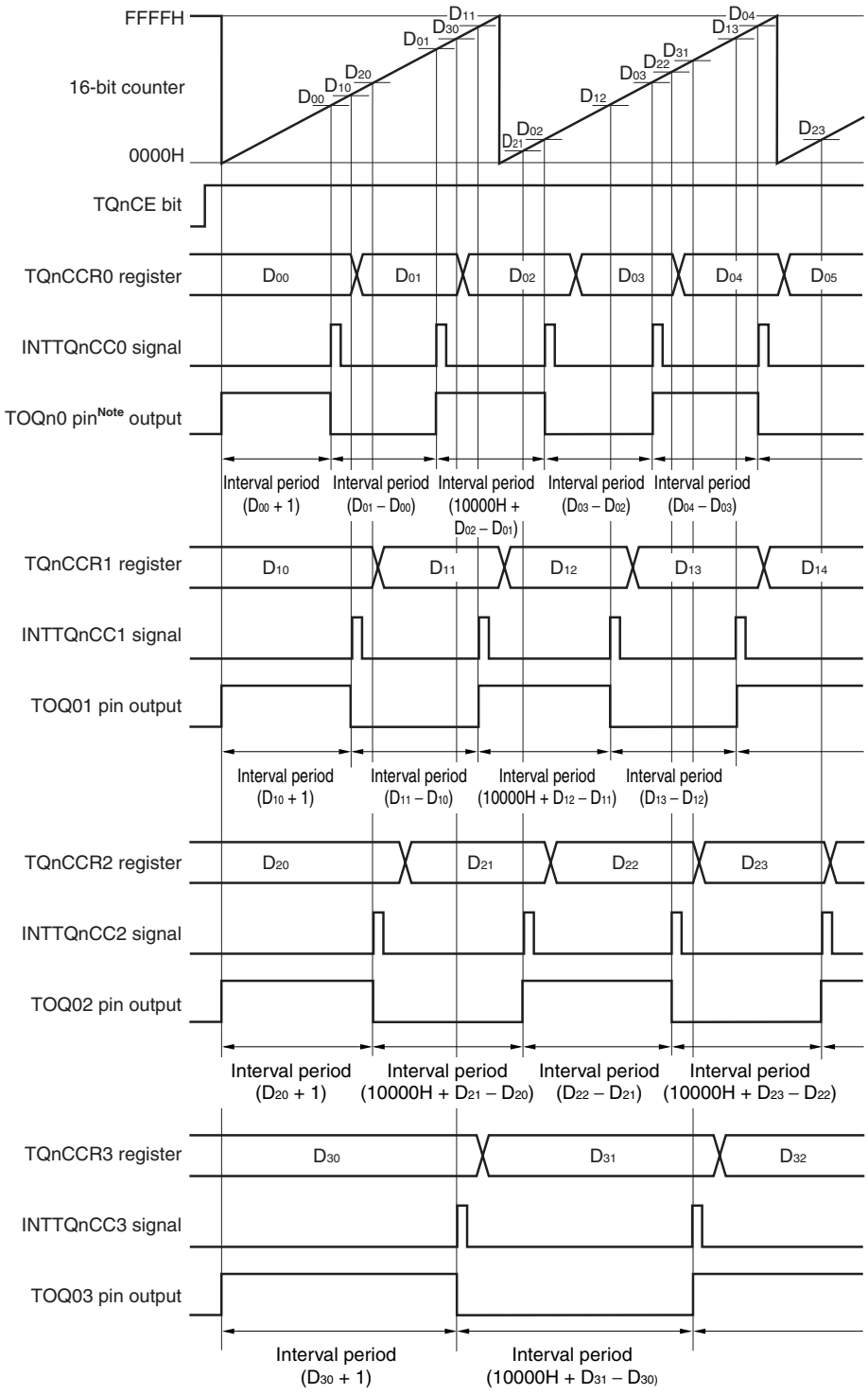
Figure 7-38. Software Processing Flow in Free-Running Timer Mode (Capture Function) (2/2)



(2) Operation timing in free-running timer mode

(a) Interval operation with compare register

When 16-bit timer/event counter Q is used as an interval timer with the TQnCCRa register used as a compare register, software processing is necessary for setting a comparison value to generate the next interrupt request signal each time the INTTQnCCa signal has been detected.



Note The TOQ10 pin is provided only in the V850E/IA4.

When performing an interval operation in the free-running timer mode, two intervals can be set with one channel.

To perform the interval operation, the value of the corresponding TQnCCRa register must be re-set in the interrupt servicing that is executed when the INTTQnCCa signal is detected.

The set value for re-setting the TQnCCRa register can be calculated by the following expression, where "D_a" is the interval period.

Compare register default value: $D_a - 1$

Value set to compare register second and subsequent time: Previous set value + D_a

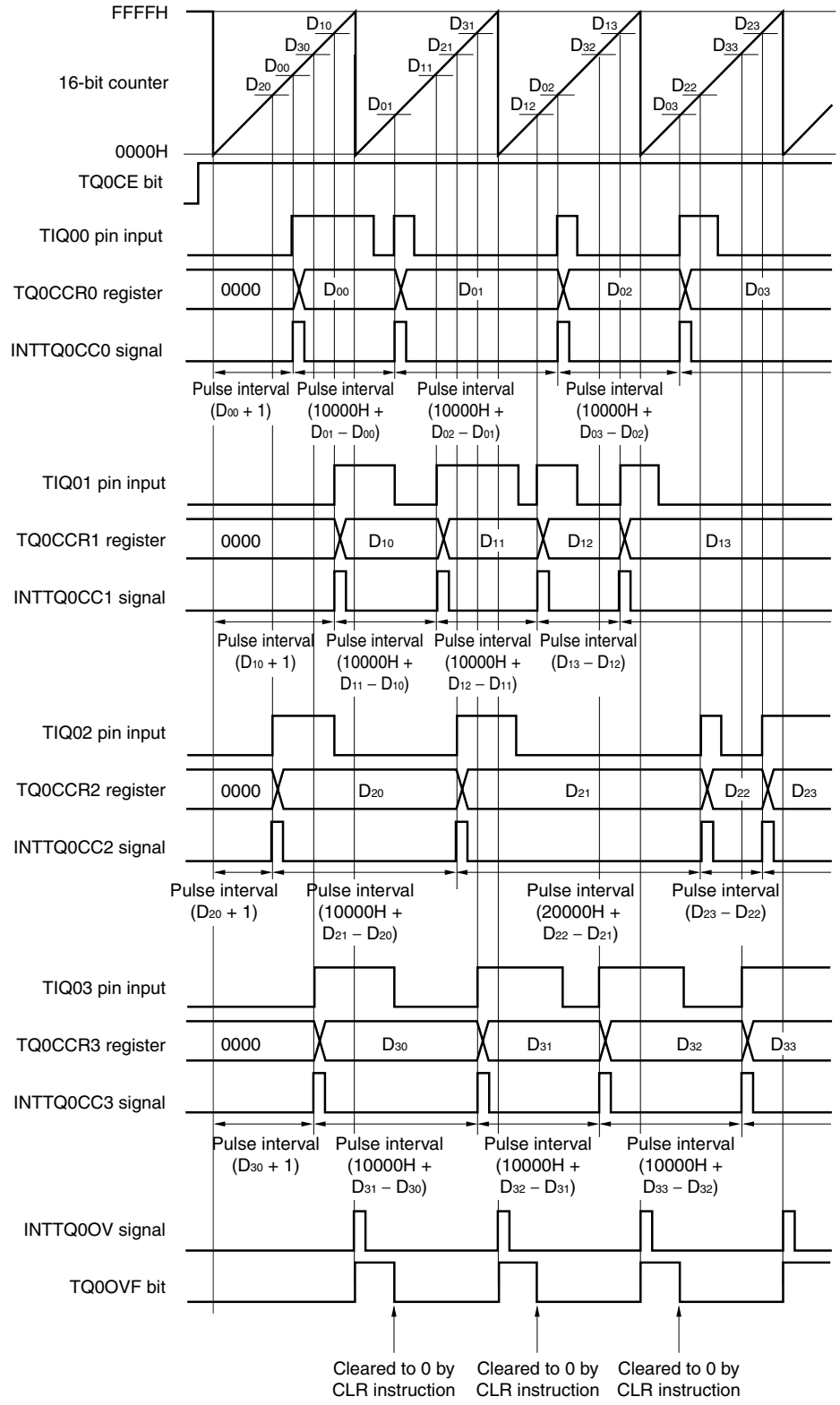
(If the calculation result is greater than FFFFH, subtract 10000H from the result and set this value to the register.)

Remark n = 0, 1

a = 0 to 3

(b) Pulse width measurement with capture register

When pulse width measurement is performed with the TQ0CCRa register used as a capture register, software processing is necessary for reading the capture register each time the INTTQ0CCa signal has been detected and for calculating an interval.



When executing pulse width measurement in the free-running timer mode, four pulse widths can be measured with one channel.

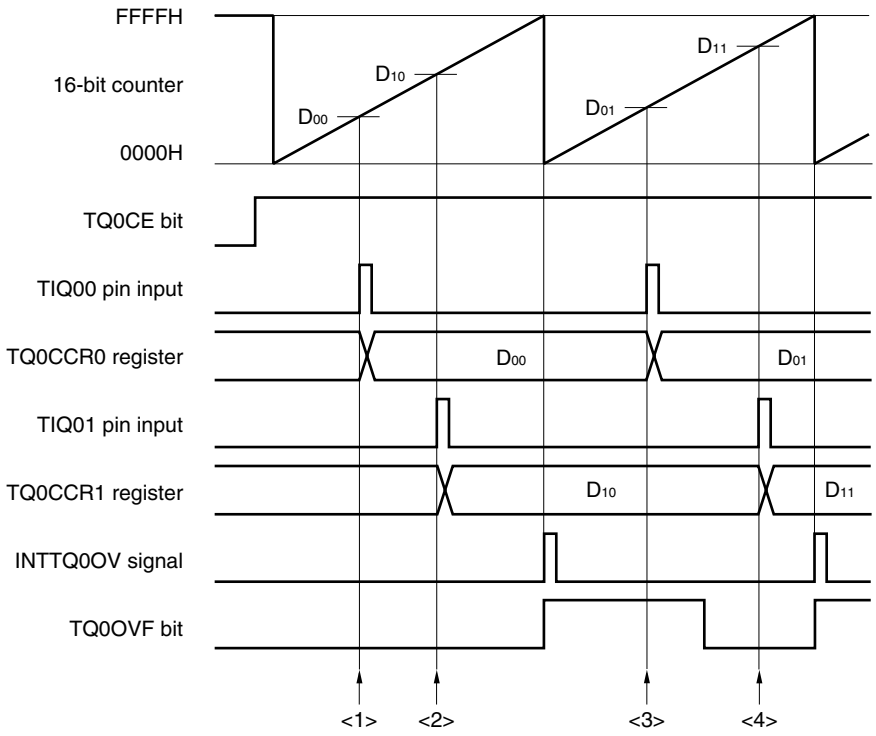
To measure a pulse width, the pulse width can be calculated by reading the value of the TQ0CCRa register in synchronization with the INTTQ0CCa signal, and calculating the difference between the read value and the previously read value.

Remark a = 0 to 3

(c) Processing of overflow when two or more capture registers are used

Care must be exercised in processing the overflow flag when two or more capture registers are used. First, an example of incorrect processing is shown below.

Example of incorrect processing when two or more capture registers are used

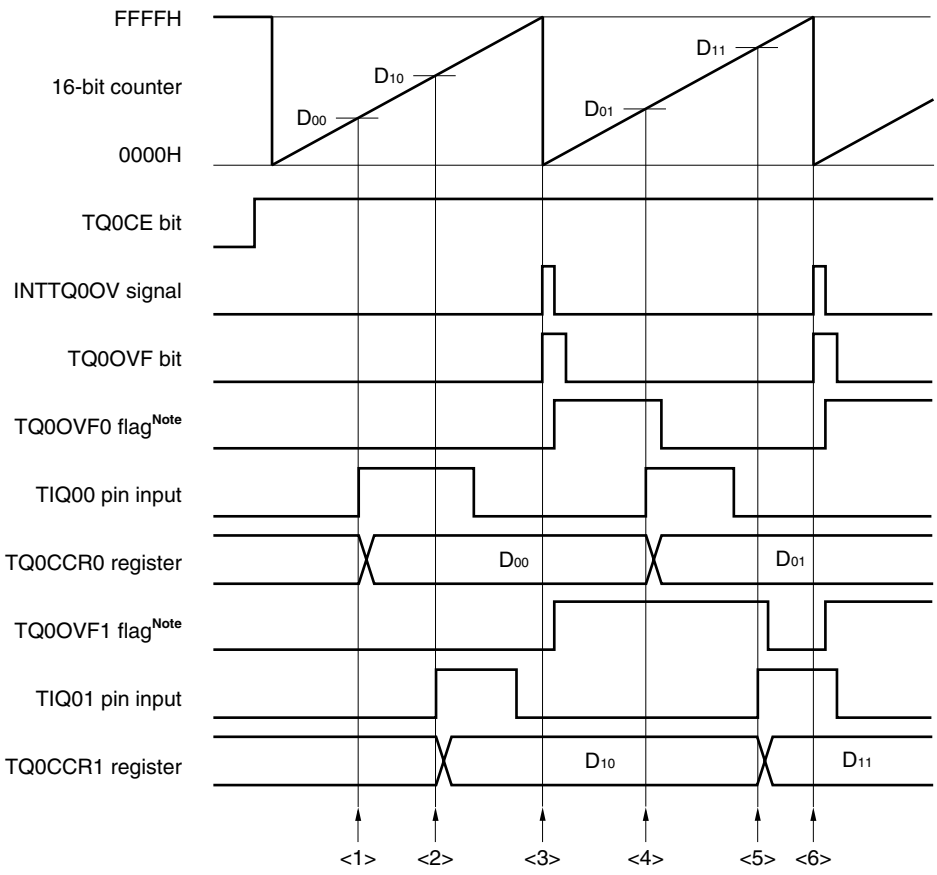


The following problem may occur when two pulse widths are measured in the free-running timer mode.

<1> Read the TQ0CCR0 register (setting of the default value of the TIQ00 pin input).
<2> Read the TQ0CCR1 register (setting of the default value of the TIQ01 pin input).
<3> Read the TQ0CCR0 register.
Read the overflow flag. If the overflow flag is 1, clear it to 0.
Because the overflow flag is 1, the pulse width can be calculated by (10000H + $D_{01} - D_{00}$).
<4> Read the TQ0CCR1 register.
Read the overflow flag. Because the flag is cleared in <3>, 0 is read.
Because the overflow flag is 0, the pulse width can be calculated by ($D_{11} - D_{10}$) (incorrect).

When two or more capture registers are used, and if the overflow flag is cleared to 0 by one capture register, the other capture register may not obtain the correct pulse width.
Use software when using two or more capture registers. An example of how to use software is shown below.

Example when two capture registers are used (using overflow interrupt)



Note The TQ0OVF0 and TQ0OVF1 flags are set on the internal RAM by software.

<1> Read the TQ0CCR0 register (setting of the default value of the TIQ00 pin input).

<2> Read the TQ0CCR1 register (setting of the default value of the TIQ01 pin input).

<3> An overflow occurs. Set the TQ0OVF0 and TQ0OVF1 flags to 1 in the overflow interrupt servicing, and clear the overflow flag to 0.

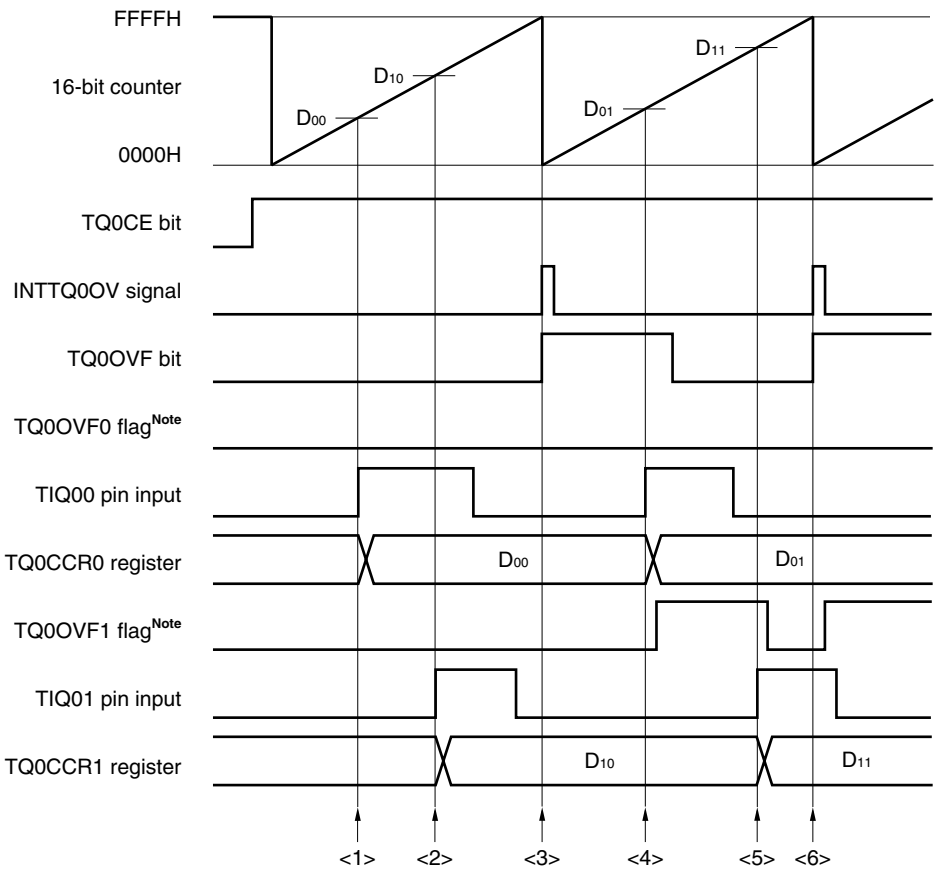
<4> Read the TQ0CCR0 register.
Read the TQ0OVF0 flag. If the TQ0OVF0 flag is 1, clear it to 0.
Because the TQ0OVF0 flag is 1, the pulse width can be calculated by (10000H + $D_{01} - D_{00}$).

<5> Read the TQ0CCR1 register.
Read the TQ0OVF1 flag. If the TQ0OVF1 flag is 1, clear it to 0 (the TQ0OVF0 flag is cleared in <4>, and the TQ0OVF1 flag remains 1).
Because the TQ0OVF1 flag is 1, the pulse width can be calculated by (10000H + $D_{11} - D_{10}$) (correct).

<6> Same as <3>

(2/2)

Example when two capture registers are used (without using overflow interrupt)



Note The TQ0OVF0 and TQ0OVF1 flags are set on the internal RAM by software.

<1> Read the TQ0CCR0 register (setting of the default value of the TIQ00 pin input).

<2> Read the TQ0CCR1 register (setting of the default value of the TIQ01 pin input).

<3> An overflow occurs. Nothing is done by software.

<4> Read the TQ0CCR0 register.

Read the overflow flag. If the overflow flag is 1, set only the TQ0OVF1 flag to 1, and clear the overflow flag to 0.

Because the overflow flag is 1, the pulse width can be calculated by (10000H + $D_{01} - D_{00}$).

<5> Read the TQ0CCR1 register.

Read the overflow flag. Because the overflow flag is cleared in <4>, 0 is read.

Read the TQ0OVF1 flag. If the TQ0OVF1 flag is 1, clear it to 0.

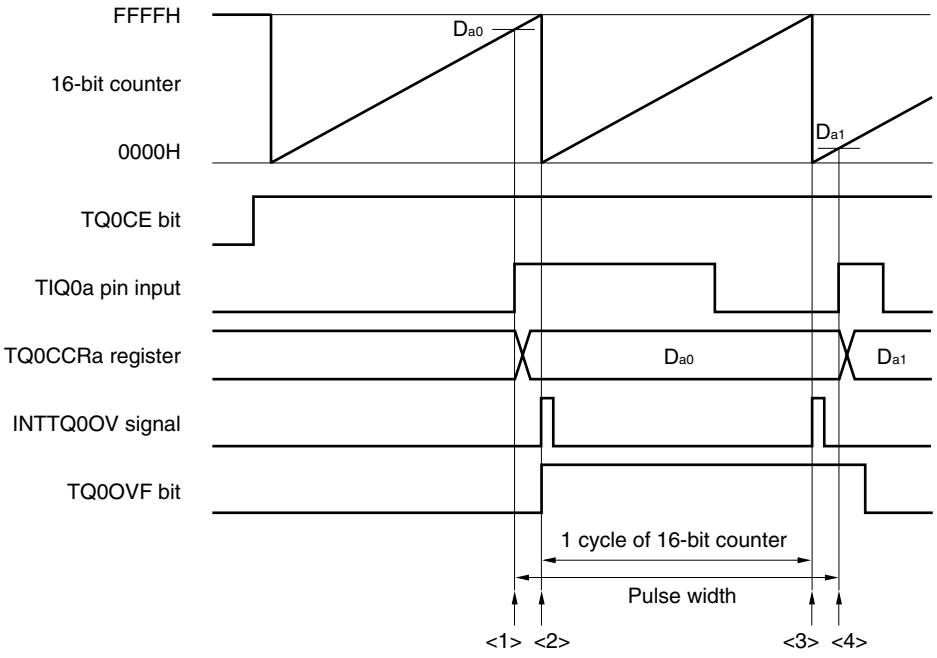
Because the TQ0OVF1 flag is 1, the pulse width can be calculated by (10000H + $D_{11} - D_{10}$) (correct).

<6> Same as <3>

(d) Processing of overflow if capture trigger interval is long

If the pulse width is greater than one cycle of the 16-bit counter, care must be exercised because an overflow may occur more than once from the first capture trigger to the next. First, an example of incorrect processing is shown below.

Example of incorrect processing when capture trigger interval is long



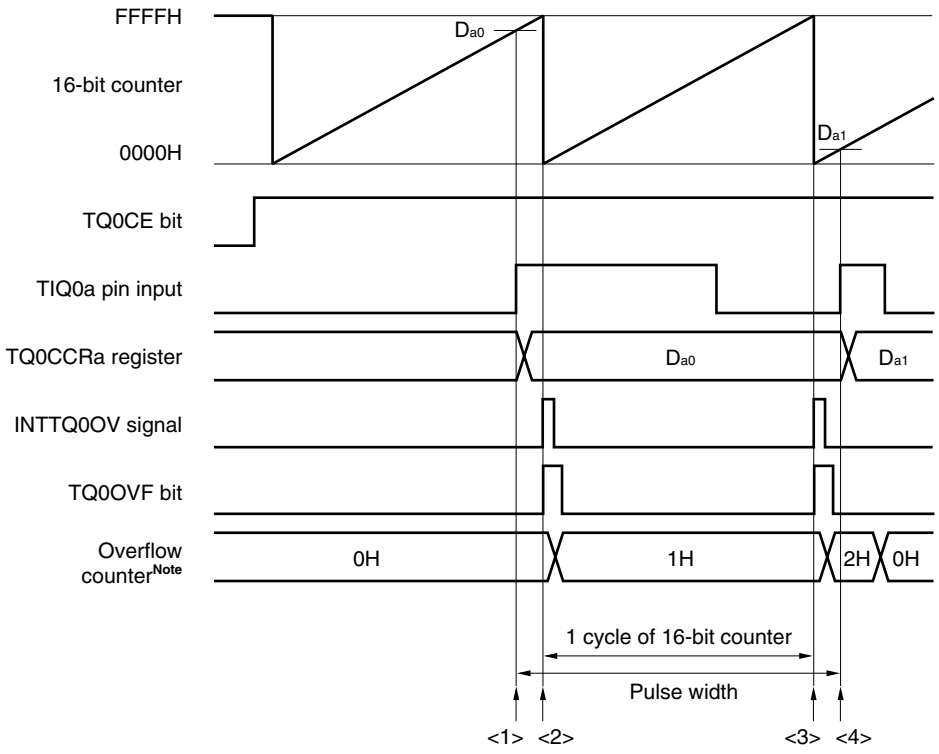
The following problem may occur when a long pulse width is measured in the free-running timer mode.

<1> Read the TQ0CCRa register (setting of the default value of the TIQ0a pin input).
<2> An overflow occurs. Nothing is done by software.
<3> An overflow occurs a second time. Nothing is done by software.
<4> Read the TQ0CCRa register.
Read the overflow flag. If the overflow flag is 1, clear it to 0.
Because the overflow flag is 1, the pulse width can be calculated by (10000H + $D_{a1} - D_{a0}$) (incorrect).
Actually, the pulse width must be (20000H + $D_{a1} - D_{a0}$) because an overflow occurs twice.

Remark a = 0 to 3

If an overflow occurs twice or more when the capture trigger interval is long, the correct pulse width may not be obtained.
If the capture trigger interval is long, slow the count clock to lengthen one cycle of the 16-bit counter, or use software. An example of how to use software is shown below.

Example when capture trigger interval is long



Note The overflow counter is set arbitrarily by software on the internal RAM.

<1> Read the TQ0CCRa register (setting of the default value of the TIQ0a pin input).

<2> An overflow occurs. Increment the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<3> An overflow occurs a second time. Increment (+1) the overflow counter and clear the overflow flag to 0 in the overflow interrupt servicing.

<4> Read the TQ0CCRa register.

Read the overflow counter.

→ When the overflow counter is "N", the pulse width can be calculated by ($N \times 10000H + D_{a1} - D_{a0}$).

In this example, the pulse width is ($20000H + D_{a1} - D_{a0}$) because an overflow occurs twice.

Clear the overflow counter (0H).

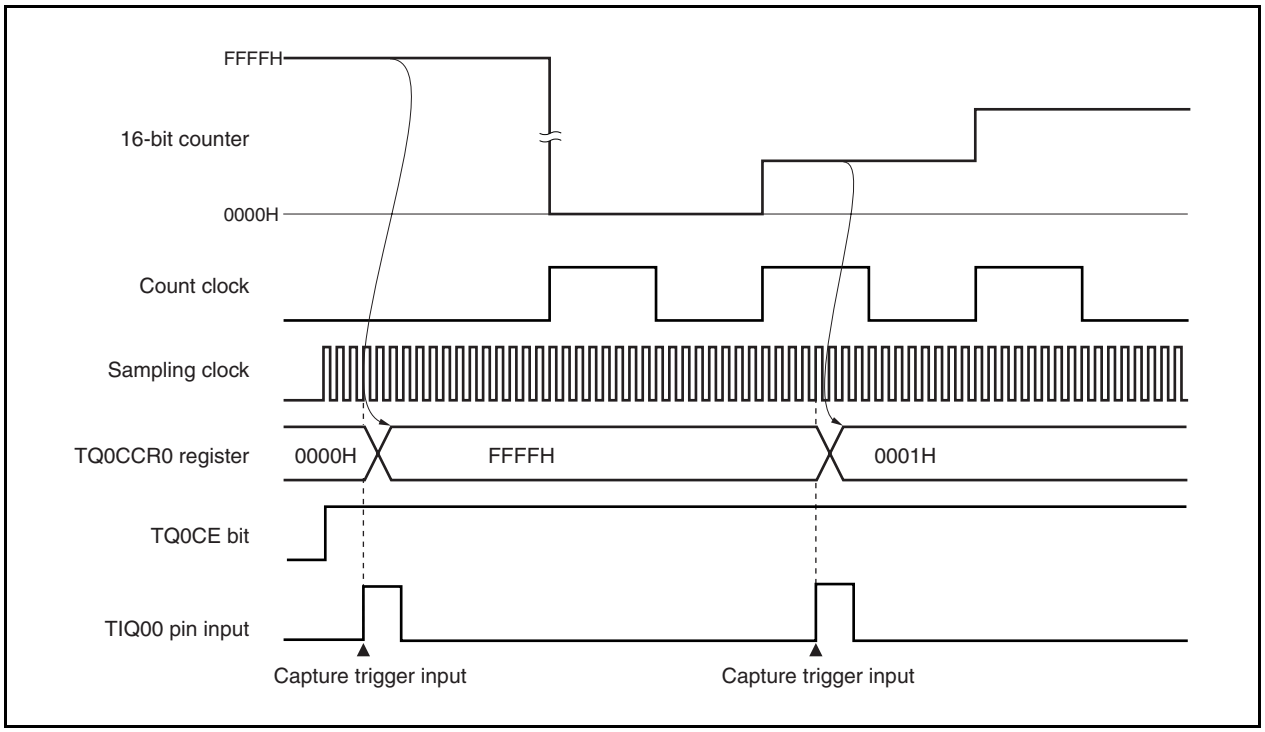
Remark a = 0 to 3

(e) Clearing overflow flag

The overflow flag can be cleared to 0 by clearing the TQnOVF bit to 0 with the CLR instruction after reading the TQnOVF bit when it is 1 and by writing 8-bit data (bit 0 is 0) to the TQnOPT0 register after reading the TQnOVF bit when it is 1.

(3) Note on capture operation

If the capture operation is used and if a slow clock is selected as the count clock, FFFFH, not 0000H, may be captured to the TQ0CCRa register if the capture trigger is input immediately after the TQ0CTL0.TQ0CE bit is set to 1 (a = 0 to 3).



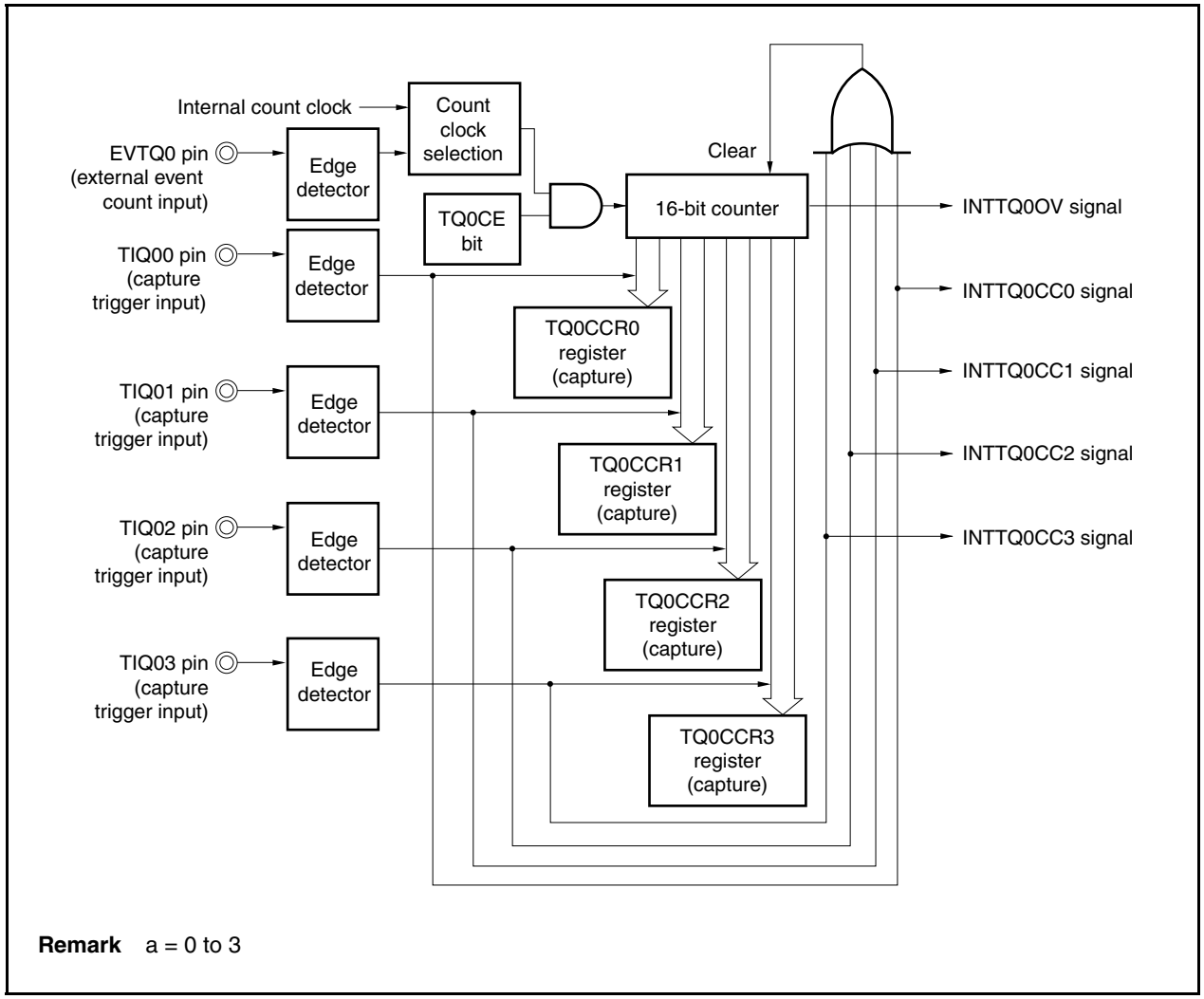
7.6.7 Pulse width measurement mode (TQ0MD2 to TQ0MD0 bits = 110)

In the pulse width measurement mode, 16-bit timer/event counter Q starts counting when the TQ0CTL0.TQ0CE bit is set to 1. Each time the valid edge input to the TIQ0a pin has been detected, the count value of the 16-bit counter is stored in the TQ0CCRa register, and the 16-bit counter is cleared to 0000H.

The interval of the valid edge can be measured by reading the TQ0CCRa register after a capture interrupt request signal (INTTQ0CCa) occurs.

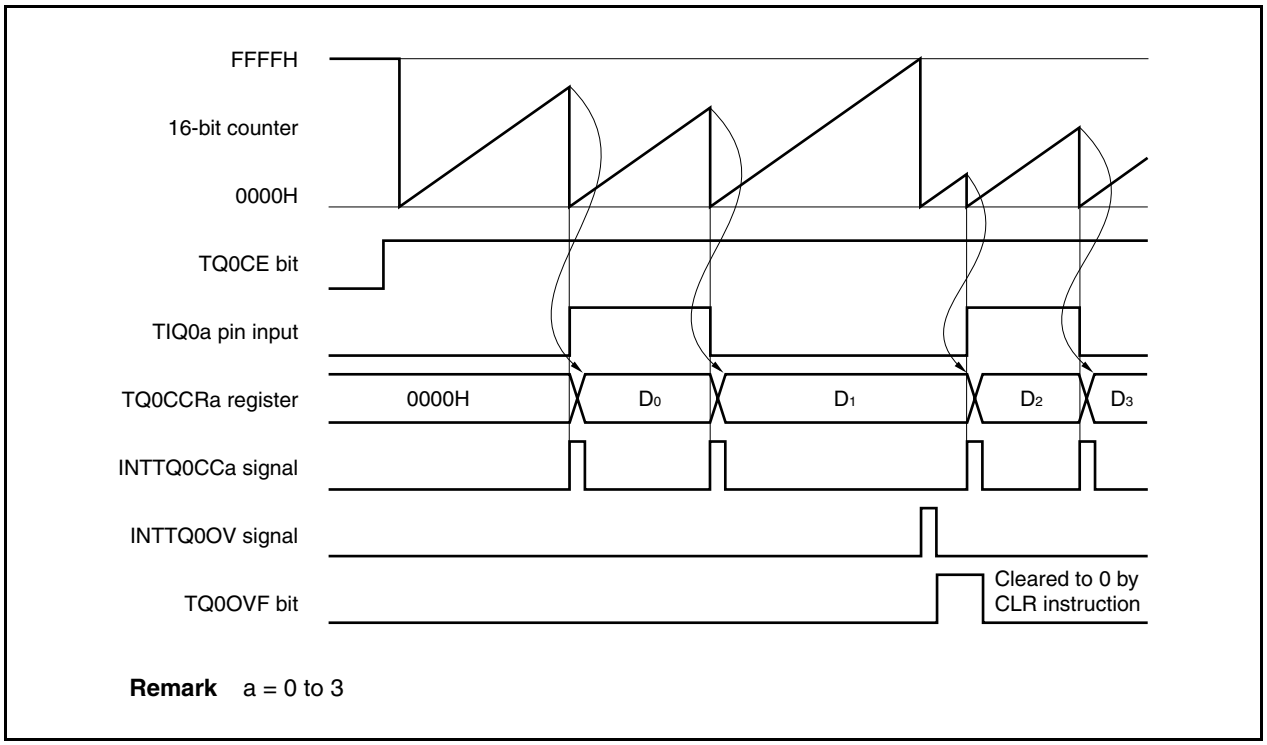
As shown in Figure 7-40, select either of the TIQ00 to TIQ03 pins as the capture trigger input pin. Specify “No edge detection” by using the TQ0IOC1 register for the unused pins.

Figure 7-39. Configuration in Pulse Width Measurement Mode



Remark a = 0 to 3

Figure 7-40. Basic Timing in Pulse Width Measurement Mode



Remark a = 0 to 3

When the TQ0CE bit is set to 1, the 16-bit counter starts counting. When the valid edge input to the TIQ0a pin is later detected, the count value of the 16-bit counter is stored in the TQ0CCRa register, the 16-bit counter is cleared to 0000H, and a capture interrupt request signal (INTTQ0CCa) is generated.

The pulse width is calculated as follows.

<R> Pulse width = Captured value × Count clock cycle

If the valid edge is not input to the TIQ0m pin even when the 16-bit counter counted up to FFFFH, an overflow interrupt request signal (INTTQ0OV) is generated at the next count clock, and the counter is cleared to 0000H and continues counting. At this time, the overflow flag (TQ0OPT0.TQ0OVF bit) is also set to 1. Clear the overflow flag to 0 by executing the CLR instruction via software.

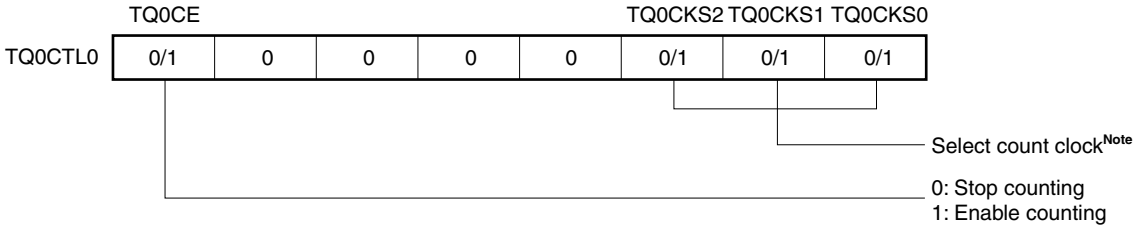
If the overflow flag is set to 1, the pulse width can be calculated as follows.

<R> Pulse width = (10000H × Number of times for which TQ0OVF bit is set to 1 + Captured value) × Count clock cycle

Remark a = 0 to 3

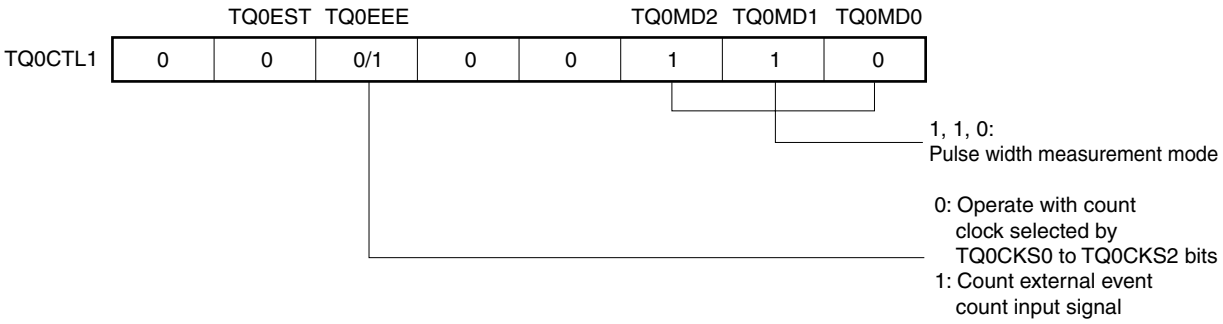
Figure 7-41. Register Setting in Pulse Width Measurement Mode (1/2)

(a) TMQ0 control register 0 (TQ0CTL0)

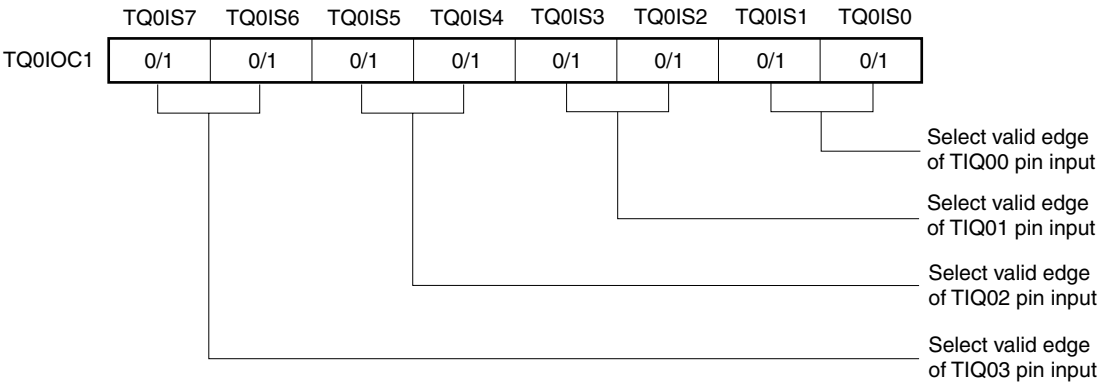


Note Setting is invalid when the TQ0CTL1.TQ0EEE bit = 1.

(b) TMQ0 control register 1 (TQ0CTL1)



(c) TMQ0 I/O control register 1 (TQ0IOC1)



(d) TMQ0 I/O control register 2 (TQ0IOC2)

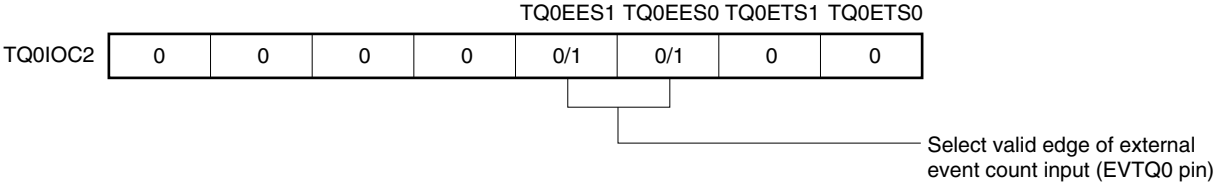


Figure 7-41. Register Setting in Pulse Width Measurement Mode (2/2)

(e) TMQ0 option register 0 (TQ0OPT0)

	TQ0CCS3	TQ0CCS2	TQ0CCS1	TQ0CCS0		TQ0CMS	TQ0CUF	TQ0OVF
TQ0OPT0	0	0	0	0	0	0	0	0/1

Overflow flag

(f) TMQ0 counter read buffer register (TQ0CNT)

The value of the 16-bit counter can be read by reading the TQ0CNT register.

(g) TMQ0 capture/compare registers 0 to 3 (TQ0CCR0 to TQ0CCR3)

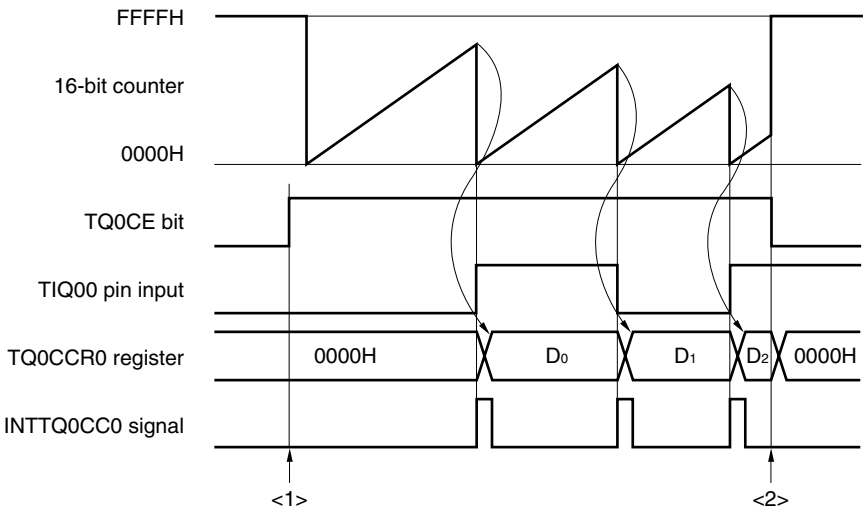
These registers store the count value of the 16-bit counter when the valid edge input to the TIQ0a pin is detected.

Remarks 1. TMQ0 I/O control register 0 (TQ0IOC0) is not used in the pulse width measurement mode.

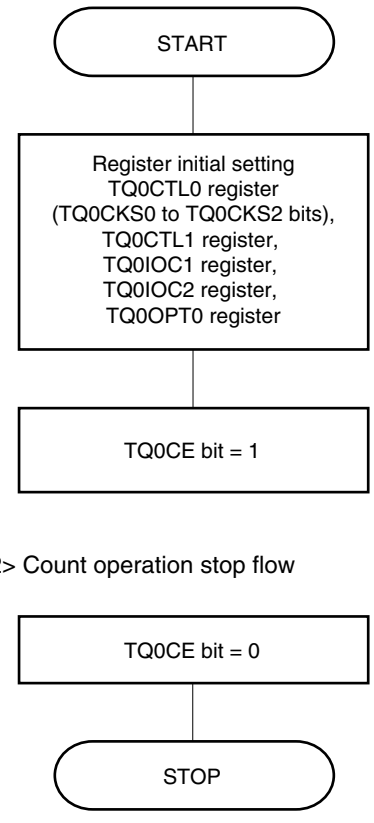
2. a = 0 to 3

(1) Operation flow in pulse width measurement mode

Figure 7-42. Software Processing Flow in Pulse Width Measurement Mode



<1> Count operation start flow



Initial setting of these registers is performed before setting the TQ0CE bit to 1.

The TQ0CKS0 to TQ0CKS2 bits can be set at the same time when counting has been started (TQ0CE bit = 1).

<2> Count operation stop flow

The counter is initialized and counting is stopped by clearing the TQ0CE bit to 0.

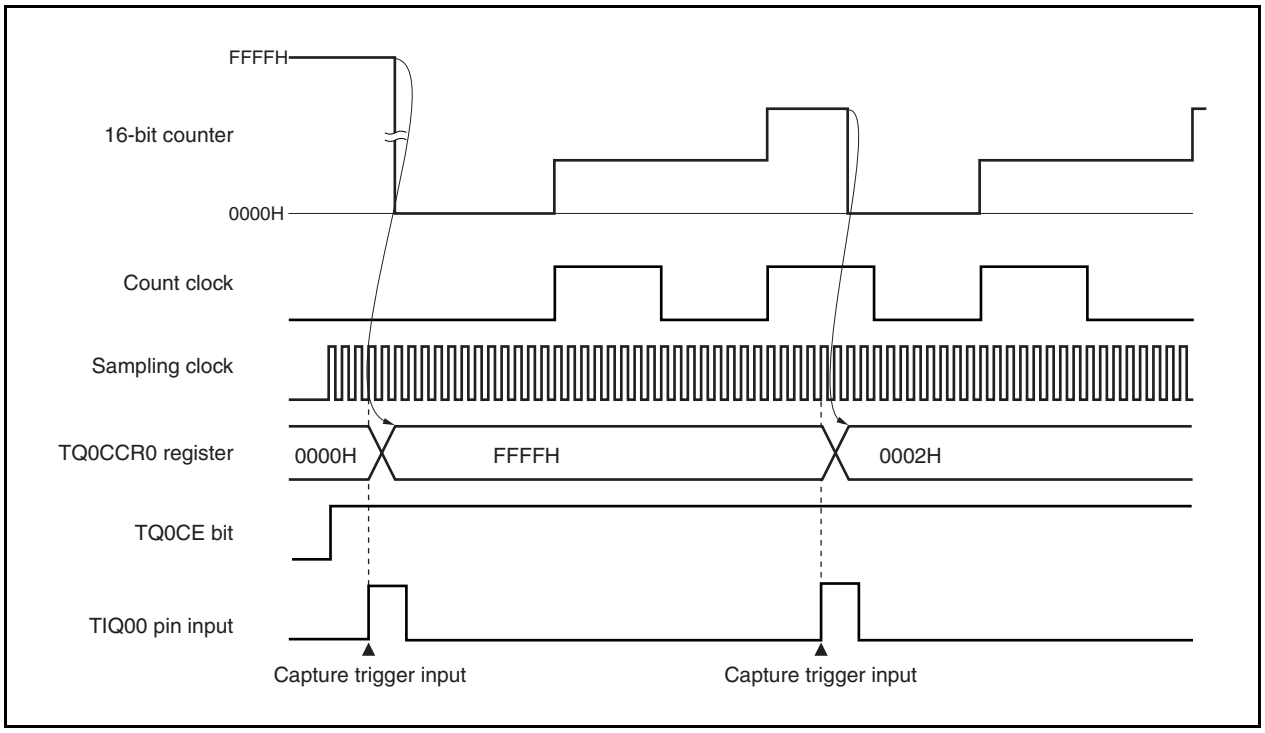
(2) Operation timing in pulse width measurement mode

(a) Clearing overflow flag

The overflow flag can be cleared to 0 by clearing the TQ0OVF bit to 0 with the CLR instruction after reading the TQ0OVF bit when it is 1 and by writing 8-bit data (bit 0 is 0) to the TQ0OPT0 register after reading the TQ0OVF bit when it is 1.

(3) Note

If a slow clock is selected as the count clock, FFFFH, not 0000H, may be captured to the TQ0CCRa register if the capture trigger is input immediately after the TQ0CTL0.TQ0CE bit is set to 1 (a = 0 to 3).



CHAPTER 8 16-BIT 2-PHASE ENCODER INPUT UP/DOWN COUNTER/ GENERAL-PURPOSE TIMER (TIMER ENC1n)

8.1 Functions

The 16-bit 2-phase encoder input up/down counter/general-purpose timer (timer ENC1n) performs the following operations (V850E/IA3: n = 0, V850E/IA4: n = 0, 1).

- General-purpose timer mode (see **8.5.1 Operation in general-purpose timer mode**)
  Free-running timer
<R> PWM output
- Up/down counter mode (see **8.5.2 Operation in UDC mode**)
  UDC mode A (mode 1, mode 2, mode 3, mode 4)
  UDC mode B (mode 1, mode 2, mode 3, mode 4)

8.2 Features

- Timer ENC1n
  V850E/IA3: 1 channel (timer ENC10)
  V850E/IA4: 2 channels (timers ENC10, ENC11)
- Compare registers: 2 each
- Capture/compare registers: 2 each
- Interrupt request sources
  - Capture/compare match interrupt request: 2 types each
  - Compare match interrupt request: 2 types each
- Capture request signal: 2 types each
  - The TMENC1n value can be latched using the valid edge of the TCLR1n and TCUD1n pins corresponding to the capture/compare register as the capture trigger.
- Count clock selectable through division by prescaler
- Timer clear
  The following timer clear operations are performed according to the mode that is used.

  (a) General-purpose timer mode: Timer clear operation is possible upon occurrence of match with CM1n0 register set value.
  (b) Up/down counter mode: The timer clear operation can be selected from among the following four conditions.
    (i) Timer clear performed upon occurrence of match with the CM1n0 register set value during TMENC1n count up operation, and timer clear performed upon occurrence of match with the CM1n1 register set value during TMENC1n count down operation.
    (ii) Timer clear performed only by external input.
    (iii) Timer clear performed upon occurrence of match between TMENC1n count value and the CM1n0 register set value.
    (iv) Timer clear performed upon occurrence of external input and match between TMENC1n count value and the CM1n0 register set value.

<R>

- PWM output function

  A 16-bit resolution PWM output waveform can be output from the TO1n pin in the general-purpose timer mode.

Caution In the μPD70F3186 (V850E/IA4), because the I/O pins (TIUD11, TO11, TCUD11, TCLR11) of timer ENC11 and input pins (DDI, DCK, DMS) of the on-chip debug function are the alternate-function pin, two functions cannot be used at the same time.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

8.3 Configuration

The basic configuration is shown below.

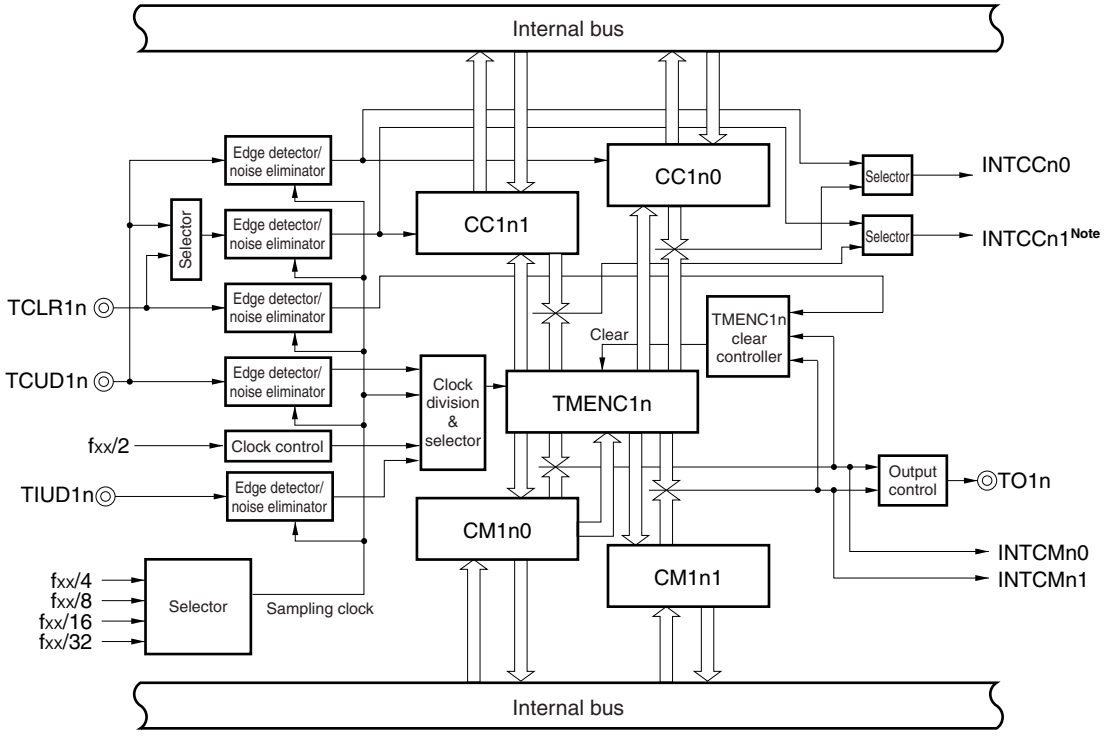
Table 8-1. Timer ENC1n Configuration

Timer	Count Clock	Register	Read/Write	Generated Interrupt Request Signal	Capture Trigger
Timer ENC1n	fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128, fxx/256	TMENC1n	Read/write	–	–
		CM1n0	Read/write	INTCMn0	–
		CM1n1	Read/write	INTCMn1	–
		CC1n0	Read/write	INTCCn0	TCUD1n
		CC1n1	Read/write	INTCCn1	TCLR1n or TCUD1n

Remarks 1. fxx: Peripheral clock

2. V850E/IA3: n = 0, V850E/IA4: n = 0, 1

Figure 8-1. Block Diagram of Timer ENC1n



Note The INTCCn1 interrupt request signal is the capture trigger signal from the TCLR1n pin or capture trigger signal from the TCUD1n pin selected by the CSL1n.CSL1n0 bit.

Remarks 1. fxx: Peripheral clock

2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

(1) Timer ENC1n (TMENC1n)

TMENC1n is a general-purpose timer (in general-purpose mode) and 2-phase encoder input up/down counter (in UDC mode).

This timer counts up in the general-purpose operation mode and counts up/down in the UDC mode.

It can be read or written in 16-bit units.

Reset sets TMENC1n to 0000H.

Cautions 1. Writing to TMENC1n is enabled only when the TMC1n.TM1CEn bit is 0 (count operation disabled).

2. Continuous reading of TMENC1n is prohibited. If TMENC1n is continuously read, the second value read may differ from the actual value. If TMENC1n must be read twice, be sure to read another register between the first and the second read operation.

3. Writing the same value to the TMENC1n, CC1n0, and CC1n1 registers, and the STATUS1n register is prohibited.

Writing the same value to the CCR1n, TUM1n, TMC1n, SESA1n, and PRM1n registers, and CM1n0 and CM1n1 registers is permitted (writing the same value is guaranteed even during a count operation).

After reset: 0000H R/W Address: TMENC10 FFFFF580H, TMENC11 FFFFF5A0H[Note]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMENC1n																

V850E/IA3 n = 0

V850E/IA4 n = 0, 1

Note V850E/IA4 only

TMENC1n start and stop is controlled by the TMC1n.TM1CEn bit.

The TMENC1n operation consists of the following two modes.

(a) General-purpose timer mode

<R> In the general-purpose timer mode, TMENC1n operates as a 16-bit interval timer, free-running timer, or PWM output.

Counting is performed based on the clock selected by software. Division by the prescaler can be selected for the count clock from among fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128, or fxx/256, using the PRM1n.PRM1n2 to PRM1n.PRM1n0 bits (fxx: Peripheral clock).

(b) Up/down counter mode (UDC mode)

In the UDC mode, TMENC1n functions as a 16-bit up/down counter that performs counting based on the TCUD1n and TIUD1n input signals. This mode is divided into the UDC mode A and UDC mode B, depending on the condition of clearing TMENC1n.

Cautions 1. The TCUD1n pin is used alternately for the UDC mode and the external capture function. Therefore, in the UDC mode, the external capture function cannot be used.

2. The TCLR1n pin is used alternately for the UDC mode and the external capture function. Therefore, when the TCLR1n input is used in UDC mode A, the external capture function cannot be used.

The conditions for clearing TMENC1n are as follows, according to the operation mode.

Table 8-2. TMENC1n Clear Conditions

Operation Mode	TUM1n Register		TMC1n Register			TMENC1n Clear
	CMDn Bit	MSELn Bit	ENMDn Bit	CLRn1 Bit	CLRn0 Bit	
General-purpose timer mode	0	0	0	×	×	Clearing not performed (free-running timer)
			1	×	×	Cleared upon match with the CM1n0 register set value
UDC mode A	1	0	×	0	0	Cleared only by TCLR1n input
			×	0	1	Cleared upon match with the CM1n0 register set value during count up operation
			×	1	0	Cleared by TCLR1n input or upon match with the CM1n0 register set value during count up operation
			×	1	1	Clearing not performed
UDC mode B	1	1	×	×	×	Cleared upon match with the CM1n0 register set value during count up operation or upon match with the CM1n1 register set value during count down operation
Other than above						Setting prohibited

Remarks 1. ×: Indicates that the set value of that bit is ignored.

2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

8.4 Control Registers

(1) Timer unit mode register 1n (TUM1n)

The TUM1n register is an 8-bit register that specifies the TMENC1n operation mode and controls the output pin operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. Changing the value of the TUM1n register during TMENC1n operation (TMC1n.TM1CEn bit = 1) is prohibited.

2. When the CMDn bit = 0 (general-purpose timer mode), setting the MSELn bit = 1 (UDC mode B) is prohibited.

After reset: 00H R/W Address: TUM10 FFFFF58BH, TUM11 FFFFF5ABH[Note]

	7	6	5	4	3	2	1	0
TUM1n	CMDn	0	0	0	TOEn	ALVT10n	0	MSELn

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

CMDn	TMENC1n operation mode specification
0	General-purpose timer mode (count up)
1	UDC mode (count up/down)

TOEn	Specification of timer output (TO1n) enable
0	Timer output disabled
1	Timer output enabled
When the CMDn bit = 1 (UDC mode), timer output is not performed regardless of the setting of the TOEn bit. At this time, timer output is the inverted phase level of the level set by the ALVT10n bit.	

ALVT10n	Specification of timer output (TO1n) active level
0	Active level is high level
1	Active level is low level
When the CMDn bit = 1 (UDC mode), timer output is not performed regardless of the setting of the TOEn bit. At this time, timer output is the inverted phase level of the level set by the ALVT10n bit.	

MSELn	Specification of operation in UDC mode (count up/down)
0	UDC mode A TMENC1n can be cleared by setting the TMC1n.CLRn1 and TMC1n.CLRn0 bits.
1	UDC mode B TMENC1n is cleared in the following cases. • Upon match with the CM1n0 register during TMENC1n count up operation • Upon match with the CM1n1 register during TMENC1n count down operation
When UDC mode B is set, the TMC1n.ENMDn, TMC1n.CLRn1, and TMC1n.CLRn0 bits become invalid.	

Note V850E/IA4 only

(2) Timer control register 1n (TMC1n)

The TMC1n register enables/disables TMENC1n operation and sets transfer and timer clear operations.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Caution Changing the values of the TMC1n register bits other than the TM1CEn bit during TMENC1n operation (TM1CEn bit = 1) is prohibited.

After reset: 00H R/W Address: TMC10 FFFFF58CH, TMC11 FFFFF5ACH[Note]

	7	<6>	5	4	3	2	1	0
TMC1n	0	TM1CEn	0	0	RLENn	ENMDn	CLRn1	CLRn0

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

TM1CEn	TMENC1n operation control
0	Count operation disabled
1	Count operation enabled

RLENn	Specification of transfer operation from the CM1n0 register to TMENC1n
0	Transfer operation disabled
1	Transfer operation enabled

- When RLENn = 1, the value set to the CM1n0 register is transferred to TMENC1n by a TMENC1n underflow.
- The RLENn bit is valid only in UDC mode A (TUM1n.CMDn bit = 1, MSELn bit = 0). In the general-purpose timer mode (CMDn bit = 0) and in UDC mode B (CMDn bit = 1, MSELn bit =1), a transfer operation is not performed even if the RLENn bit is set (1).

ENMDn	Control of TMENC1n clear operation in general-purpose timer mode
0	Clear disabled (free-running mode) Clearing is not performed even when TMENC1n and the CM1n0 register values match.
1	Clear enabled Clearing is performed upon match of TMENC1n and the CM1n0 register values.

In UDC mode (TUM1n.CMDn bit = 1), the ENMDn bit setting becomes invalid.

CLRn1	CLRn0	TMENC1n clear source specification
0	0	Cleared only by external input (TCLR1n)
0	1	Cleared upon match of TMENC1n count value and the CM1n0 register set value
1	0	Cleared by TCLR1n input or upon match of TMENC1n count value and the CM1n0 register set value
1	1	Not cleared

- Clearing by match of the TMENC1n count value and the CM1n0 register set value is valid only during a TMENC1n count up operation (TMENC1n is not cleared during a TMENC1n count down operation).
- In general-purpose timer mode (TUM1n.CMDn bit = 0), the CLRn1 and CLRn0 bit settings are invalid.
- In UDC mode B (TUM1n.MSELn bit = 1), the CLRn1 and CLRn0 bit settings are invalid.
- When clearing by TCLR1n has been enabled by bits CLRn1 and CLRn0, clearing is performed regardless of whether the value of the TM1CEn bit is 1 or 0.

Note V850E/IA4 only

(3) Capture/compare control register 1n (CCR1n)

The CCR1n register specifies the operation mode of the CC1n0 and CC1n1 registers.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. Overwriting the CCR1n register during TMENC1n operation (TMC1n.TM1CEn bit = 1) is prohibited.

2. The TCUD1n pin is used alternately for the UDC mode and the external capture function. Therefore, in the UDC mode, the external capture function cannot be used.

3. The TCLR1n pin is used alternately for the UDC mode and the external capture function. Therefore, when the TCLR1n input is used in UDC mode A, the external capture function cannot be used.

After reset: 00H R/W Address: CCR10 FFFFF58AH, CCR11 FFFFF5AAH[Note]

	7	6	5	4	3	2	1	0
CCR1n	0	0	0	0	0	0	CMSn1	CMSn0

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

CMSn1	CC1n1 register operation mode specification
0	Operates as capture register
1	Operates as compare register

CMSn0	CC1n0 register operation mode specification
0	Operates as capture register
1	Operates as compare register

Note V850E/IA4 only

(4) Valid edge select register 1n (SESA1n)

The SESA1n register specifies the valid edge of external interrupt request signals from the external pins (TIUD1n, TCUD1n, TCLR1n).

The valid edge (rising edge, falling edge, or both edges) can be specified independently for each pin.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. Changing the values of the SESA1n register bits during TMENC1n operation (TMC1n.TM1CEn bit = 1) is prohibited.

2. Before setting the trigger mode of the TIUD1n, TCUD1n, and TCLR1n pins, set the PMC3 and PMC5 registers. If the PMC3 and PMC5 registers are set after the SESA1n register has been set, an illegal interrupt, incorrect counting, and incorrect clearing may occur, depending on the timing of setting the PMC3 and PMC5 registers.

After reset: 00H R/W Address: SESA10 FFFFF58DH, SESA11 FFFFF5ADH[Note]

SESA1n

7	6	5	4	3	2	1	0
TESUDn1	TESUDn0	CESUDn1	CESUDn0	IESUD1n1	IESUD1n0	IESUD0n1	IESUD0n0

V850E/IA3 n = 0: TESUDn1, TESUDn0 → TIUD1n, TCUD1n; CESUDn1, CESUDn0 → TCLR1n; IESUD1n1, IESUD1n0 → CC1n1 capture trigger; IESUD0n1, IESUD0n0 → CC1n0 capture trigger

V850E/IA4 n = 0, 1

TESUDn1	TESUDn0	Specification of valid edge of TIUD1n and TCUD1n pins
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

- The settings of the TESUDn1 and TESUDn0 bits are only valid in UDC mode A and UDC mode B.
- If mode 4 is specified as the operation mode of TMENC1n (specified by the PRM1n.PRM1n2 to PRM1n.PRM1n0 bits), the valid edge specifications for the TIUD1n and TCUD1n pins (TESUDn1 and TESUDn0 bits) are not valid.

CESUDn1	CESUDn0	Specification of valid edge of TCLR1n pin
0	0	Falling edge (TMENC1n cleared after edge detection)
0	1	Rising edge (TMENC1n cleared after edge detection)
1	0	Low level (TMENC1n cleared status held)
1	1	High level (TMENC1n cleared status held)

- The settings of the CESUDn1 and CESUDn0 bits are valid only in UDC mode A.

IESUD1n1	IESUD1n0	Specification of valid edge of CC1n1 register capture trigger
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

The TCLR1n pin or TCUD1n pin can be selected by the CSL1n.CSL1n0 bit to input the capture trigger signal to the CC1n1 register.

IESUD0n1	IESUD0n0	Specification of valid edge of CC1n0 register capture trigger
0	0	Falling edge
0	1	Rising edge
1	0	Setting prohibited
1	1	Both rising and falling edges

The capture trigger signal of the CC1n0 register is input from the TCUD1n pin.

Note V850E/IA4 only

(5) Prescaler mode register 1n (PRM1n)

The PRM1n register is used to perform the following selections.

- Selection of count clock in general-purpose timer mode (TUM1n.CMDn bit = 0)
- Selection of count operation mode in UDC mode (CMDn bit = 1)

This register can be read or written in 8-bit or 1-bit units.
Reset sets this register to 07H.

Cautions 1. Overwriting the PRM1n register during TMENC1n operation (TMC1n.TM1CEn bit = 1) is prohibited.

2. In UDC mode (TUM1n.CMDn bit = 1), setting the values of the PRM1n2 bit to 0 is prohibited.

3. When TMENC1n is in mode 4, specification of the valid edge for the TIUD1n and TCUD1n pins is invalid.

After reset: 07H R/W Address: PRM10 FFFFF58EH, PRM11 FFFFF5AEH[Note]

	7	6	5	4	3	2	1	0
PRM1n	0	0	0	0	0	PRM1n2	PRM1n1	PRM1n0

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

PRM1n2	PRM1n1	PRM1n0	CMDn = 0	CMDn = 1	
			Count clock	Count clock	UDC mode
0	0	0	Setting prohibited	Setting prohibited	
0	0	1	fxx/4		
0	1	0	fxx/8		
0	1	1	fxx/16		
1	0	0	fxx/32	TIUD1n	Mode 1
1	0	1	fxx/64		Mode 2
1	1	0	fxx/128		Mode 3
1	1	1	fxx/256		Mode 4

Note V850E/IA4 only

Remark fxx: Peripheral clock

(a) In general-purpose timer mode (TUM1n.CMDn bit = 0)

The count clock is specified by the PRM1n2 to PRM1n0 bits.

(b) UDC mode (TUM1n.CMDn bit = 1)

The TMENC1n count triggers in the UDC mode are as follows.

Operation Mode	TMENC1n Operation
Mode 1	Count down when TCUD1n = high level Count up when TCUD1n = low level
Mode 2	Count up upon detection of valid edge of TIUD1n input Count down upon detection of valid edge of TCUD1n input
Mode 3	Count up upon detection of valid edge of TIUD1n input when TCUD1n = high level Count down upon detection of valid edge of TIUD1n input when TCUD1n = low level
Mode 4	Automatic judgment upon detection of both edges of TIUD1n input and both edges of TCUD1n input

(6) Status register 1n (STATUS1n)

The STATUS1n register indicates the operating status of TMENC1n.

This register is read-only in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R Address: STATUS10 FFFFF58FH, STATUS11 FFFFF5AFH[Note]

	7	6	5	4	3	<2>	<1>	<0>
STATUS1n	0	0	0	0	0	TM1UDFn	TM1OVFn	TM1UBDn

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

TM1UDFn	TMENC1n underflow flag
0	No TMENC1n count underflow
1	TMENC1n count underflow
The TM1UDFn bit is cleared (0) upon completion of a read access to the STATUS1n register from the CPU.	

TM1OVFn	TMENC1n overflow flag
0	No TMENC1n count overflow
1	TMENC1n count overflow
The TM1OVFn bit is cleared (0) upon completion of a read access to the STATUS1n register from the CPU.	

TM1UBDn	TMENC1n count up/down operation status
0	TMENC1n count up in progress
1	TMENC1n count down in progress
The state of the TM1UBDn bit differs according to the mode as follows. • The TM1UBDn bit is fixed to 0 in general-purpose timer mode (TUM1n.CMDn bit = 0). • The TM1UBDn bit indicates the TMENC1n count up/down status in UDC mode (TUM1n.CMDn bit = 1).	

Note V850E/IA4 only

(7) CC1n1 capture input select register (CSL1n)

The CSL1n register is used to select the TCLR1n or TCUD1n pin to input a capture signal when the CC1n1 register is used as a capture register.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: CSL10 FFFFF596H, CSL11 FFFFF5B6H[Note]

	7	6	5	4	3	2	1	0
CSL1n	0	0	0	0	0	0	0	CSL1n0

(V850E/IA3 n = 0)

CSL1n0	Selection of capture input signal of CC1n1 register
0	TCLR1n input
1	TCUD1n input

(V850E/IA4 n = 0, 1)

Note V850E/IA4 only

(8) Noise elimination time select register 1n (NRC1n)

The NRC1n register selects the sampling clock that is used to eliminate digital noise on the TIUD1n, TCUD1n, or TCLR1n pin. If a level is not detected on these pins five times in a row at the clock selected by the NRC1n register, the signal is eliminated as noise.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. If the input pulse is 4- to 5-clock width, it is undefined whether the pulse is detected as a valid edge or eliminated as noise. So that the pulse is actually detected as a valid edge, a pulse level must be input for the duration of 5 clocks or more.

2. If noise is generated in synchronization with the sampling clock, eliminate the noise by attaching a filter to the input pin.

3. Noise is not eliminated if the pin is used as a normal input port pin.

4. The noise elimination function starts operating when the TMC1n.TM1CEn bit is set to 1 (enabling count operations).

After reset: 00H R/W Address: NRC10 FFFFF598H, NRC11 FFFFF5B8H[Note]

	7	6	5	4	3	2	1	0
NRC1n	0	0	0	0	0	0	NRC1n1	NRC1n0

(V850E/IA3 n = 0)

NRC1n1	NRC1n0	Sampling clock selection
0	0	$f_{XX}/32$
0	1	$f_{XX}/16$
1	0	$f_{XX}/8$
1	1	$f_{XX}/4$

(V850E/IA4 n = 0, 1)

Note V850E/IA4 only

(9) Compare register 1n0 (CM1n0)

The CM1n0 register is a 16-bit register that always compares its value with the value of TMENC1n. When the value of the compare register matches the value of TMENC1n, an interrupt request signal is generated. The interrupt request signal generation timing in the various modes is described below.

- In the general-purpose timer mode (TUM1n.CMDn bit = 0) and UDC mode A (TUM1n.MSELn bit = 0), an interrupt request signal (INTCMn0) is generated upon occurrence of a match.
- In UDC mode B (TUM1n.MSELn bit = 1), an interrupt request signal (INTCMn0) is generated only upon occurrence of a match during an count up operation.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

Caution When the TMC1n.TM1CEn bit is 1, it is prohibited to overwrite the value of the CM1n0 register.

After reset: 0000H R/W Address: CM100 FFFFF582H, CM110 FFFFF5A2H[Note]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CM1n0																

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

Note V850E/IA4 only

(10) Compare register 1n1 (CM1n1)

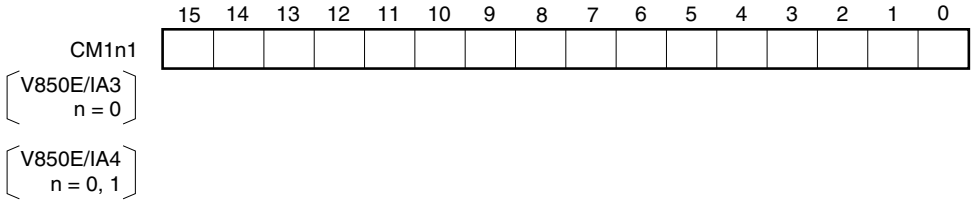
The CM1n1 register is a 16-bit register that always compares its value with the value of TMENC1n. When the value of the compare register matches the value of TMENC1n, an interrupt request signal is generated. The interrupt request signal generation timing in the various modes is described below.

- In the general-purpose timer mode (TUM1n.CMDn bit = 0) and UDC mode A (TUM1n.MSELn bit = 0), an interrupt request signal (INTCMn1) is generated upon occurrence of a match.
- In UDC mode B (TUM1n.MSELn bit = 1), an interrupt request signal (INTCMn1) is generated only upon occurrence of a match during a count down operation.

This register can be read or written in 16-bit units.
Reset sets this register to 0000H.

Caution When the TMC1n.TM1CEn bit is 1, it is prohibited to overwrite the value of the CM1n1 register.

After reset: 0000H R/W Address: CM101 FFFFF584H, CM111 FFFFF5A4H[Note]



Note V850E/IA4 only

(11) Capture/compare register 1n0 (CC1n0)

The CC1n0 register is a 16-bit register. This register can be specified as a capture register or as a compare register using the CCR1n register.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

Cautions 1. When used as a capture register (CCR1n.CMSn0 bit = 0), write access is prohibited.

2. When used as a compare register (CCR1n.CMSn0 bit = 1) and while TMENC1n is operating (TMC1n.TM1CEn bit = 1), overwriting the CC1n0 register values is prohibited.

3. When TMENC1n is stopped (TMC1n.TM1CEn bit = 0), the capture trigger is disabled.

4. When the operation mode is changed from capture register to compare register, set a new compare value.

5. Continuous reading of the CC1n0 register is prohibited. If the CC1n0 register is continuously read, the second read value may differ from the actual value. If the CC1n0 register must be read twice, be sure to read another register between the first and the second read operation.

After reset: 0000H R/W Address: CC100 FFFFF586H, CC110 FFFFF5A6H[Note]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1n0																

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

Note V850E/IA4 only

(a) When set as a capture register

When CC1n0 is set as a capture register, the valid edge of the corresponding external interrupt request signal (TCUD1n) is detected as the capture trigger. TMENC1n latches the count value in synchronization with the capture trigger (capture operation). The latched value is held in the capture register until the next capture operation.

The valid edge of external interrupt request signals (rising edge, falling edge, both rising/falling edges) is selected by the SESA1n register.

When the CC1n0 register is specified as a capture register, interrupt request signals are generated upon detection of the valid edge of the TCUD1n signal.

Caution The TCUD1n pin is used alternately for the UDC mode and the external capture function. Therefore, in the UDC mode, the external capture function cannot be used.

(b) When set as a compare register

When the CC1n0 register is set as a compare register, it always compares its own value with the value of TMENC1n. If the value of the CC1n0 register matches the value of TMENC1n, the CC1n0 register generates an interrupt request signal (INTCCn0).

(12) Capture/compare register 1n1 (CC1n1)

The CC1n1 register is a 16-bit register. This register can be specified as a capture register or as a compare register using the CCR1n register.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

Cautions 1. When used as a capture register (CCR1n.CMSn1 bit = 0), write access is prohibited.

2. When used as a compare register (CCR1n.CMSn1 bit = 1) and while TMENC1n is operating (TMC1n.TM1CEn bit = 1), overwriting the CC1n1 register values is prohibited.

3. When TMENC1n is stopped (TMC1n.TM1CEn bit = 0), the capture trigger is disabled.

4. When the operation mode is changed from capture register to compare register, newly set a compare value.

5. Continuous reading of the CC1n1 register is prohibited. If the CC1n1 register is continuously read, the second read value may differ from the actual value. If the CC1n1 register must be read twice, be sure to read another register between the first and the second read operation.

After reset: 0000H R/W Address: CC101 FFFFF588H, CC111 FFFFF5A8H[Note]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1n1																

(V850E/IA3 n = 0)

(V850E/IA4 n = 0, 1)

Note V850E/IA4 only

(a) When set as a capture register

When CC1n1 is set as a capture register, the valid edge of the corresponding external interrupt request signal (TCLR1n) is detected as the capture trigger. TMENC1n latches the count value in synchronization with the capture trigger (capture operation). The latched value is held in the capture register until the next capture operation.

The valid edge of external interrupt request signals (rising edge, falling edge, both rising/falling edges) is selected by the SESA1n register.

If the CC1n1 register is specified as a capture register, interrupt request signals are generated when the valid edge of either the TCLR1n or TCUD1n signal is detected.

Cautions 1. The TCUD1n pin is used alternately for the UDC mode and the external capture function. Therefore, in the UDC mode, the external capture function cannot be used.

2. The TCLR1n pin is used alternately for the UDC mode and the external capture function. Therefore, when the TCLR1n input is used in UDC mode A, the external capture function cannot be used.

(b) When set as a compare register

When the CC1n1 register is set as a compare register, it always compares its own value with the value of TMENC1n. If the value of the CC1n1 register matches the value of TMENC1n, the CC1n1 register generates an interrupt request signal (INTCCn1).

8.5 Operation

8.5.1 Operation in general-purpose timer mode

TMENC1n can perform the following operations in the general-purpose timer mode.

(1) Interval operation (when TMC1n. ENMDn bit = 1)

TMENC1n and the CM1n0 register always compare their values and the INTCMn0 interrupt request signal is generated upon occurrence of a match. TMENC1n is cleared (0000H) at the count clock following the match. Furthermore, when one more count clock is input, TMENC1n counts up to 0001H.

The interval time can be calculated by the following formula.

Interval time = (CM1n0 register value + 1) × TMENC1n count clock rate

(2) Free-running operation (when TMC1n.ENMDn bit = 0)

TMENC1n performs a full count operation from 0000H to FFFFH, and after the STATUS1n.TM1OVFn bit is set (1), TMENC1n is cleared to 0000H at the next count clock and resumes counting.

The free-running cycle can be calculated by the following formula.

Free-running cycle = 65,536 × TMENC1n count clock rate

(3) Compare function

TMENC1n connects two compare register (CM1n0, CM1n1) channels and two capture/compare register (CC1n0, CC1n1) channels.

When the TMENC1n count value and the set value of one of the compare registers match, a match interrupt request signal (INTCMn0, INTCMn1, INTCCn0[Note], INTCCn1[Note]) is output. Particularly in the case of a interval operation, TMENC1n is cleared upon generation of the INTCMn0 interrupt.

Note This match interrupt request signal is generated when the CC1n0 and CC1n1 registers are set to the compare register mode.

(4) Capture function

TMENC1n connects two capture/compare register (CC1n0, CC1n1) channels.

When the CC1n0 and CC1n1 registers are set to the capture register mode, the value of TMENC1n is captured in synchronization with the corresponding capture trigger signal.

Furthermore, an interrupt request signal (INTCCn0, INTCCn1) is generated by the valid edge of the TCUD1n and TCLR1n input signals specified as the capture trigger signals.

Table 8-3. Capture Trigger Signal to 16-Bit Capture Register

Capture Register	Capture Trigger Signal
CC1n0	TCUD1n
CC1n1	TCUD1n or TCLR1n

Remark The CC1n0 and CC1n1 registers are capture/compare registers. Which of these registers is used is specified by the CCR1n register.

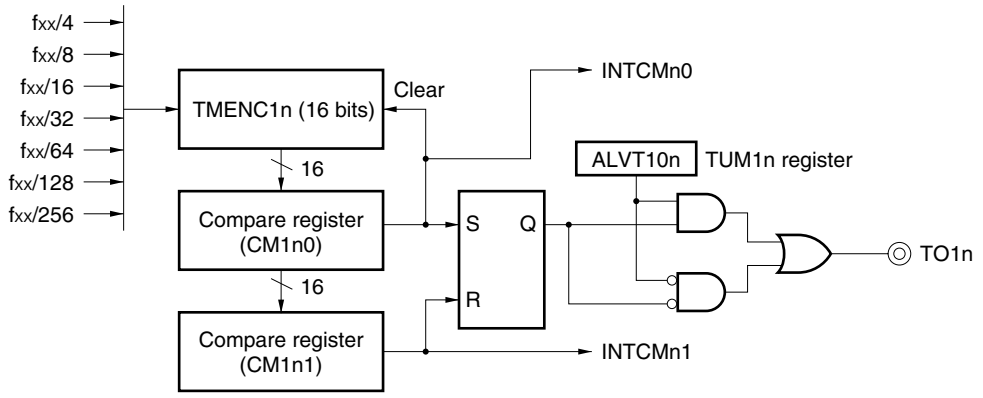
The valid edge of the capture trigger is specified by the SESA1n register. If both the rising and falling edges are selected as the capture trigger, it is possible to measure the width of a pulse input externally. If a single edge is selected as the capture trigger, the input pulse cycle can be measured.

<R> **(5) PWM output operation**

PWM output operation is performed from the TO1n pin by setting TMENC1n to the general-purpose timer mode (TUM1n.CMDn bit = 0) using TUM1n register.

The resolution is 16 bits, and the count clock can be selected from among seven internal clocks (fxx/4, fxx/8, fxx/16, fxx/32, fxx/64, fxx/128, fxx/256).

Figure 8-2. TMENC1n Block Diagram (During PWM Output Operation)



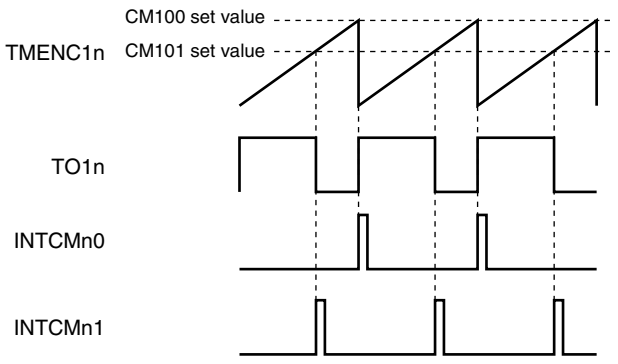
Remark fxx: Base clock

(a) Description of operation

The CM1n0 register is a compare register used to set the PWM output cycle. When the value of this register matches the value of TMENC1n, the INTCMn0 interrupt is generated. The compare match is saved by hardware, and TMENC1n is cleared at the next count clock after the match.

The CM1n1 register is a compare register used to set the PWM output duty. Set the duty required for the PWM cycle.

Figure 8-3. PWM Signal Output Example (When ALVT10n Bit = 0 Is Set)



Cautions 1. Changing the values of the CM1n0 and CM1n1 registers is prohibited during TMENC1n operation (TMC1n.TM1CEn bit = 1).

2. Changing the value of the TUM1n.ALVT10n bit is prohibited during TMENC1n operation.

3. PWM signal output is performed from the second PWM cycle after the TM1CEn bit is set (to 1).

8.5.2 Operation in UDC mode

(1) Overview of operation in UDC mode

The count clock input to TMENC1n in the UDC mode (TUM1n.CMDn bit = 1) can only be externally input from the TIUD1n and TCUD1n pins. Up/count down judgment in the UDC mode is determined based on the phase difference of the TIUD1n and TCUD1n pin inputs according to the PRM1n register setting (there is a total of four choices).

Table 8-4. List of Count Operations in UDC Mode

PRM1n Register			Operation Mode	TMENC1n Operation
PRM1n2	PRM1n1	PRM1n0		
1	0	0	Mode 1	Count down when TCUD1n = high level Count up when TCUD1n = low level
1	0	1	Mode 2	Count up upon detection of valid edge of TIUD1n input Count down upon detection of valid edge of TCUD1n input
1	1	0	Mode 3	Count up upon detection of valid edge of TIUD1n input when TCUD1n = high level Count down upon detection of valid edge of TIUD1n input when TCUD1n = low level
1	1	1	Mode 4	Automatic judgment upon detection of both edges of TIUD1n input and both edges of TCUD1n input

The UDC mode is further divided into two modes according to the TMENC1n clear conditions (a count operation is performed only with TIUD1n and TCUD1n input in both modes).

- **UDC mode A (TUM1n.CMDn bit = 1, MSELn bit = 0)**

  The TMENC1n clear source can be selected as only external clear input (TCLR1n), the signal indicating a match between the TMENC1n count value and the CM1n0 register set value during an count up operation, or the logical sum (OR) of the two signals, using the TMC1n.CLRn1 and TMC1n.CLRn0 bits.

  TMENC1n can transfer the value of the CM1n0 register upon occurrence of a TMENC1n underflow.

- **UDC mode B (TUM1n.CMDn bit = 1, MSELn bit = 1)**

  The status of TMENC1n after a match of the TMENC1n count value and the CM1n0 register set value is as follows.

  <1> In the case of an count up operation, TMENC1n is cleared (0000H), and the INTCMn0 interrupt request signal is generated.

  <2> In the case of a count down operation, the TMENC1n count value is decremented (−1).

  The status of TMENC1n after a match of the TMENC1n count value and the CM1n1 register set value is as follows.

  <1> In the case of an count up operation, the TMENC1n count value is incremented (+1).

  <2> In the case of a count down operation, TMENC1n is cleared (0000H), and the INTCMn1 interrupt request signal is generated.

(2) Up/count down operation in UDC mode

TMENC1n up/count down judgment in the UDC mode is determined based on the phase difference of the TIUD1n and TCUD1n pin inputs according to the PRM1n register setting.

(a) Mode 1 (PRM1n.PRM1n2 bit = 1, PRM1n.PRM1n1 bit = 0, PRM1n.PRM1n0 bit = 0)

In mode 1, the following count operations are performed based on the level of the TCUD1n pin upon detection of the valid edge of the TIUD1n pin.

- TMENC1n count down operation when TCUD1n pin = high level
- TMENC1n count up operation when TCUD1n pin = low level

Figure 8-4. Mode 1 (When Rising Edge Is Specified as Valid Edge of TIUD1n Pin)

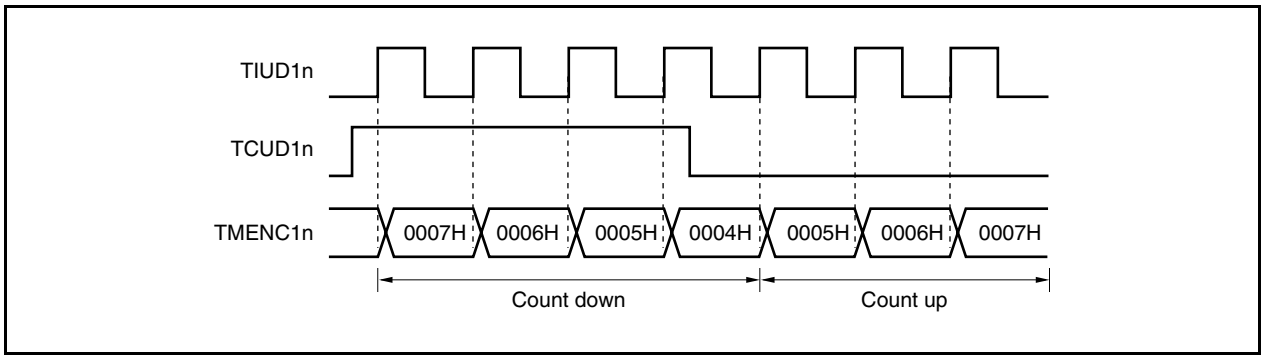
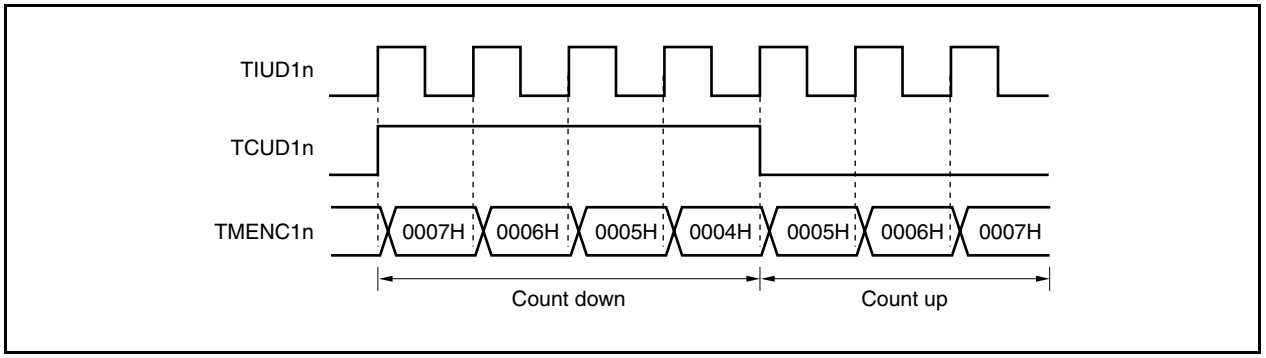


Figure 8-5. Mode 1 (When Rising Edge Is Specified as Valid Edge of TIUD1n Pin): In Case of Simultaneous TIUD1n, TCUD1n Pin Edge Timing



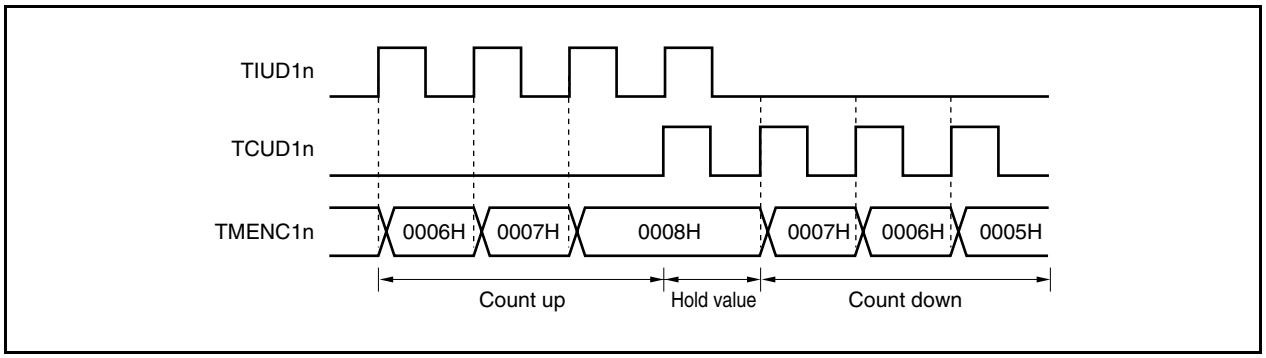
(b) Mode 2 (PRM1n.PRM1n2 bit = 1, PRM1n.PRM1n1 bit = 0, PRM1n.PRM1n0 bit = 1)

The count conditions in mode 2 are as follows.

- TMENC1n count up upon detection of valid edge of TIUD1n pin
- TMENC1n count down upon detection of valid edge of TCUD1n pin

Caution If the count clock is simultaneously input to the TIUD1n pin and the TCUD1n pin, a count operation is not performed and the immediately preceding value is held.

Figure 8-6. Mode 2 (When Rising Edge Is Specified as Valid Edge of TIUD1n, TCUD1n Pins)



(c) Mode 3 (PRM1n.PRM1n2 = 1, PRM1n.PRM1n1 = 1, PRM1n.PRM1n0 = 0)

In mode 3, when two signals 90 degrees out of phase are input to the TIUD1n and TCUD1n pins, the level of the TCUD1n pin is sampled at the input of the valid edge of the TIUD1n pin (see **Figure 8-7**).

If the TCUD1n pin level sampled at the valid edge input to the TIUD1n pin is low, TMENC1n counts down when the valid edge is input to the TIUD1n pin.

If the TCUD1n pin level sampled at the valid edge input to the TIUD1n pin is high, TMENC1n counts up when the valid edge is input to the TIUD1n pin.

Figure 8-7. Mode 3 (When Rising Edge Is Specified as Valid Edge of TIUD1n Pin)

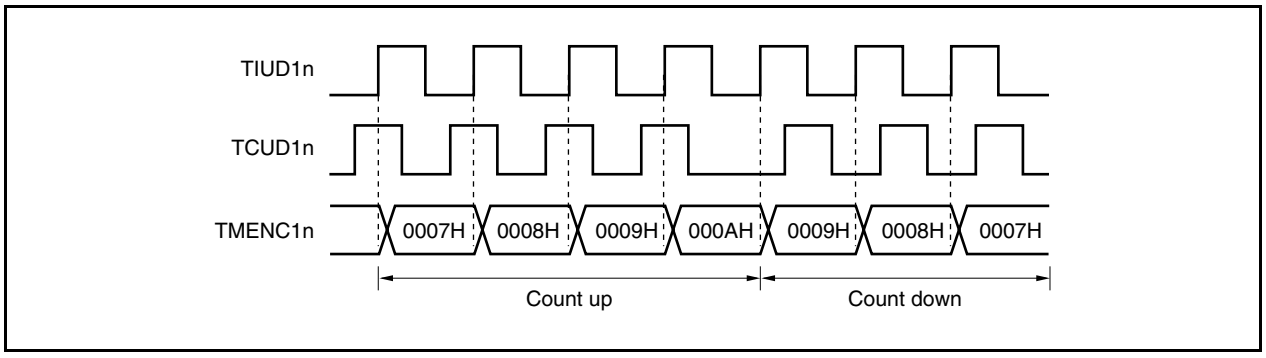
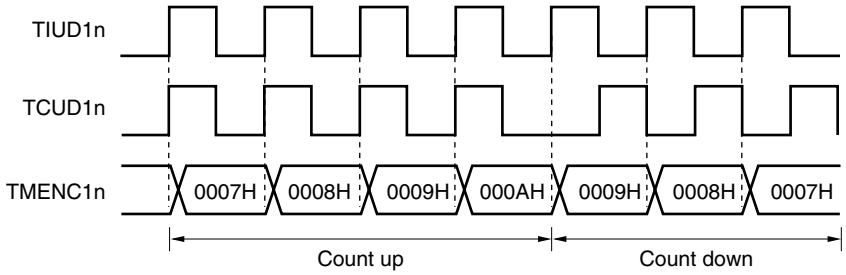


Figure 8-8. Mode 3 (When Rising Edge Is Specified as Valid Edge of TIUD1n Pin): In Case of Simultaneous TIUD1n, TCUD1n Pin Edge Timing

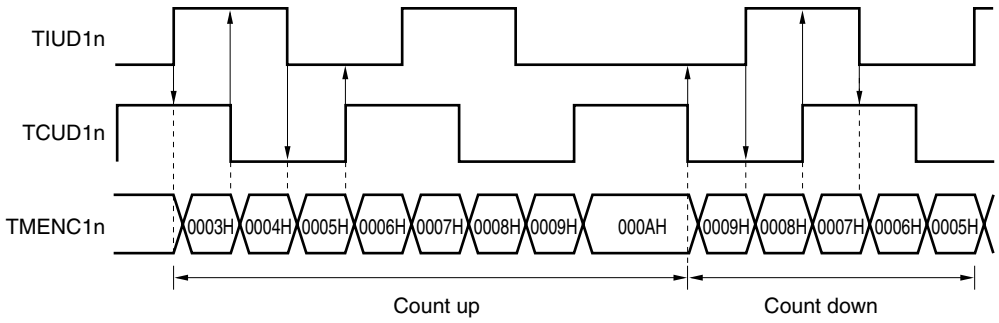


(d) Mode 4 (PRM1n.PRM1n2 = 1, PRM1n.PRM1n1 = 1, PRM1n.PRM1n0 = 1)

In mode 4, when two signals out of phase are input to the TIUD1n and TCUD1n pins, the up/down operation is automatically judged and counting is performed according to the timing shown in Figure 8-9.

In mode 4, counting is executed at both the rising and falling edges of the two signals input to the TIUD1n and TCUD1n pins. Therefore, TMENC1n counts four times per cycle of an input signal (×4 count).

Figure 8-9. Mode 4



Cautions 1. When mode 4 is specified as the operation mode of TMENC1n, the valid edge specifications for the TIUD1n and TCUD1n pins are not valid.

2. If the TIUD1n pin edge and TCUD1n pin edge are input simultaneously in mode 4, TMENC1n continues the same count operation (up or down) it was performing immediately before the input.

(3) Operation in UDC mode A

(a) Interval operation

The operations at the count clock following a match of the TMENC1n count value and the CM1n0 register set value are as follows.

- In case of count up operation: TMENC1n is cleared (0000H) and the INTCMn0 interrupt request signal is generated.
- In case of count down operation: The TMENC1n count value is decremented (−1) and the INTCMn0 interrupt request signal is generated.

Remark The interval operation can be combined with a transfer operation.

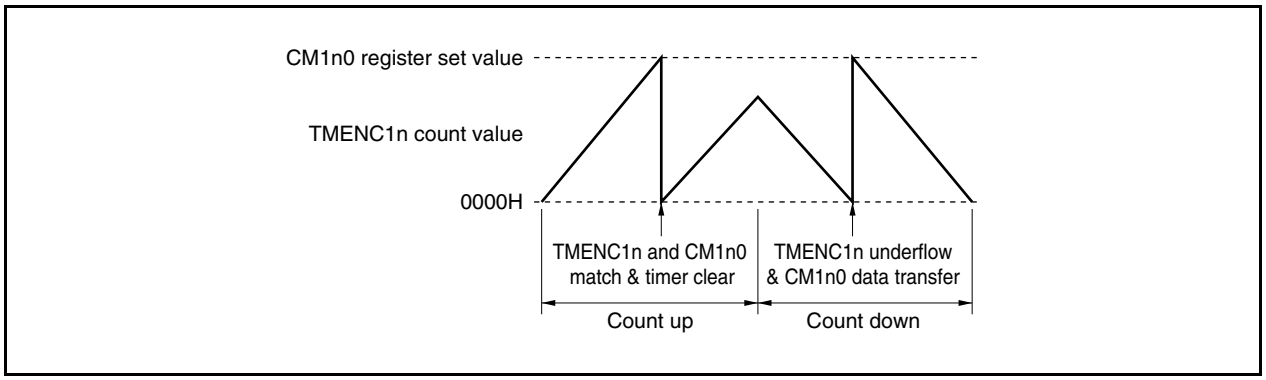
(b) Transfer operation

If TMENC1n = 0000H during counting down when the TMC1n.RLENn bit = 1, the set value of the CM1n0 register is transferred to TMENC1n at the next count clock.

Remarks 1. Transfer enable/disable can be set using the TMC1n.RLENn bit.

2. The transfer operation can be combined with an interval operation.

Figure 8-10. Example of TMENC1n Operation When Interval Operation and Transfer Operation Are Combined



(c) Compare function

TMENC1n connects two compare register (CM1n0, CM1n1) channels and two capture/compare register (CC1n0, CC1n1) channels.

When the TMENC1n count value and the set value of one of the compare registers match, a match interrupt request signal (INTCMn0, INTCMn1, INTCCn0[Note], INTCCn1[Note]) is output.

Note This match interrupt request signal is generated when the CC1n0 and CC1n1 registers are set to the compare register mode.

(d) Capture function

TMENC1n connects two capture/compare register (CC1n0, CC1n1) channels.

When the CC1n0 and CC1n1 registers are set to the capture register mode, the value of TMENC1n is captured in synchronization with the corresponding capture trigger signal. Also, a capture interrupt request signal (INTCCn0, INTCCn1) is generated upon detection of the valid edge.

(4) Operation in UDC mode B

(a) Basic operation

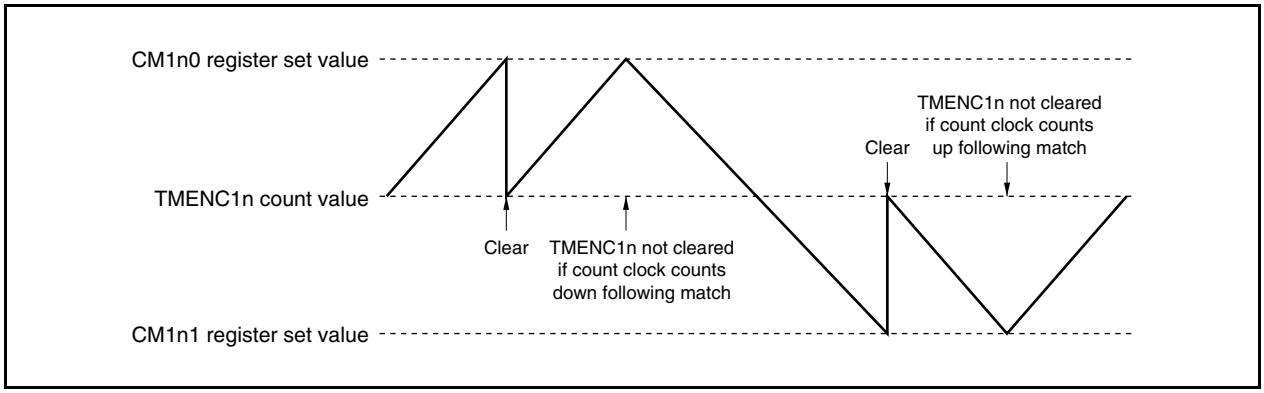
The operations at the next count clock after the count value of TMENC1n and the CM1n0 register set value match when TMENC1n is in UDC mode B are as follows.

- In case of count up operation: TMENC1n is cleared (0000H) and the INTCMn0 interrupt request signal is generated.
- In case of count down operation: The TMENC1n count value is decremented (−1).

The operations at the next count clock after the count value of TMENC1n and the CM1n1 register set value match when TMENC1n is in UDC mode B are as follows.

- In case of count up operation: The TMENC1n count value is incremented (+1).
- In case of count down operation: TMENC1n is cleared (0000H) and the INTCMn1 interrupt request signal is generated.

Figure 8-11. Example of TMENC1n Operation in UDC Mode



(b) Compare function

TMENC1n connects two compare register (CM1n0, CM1n1) channels and two capture/compare register (CC1n0, CC1n1) channels.

When the TMENC1n count value and the set value of one of the compare registers match, a match interrupt request signal (INTCMn0 (only during count up operation), INTCMn1 (only during count down operation), INTCCn0[Note], INTCCn1[Note]) is output.

Note This match interrupt request signal is generated when the CC1n0 and CC1n1 registers are set to the compare register mode.

(c) Capture function

TMENC1n connects two capture/compare register (CC1n0, CC1n1) channels.

When the CC1n0 and CC1n1 registers are set to the capture register mode, the value of TMENC1n is captured in synchronization with the corresponding capture trigger signal. Also, a capture interrupt request signal (INTCCn0, INTCCn1) is generated upon detection of the valid edge.

8.6 Supplementary Description of Internal Operation

8.6.1 Clearing of count value in UDC mode B

When TMENC1n is in UDC mode B, the conditions to clear the count value are as follows.

- In case of TMENC1n count up operation: TMENC1n count value is cleared upon match with CM1n0 register
- In case of TMENC1n count down operation: TMENC1n count value is cleared upon match with CM1n1 register

Figure 8-12. Clear Operation After Match of CM1n0 Register Set Value and TMENC1n Count Value

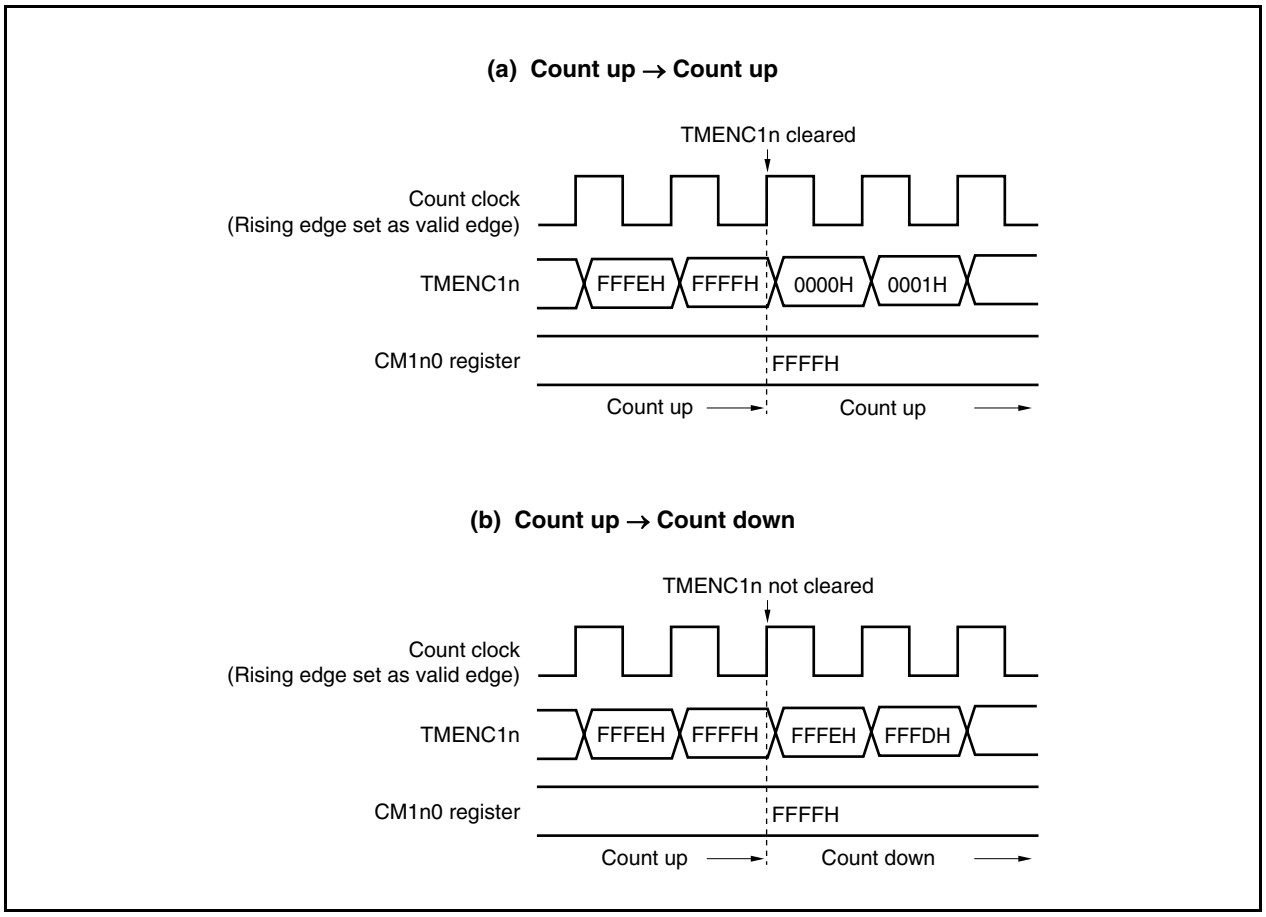
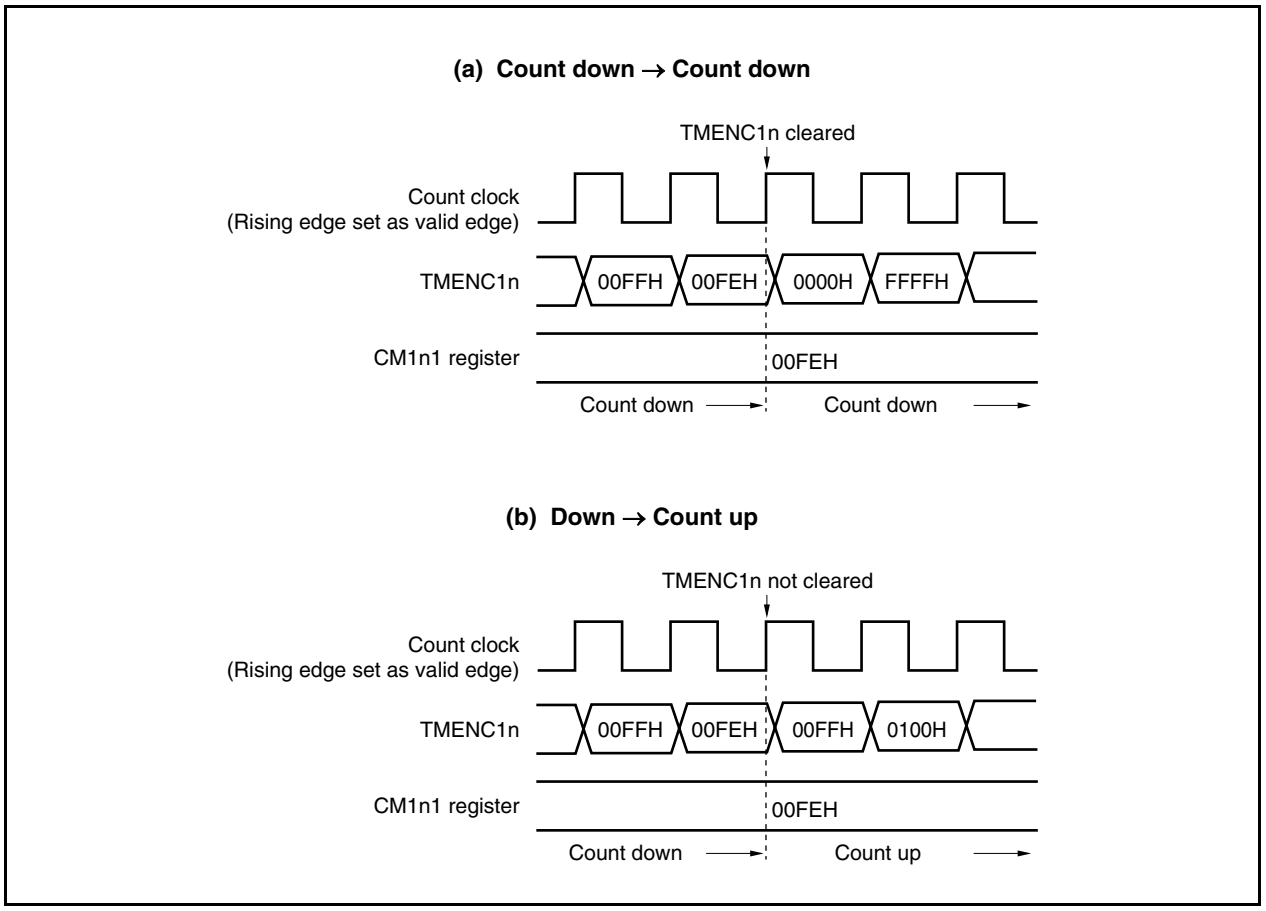


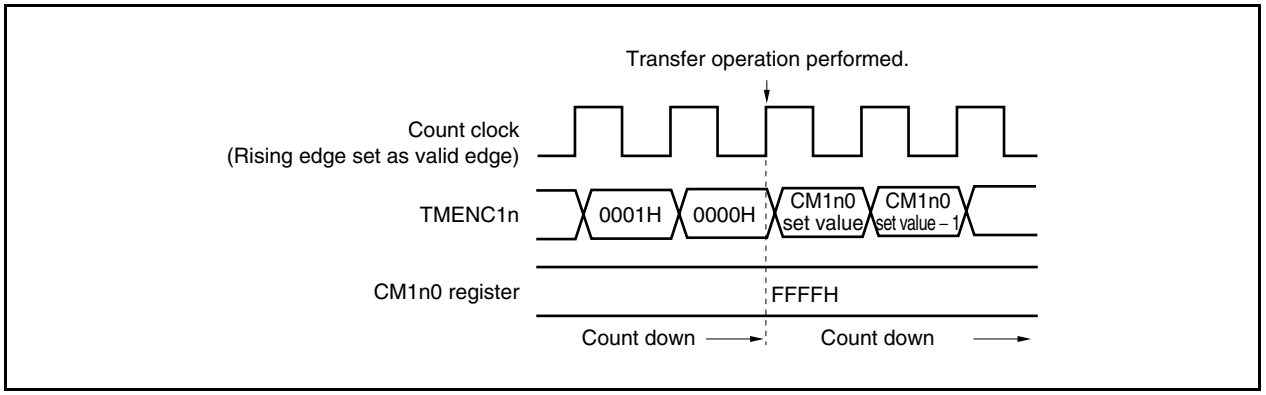
Figure 8-13. Clear Operation After Match of CM1n1 Register Set Value and TMENC1n Count Value



8.6.2 Transfer operation

If TMENC1n = 0000H during counting down when the TMC1n.RLENn bit = 1 in UDC mode A, the set value of the CM1n0 register is transferred to TMENC1n at the next count clock. The transfer operation is not performed during counting up.

Figure 8-14. Internal Operation During Transfer Operation

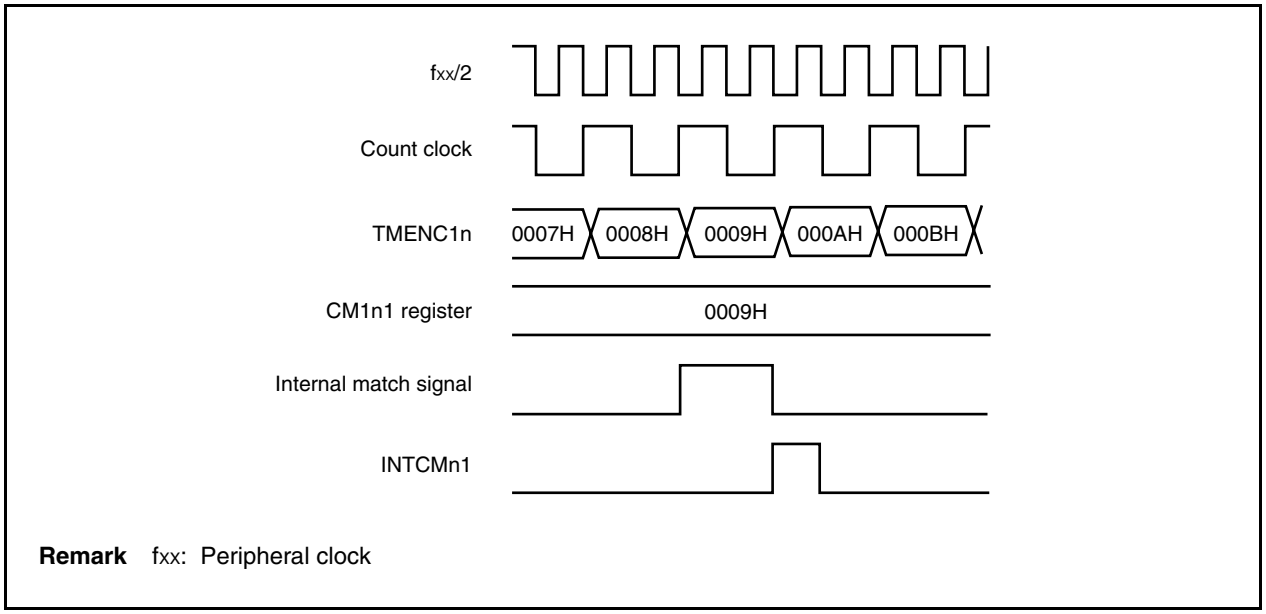


8.6.3 Interrupt request signal output upon compare match

An interrupt request signal is output when the count value of TMENC1n matches the set value of the CM1n0, CM1n1, CC1n0[Note], or CC1n1[Note] register. The interrupt generation timing is as follows.

Note When the CC1n0 and CC1n1 registers are set to the compare register mode.

Figure 8-15. Interrupt Request Signal Output upon Compare Match (CM1n1 with Operation Mode Set to General-Purpose Timer Mode and Count Clock Set to fxx/4)



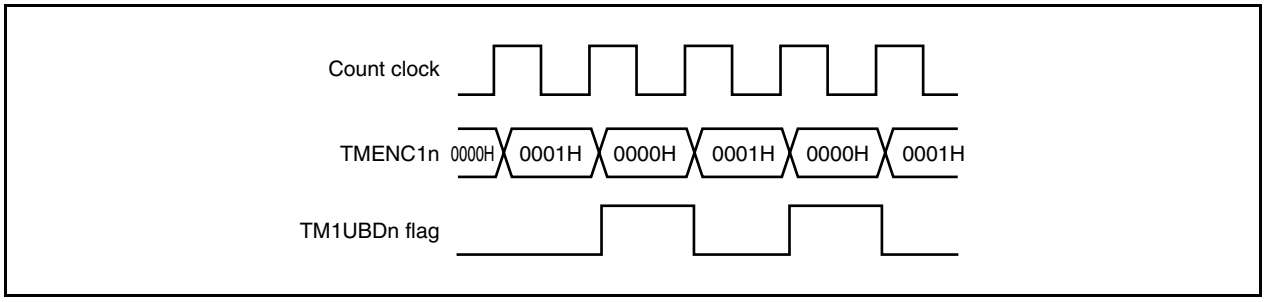
Remark fxx: Peripheral clock

An interrupt request signal such as the one illustrated in Figure 8-15 is output at the next count clock following a match of the TMENC1n count value and the set value of the corresponding compare register.

8.6.4 TM1UBDn flag (bit 0 of STATUS1n register) operation

In the UDC mode (TUM1n.CMDn bit = 1), the TM1UBDn flag changes as follows during a TMENC1n count up/count down operation at every internal operation clock.

Figure 8-16. TM1UBDn Flag Operation



CHAPTER 9 16-BIT INTERVAL TIMER M (TMM)

9.1 Overview

- Interval function
- 8 clocks selectable
- 16-bit counter × 1
  (The 16-bit counter cannot be read during timer count operation.)
- Compare register × 1
  (The compare register cannot be written during timer counter operation.)
- Compare match interrupt × 1

Timer M supports only the clear & start mode. The free-running timer mode is not supported.

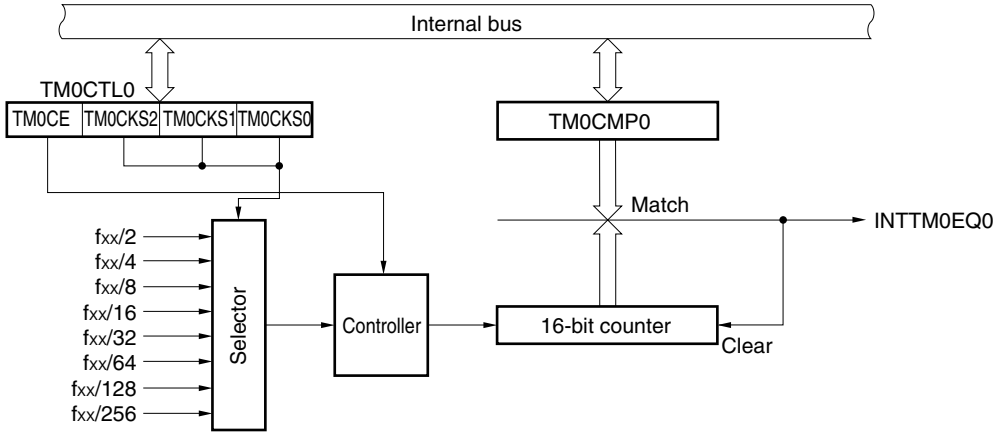
9.2 Configuration

TMM0 includes the following hardware.

Table 9-1. Configuration of TMM0

Item	Configuration
Timer register	16-bit counter
Register	TMM0 compare register 0 (TM0CMP0)
Control register	TMM0 control register 0 (TM0CTL0)

Figure 9-1. Block Diagram of TMM0



Remark fxx: Peripheral clock frequency

(1) 16-bit counter

This is a 16-bit counter that counts the internal clock.

The 16-bit counter cannot be read or written.

(2) TMM0 compare register 0 (TM0CMP0)

The TM0CMP0 register is a 16-bit compare register.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

The same value can always be written to the TM0CMP0 register by software.

Rewriting the TM0CMP0 register is prohibited during TMM0 operation (TM0CTL0.TM0CE bit = 1).

After reset: 0000H R/W Address: FFFFF544H

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM0CMP0																

9.3 Control Register

(1) TMM0 control register 0 (TM0CTL0)

The TM0CTL0 register is an 8-bit register that controls the TMM0 operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

The same value can always be written to the TM0CTL0 register by software.

After reset: 00H R/W Address: FFFFF540H

	<7>	6	5	4	3	2	1	0
TM0CTL0	TM0CE	0	0	0	0	TM0CKS2	TM0CKS1	TM0CKS0

TM0CE	Internal clock operation enable/disable specification
0	TMM0 operation disabled (16-bit counter reset asynchronously)
1	TMM0 operation enabled. Start operation clock supply. Start TMM0 operation.
The internal clock control and internal circuit reset for TMM0 are performed asynchronously with the TM0CE bit. When the TM0CE bit is cleared to 0, the internal clock of TMM0 is stopped (fixed to low level) and 16-bit counter is reset asynchronously.	

TM0CKS2	TM0CKS1	TM0CKS0	Count clock selection
0	0	0	$f_{XX}/2$
0	0	1	$f_{XX}/4$
0	1	0	$f_{XX}/8$
0	1	1	$f_{XX}/16$
1	0	0	$f_{XX}/32$
1	0	1	$f_{XX}/64$
1	1	0	$f_{XX}/128$
1	1	1	$f_{XX}/256$

Cautions 1. Set the TM0CKS2 to TM0CKS0 bits when the TM0CE bit = 0. However, when changing the value of TM0CE from 0 to 1, it is impossible to set the value of the TM0CKS2 to TM0CKS0 bits simultaneously.

2. Be sure to clear bits 3 to 6 to "0".

Remark f_{XX}: Peripheral clock frequency

9.4 Operation

9.4.1 Interval timer mode

In the interval timer mode, an interrupt request signal (INTTM0EQ0) is generated at the interval set by the TM0CMP0 register if the TM0CTL0.TM0CE bit is set to 1.

Figure 9-2. Configuration of Interval Timer

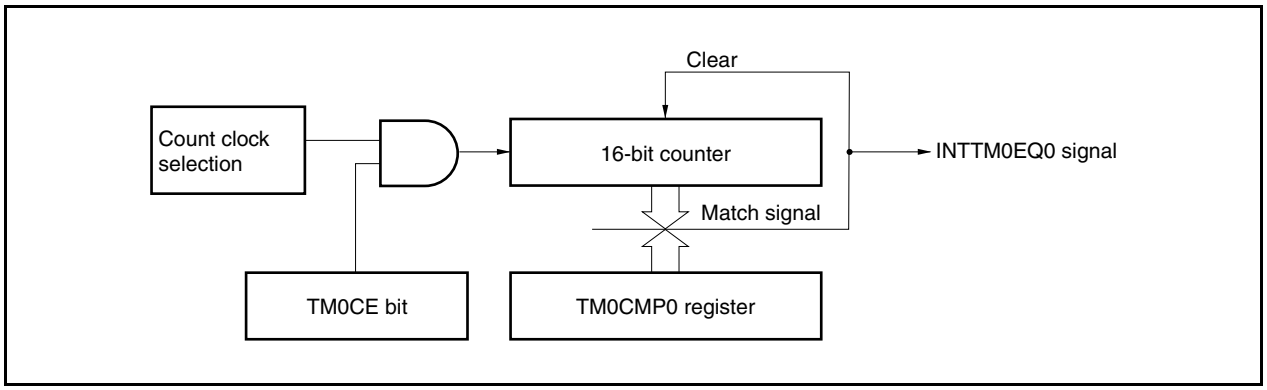
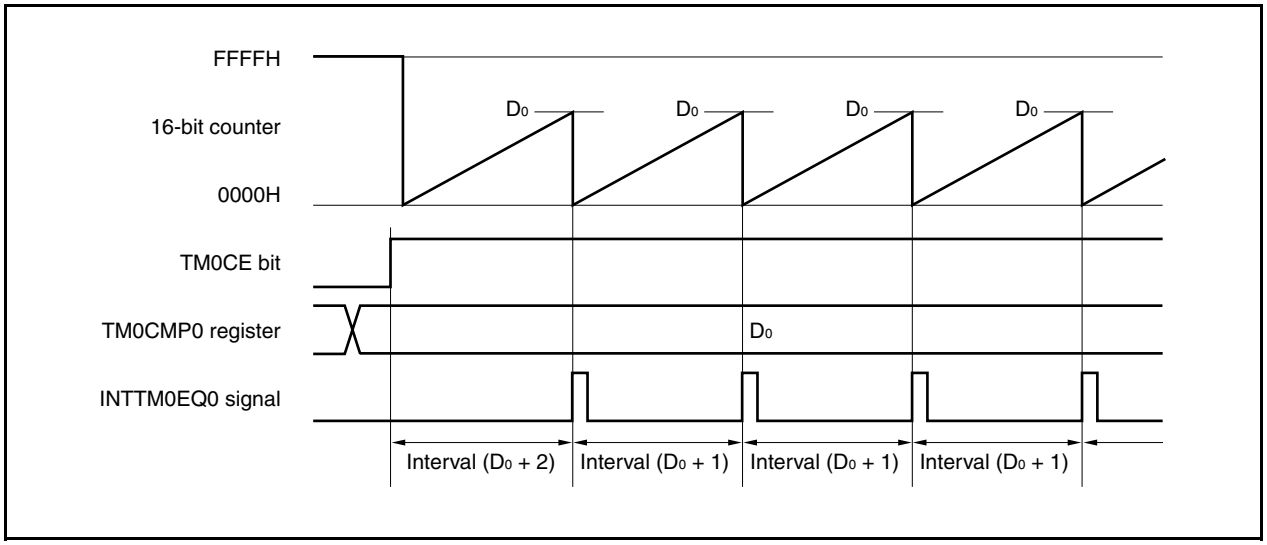


Figure 9-3. Basic Timing of Operation in Interval Timer Mode



When the TM0CE bit is set to 1, the value of the 16-bit counter is cleared from FFFFH to 0000H in synchronization with the count clock, and the counter starts counting.

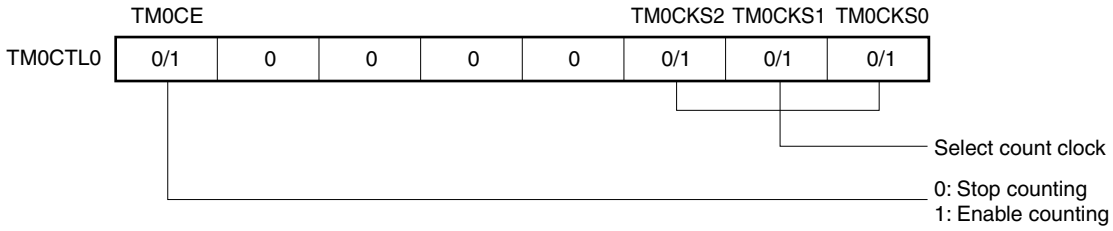
When the count value of the 16-bit counter matches the value of the TM0CMP0 register, the 16-bit counter is cleared to 0000H, and a compare match interrupt request signal (INTTM0EQ0) is generated.

The interval can be calculated by the following expression.

Interval = (Set value of TM0CMP0 register + 1) × Count clock cycle

Figure 9-4. Register Setting for Interval Timer Mode Operation

(a) TMM0 control register 0 (TM0CTL0)



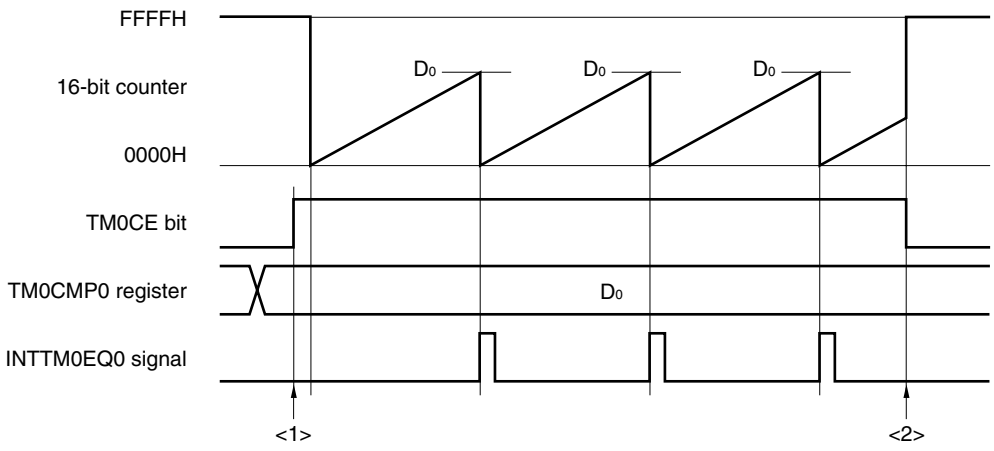
(b) TMM0 compare register 0 (TM0CMP0)

If the TM0CMP0 register is set to D_0, the interval is as follows.

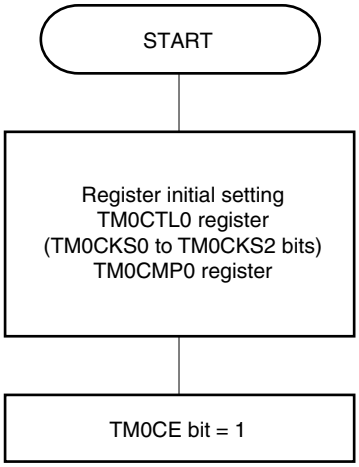
Interval = (D_0 + 1) × Count clock cycle

(1) Interval timer mode operation flow

Figure 9-5. Software Processing Flow in Interval Timer Mode



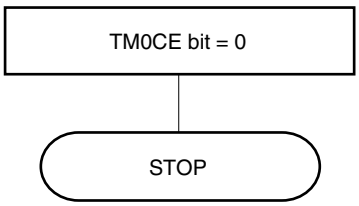
<1> Count operation start flow



Initial setting of these registers is performed before setting the TM0CE bit to 1.

The TM0CKS0 to TM0CKS2 bits cannot be set at the same time when counting has been started (TM0CE bit = 1).

<2> Count operation stop flow

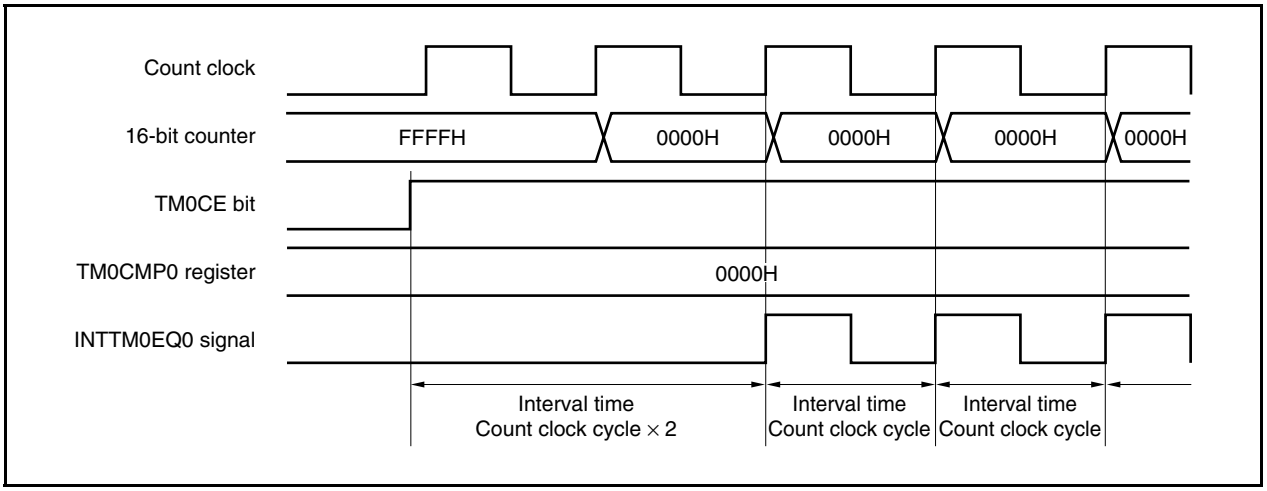


The counter is initialized and counting is stopped by clearing the TM0CE bit to 0.

(2) Interval timer mode operation timing

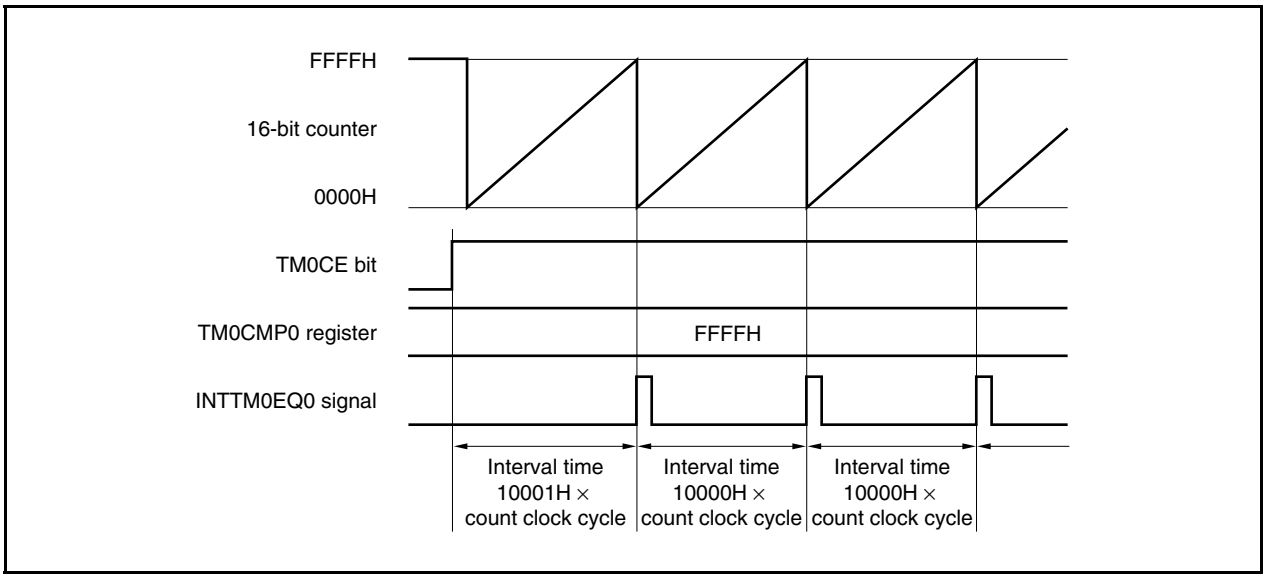
(a) Operation if TM0CMP0 register is set to 0000H

If the TM0CMP0 register is set to 0000H, the INTTM0EQ0 signal is generated at each count clock. The value of the 16-bit counter is always 0000H.



(b) Operation if TM0CMP0 register is set to FFFFH

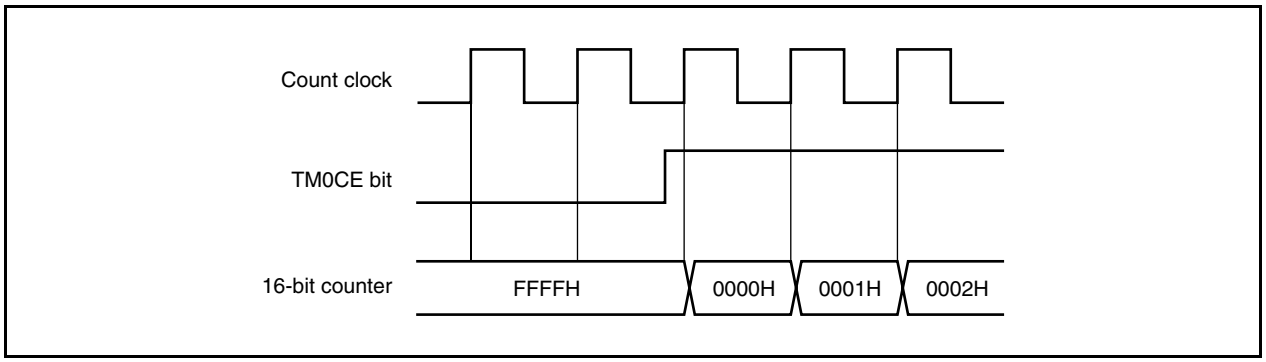
If the TM0CMP0 register is set to FFFFH, the 16-bit counter counts up to FFFFH. The counter is cleared to 0000H in synchronization with the next count-up timing. The INTTM0EQ0 signal is generated.



9.5 Cautions

(1) Error on starting timer

It takes one clock to generate the first compare match interrupt request signal (INTTM0EQ0) after the TM0CTL0.TM0CE bit is set to 1 and TMM0 is started. This is because the value of the 16-bit counter is FFFFH when the TM0CE bit = 0 and TMM0 is started asynchronously to the count clock.



(2) Rewriting the TM0CMP0 and TM0CTL0 registers is prohibited while TMM0 is operating.

If these registers are rewritten while the TM0CTL0.TM0CE bit is 1, the operation cannot be guaranteed.

If they are rewritten by mistake, clear the TM0CE bit to 0, and re-set the registers.

CHAPTER 10 MOTOR CONTROL FUNCTION

10.1 Functional Overview

Timer Qn (TMQn) and the TMQn option (TMQOPn) can be used as an inverter function that controls a motor. It performs a tuning operation with timer Pn (TMPn) and A/D conversion of A/D converters 0 and 1 can be started when the value of TMQn matches the value of TMPn. The following operations can be performed as motor control functions.

- 6-phase PWM output function with 16-bit accuracy (with dead-timer, for upper and lower arms)
- Timer tuning operation function (tunable with TMPn)
- Period setting function (period can be changed during operation of crest or valley interrupt)
- Compare register rewriting: Anytime rewrite, batch write, or intermittent rewrite (selectable during TMQn operation)
- Interrupt and transfer culling functions
- Dead-time setting function
- A/D trigger timing function of A/D converters 0 and 1 (four types of timing can be generated)
- 0% output and 100% output available
- 0% output and 100% output selectable by crest interrupt and valley interrupt
- Forced output stop function
  - At valid edge detection by external pin input (TOQnOFF, TOPmOFF)
  - At overvoltage detection by comparator function of A/D converter
  - At main clock oscillation stop detection by clock monitor function

Remark V850E/IA3: n = 0, m = 2
V850E/IA4: n = 0, 1, m = 2, 3

10.2 Configuration

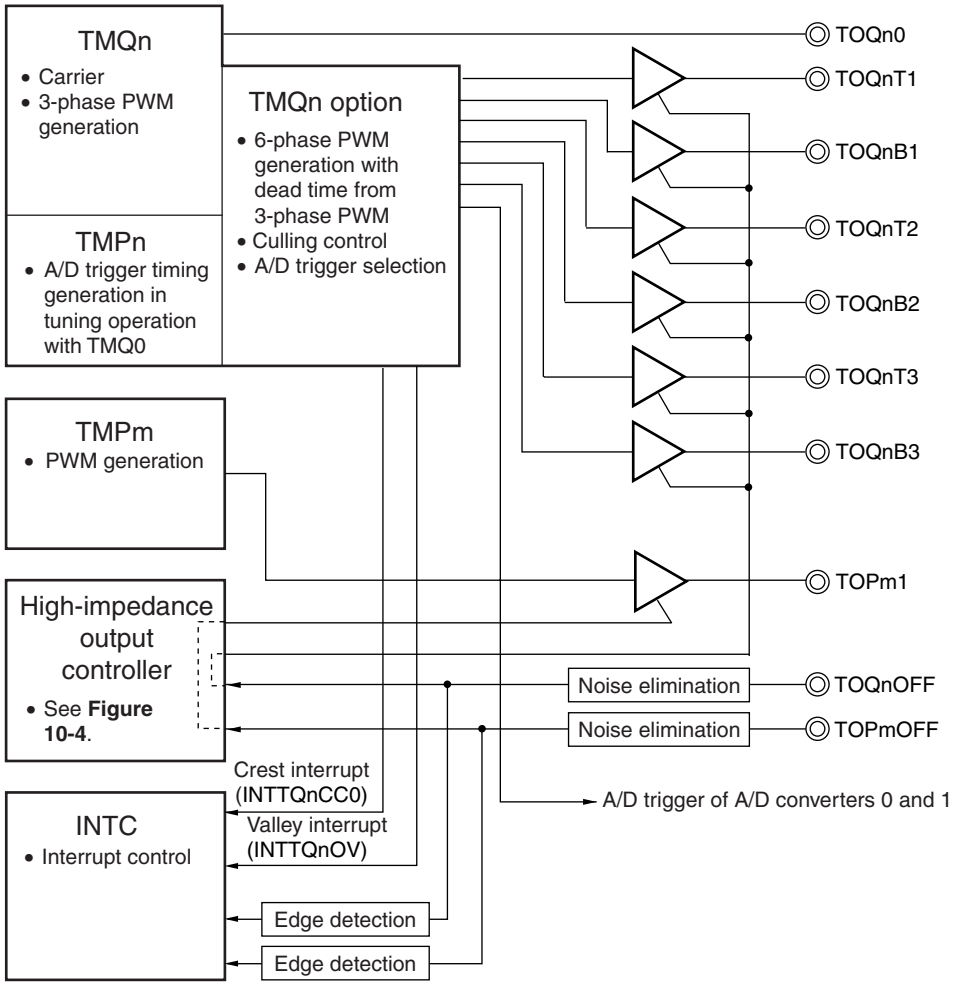
The motor control function consists of the following hardware.

Item	Configuration
Timer register	Dead-time counter m
Compare register	TMQn dead-time compare register (TQnDTC register)
Control registers	TMQa option register 0 (TQaOPT0) TMQn option register 1 (TQnOPT1) TMQn option register 2 (TQnOPT2) TMQn option register 3 (TQnOPT3) TMQn I/O control register 3 (TQnIOC3) High-impedance output control registers 0, 1 (HZAyCTL0, HZAyCTL1)

Remark V850E/IA3: m = 0 to 3, n = 0, y = 0, 2, a = 0, 1
V850E/IA4: m = 0 to 3, n = 0, 1, y = 0 to 2, a = 0, 1

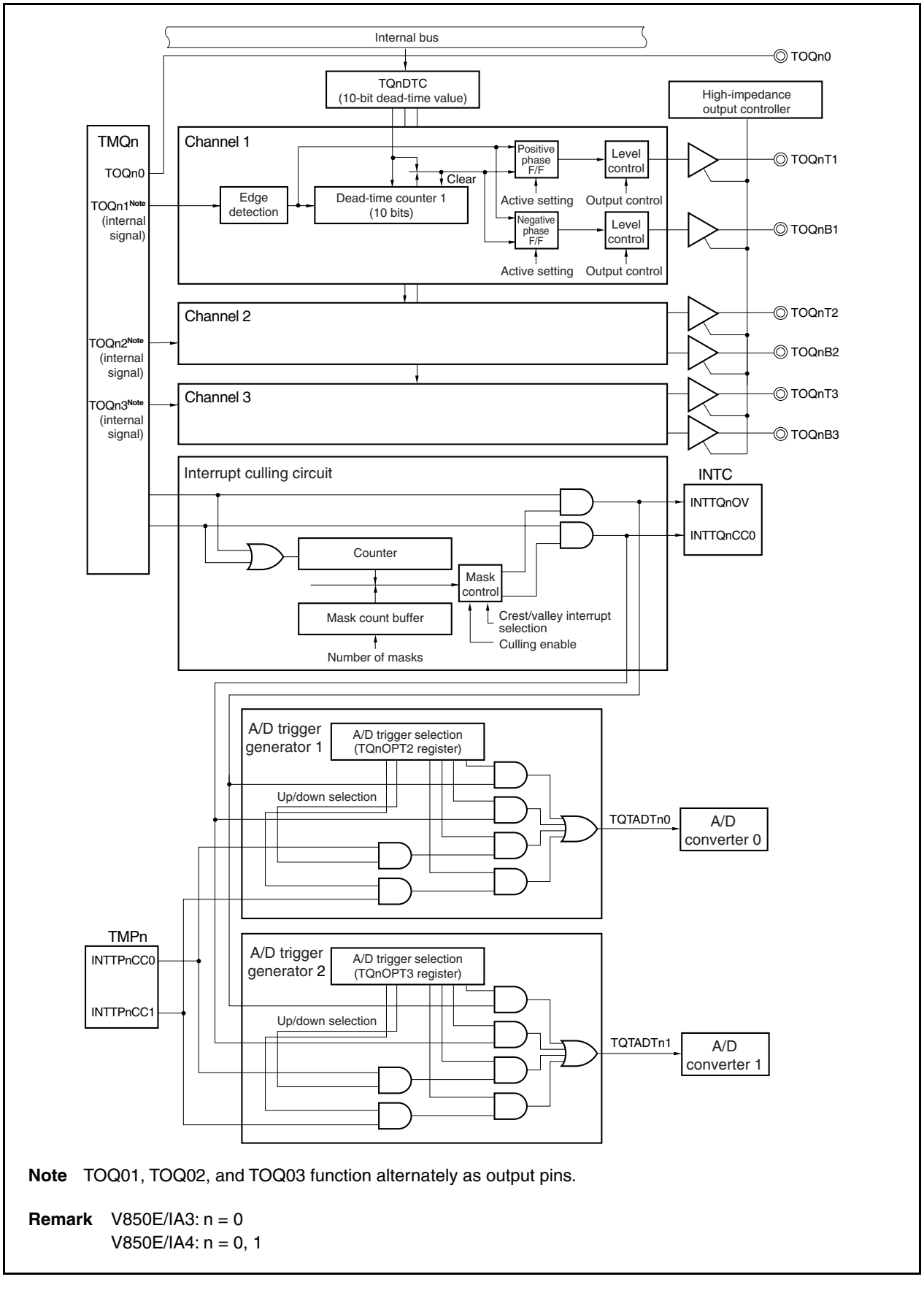
- 6-phase PWM output can be produced with dead time by using the output of TMQn (TOQn1, TOQn2, TOQn3)
- The output level of the 6-phase PWM output can be set individually.
- The 16-bit timer/counter of TMQn counts up/down triangular waves. When the timer/counter underflows and when a period match occurs, an interrupt is generated. Interrupt generation, however, can be suppressed up to 31 times.
- TMPn can execute counting at the same time as TMQn (timer tuning operation function). TMPn can be set in four ways as it can generate two types of A/D trigger sources (INTTPnCC0 and INTTPnCC1), and two types of interrupts: on underflow interrupt (INTTQnOV) and period match interrupt (INTTQnCC0).

Figure 10-1. Block Diagram of Motor Control



Remark V850E/IA3: n = 0, m = 2
V850E/IA4: n = 0, 1, m = 2, 3

Figure 10-2. TMQn Option



Note TOQ01, TOQ02, and TOQ03 function alternately as output pins.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

(1) TMQn dead-time compare register (TQnDTC)

The TQnDTC register is a 10-bit compare register that specifies a dead-time value.

Rewriting this register is prohibited when the TQnCTL0.TQnCE bit = 1.

This register can be read or written in 16-bit units.

Reset sets this register to 0000H.

<R> **Caution To generate the dead-time period, set the TQnDTC register to 1 or greater.**
When the operation is stopped (TQnCTL0.TQnCE bit = 0), the dead-time period is not generated and the output level of the TOQnT1 to TOQnT3 pins and TOQnB1 to TOQnB3 pins will be in the initial status. For the system protection, therefore, before operation is stopped, set the TOQnT1 to TOQnT3 and TOQnB1 to TOQnB3 pins to the high impedance state, or set the output level of pins and switch them to the port mode.
If a dead time period is not needed, set the TQnDTC register to 0.

After reset: 0000H R/W Address: TQ0DTC FFFFF5E4H, TQ1DTC FFFFF624H[Note]

	15	10	9	0
TQnDTC	000000		TQnDTC9 to TQnDTC0	

V850E/IA3
n = 0

V850E/IA4
n = 0, 1

Note V850E/IA4 only

(2) Dead-time counters 1 to 3

The dead-time counters are 10-bit counters that count dead time.

These counters are cleared or count up at the rising or falling edge of the TOQnm output signal by TMQn, and are cleared or stopped when their count value matches the value of the TQnDTC register. The count clock of these counters is the same as that set by the TQnCTL0.TQnCKS2 to TQnCTL0.TQnCKS0 bits of TMQn.

Remarks 1. The operation differs when the TQnOPT2.TQnDTM bit = 1. For details, see **10.4.2 (4) Automatic dead-time width narrowing function (TQnOPT2.TQnDTM bit = 1)**.

2. V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

10.3 Control Registers

(1) TMQn option register 0 (TQnOPT0)

The TQnOPT0 register is an 8-bit register that controls the timer Q option function.

This register can be read or written in 8-bit or 1-bit units. However, the TQnCUF bit is read-only.

Reset sets this register to 00H.

Caution The TQnCMS and TQnCUF bits can be set only in the 6-phase PWM output mode. Be sure to clear these bits to 0 when TMQn is used alone (V850E/IA3: n = 0, V850E/IA4: n = 0, 1).

After reset: 00H R/W Address: TQ0OPT0 FFFFF5C5H, TQ1OPT0 FFFFF605H

TQnOPT0 (n = 0, 1)

<7>	<6>	<5>	<4>	3	<2>	<1>	<0>
TQ0CCS3 Notes 1, 2	TQ0CCS2 Notes 1, 2	TQ0CCS1 Notes 1, 2	TQ0CCS0 Notes 1, 2	0	TQnCMS Note 3	TQnCUF Note 3	TQnOVF Note 4

TQnCMS Note 3	Compare register rewrite mode selection
0	Batch write mode (transfer operation)
1	Anytime write mode
• The TQnCMS bit is valid only when the 6-phase PWM output mode is set (when the TQnCTL1.TQnMD2 to TQnCTL1.TQnMD0 bits = 111). Clear the TQnCMS bit to 0 in any other mode. • The TQnCMS bit can be rewritten while the timer is operating (when the TQnCTL0.TQnCE bit = 1). • The following compare registers are rewritten in the batch write mode. TQnCCR0 to TQnCCR3, TPnCCR0, TPnCCR1, and TQnOPT1 registers	

TQnCUF Note 3	Count-up/count-down flag of timer Qn
0	Timer Qn is counting up.
1	Timer Qn is counting down.
The TQnCUF bit is valid only when the 6-phase PWM output mode is set (when the TQnCTL1.TQnMD2 to TQnCTL1.TQnMD0 bits = 111).	

Notes 1. Valid only for TMQ0. Be sure to clear bits 7 to 4 of TMQ1 to 0.

2. Be sure to clear the TQ0CCS3 to TQ0CCS0 bits to 0 in the 6-phase PWM output mode.

3. In the V850E/IA3, clear bits 2 and 1 of TMQ1 to 0.

4. For details of the TQnOVF bit, see **CHAPTER 7 16-BIT TIMER/EVENT COUNTER Q (TMQ)**.

(2) TMQn option register 1 (TQnOPT1)

The TQnOPT1 register is an 8-bit register that controls the interrupt request signal generated by the timer Qn option function.

<R> The TQnOPT1 register generates the signals output to the interrupt culling circuit, A/D trigger generator 1, and A/D trigger generator 2 shown in Figure 10-2.

This register can be rewritten when the TQnCTL0.TQnCE bit is 1.

Two rewriting modes (batch write mode and anytime write mode) can be selected, depending on the setting of the TQnOPT0.TQnCMS bit.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: TQ0OPT1 FFFFF5E0H, TQ1OPT1 FFFFF620H[Note 1]

	<7>	<6>	5	4	3	2	1	0
TQnOPT1	TQnICE	TQnIOE	0	TQnID4	TQnID3	TQnID2	TQnID1	TQnID0

(V850E/IA3 n = 0)

<R> (V850E/IA4 n = 0, 1)

TQnICE	Crest interrupt (INTTQnCC0 signal) enable[Note 2]
0	Do not use INTTQnCC0 signal (do not use it as count signal for interrupt culling).
1	Use INTTQnCC0 signal (use it as count signal for interrupt culling).

<R>

TQnIOE	Valley interrupt (INTTQnOV signal) enable[Note 2]
0	Do not use INTTQnOV signal (do not use it as count signal for interrupt culling).
1	Use INTTQnOV signal (use it as count signal for interrupt culling).

TQnID4	TQnID3	TQnID2	TQnID1	TQnID0	Number of times of interrupt
0	0	0	0	0	Not culled (all interrupts are output)
0	0	0	0	1	1 masked (one of two interrupts is output)
0	0	0	1	0	2 masked (one of three interrupts is output)
0	0	0	1	1	3 masked (one of four interrupts is output)
:	:	:	:	:	:
1	1	1	0	0	28 masked (one of 29 interrupts is output)
1	1	1	0	1	29 masked (one of 30 interrupts is output)
1	1	1	1	0	30 masked (one of 31 interrupts is output)
1	1	1	1	1	31 masked (one of 32 interrupts is output)

Notes 1. V850E/IA4 only

<R> **2.** When using the crest interrupt (INTTQnCC0 signal) and the valley interrupt (INTTQnOV signal) as the count signal for interrupt culling or as the A/D trigger signal, set the signal to be used to 1.
A/D trigger is generated at the culled interrupt timing.

(3) TMQn option register 2 (TQnOPT2)

The TQnOPT2 register is an 8-bit register that controls the timer Q option function.

This register can be rewritten when the TQnCTL0.TQnCE bit is 1. However, rewriting the TQnDTM bit is prohibited when the TQnCE bit is 1. The same value can be rewritten.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

(1/2)

After reset: 00H R/W Address: TQ0OPT2 FFFFF5E1H, TQ1OPT2 FFFFF621H[Note]

TQnOPT2

<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
TQnRDE	TQnDTM	TQnATM03	TQnATM02	TQnAT03	TQnAT02	TQnAT01	TQnAT00

V850E/IA3
n = 0
m = 1 to 3

V850E/IA4
n = 0, 1
m = 1 to 3

TQnRDE	Transfer culling enable
0	Do not cull transfer (transfer timing is generated every time at crest and valley).
1	Cull transfer at the same interval as interrupt culling set by the TQnOPT1 register.

TQnDTM	Dead-time counter operation mode selection
0	Dead-time counter counts up normally and, if TOQnm output of TMQn is at a narrow interval (TOQnm output width < dead-time width), the dead-time counter is cleared and counts up again.
1	Dead-time counter counts up normally and, if TOQnm output of TMQn is at a narrow interval (TOQnm output width < dead-time width), the dead-time counter counts down and the dead-time control width is automatically narrowed.
Rewriting the TQnDTM bit is disabled during timer operation. If it is rewritten by mistake, stop the timer operation by clearing the TQnCE bit to 0, and re-set the TQnDTM bit.	

Note V850E/IA4 only

Cautions 1. When using interrupt culling (the TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits are set to other than 00000), be sure to set the TQnRDE bit to 1.
Therefore, the interrupt and transfer are generated at the same timing. The interrupt and transfer cannot be set separately. If the interrupt and transfer are set separately (TQnRDE bit = 0), transfer is not performed normally.

<R> **2. To generate the dead-time period, set the TQnDTC register to 1 or greater.**
When the operation is stopped (TQnCTL0.TQnCE bit = 0), the dead-time period is not generated and the output level of the TOQnT1 to TOQnT3 pins and TOQnB1 to TOQnB3 pins will be in the initial status. For the system protection, therefore, before operation is stopped, set the TOQnT1 to TOQnT3 and TOQnB1 to TOQnB3 pins to the high impedance state, or set the output level of pins and switch them to the port mode.
If a dead time period is not needed, set the TQnDTC register to 0.

(2/2)

TQnATM03	TQnATM03 mode selection
0	Output A/D trigger signal (TQTADTn0) for INTTPnCC1 interrupt while dead-time counter is counting up.
1	Output A/D trigger signal (TQTADTn0) for INTTPnCC1 interrupt while dead-time counter is counting down.

TQnATM02	TQnATM02 mode selection
0	Output A/D trigger signal (TQTADTn0) for INTTPnCC0 interrupt while dead-time counter is counting up.
1	Output A/D trigger signal (TQTADTn0) for INTTPnCC0 interrupt while dead-time counter is counting down.

TQnAT03Note	A/D trigger output control 3
0	Disable output of A/D trigger signal (TQTADTn0) for INTTPnCC1 interrupt.
1	Enable output of A/D trigger signal (TQTADTn0) for INTTPnCC1 interrupt.

TQnAT02Note	A/D trigger output control 2
0	Disable output of A/D trigger signal (TQTADTn0) for INTTPnCC0 interrupt.
1	Enable output of A/D trigger signal (TQTADTn0) for INTTPnCC0 interrupt.

TQnAT01Note	A/D trigger output control 1
0	Disable output of A/D trigger signal (TQTADTn0) for INTTQnCC0 (crest interrupt).
1	Enable output of A/D trigger signal (TQTADTn0) for INTTQnCC0 (crest interrupt).

TQnAT00Note	A/D trigger output control 0
0	Disable output of A/D trigger signal (TQTADTn0) for INTTQnOV (valley interrupt).
1	Enable output of A/D trigger signal (TQTADTn0) for INTTQnOV (valley interrupt).

Note For the setting of the TQnAT03 to TQnAT00 bits, see **CHAPTER 12 A/D CONVERTERS 0 AND 1**.

(4) TMQn option register 3 (TQnOPT3)

The TQnOPT3 register is an 8-bit register that controls the timer Qn option function.

This register can be rewritten when the TQnCTL0.TQnCE bit is 1.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: TQ0OPT3 FFFFF5E3H, TQ1OPT3 FFFFF623H[Note 1]

	7	6	<5>	<4>	<3>	<2>	<1>	<0>
TQnOPT3	0	0	TQnATM13	TQnATM12	TQnATAT13	TQnATAT12	TQnATAT11	TQnATAT10

V850E/IA3 n = 0

V850E/IA4 n = 0, 1

TQnATM13	TQnATM13 mode selection
0	Output A/D trigger signal (TQTADTn1) of INTTPnCC1 interrupt while dead-time counter is counting up.
1	Output A/D trigger signal (TQTADTn1) of INTTPnCC1 interrupt while dead-time counter is counting down.

TQnATM12	TQnATM12 mode selection
0	Output A/D trigger signal (TQTADTn1) of INTTPnCC0 interrupt while dead-time counter is counting up.
1	Output A/D trigger signal (TQTADTn1) of INTTPnCC0 interrupt while dead-time counter is counting down.

TQnAT13[Note 2]	A/D trigger output control 3
0	Disable output of A/D trigger signal (TQTADTn1) for INTTPnCC1 interrupt.
1	Enable output of A/D trigger signal (TQTADTn1) for INTTPnCC1 interrupt.

TQnAT12[Note 2]	A/D trigger output control 2
0	Disable output of A/D trigger signal (TQTADTn1) for INTTPnCC0 interrupt.
1	Enable output of A/D trigger signal (TQTADTn1) for INTTPnCC0 interrupt.

TQnAT11[Note 2]	A/D trigger output control 1
0	Disable output of A/D trigger signal (TQTADTn1) for INTTQnCC0 interrupt (crest interrupt).
1	Enable output of A/D trigger signal (TQTADTn1) for INTTQnCC0 interrupt (crest interrupt).

TQnAT10[Note 2]	A/D trigger output control 0
0	Disable output of A/D trigger signal (TQTADTn1) for INTTQnOV interrupt (valley interrupt).
1	Enable output of A/D trigger signal (TQTADTn1) for INTTQnOV interrupt (valley interrupt).

Notes 1. V850E/IA4 only

2. For the setting of the TQnAT13 to TQnAT10 bits, see **CHAPTER 12 A/D CONVERTERS 0 AND 1**.

(5) TMQn I/O control register 3 (TQnIOC3)

The TQnIOC3 register is an 8-bit register that controls the output of the timer Qn option function.

To output from the TOQnTm pin, set the TQnIOC0.TQnOEm bit to 1 and then set the TQnIOC3 register.

The TQnIOC3 register can be rewritten only when the TQnCTL0.TQnCE bit is 0.

Rewriting each bit of the TQnIOC3 register is prohibited when the TQnCTL0.TQnCE bit is 1; however the same value can be rewritten to each bit of the TQnIOC3 register when the TQnCTL0.TQnCE bit is 1.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to A8H.

Caution Set the TQnIOC3 register to the default value (A8H) when the timer is used in a mode other than the 6-phase PWM output mode.

Remark Set the output level of the TOQnTm pin by the TQnIOC0 register.

After reset: A8H R/W Address: TQ0IOC3 FFFFF5E2H, TQ1IOC3 FFFFF622H[Note]

	<7>	<6>	<5>	<4>	<3>	<2>	1	0
TQnIOC3	TQnOLB3	TQnOEB3	TQnOLB2	TQnOEB2	TQnOLB1	TQnOEB1	0	0

V850E/IA3
n = 0
m = 1 to 3

V850E/IA4
n = 0, 1
m = 1 to 3

TQnOLBm	Setting of TOQnBm pin output level
0	Disable inversion of output of TOQnBm pin
1	Enable inversion of output of TOQnBm pin

TQnOEBm	Setting of TOQnBm pin output
0	Disable TOQnBm pin output. • When TQnOLBm bit = 0, low level is output from TOQnBm pin. • When TQnOLBm bit = 1, high level is output from TOQnBm pin.
1	Enable TOQnBm pin output.

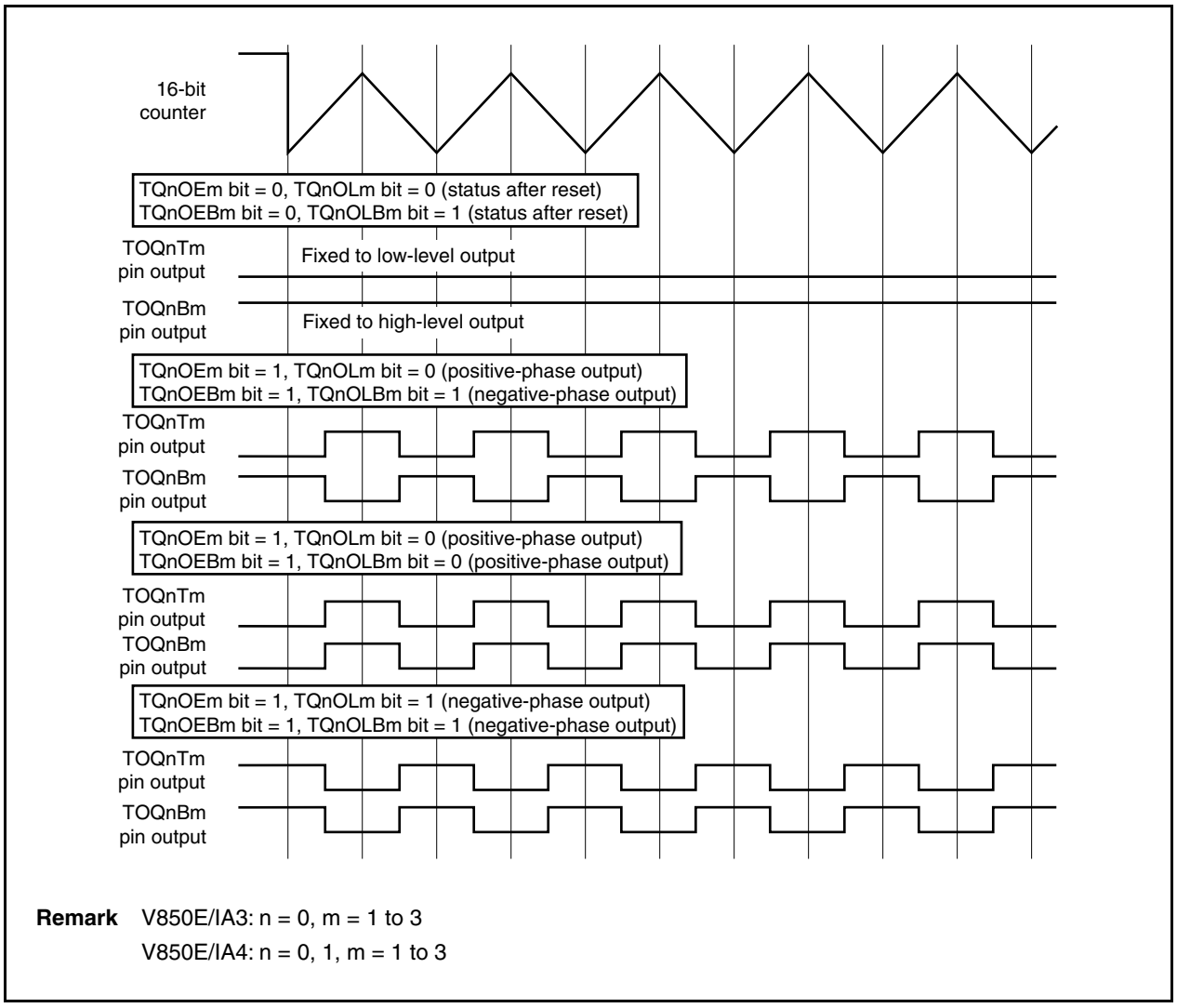
Note V850E/IA4 only

(a) Output from TOQnTm and TOQnBm pins

The TOQnTm pin output is controlled by the TQnIOC0.TQnOLm and TQnIOC0.TQnOEm bits. The TOQnBm pin output is controlled by the TQnIOC3.TQnOLBm and TQnIOC3.TQnOEBm bits.

A timer output with each setting in the 6-phase PWM output mode is shown below.

<R> **Figure 10-3. TOQnTm and TOQnBm Pin Output Control (Without Dead Time)**



Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

Table 10-1. TOQnTm Pin Output

TQnOLm Bit	TQnOEm Bit	TQnCE Bit	TOQnTm Pin Output
0	0	x	Low-level output
	1	0	Low-level output
		1	TOQnTm positive-phase output
1	0	x	High-level output
	1	0	High-level output
		1	TOQnTm negative-phase output

Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

Table 10-2. TOQnBm Pin Output

TQnOLBm Bit	TQnOEBm Bit	TQnCE Bit	TOQnBm Pin Output
0	0	x	Low-level output
	1	0	Low-level output
		1	TOQnBm positive-phase output
1	0	x	High-level output
	1	0	High-level output
		1	TOQnBm negative-phase output

Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

(6) High-impedance output control registers 00, 01, 10, 11, 20, 21 (HZAmCTL0, HZAmCTL1)

The HZAmCTL0 and HZAmCTL1 registers are 8-bit registers that control the high-impedance state of the output buffer.

These registers can be read or written in 8-bit or 1-bit units. However, the HZAmDCFn bit is a read-only bit and cannot be written.

16-bit access is not possible.

Reset sets these registers to 00H.

The same value can be always rewritten to the HZAmCTLn register by software.

The relationship between detection factor and the control registers is shown below.

Pins Subject to High-Impedance Control	High-Impedance Control Factor		Control Register
	External Pin	A/D Unit (Comparator)	
When TOQ0T1 to TOQ0T3 are output When TOQ0B1 to TOQ0B3 are output	TOQ0OFF	–	HZA0CTL0
	–	When reference voltage of ANI00, ANI01, or ANI02[Note] input is exceeded	HZA2CTL0
When TOP21 is output	TOP2OFF	–	HZA0CTL1
When TOQ1T1[Note], TOQ1T2[Note], and TOQ1T3[Note] are output When TOQ1B1[Note], TOQ1B2[Note], and TOQ1B3[Note] are output	TOQ1OFF[Note]	–	HZA1CTL0[Note]
	–	When reference voltage of ANI10 to ANI12 inputs is exceeded	HZA2CTL1
When TOP31[Note] is output	TOP3OFF[Note]	–	HZA1CTL1[Note]

Note V850E/IA4 only

Caution High-impedance control is performed only when a port pin is set to function as indicated in the above table.

(1/2)

After reset: 00H R/W Address: HZA0CTL0 FFFFF5F0H, HZA0CTL1 FFFFF5F1H, HZA1CTL0 FFFFF630H[Note 1], HZA1CTL1 FFFFF631H[Note 1], HZA2CTL0 FFFFF638H, HZA2CTL1 FFFFF639H

HZAmCTLn (V850E/IA3 n = 0, 1 m = 0, 2) (V850E/IA4 n = 0, 1 m = 0 to 2)

<7>	<6>	5	4	<3>	<2>	1	<0>
HZAmDCEn	HZAmDCMn	HZAmDCNn	HZAmDCPn	HZAmDCTn	HZAmDCCn	0	HZAmDCFn

HZAmDCEn	High-impedance output control
0	Disable high-impedance output control operation. Pins can function as output pins.
1	Enable high-impedance output control operation.

HZAmDCMn	Condition of clearing high-impedance state by HZAmDCCn bit
0	Setting of the HZAmDCCn bit is valid regardless of the external pin[Note 2] input.
1	Setting of the HZAmDCCn bit is invalid while the external pin[Note 2] input holds a level detected as abnormal (active level).
Rewrite the HZAmDCMn bit when the HZAmDCEn bit = 0.	

HZAmDCNn	HZAmDCPn	External pin[Note 2] input edge specification
0	0	No valid edge (setting the HZAmDCFn bit by external pin[Note 2] input is prohibited).
0	1	Rising edge of the external pin[Note 2] input is valid (abnormality is detected by rising edge input)[Note 3].
1	0	Falling edge of the external pin[Note 2] input is valid (abnormality is detected by falling edge input).
1	1	Setting prohibited

- Rewrite the HZAmDCNn and HZAmDCPn bits when the HZAmDCEn bit is 0.
- For the edge specification of the INTP0 to INTP3 pins, see **17.4.2 (1) External interrupt rising edge specification register 0 (INTR0), external interrupt falling edge specification register 0 (INTF0)**.
- The edge of the external pins must be specified starting from the TOQnOFF and TOPmOFF pins. Then the edge of the external pins other than the TOQnOFF and TOPmOFF pins must be specified. Otherwise, the undefined edge may be detected when the edges of the TOQnOFF and TOPmOFF pins are specified.
- High-impedance output control is performed when the valid edge is input after the operation is enabled (by setting HZAmDCEn bit to 1). If the external pin[Note 2] is at the active level when the operation is enabled, therefore, high-impedance output control is not performed.

Notes 1. V850E/IA4 only

2. • V850E/IA3
HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
HZA2CTL0: ANI00, ANI01 pins, HZA2CTL1: ANI10 to ANI12 pins
• V850E/IA4
HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
HZA1CTL0: TOQ1OFF pin, HZA1CTL1: TOP3OFF pin,
HZA2CTL0: ANI00 to ANI02 pins, HZA2CTL1: ANI10 to ANI12 pins

3. When using the comparator output, set the rising edge input.

(2/2)

HZAmDCTn	High-impedance output trigger bit
0	No operation
1	Pins are made to go into a high-impedance state by software and the HZAmDCFn bit is set to 1.

- If an edge indicating abnormality is input to the external pin[Note 2] (which is detected according to the setting of the HZAmDCNn and HZAmDCPn bits), the HZAmDCTn bit is invalid even if it is set to 1.
- The HZAmDCTn bit is always 0 when it is read because it is a software-triggered bit.
- The HZAmDCTn bit is invalid even if it is set to 1 when the HZAmDCEn bit = 0.
- Simultaneously setting the HZAmDCTn and HZAmDCCn bits to 1 is prohibited.

HZAmDCCn	High-impedance output control clear bit
0	No operation
1	Pins that have gone into a high-impedance state are output-enabled by software and the HZAmDCFn bit is cleared to 0.

- Pins can function as output pins when the HZAmDCM bit = 0, regardless of the status of the external pin[Note].
- If an edge indicating abnormality is input to the external pin[Note] (which is set by the HZAmDCNn and HZAmDCPn bits) when the HZAmDCM bit = 1, the HZAmDCCn bit is invalid even if it is set to 1.
- The HZAmDCCn bit is always 0 when it is read.
- The HZAmDCCn bit is invalid even if it is set to 1 when the HZAmDCEn bit = 0.
- Simultaneously setting the HZAmDCTn and HZAmDCCn bits to 1 is prohibited.

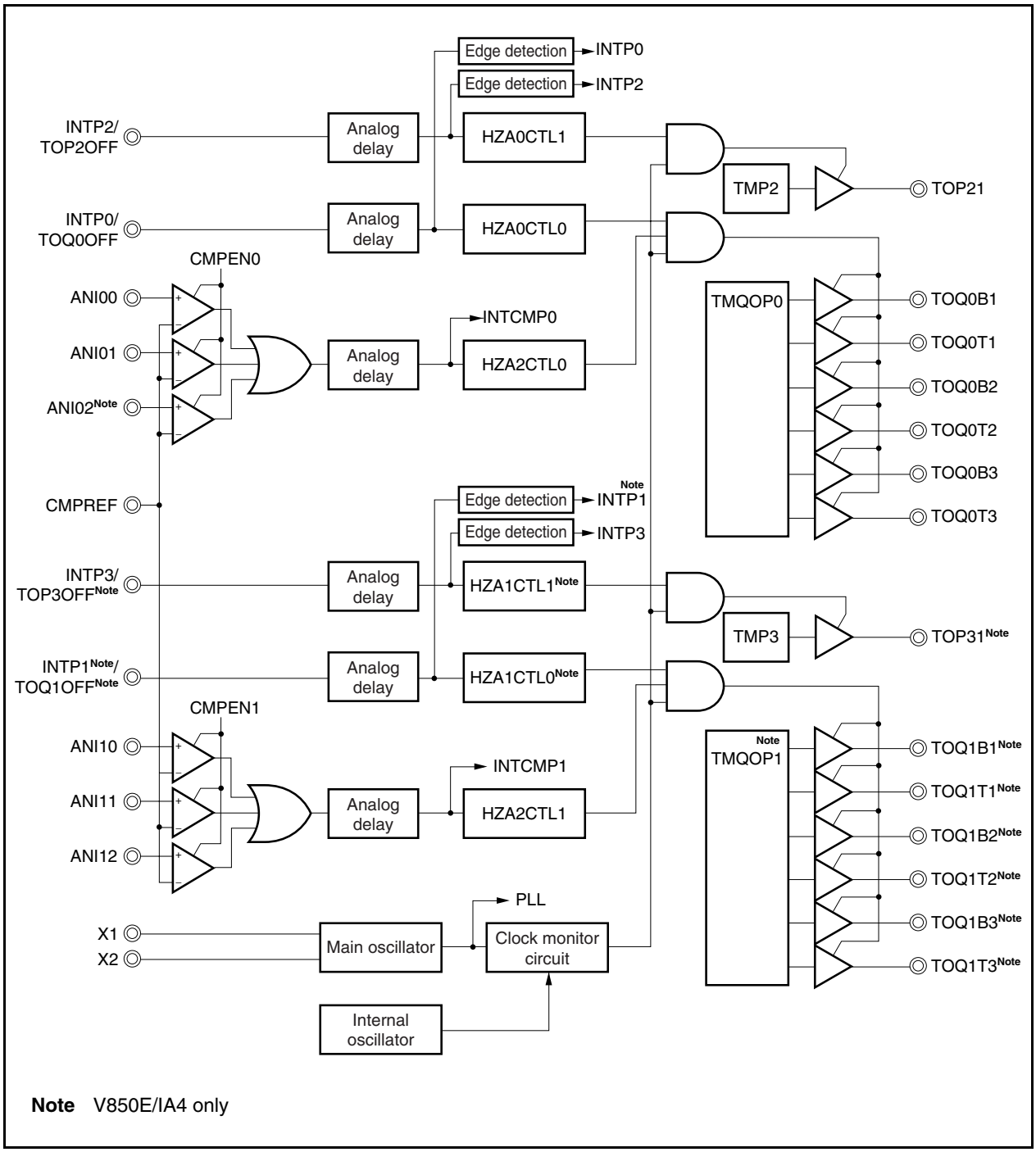
HZAmDCFn	High-impedance output status flag
0	Indicates that output of the pin is enabled. • This bit is cleared to 0 when the HZAmDCEn bit = 0. • This bit is cleared to 0 when the HZAmDCCn bit = 1.
1	Indicates that the pin goes into a high-impedance state. • This bit is set to 1 when the HZAmDCTn bit = 1. • This bit is set to 1 when an edge indicating abnormality is input to the external pin[Note] (which is detected according to the setting of the HZAmDCNn and HZAmDCPn bits).

Note
- V850E/IA3

  HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
  HZA2CTL0: ANI00, ANI01 pins, HZA2CTL1: ANI10 to ANI12 pins
- V850E/IA4

  HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
  HZA1CTL0: TOQ1OFF pin, HZA1CTL1: TOP3OFF pin,
  HZA2CTL0: ANI00 to ANI02 pins, HZA2CTL1: ANI10 to ANI12 pins

Figure 10-4. High-Impedance Output Controller Configuration



Note V850E/IA4 only

(a) Setting procedure

(i) Setting of high-impedance control operation

<1> Set the HZAmDCMn, HZAmDCNn, and HZAmDCPn bits.

<2> Set the HZAmDCEn bit to 1 (enable high-impedance control).

(ii) Changing setting after enabling high-impedance control operation

<1> Clear the HZAmDCEn bit to 0 (to stop the high-impedance control operation).

<2> Change the setting of the HZAmDCMn, HZAmDCNn, and HZAmDCPn bits.

<3> Set the HZAmDCEn bit to 1 (to enable the high-impedance control operation again).

(iii) Resuming output when pins are in high-impedance state

If the HZAmDCMn bit is 1, set the HZAmDCCn bit to 1 to clear the high-impedance state after the valid edge of the external pin[Note] is detected. However, the high-impedance state cannot be cleared unless this bit is set while the input level of the external pin[Note] is inactive.

<1> Set the HZAmDCCn bit to 1 (command signal to clear the high-impedance state).

<2> Read the HZAmDCFn bit and check the flag status.

<3> Return to <1> if the HZAmDCFn bit is 1. The input level of the external pin[Note] must be checked. The pin can function as an output pin if the HZAmDCFn bit is 0.

(iv) To make the pin to go into a high-impedance state by software

The HZAmDCTn bit must be set to 1 by software to make the pin to go into a high-impedance state while the input level of the external pin[Note] is inactive. The following procedure is an example in which the setting is not dependent upon the setting of the HZAmDCMn bit.

<1> Set the HZAmDCTn bit to 1 (high-impedance output command).

<2> Read the HZAmDCFn bit to check the flag status.

<3> Return to <1> if the HZAmDCFn bit is 0. The input level of the external pin[Note] must be checked. The pin is in a high-impedance state if the HZAmDCFn bit is 1.

However, if the external pin[Note] is not used with the HZAmDCPn bit and HZAmDCNn bit cleared to 0, the pin goes into a high-impedance state when the HZAmDCTn bit is set to 1.

Note • V850E/IA3

HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
HZA2CTL0: ANI00, ANI01 pins, HZA2CTL1: ANI10 to ANI12 pins

• V850E/IA4

HZA0CTL0: TOQ0OFF pin, HZA0CTL1: TOP2OFF pin,
HZA1CTL0: TOQ1OFF pin, HZA1CTL1: TOP3OFF pin,
HZA2CTL0: ANI00 to ANI02 pins, HZA2CTL1: ANI10 to ANI12 pins

10.4 Operation

10.4.1 System outline

(1) Outline of 6-phase PWM output

The 6-phase PWM output mode is used to generate a 6-phase PWM output wave, by using TMQn and the TMQn option in combination.

The 6-phase PWM output mode is enabled by setting the TQnCTL1.TQnMD2 to TQnCTL1.TQnMD0 bits of TMQn to “111”.

One 16-bit counter and four 16-bit compare registers of TMQn are used to generate a basic 3-phase wave.

The functions of the compare registers are as follows.

TMPn can perform a tuning operation with TMQn to start a conversion trigger source for A/D converters 0 and 1.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Compare Register	Function	Settable Range
TQnCCR0 register	Setting of cycle	0002H ≤ m ≤ FFFEH
TQnCCR1 register	Specifying output width of phase U	0000H ≤ i ≤ m + 1
TQnCCR2 register	Specifying output width of phase V	0000H ≤ j ≤ m + 1
TQnCCR3 register	Specifying output width of phase W	0000H ≤ k ≤ m + 1

Remark m = Set value of TQnCCR0 register
i = Set value of TQnCCR1 register
j = Set value of TQnCCR2 register
k = Set value of TQnCCR3 register

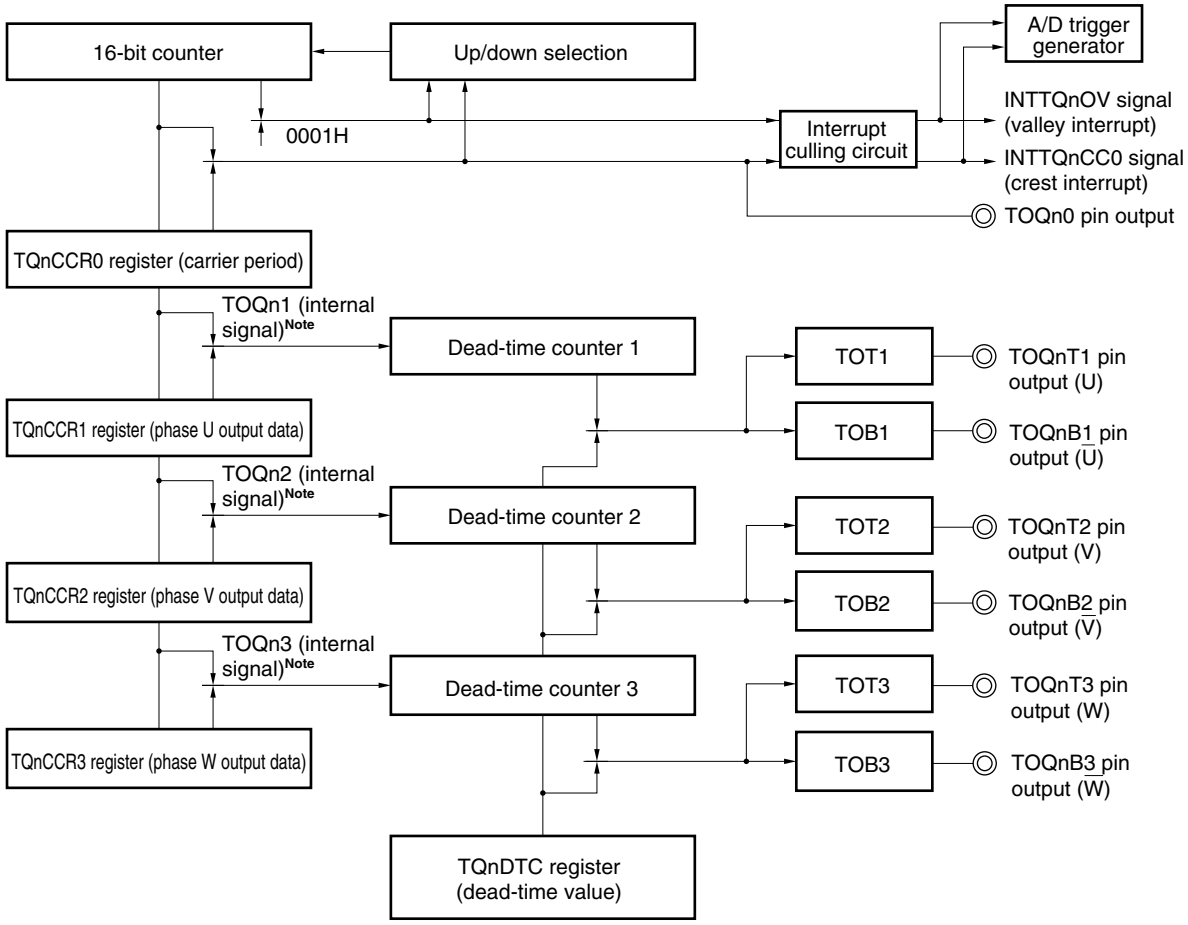
A dead-time interval is generated from the basic 3-phase wave generated by using three 10-bit dead-time counters and one compare register to create a wave with a reverse phase to that of the basic 3-phase wave. Then a 6-phase PWM output wave (U, $\overline{U}$, V, $\overline{V}$, W, and $\overline{W}$) is generated.

The 16-bit counter for generating the basic 3-phase wave counts up or down. After the operation has been started, this counter counts up. When its count value matches the cycle set to the TQnCCR0 register, the counter starts counting down. When the count value matches 0001H, the counter counts up again. This means that a value two times higher than the value set to the TQnCCR0 register +1 is the carrier cycle.

10-bit dead-time counters 1 to 3 that generate the dead-time interval count up. Therefore, the value set to the TMQn dead-time compare register (TQnDTC) is used as a dead-time value as is. Because three counters are used, dead time can be generated independently in phases U, V, and W. However, because there is only one register that specifies a dead-time value (TQnDTC), the same dead-time value is used in the three phases.

<R>

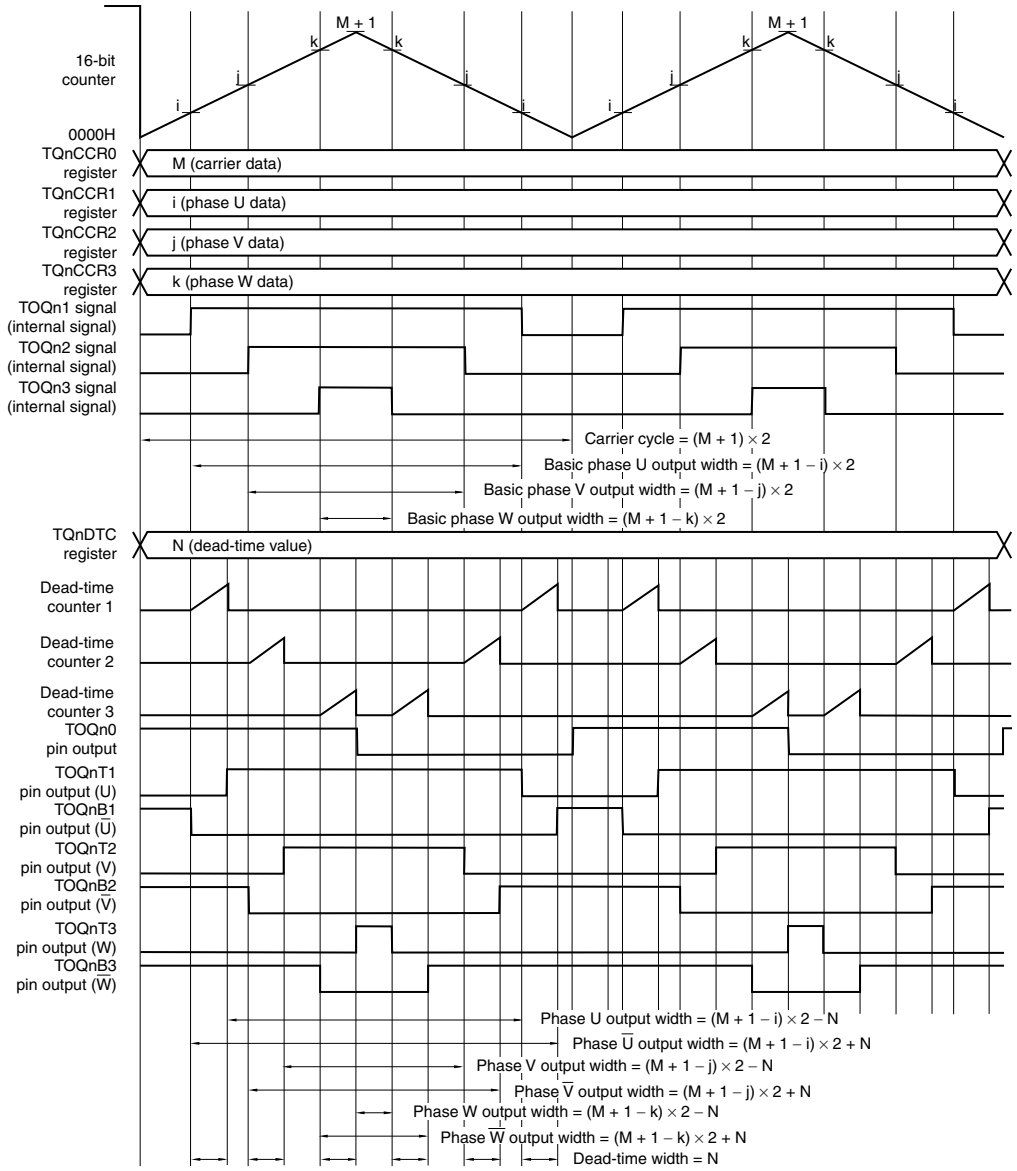
Figure 10-5. Outline of 6-Phase PWM Output Mode



Note TOQ01, TOQ02, and TOQ03 function alternately as output pins.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Figure 10-6. Timing Chart of 6-Phase PWM Output Mode



Cautions 1. Set the value "M" of the TQnCCR0 register in a range of 0002H ≤ M ≤ FFFEH in the 6-phase PWM output mode.

2. Only a value of up to "M + 1" can be set to the TQnCCR1, TQnCCR2, and TQnCCR3 registers.

3. The output is 100% if "0000H" is set to the TQnCCR1, TQnCCR2, and TQnCCR3 registers. The output is 0% if "M + 1" is set to the TQnCCR1, TQnCCR2, and TQnCCR3 registers. The output (duty 50%) rises at the crest (M + 1) of the 16-bit counter and falls at the valley (0000H) if "M + 2" or higher is set to the TQnCCR1, TQnCCR2, and TQnCCR3 registers.

4. If the operation value of an equation (such as (M + 1 − i) × 2 − N) of the output width of phases U, V, and W is 0 or lower, it is converged to 0 (100% output). If the operation value is higher than "(M + 1) × 2", it is converged to (M + 1) × 2 (0% output).

Remark V850E/IA3: n = 0 V850E/IA4: n = 0, 1

(2) Interrupt requests

Two types of interrupt requests are available: the INTTQnCC0 (crest interrupt) signal and INTTQnOV (valley interrupt) signal.

The INTTQnCC0 and INTTQnOV signals can be culled by using the TQnOPT1 register.

For details of culling interrupts, see **10.4.3 Interrupt culling function**.

- INTTQnCC0 (crest interrupt) signal: Interrupt signal indicating matching between the value of the 16-bit counter that counts up and the value of the TQnCCR0 register
- INTTQnOV (valley interrupt) signal: Interrupt signal indicating matching between the value of the 16-bit counter that counts down and the value 0001H

(3) Rewriting registers during timer operation

The following registers have a buffer register and can be rewritten in the anytime rewriting mode, batch rewrite mode, or intermittent batch rewrite mode.

Related Unit	Register
Timer Pn	TMPn capture/compare register 0 (TPnCCR0) TMPn capture/compare register 1 (TPnCCR1)
Timer Qn	TMQn capture/compare register 0 (TQnCCR0) TMQn capture/compare register 1 (TQnCCR1) TMQn capture/compare register 2 (TQnCCR2) TMQn capture/compare register 3 (TQnCCR3)
Timer Qn option	TMQn option register 1 (TQnOPT1)

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

For details of the transfer function of the compare register, see **10.4.4 Operation to rewrite register with transfer function**.

(4) Counting-up/down operation of 16-bit counter

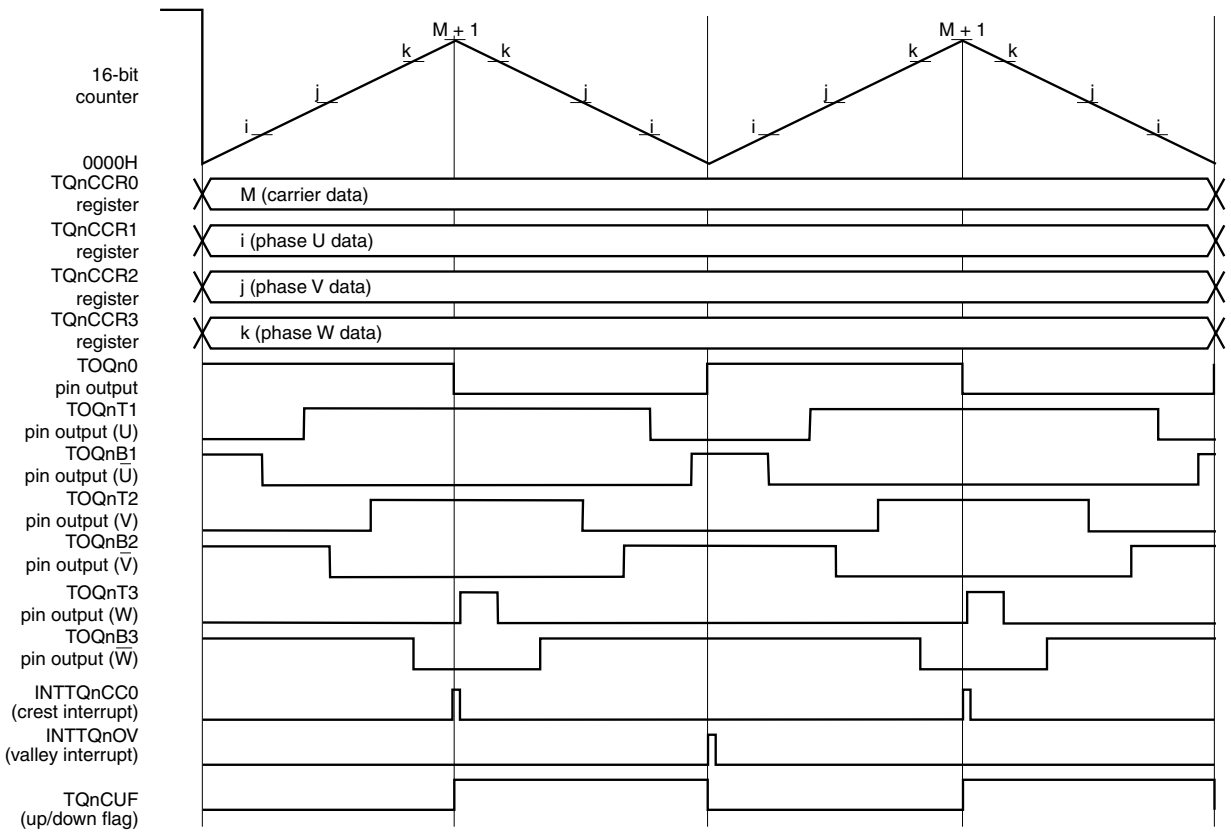
The operation status of the 16-bit counter can be checked by using the TQnCUF bit of TMQn option register 0 (TQnOPT0).

Status of TQnCUF Bit	Status of 16-Bit Counter	Range of 16-Bit Counter Value
TQnCUF bit = 0	Counting up	0000H – m
TQnCUF bit = 1	Counting down	(m+1) – 0001H

Remarks 1. m = Set value of TQnCCR0 register

2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Figure 10-7. Interrupt and Up/Down Flag



Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

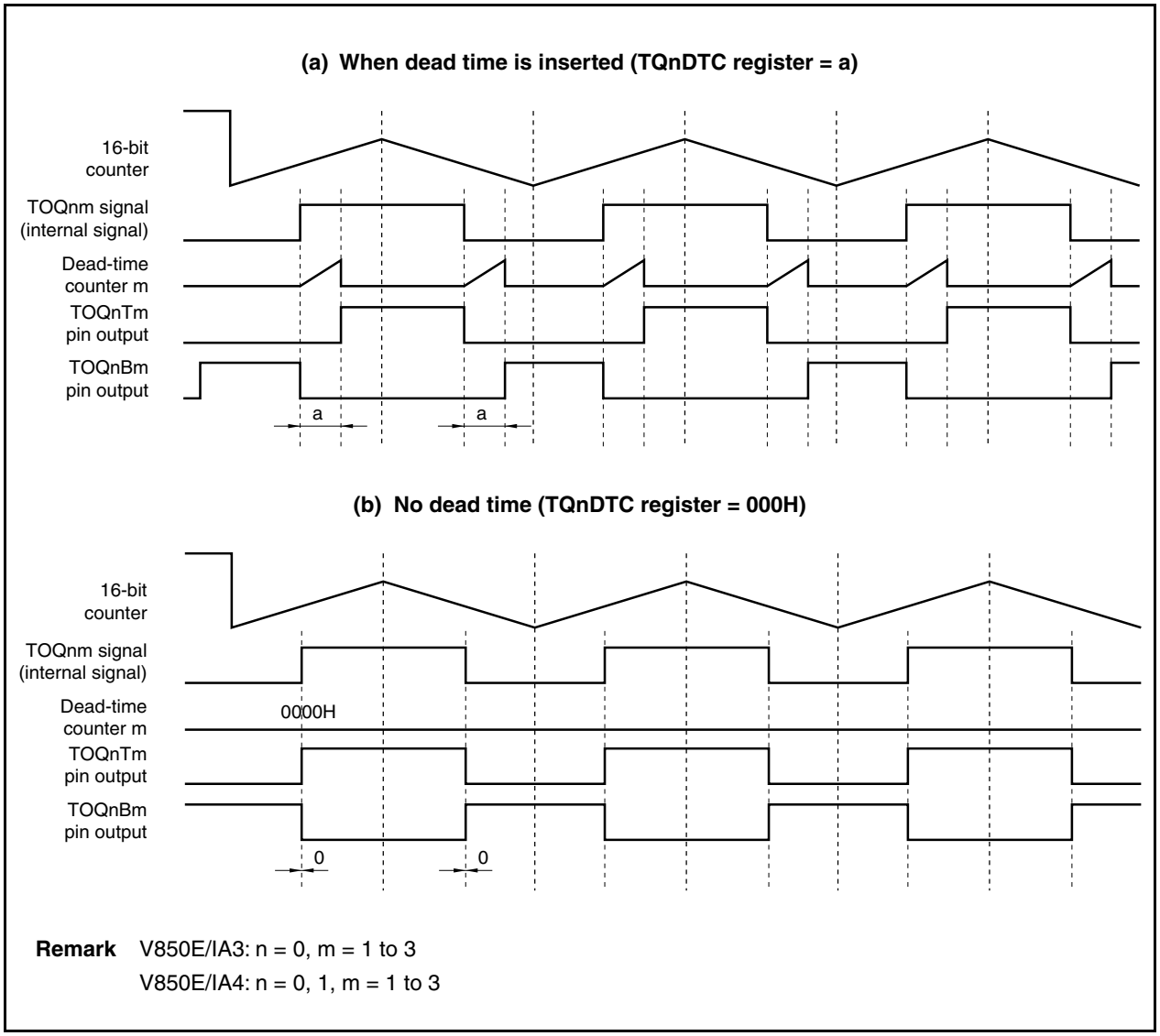
10.4.2 Dead-time control (generation of negative-phase wave signal)

(1) Dead-time control mechanism

In the 6-phase PWM output mode, compare registers 1 to 3 (TQnCCR1, TQnCCR2, and TQnCCR3) are used to set the duty factor, and compare register 0 (TQnCCR0) is used to set the cycle. By setting these four registers and by starting the operation of TMQ, three types of PWM output waves (basic 3-phase waves) with a variable duty factor are generated. These three PWM output waves are input to the timer Q option unit (TMQOPn) and their inverted signal with dead-time is created to generate three sets of (six) PWM waves.

The TMQOPn unit consists of three 10-bit counters (dead-time counters 1 to 3) that operate in synchronization with the count clock of TMQn, and a TMQn dead-time compare register (TQnDTC) that specifies dead time. If "a" is set to the TQnDTC register, the dead-time value is "a", and interval "a" is created between a positive-phase wave and a negative-phase wave.

Figure 10-8. PWM Output Wave with Dead Time (1)



Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

(2) PWM output of 0%/100%

The V850E/IA3 and V850E/IA4 are capable of 0% wave output and 100% wave output for PWM output.

A low level is continuously output from TOQnTm pin as the 0% wave output. A high level is continuously output from TOQnTm pin as the 100% wave output.

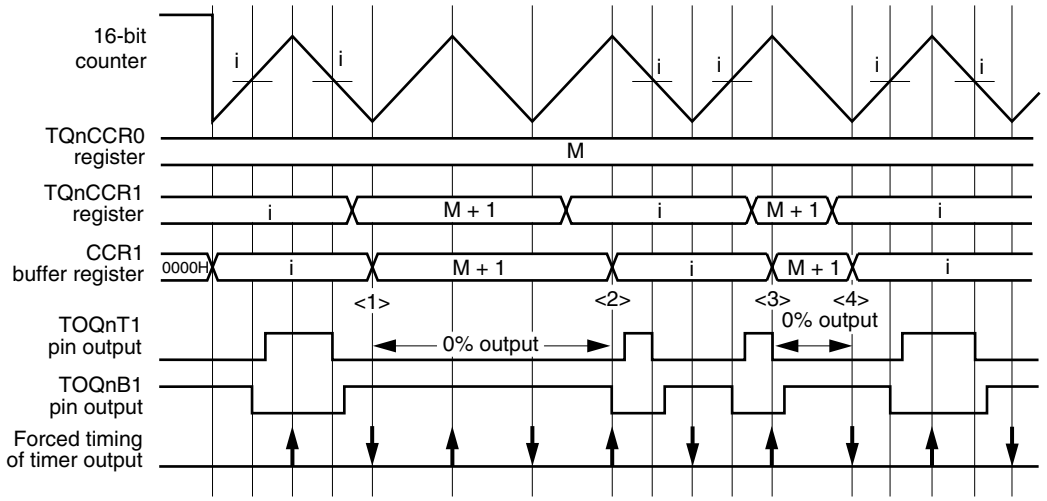
The 0% wave is output by setting the TQnCCRm register to "M + 1" when the TQnCCR0 register = M.

The 100% wave is output by setting the TQnCCRm register to "0000H".

Rewriting the TQnCCRm register is enabled while the timer is operating, and 0% wave output or 100% wave output can be selected at the point of the crest interrupt (INTTQnCC0) and valley interrupt (INTTQnOV).

Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

<R> **Figure 10-9. 0% PWM Output Waveform (With Dead Time)**



<1> 0% output is selected by the valley interrupt (without a match with the 16-bit counter).
The valley interrupt forcibly lowers the timer output. This produces the 0% output.

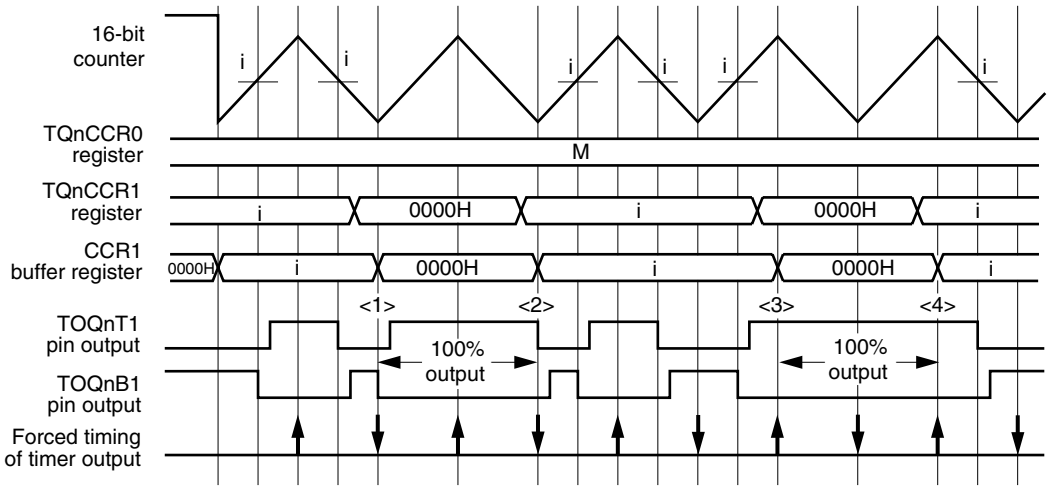
<2> 0% output is canceled by the crest interrupt (without a match with the 16-bit counter).
The crest interrupt forcibly raises the timer output. This cancels the 0% output.

<3> 0% output is selected by the crest interrupt (with a match with the 16-bit counter).
The crest interrupt forcibly raises the timer output, but lowering the timer output takes precedence when the value of the TQnCCRm register matches the value of the 16-bit counter. As a result, the 0% wave is output.

<4> 0% output is canceled by the valley interrupt (without a match with the 16-bit counter).
The valley interrupt forcibly lowers the timer output. This cancels the 0% output.

Remark ↑ means forced raising and ↓ means forced lowering.

<R> **Figure 10-10. 100% PWM Output Waveform (With Dead Time)**



<1> 100% output is selected by the valley interrupt (with a match with the 16-bit counter).
The valley interrupt forcibly lowers the timer output, but raising the timer output takes precedence when the value of the TQnCCRm register matches the value of the 16-bit counter. As a result, the 100% output is produced.

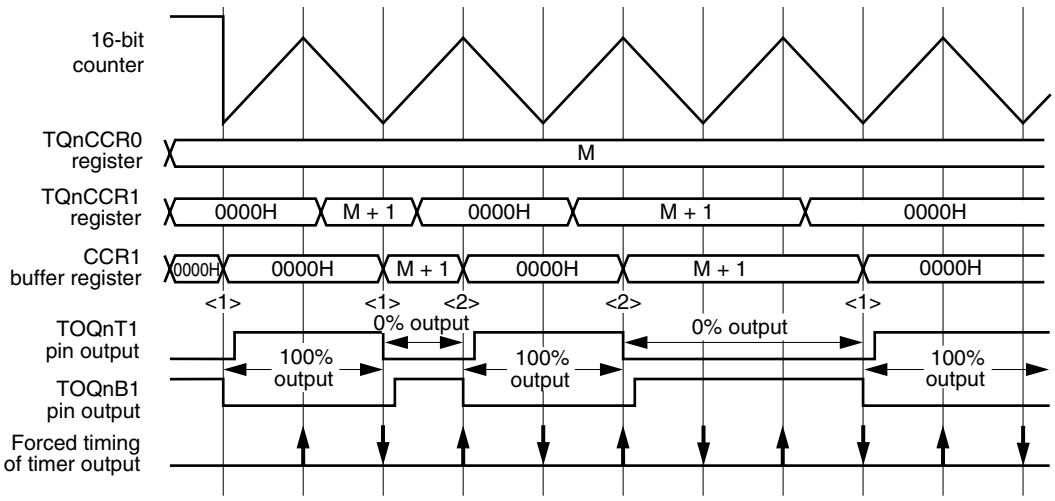
<2> 100% output is canceled by the valley interrupt (without a match with the 16-bit counter).
The valley interrupt forcibly lowers the timer output. This cancels the 100% output.

<3> 100% output is selected by the crest interrupt (without a match with the 16-bit counter).
The crest interrupt forcibly raises the timer output. This produces the 100% output.

<4> 100% output is canceled by the crest interrupt (without a match with the 16-bit counter).
The crest interrupt forcibly raises the timer output. This cancels the 100% output.

Remark ↑ means forced raising and ↓ means forced lowering.

<R> **Figure 10-11. PWM Output Waveform from 0% to 100% and from 100% to 0% (With Dead Time)**



<1> The valley interrupt selects 100% ←→ 0% or 0% ←→ 100% output.
Output can be selected from 100% ←→ 0% or 0% ←→ 100% immediately after the timer has been started.

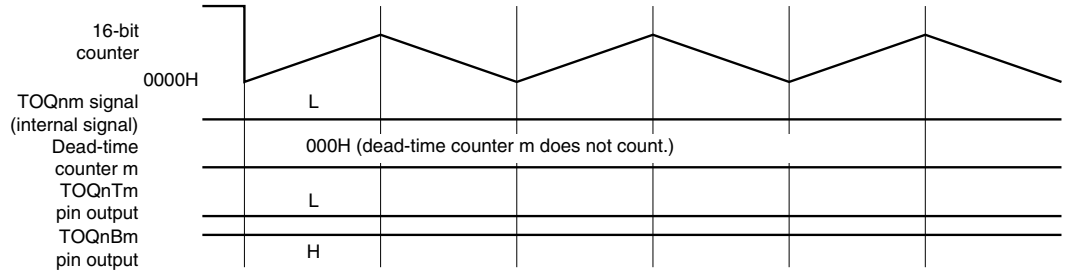
<2> The crest interrupt selects 100% ←→ 0% output.
The crest interrupt selects 100% → 0% output by using the timer output forced raising function and by a match between the 16-bit counter value and the TQnCCR0 register value.

(3) Output wave in vicinity of 0% and 100% output

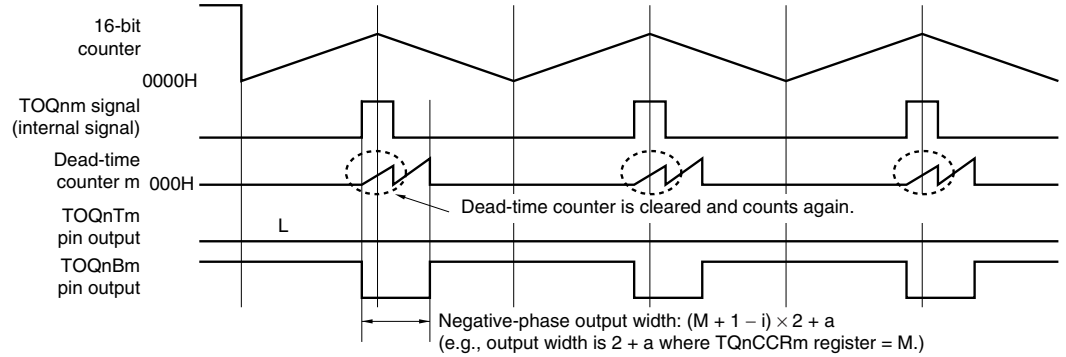
If an interrupt is generated because the value of the 16-bit counter matches the value of the compare register while dead time is being counted, the dead-time counter is cleared and starts its count operation again.
The output waveform of dead-time control in the vicinity of 0% and 100% output is shown below.

Figure 10-12. PWM Output Waveform with Dead Time (2)

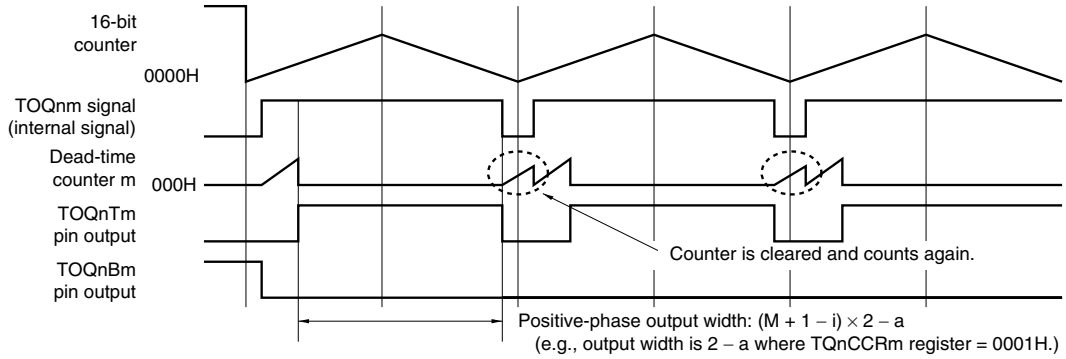
(a) 0% output (TQnCCRm register = M + 1, TQnCCR0 register = M, TQnDTC register = a)



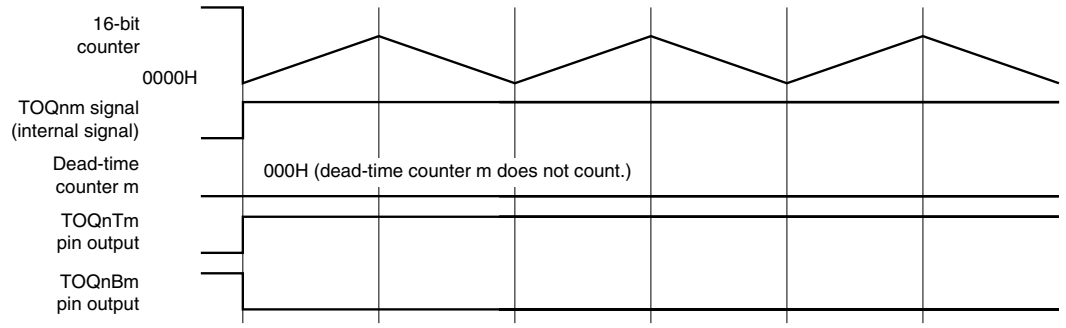
(b) In vicinity of 0% output (TQnCCRm register = i ≥ M + 1 − a/2, TQnCCR0 register = M, TQnDTC register = a)



(c) In vicinity of 100% output (TQnCCRm register = i ≤ a/2, TQnCCR0 register = M, TQnDTC register = a)



(d) 100% output (TQnCCRm register = 0000H, TQnCCR0 register = M, TQnDTC register = a)



Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

(4) Automatic dead-time width narrowing function (TQnOPT2.TQnDTM bit = 1)

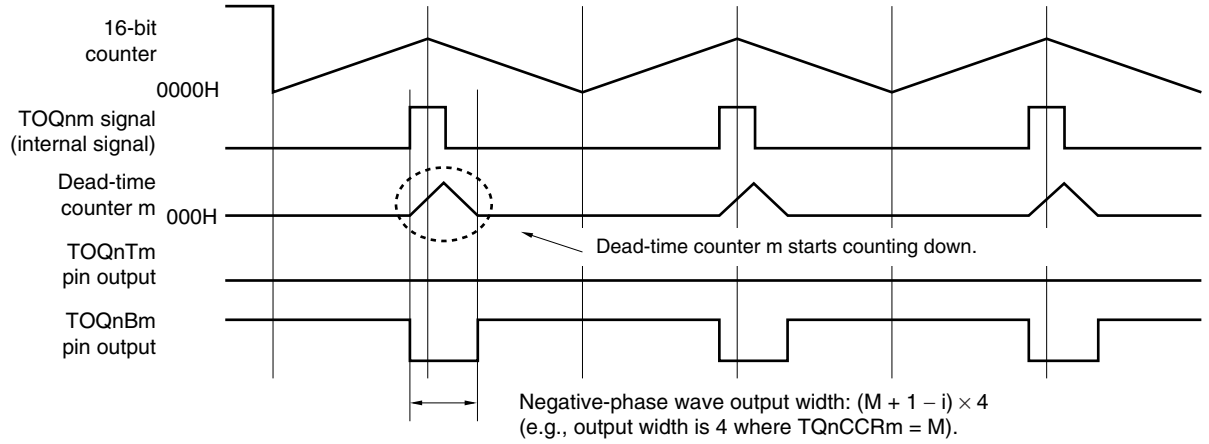
The dead-time width can be automatically narrowed in the vicinity of 0% output or 100% output by setting the TQnOPT2.TQnDTM bit to 1.

By setting the TQnDTM bit to 1, the dead-time counter is not cleared, but starts counting down if the TOQnm (internal signal) output of timer Q changes during dead-time counting.

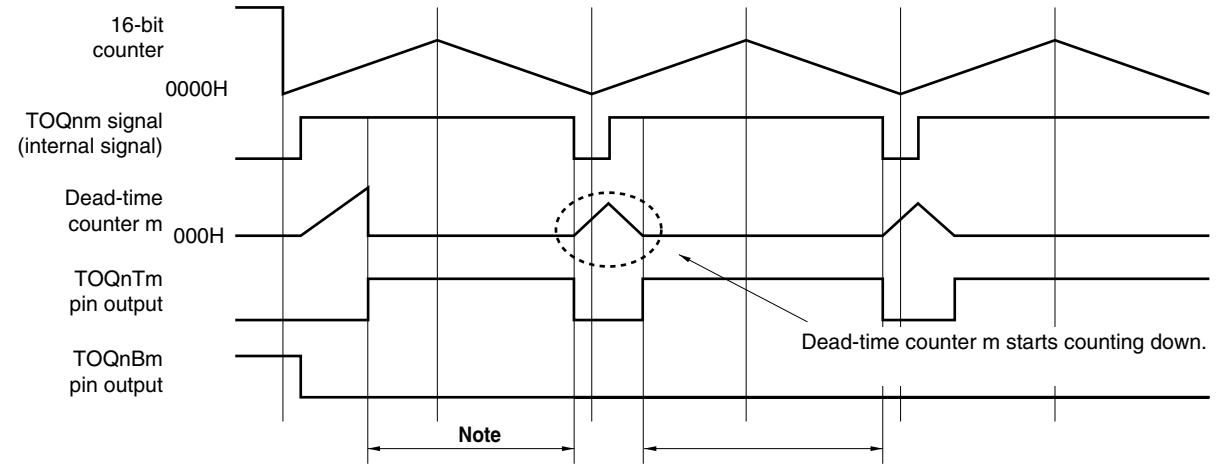
The following timing chart shows the operation of the dead-time counter when the TQnDTM bit is set to 1.

Figure 10-13. Operation of Dead-Time Counter m (1)

(a) In vicinity of 0% output (TQnCCRm register = i ≥ M + 1 − a/2, TQnCCR0 register = M, TQnDTC register = a)



(b) In vicinity of 100% output (TQnCCRm register = i ≤ a/2, TQnCCR0 register = M, TQnDTC register = a)



Note The output width of the first wave differs from that of the second and subsequent waves immediately after the TQnCTL0.TQnCE bit has been set. The first wave is shorter than the second wave because the dead time is fully counted.

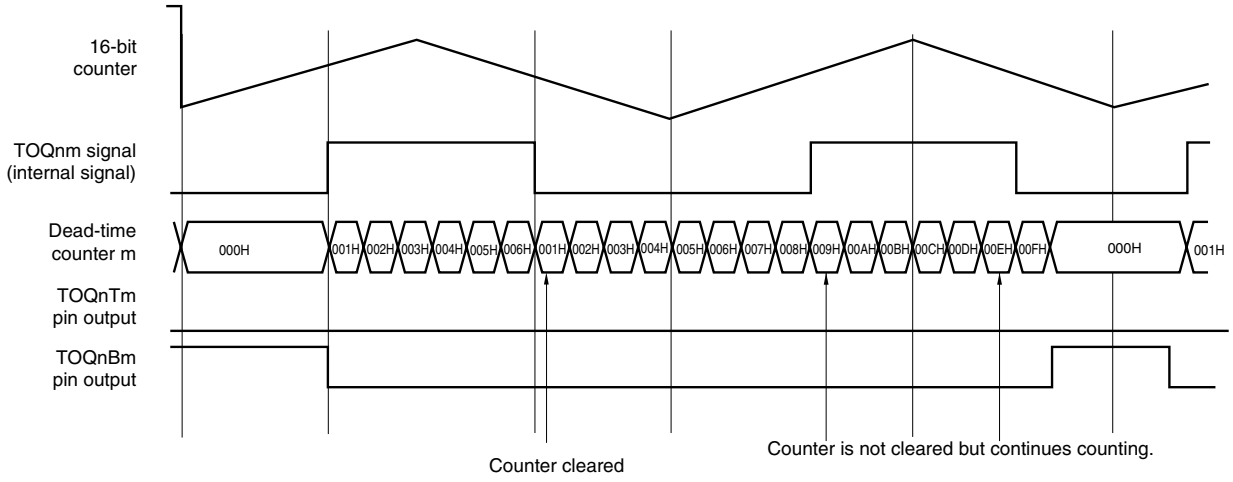
Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

(5) Dead-time control in case of incorrect setting

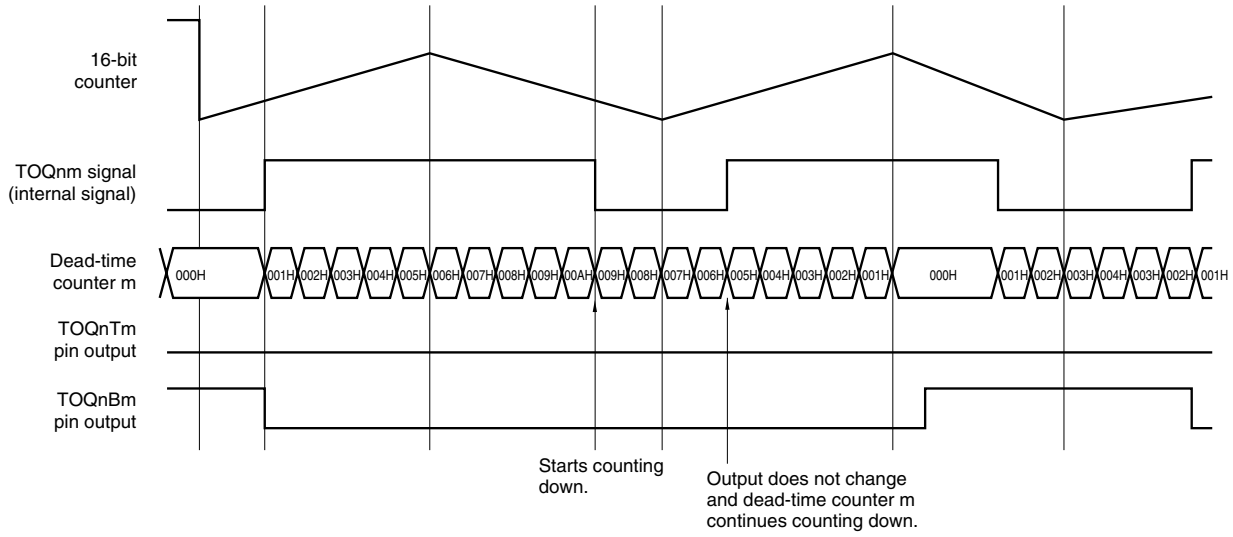
Usually, the TOQnm (internal signal) output of TMQn changes only once during dead-time counting, only in the vicinity of 0% and 100% output. This section shows an example where the TQnCCR0 register (carrier cycle) and TQnDTC register (dead-time value) are incorrectly set. If these registers are incorrectly set, the TOQnm (internal signal) output of TMQn changes more than once during dead-time counting. The following flowchart shows the 6-phase PWM output wave in this case.

Figure 10-14. Operation of Dead-Time Counter m (2)

(a) When TQnOPT2.TQnDTM bit = 0, TQnCCR0 register = 0006H, TQnDTC register = 000FH, TQnCCRm register = 0004H



(b) When TQnOPT2.TQnDTM bit = 1, TQnCCR0 register = 0006H, TQnDTC register = 000FH, TQnCCRm register = 0002H



Remark V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

10.4.3 Interrupt culling function

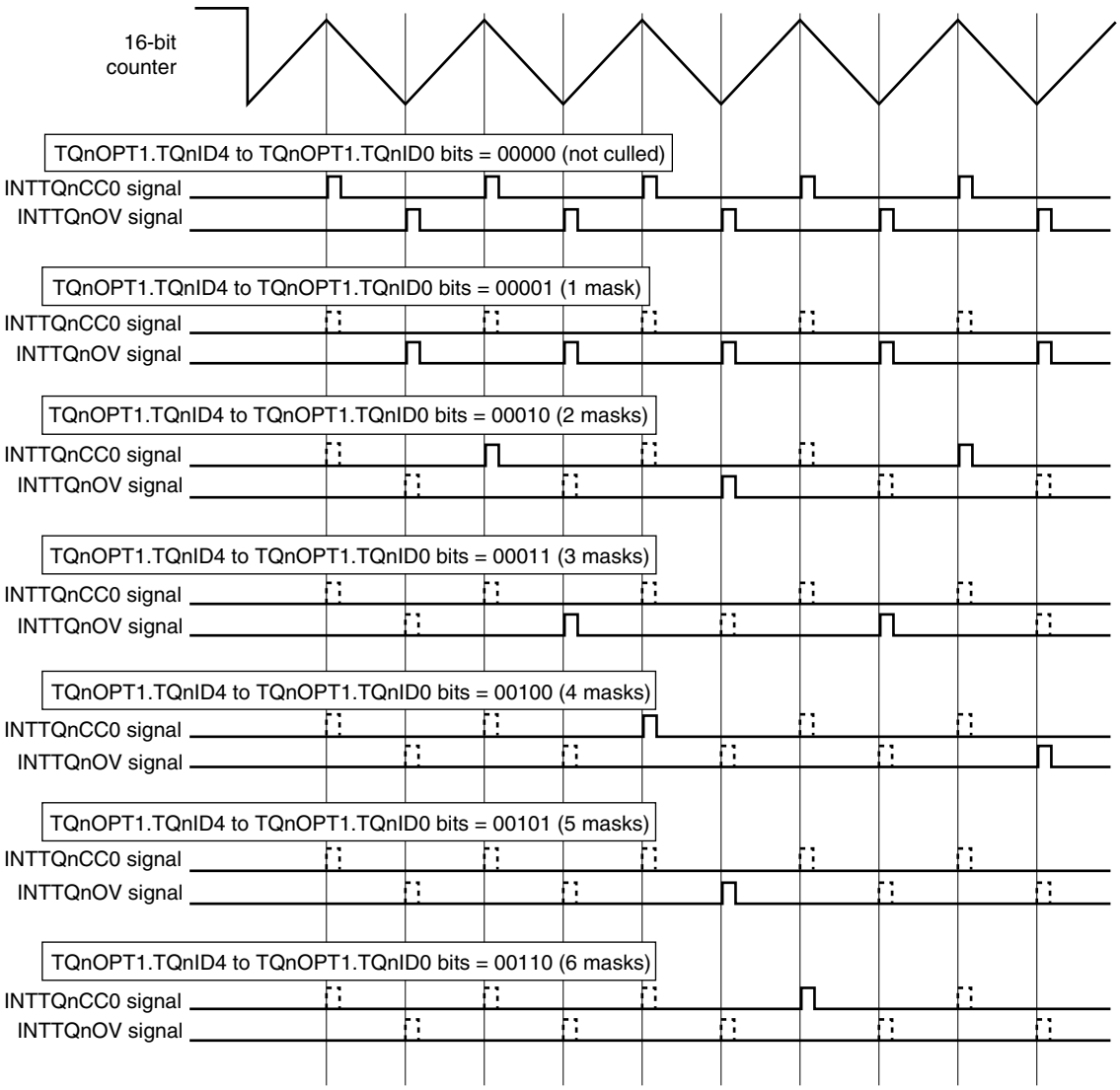
- The interrupts to be culled are INTTQnCC0 (crest interrupt) and INTTQnOV (valley interrupt).
- The TQnOPT1.TQnICE bit is used to enable output of the INTTQnCC0 interrupt and the number of times the interrupt is to be culled.
- The TQnOPT1.TQnIOE bit is used to enable output of the INTTQnOV interrupt and the number of times the interrupt is to be culled.
- <R> The TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits are used to specify the number of times for which an interrupt, subject to counting of culling, is counted.
  The interrupt is masked for the specified culling count and the masked interrupt occurs at the next occurrence timing.
- The TQnOPT2.TQnRDE bit is used to specify whether transfer is to be culled or not.
  If it is specified that transfer is to be culled, transfer is executed at the same timing as the interrupt output after culling. If it is specified that transfer is not to be culled, transfer is executed at the transfer timing after the TQnCCR1 register has been written.
- The TQnOPT0.TQnCMS bit is used to specify whether the registers with a transfer function are batch rewritten or anytime rewritten.
  The values of the registers are updated in synchronization with transferring when the TQnCMS bit is 0. When the TQnCMS bit is 1, the values of the registers are immediately updated when a new value is written to the registers.
  Transfer is performed from the TQnCCRm register to the CCRm buffer register in synchronization with interrupt culling timing.

Cautions 1. When using the interrupt culling function in the batch rewrite mode (transfer mode), execute the function in the intermittent batch rewrite mode (transfer culling mode).

2. An interrupt is generated at the timing after culling.

(1) Interrupt culling operation

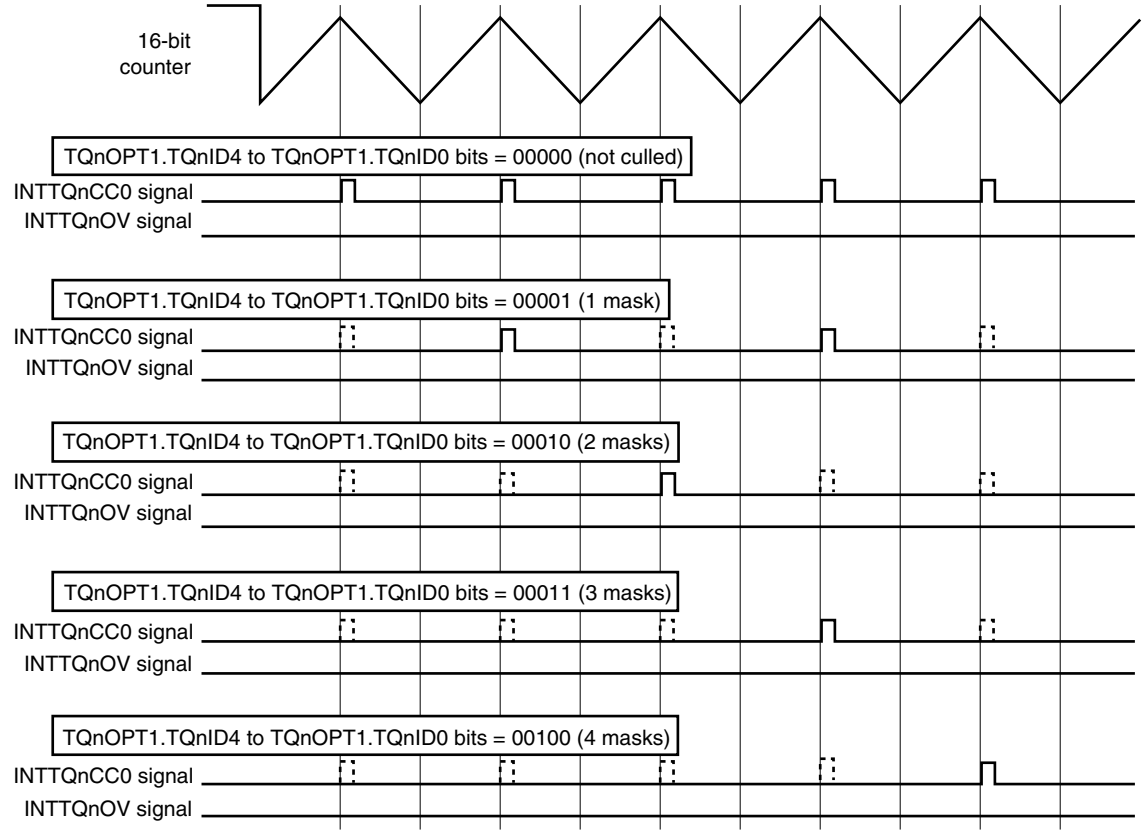
Figure 10-15. Interrupt Culling Operation When TQnOPT1.TQnICE Bit = 1, TQnOPT1.TQnIOE Bit = 1, TQnOPT2.TQnRDE Bit = 1 (Crest/Valley Interrupt Output)



Remarks 1. : Culled interrupt

2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Figure 10-16. Interrupt Culling Operation When TQnOPT1.TQnICE Bit = 1, TQnOPT1.TQnIOE Bit = 0, TQnOPT2.TQnRDE Bit = 1 (Crest Interrupt Output)

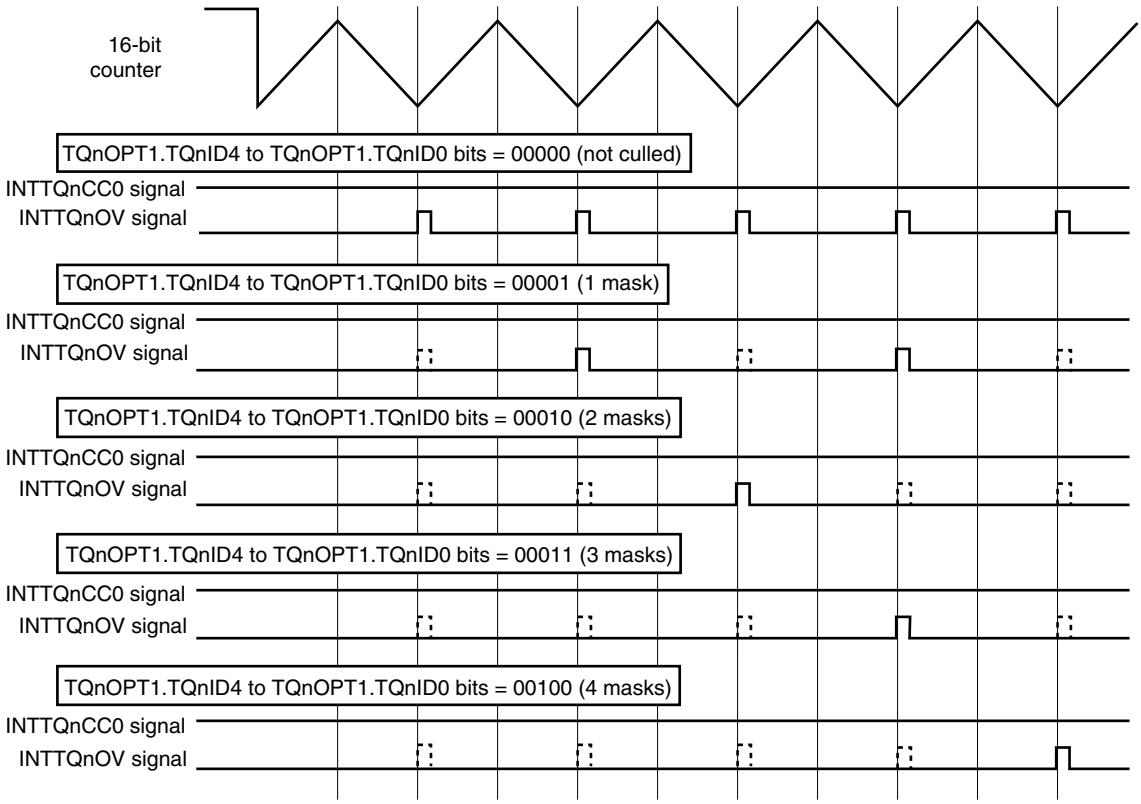


Remarks 1. : Culled interrupt

2. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

Figure 10-17. Interrupt Culling Operation When TQnOPT1.TQnICE Bit = 0, TQnOPT1.TQnIOE Bit = 1, TQnOPT2.TQnRDE Bit = 1 (Valley Interrupt Output)



Remarks 1. : Culled interrupt

2. V850E/IA3: n = 0

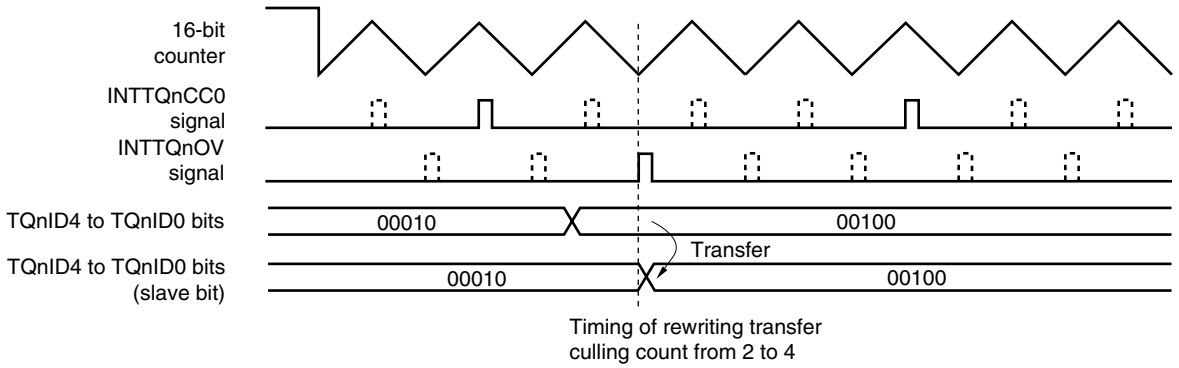
V850E/IA4: n = 0, 1

(2) To alternately output crest interrupt (INTTQnCC0) and valley interrupt (INTTQnOV)

To alternately output the crest and valley interrupts, set both the TQnOPT1.TQnICE and TQnOPT1.TQnIOE bits to 1.

Figure 10-18. Crest/Valley Interrupt Output

(a) TQnOPT0.TQnCMS bit = 0, TQnOPT2.TQnRDE bit = 1 (with transfer culling control)



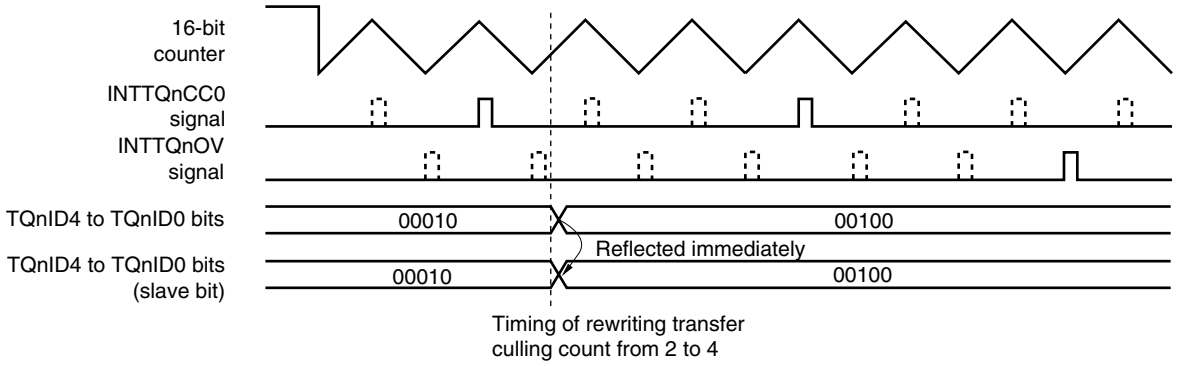
Remarks 1. Transfer is performed when the culled interrupt is output. The other transfer timing is ignored.

2. ⊓̄!: Culled interrupt

3. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

(b) TQnCMS bit = 1, TQnRDE bit = 0 or 1 (without transfer control)



Remarks 1. Rewriting is reflected immediately. The transfer timing is ignored.

2. ⊓̄!: Culled interrupt

3. V850E/IA3: n = 0

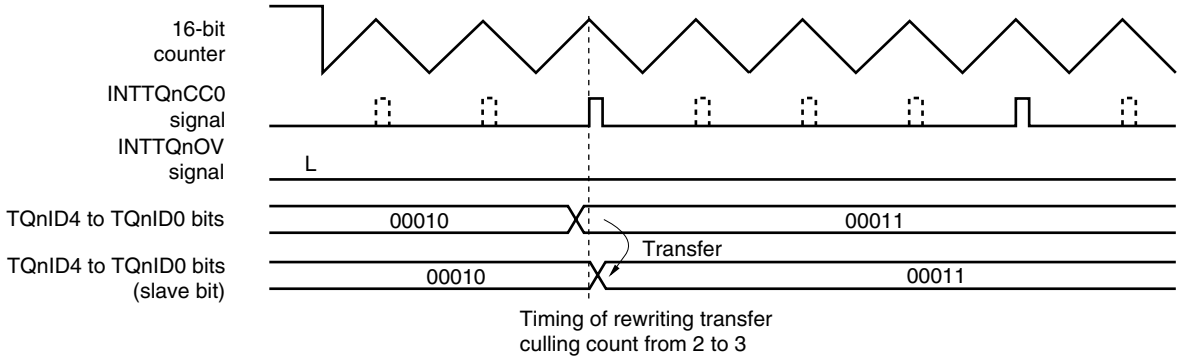
V850E/IA4: n = 0, 1

(3) To output only crest interrupt (INTTQnCC0)

Set the TQnOPT1.TQnICE bit to 1 and clear the TQnIOE bit to 0.

Figure 10-19. Crest Interrupt Output

(a) TQnOPT0.TQnCMS bit = 0, TQnOPT2.TQnRDE bit = 1 (with transfer culling control)



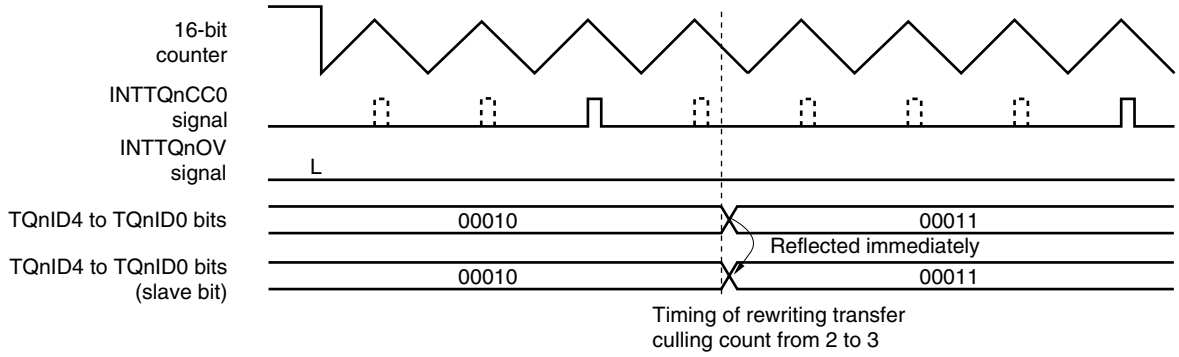
Remarks 1. Transfer is performed when the culled interrupt is output. The other transfer timing is ignored.

2. : Culled interrupt

3. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

(b) TQnOPT0.TQnCMS bit = 1, TQnOPT0.TQnRDE bit = 0 or 1 (without transfer control)



Remarks 1. Rewriting is reflected immediately. The transfer timing is ignored.

2. : Culled interrupt

3. V850E/IA3: n = 0

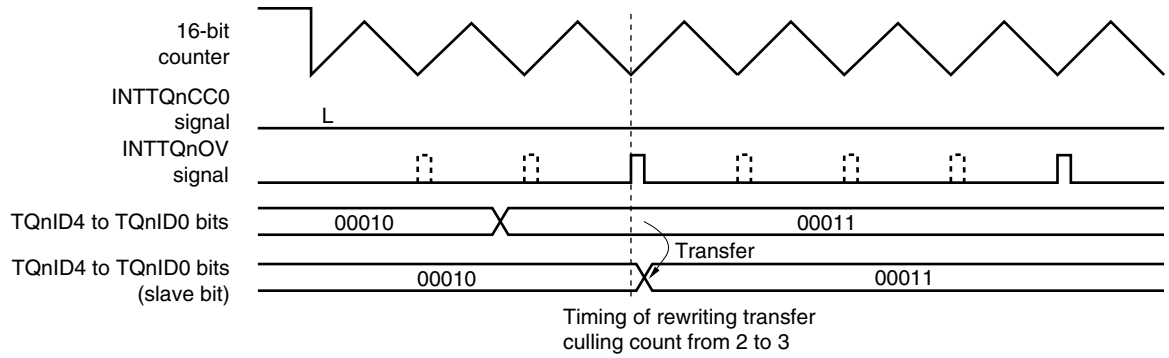
V850E/IA4: n = 0, 1

(4) To output only valley interrupt (INTTQnOV)

Clear the TQnOPT1.TQnICE bit to 0 and set the TQnIOE bit to 1.

Figure 10-20. Valley Interrupt Output

(a) TQnOPT0.TQnCMS bit = 0, TQnOPT2.TQnRDE bit = 1 (with transfer culling control)



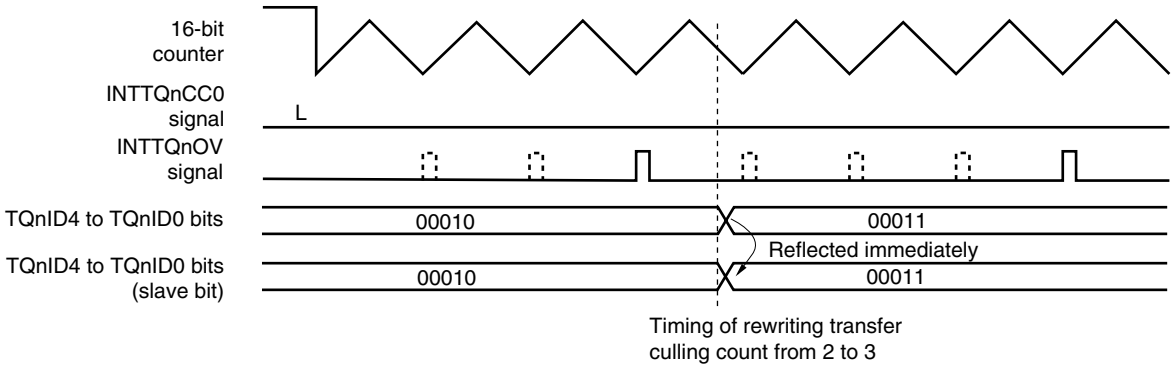
Remarks 1. Transfer is performed when the culled interrupt is output. The other transfer timing is ignored.

2. : Culled interrupt

3. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

(b) TQnOPT0.TQnCMS bit = 1, TQnOPT0.TQnRDE bit = 0 or 1 (without transfer control)



Remarks 1. Rewriting is reflected immediately. The transfer timing is ignored.

2. : Culled interrupt

3. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

10.4.4 Operation to rewrite register with transfer function

The following seven registers are provided with a transfer function and used to control a motor. Each of registers has a buffer register.

- TQnCCR0: Register that specifies the cycle of the 16-bit counter (TMQ)
- TQnCCR1: Register that specifies the duty factor of TOQnT1 (U) and TOQnB1 ($\overline{U}$)
- TQnCCR2: Register that specifies the duty factor of TOQnT2 (V) and TOQnB2 ($\overline{V}$)
- TQnCCR3: Register that specifies the duty factor of TOQnT3 (W) and TOQnB3 ($\overline{W}$)
- TQnOPT1: Register that specifies the culling of interrupts
- TPnCCR0: Register that specifies the A/D conversion start trigger generation timing (TMPn during tuning operation)
- TPnCCR1: Register that specifies the A/D conversion start trigger generation timing (TMPn during tuning operation)

The following three rewrite modes are provided in the registers with a transfer function.

- Anytime rewriting mode

<R> This mode is specified by setting the TQnOPT0.TQnCMS bit to 1. The setting of the TQnOPT2.TQnRDE bit is ignored.

In this mode, each compare register is updated independently, and the value of the compare register is updated as soon as a new value is written to it.

- Batch rewrite mode (transfer mode)

This mode is set by clearing the TQnOPT0.TQnCMS bit to 0, the TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits to 00000, and the TQnOPT2.TQnRDE bit to 0.

When data is written to the TQnCCR1 register, the seven registers are transferred to the buffer register all at once at the next transfer timing. Unless the TQnCCR1 register is rewritten, the transfer operation is not performed even if the other six registers are rewritten.

The transfer timing is the timing of each crest (match between the 16-bit counter value and TQnCCR0 register value) and valley (match between the 16-bit counter value and 0001H) regardless of the interrupt.

- Intermittent batch rewrite mode (transfer culling mode)

This mode is set by clearing the TQnOPT0.TQnCMS bit to 0 and setting the TQnOPT2.TQnRDE bit to 1.

When data is written to the TQnCCR1 register, the seven registers are transferred to the buffer register all at once at the next transfer timing. Unless the TQnCCR1 register is rewritten, the transfer operation is not performed even if the other six registers are rewritten.

If interrupt culling is specified by the TQnOPT1 register, the transfer timing is also culled as the interrupts are culled, and the seven registers are transferred all at once at the culled timing of crest interrupt (match between the 16-bit counter value and TQnCCR0 register value) or valley interrupt (match between the 16-bit counter value and 0001H).

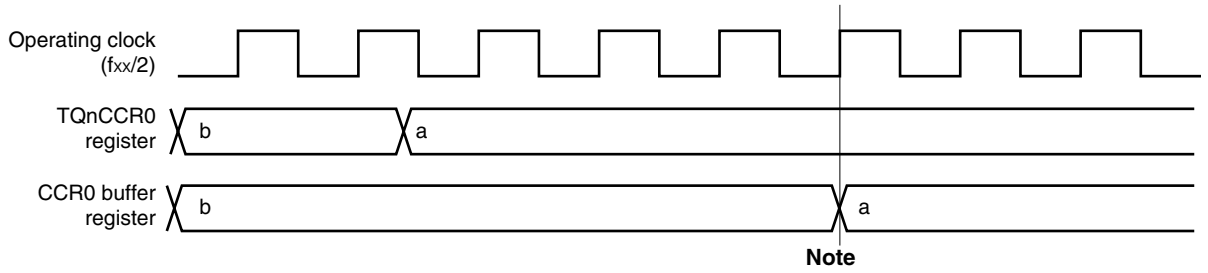
For details of the interrupt culling function, see **10.4.3 Interrupt culling function**.

(1) Anytime rewriting mode

This mode is set when the TQnOPT0.TQnCMS bit is 1. The setting of the TQnOPT2.TQnRDE bit is ignored.

In this mode, the value written to each register with a transfer function is immediately transferred to an internal buffer register and compared with the value of the counter. If a register with transfer function is rewritten in this mode after the count value of the 16-bit counter matches the value of the TQnCCRm register, the rewritten value is not reflected because the next match is ignored after the first match has occurred. If the register is rewritten during counting up, the new register value becomes valid after the counter has started counting down.

Figure 10-21. Timing of Reflecting Rewritten Value

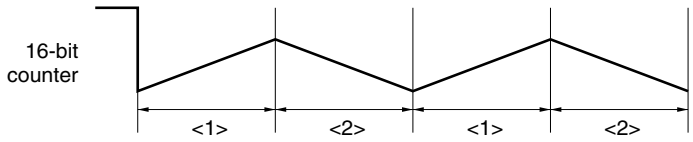


Note After the register (TQnCCR0, TQnCCR2, TQnCCR3, TQnOPT1, TPnCCR0, or TPnCCR1) has been written, the written value is transferred to an internal buffer register after four clocks of the operating clock. However, the value of only the TQnCCR1 register is transferred after 5 more clocks.

(a) Rewriting TQnCCR0 register

Even if the TQnCCR0 register is rewritten in the anytime rewriting mode, the new value may not be reflected in some cases.

Figure 10-22. Example of Rewriting TQnCCR0 Register



Rewriting during period <1> (rewriting during counting up)

If the newly rewritten value is greater than the value of the 16-bit counter, there is no problem because it will match the value of the 16-bit counter. If the new value is less than the value of the 16-bit counter, it will not match the value of the counter. As a result, the 16-bit counter overflows and continues counting up from 0000H until it matches the register value again, and the correct PWM waveform is not output.

Rewriting during period <2> (rewriting during counting down)

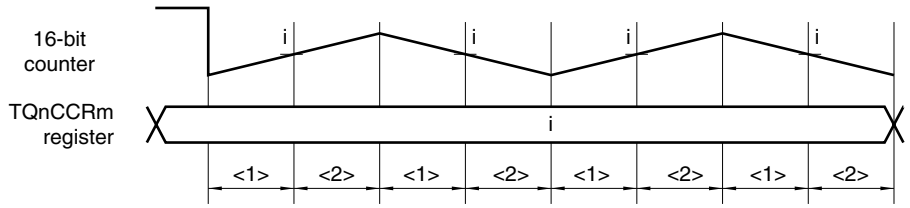
A match with the value of the 16-bit counter is ignored during counting down. Therefore, the rewritten period value is reflected starting from counting up in the next cycle as a match point.

(b) Rewriting TQnCCRm register

Figure 10-24 shows the timing of rewriting before the value of the 16-bit counter matches the value of the TQnCCRm register (<1> in Figure 10-23), and Figure 10-25 shows the timing of rewriting after the value of the 16-bit counter matches the value of the TQnCCRm register (<2> in Figure 10-23).

Figure 10-23. Basic Operation of 16-Bit Counter and TQnCCRm Register

(a) Basic figure



Remarks 1. i = Set value of TQnCCRm register

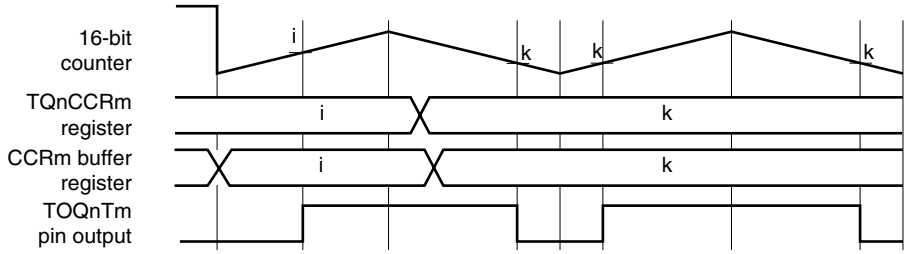
2. V850E/IA3: n = 0, m = 1 to 3

V850E/IA4: n = 0, 1, m = 1 to 3

Figure 10-24. Example of Rewriting TQnCCR1 to TQnCCR3 Registers (Rewriting Before Match Occurs)

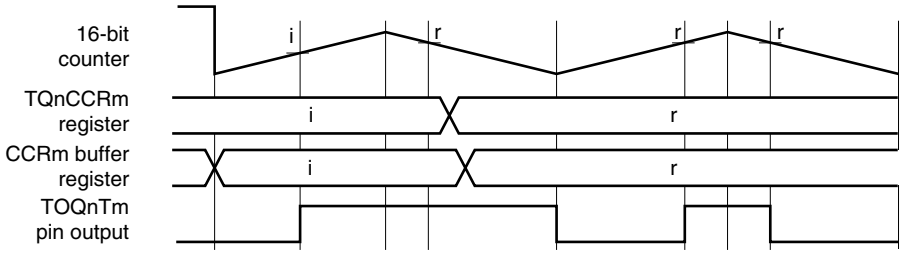
(a)

If the TQnCCRm register is rewritten before its value matches the value of the 16-bit counter, the register value will match the value of the 16-bit counter after the register has been rewritten. Consequently, the new register value is immediately reflected.



(b)

If a value less than the value of the 16-bit counter (greater if the counter is counting down) is written to the TQnCCRm register, the output waveform is as follows because the register value does not match the counter value.

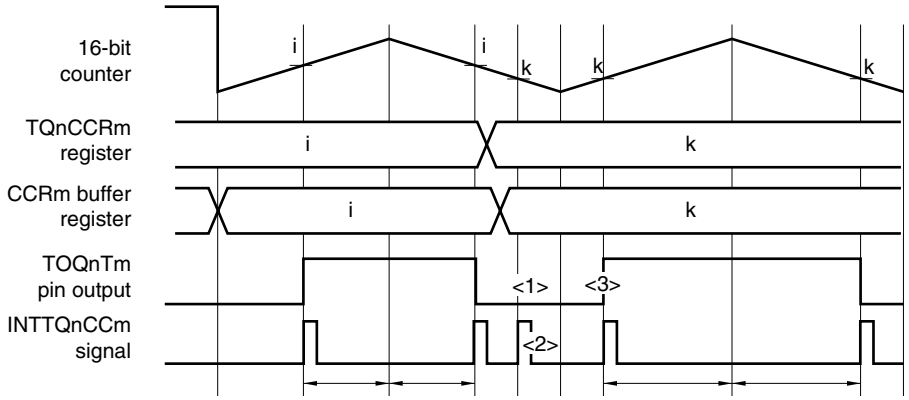


If the register value does not match the counter value, the TOQnTm pin output does not change. Even if the value of the 16-bit counter does not match the value of the TQnCCRm register, the TOQnTm pin output always changes to the high level if the crest interrupt occurs and to the low level if the valley interrupt occurs.
This is a function provided for 0% output and 100% output.
For details, see **10.4.2 (2) PWM output of 0%/100%**.

Remarks 1. i, r, k = Set values of TQnCCRm register

2. V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

Figure 10-25. Example of Rewriting TQnCCR1 to TQnCCR3 Registers (Rewriting After Match Occurs)



<1> Matching of the count value of the 16-bit counter and the value of the TQnCCRm register as a result of rewriting the register is ignored after a match signal has been generated, and the PWM output does not change.

<2> Even if the PWM output does not change, the interrupt generated upon a match between the 16-bit counter value and the TQnCCRm register value (INTTQnCCm) is output.

<3> The next match between the 16-bit counter and TQnCCRm register is valid after the counter has changed its counting direction to up or down, and the PWM output changes.

If the TQnCCRm register is rewritten after its value matches the value of the 16-bit counter, the next match is ignored after the first match occurs and the rewritten value is not reflected to the TOQnTm pin output. If the register is rewritten while the counter is counting down, the match that occurs after the counter starts counting down is valid (the match that occurs after the counter has started counting up is valid if the register is rewritten while the counter is counting up).

Remarks 1. i, r, k = Set value of TQnCCRm register

2. V850E/IA3: n = 0, m = 1 to 3
V850E/IA4: n = 0, 1, m = 1 to 3

(c) Rewriting TQnOPT1 register

The interrupt culling counter is cleared when the TQnOPT1 register is written. When the interrupt culling counter has been cleared, the measured number of times the interrupt has occurred is discarded. Consequently, the interrupt generation interval is temporarily extended.

To avoid this operation, rewrite the TQnOPT1 register in the intermittent batch rewriting mode (transfer culling mode).

For details of rewriting the TQnOPT1 register, see **10.4.3 Interrupt culling function**.

(2) Batch rewrite mode (transfer mode)

This mode is set by clearing the TQnOPT0.TQnCMS bit to 0, the TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits to 00000, and the TQnOPT2.TQnRDE bit to 0.

In this mode, the values written to each compare register are transferred to the internal buffer register all at once at the transfer timing and compared with the counter value.

(a) Rewriting procedure

If data is written to the TQnCCR1 register, the values set to the TQnCCR0 to TQnCCR3, TQnOPT1, TPnCCR0, and TPnCCR1 registers are transferred all at once to the internal buffer register at the next transfer timing. Therefore, write to the TQnCCR1 register last. Writing to the register is prohibited after the TQnCCR1 register has been written and before the transfer timing is generated (until the crest (match between the 16-bit counter value and TQnCCR0 register value) or the valley (match between the 16-bit counter value and 0001H)). The operation procedure is as follows.

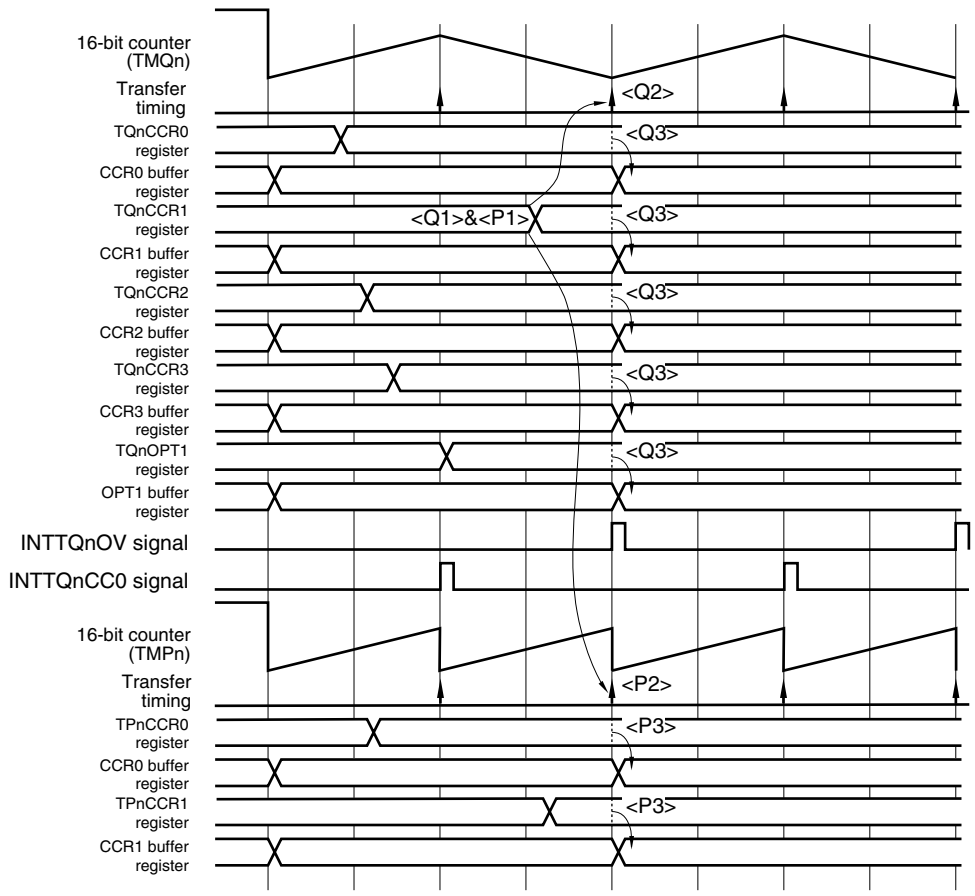
<1> Rewriting the TQnCCR0, TQnCCR2, TQnCCR3, TQnOPT1, TPnCCR0, and TPnCCR1 registers.
Do not rewrite registers that do not have to be rewritten.

<2> Rewriting the TQnCCR1 register.
Rewrite the same value to the register even when it is not necessary to rewrite the TQnCCR1 register.

<3> Holding the next rewriting pending until the transfer timing is generated.
Rewrite the register next time after the INTTQnOV or INTTQnCC0 interrupt has occurred.

<4> Return to <1>.

Figure 10-26. Basic Operation in Batch Mode



[Operation of TMQn]

<Q1> Write the TQnCCR1 register

<Q2> The target timing is the first transfer timing after a write to the TQnCCR1 register.

<Q3> The values are transferred all at once at the transfer timing.

[Operation of TMPn]

<P1> Write the TQnCCR1 register

<P2> The target timing is the first transfer timing after a write to the TQnCCR1 register.

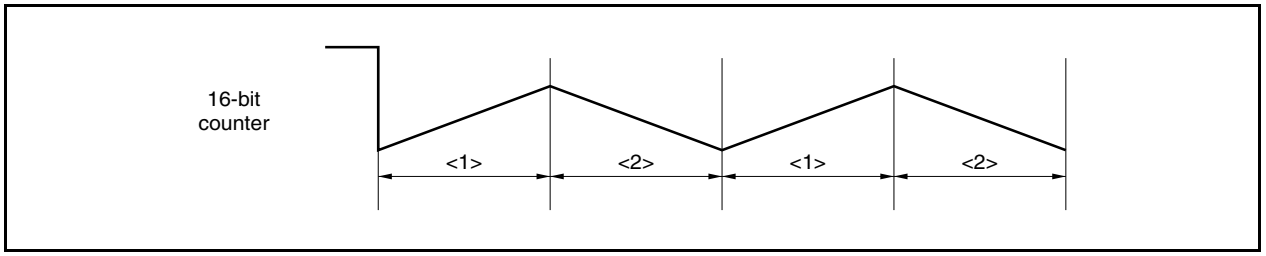
<P3> The values are transferred all at once at the transfer timing.

(b) Rewriting TQnCCR0 register

When rewriting the TQnCCR0 register in the batch rewrite mode, the output waveform differs depending on whether transfer occurs at the crest (match between the 16-bit counter value and TQnCCR0 register value) or at the valley (match between the 16-bit counter value and 0001H). Usually, it is recommended to rewrite the TQnCCR0 register while the 16-bit counter is counting down, and transfer the register value at the transfer timing of the crest timing.

Figure 10-28 shows an example of rewriting the TQnCCR0 register while the 16-bit counter is counting up (during period <1> in Figure 10-27). Figure 10-29 shows an example of rewriting the TQnCCR0 register while the counter is counting down (during period <2> in Figure 10-27).

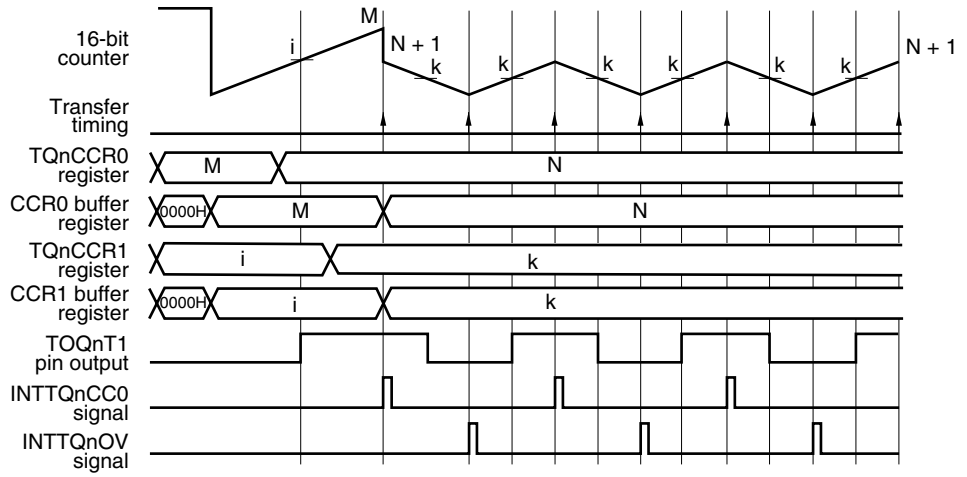
Figure 10-27. Basic Operation of 16-Bit Counter



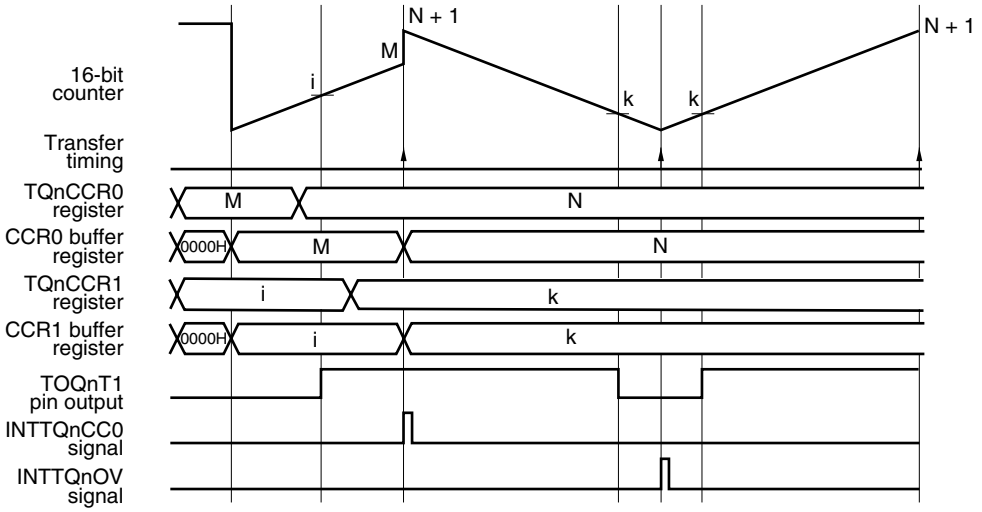
The transfer timing in Figure 10-28 is at the point where the crest timing occurs. While the 16-bit counter is counting down, the cycle changes and an asymmetrical triangular wave is output. Because the cycle changes, rewrite the duty factor (voltage data value).

Figure 10-28. Example of Rewriting TQnCCR0 Register (During Counting Up)

(a) M > N



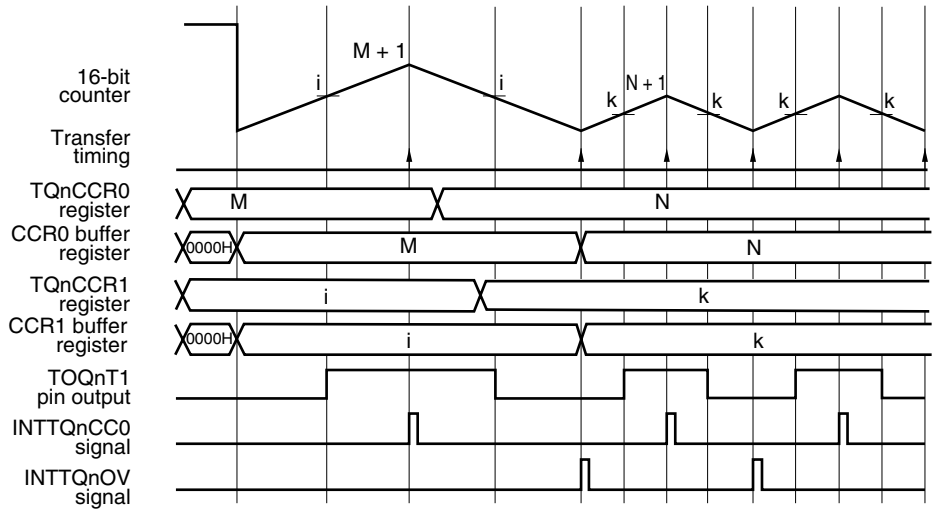
(b) M < N



Remarks 1. If transfer (match between the value of the 16-bit counter and the value of the CCR0 buffer register) occurs in the 6-phase PWM output mode, the value of the TQnCCR0 register plus 1 is loaded to the 16-bit counter. In this way, the expected wave can be output even if the cycle value is changed at the transfer timing of the crest (match between the 16-bit counter value and the TQnCCR0 register value) timing.

2. M: Value of CCR0 buffer register before rewriting
N: Value of CCR0 buffer register after rewriting

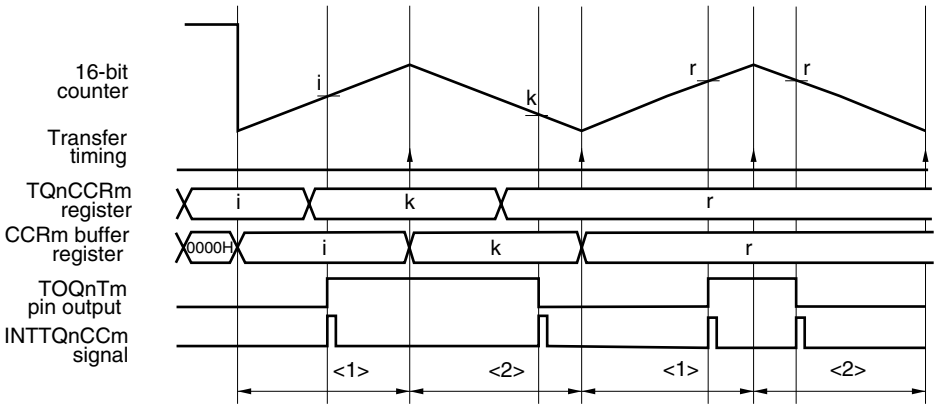
Figure 10-29. Example of Rewriting TQnCCR0 Register (During Counting Down)



Because the next transfer timing is at the point of the valley (match between the 16-bit counter value and 0001H), the cycle value changes from the next cycle and output of a symmetrical triangular wave is maintained. Because the cycle changes, rewrite the duty value (voltage data value) as required.

(c) Rewriting TQnCCRm register

Figure 10-30. Example of Rewriting TQnCCRm Register



Rewriting during period <1> (rewriting during counting up)

Because the TQnCCRm register value is transferred at the transfer timing of the crest (match between the 16-bit counter value and TQnCCR0 register value), an asymmetrical triangular wave is output.

Rewriting during period <2> (rewriting during counting down)

Because the TQnCCRm register value is transferred at the transfer timing of the valley (match between the 16-bit counter value and 0001H), a symmetrical triangular wave is output.

Remark m = 1 to 3

(d) Transferring TQnOPT1 register value

Do not set the TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits to other than 00000. When using the interrupt culling function, rewrite the TQnOPT1 register in the intermittent batch rewrite mode (transfer culling mode).

For details of rewriting the TQnOPT1 register, see **10.4.3 Interrupt culling function**.

(3) Intermittent batch rewriting mode (transfer culling mode)

This mode is set when the TQnOPT0.TQnCMS bit is 0 and the TQnOPT2.TQnRDE bit is 1.

In this mode, the values written to each compare register are transferred to the internal buffer register all at once at the culled transfer timing and compared with the counter value.

The transfer timing is the timing at which an interrupt is generated (INTTQnCC0, INTTQnOV) by interrupt culling.

For details of the interrupt culling function, see **10.4.3 Interrupt culling function**.

(a) Rewriting procedure

If data is written to the TQnCCR1 register, the TQnCCR0 to TQnCCR3, TQnOPT1, TPnCCR0, and TPnCCR1 registers are transferred all at once to the internal buffer register at the next transfer timing. Therefore, write to the TQnCCR1 register last. Writing to the register is prohibited after the TQ0CCR1 register has been written until the transfer timing is generated (until the INTTQnOV or INTTQnCC0 interrupt occurs). The operation procedure is as follows.

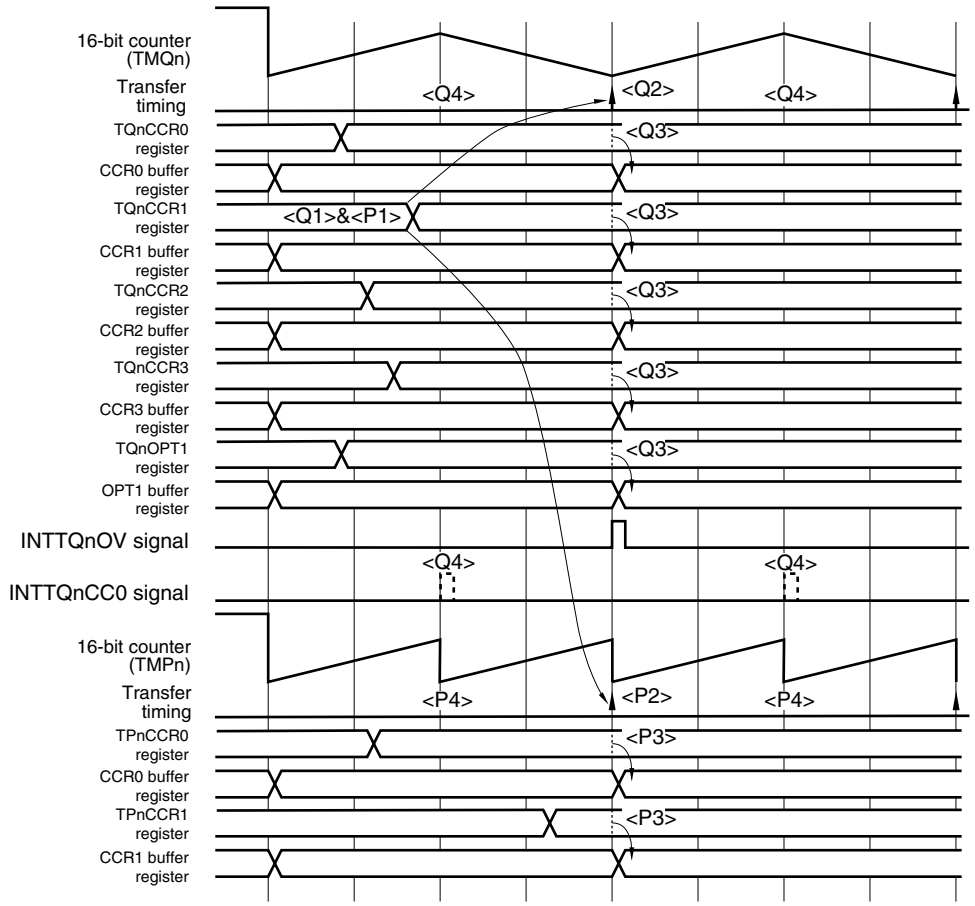
<1> Rewrite the TQnCCR0, TQnCCR2, TQnCCR3, TQnOPT1, TPnCCR0, and TPnCCR1 registers.
Do not rewrite registers that do not have to be rewritten.

<2> Rewrite the TQnCCR1 register.
Rewrite the same value to the register even when it is not necessary to rewrite the TQnCCR1 register.

<3> Hold the next rewriting pending until the transfer timing is generated.
Perform the next rewrite after the INTTQnOV or INTTQnCC0 interrupt has occurred.

<4> Return to <1>.

Figure 10-31. Basic Operation in Intermittent Batch Rewriting Mode



[TMQn operation]

<Q1> Write the TQnCCR1 register.

<Q2> Rewrite the register at the transfer timing that is generated after the TQnCCR1 register has been rewritten.

<Q3> The registers are transferred all at once at the transfer timing.

<Q4> The transfer timing is also culled as the interrupts are culled.

[TMPn operation]

<P1> Write the TQnCCR1 register.

<P2> Rewrite the register at the transfer timing that is generated after the TQnCCR1 register has been rewritten.

<P3> The registers are transferred all at once at the transfer timing.

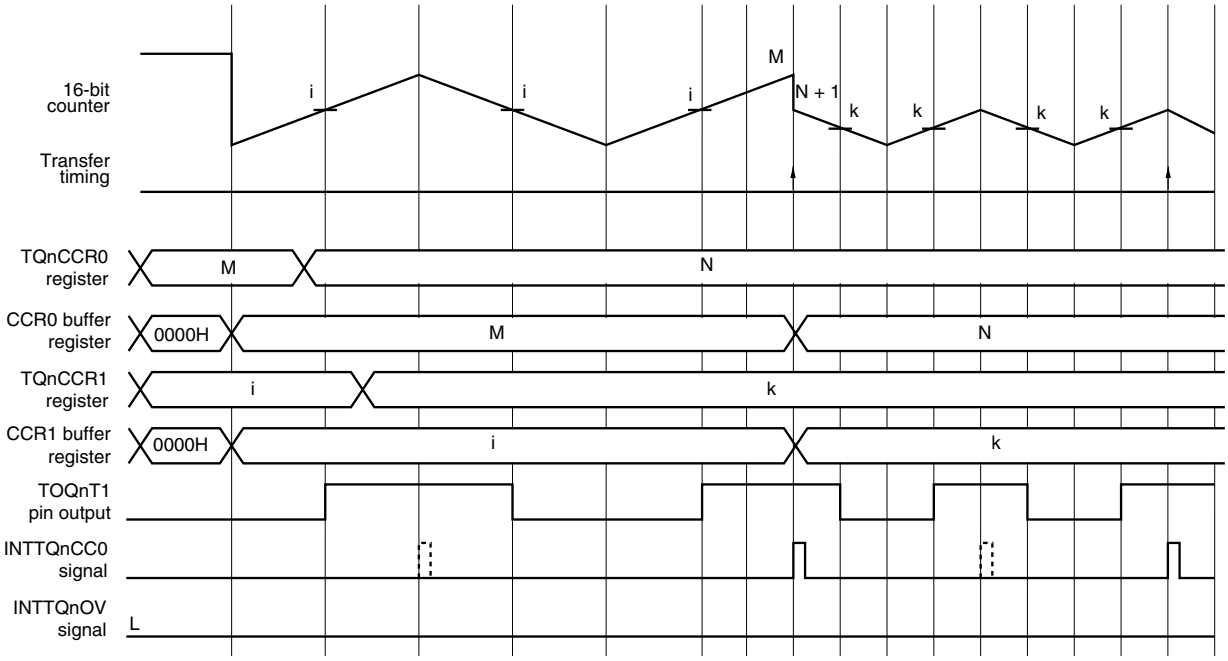
<P4> The transfer timing is also culled as the interrupts are culled.

Remark This is an example of the operation when the TQnOPT1.TQnICE bit = 1, TQnOPT1.TQnIOE bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits = 00001.

(b) Rewriting TQnCCR0 register

When rewriting the TQnCCR0 register in the intermittent batch mode, the output waveform differs depending on where the occurrence of the crest or valley interrupt is specified by the interrupt culling setting. The following figure illustrates the change of the output waveform when interrupts are culled.

Figure 10-32. Rewriting TQnCCR0 Register (When Crest Interrupt Is Set)



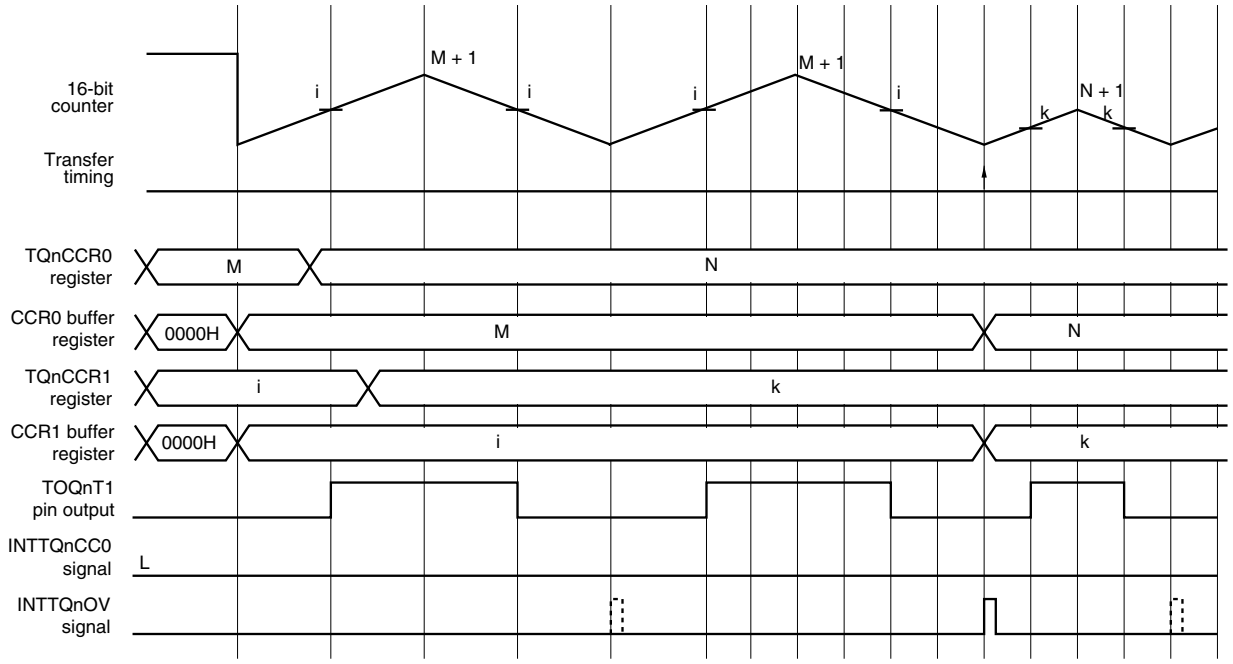
The transfer timing is generated when the crest interrupt occurs, the period of counting up and counting down changes, and an asymmetrical triangular wave is output.

Remarks 1. This is an example of the operation when the TQnOPT1.TQnICE bit = 1, TQnOPT1.TQnIOE bit = 0, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits = 00001.

2. ¦¯¦: Culled interrupt

3. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Figure 10-33. Rewriting TQnCCR0 Register (When Valley Interrupt Is Set)



The transfer timing is generated when the valley interrupt occurs, the cycle of counting up and counting down becomes identical, and a symmetrical triangular wave is output.

Remarks 1. This is an example of the operation when the TQnOPT1.TQnICE bit = 0, TQnOPT1.TQnIOE bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 bits = 00001.

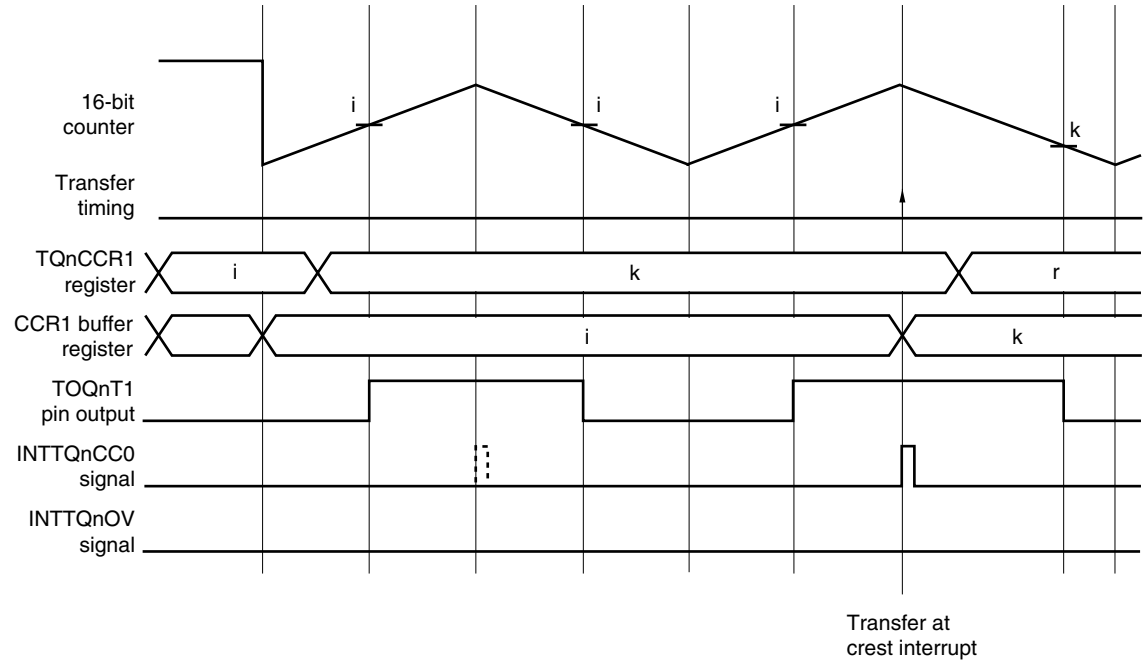
2. : Culled interrupt

3. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

(c) Rewriting TQnCCR1 to TQnCCR3 registers

- Transfer at crest when crest interrupt is set
  Because the register is transferred at the transfer timing of the crest interrupt, an asymmetrical triangular wave is output.

Figure 10-34. Rewriting TQnCCR1 Register
(TQnOPT1.TQnICE Bit = 1, TQnOPT1.TQnIOE Bit = 0, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 = 00001)

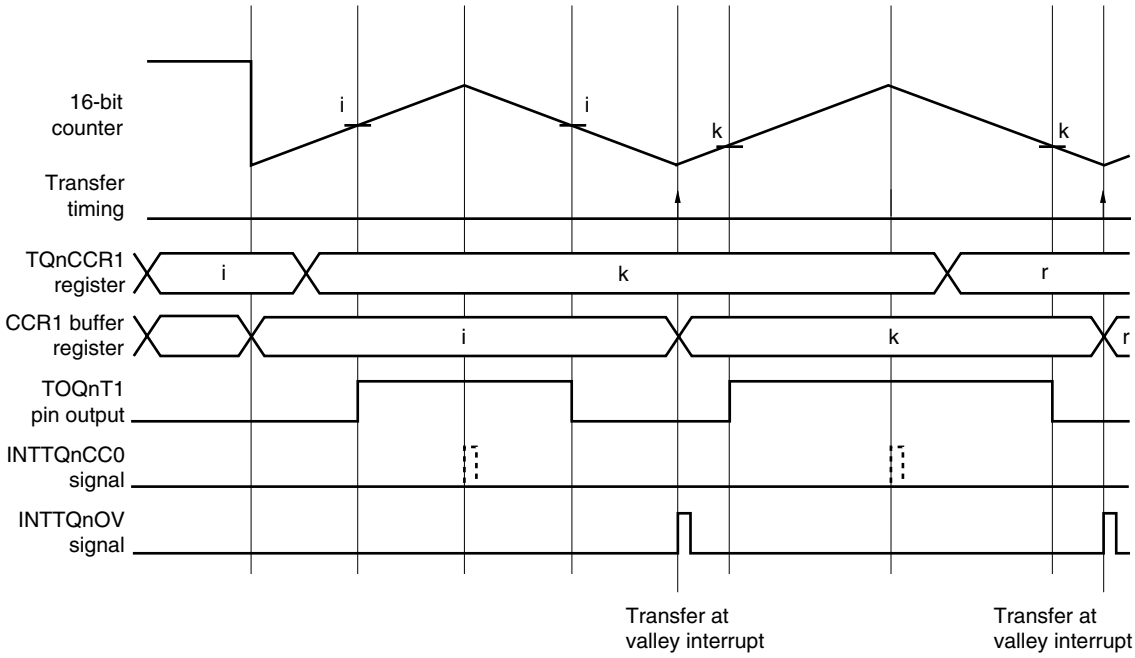


Remarks 1. : Culled interrupt
2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

- Transfer at valley when valley interrupt is set

  Because the register is transferred at the transfer timing of the valley interrupt, a symmetrical triangular wave is output.

Figure 10-35. Rewriting TQnCCR1 Register
(TQnOPT1.TQnICE Bit = 1, TQnOPT1.TQnIOE Bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 = 00001)



Remarks 1. : Culled interrupt

2. V850E/IA3: n = 0
V850E/IA4: n = 0, 1

(d) Rewriting TQnOPT1 register

Because a new interrupt culling value is transferred when the value of the interrupt culling counter matches the value of the 16-bit counter, the next interrupt and those that follow occur at the set interval.

For details of rewriting the TQnOPT1 register, see **10.4.3 Interrupt culling function**.

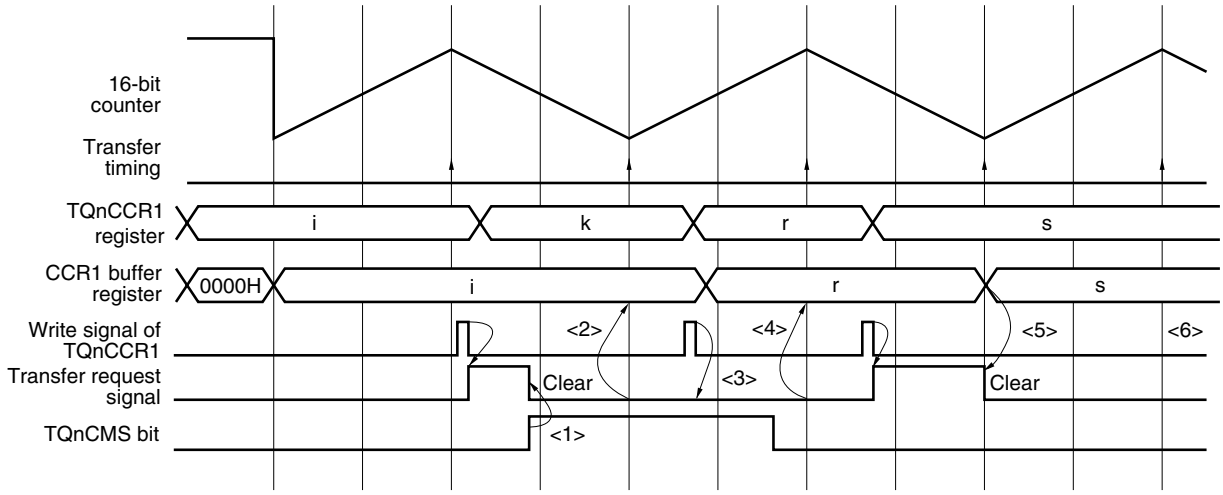
(4) Rewriting TQnOPT0.TQnCMS bit

The TQnCMS bit can select the anytime rewrite mode and batch rewrite mode. This bit can be rewritten during timer operation (when TQnCTL0.TQnCE bit = 1). However, the operation and caution illustrated in Figure 10-36 are necessary.

If the TQnCCR1 register is written when the TQnCMS bit is cleared to 0, a transfer request signal (internal signal) is set.

When the transfer request signal is set, the register is transferred at the next transfer timing, and the transfer request signal is cleared. This transfer request signal is also cleared when the TQnCMS bit is set to 1.

Figure 10-36. Rewriting TQnCMS Bit



<1> If the TQnCCR1 register is rewritten when the TQnCMS bit is 0, the transfer request signal is set.
If the TQnCMS bit is set to 1 in this status, the transfer request signal is cleared.

<2> The register is not transferred because the TQnCMS bit is set to 1 and the transfer request signal is cleared.

<3> The transfer request signal is not set even if the TQnCCR1 register is written when the TQnCMS bit is 1.

<4> The transfer request signal is not set even if the TQnCCR1 register is written when the TQnCMS bit is 1, so even if the TQnCMS bit is cleared to 0, transfer does not occur at the subsequent transfer timing.

<5> The transfer request signal is set if the TQnCCR1 register is written when the TQnCMS bit is 0.
Transfer is performed at the subsequent transfer timing and the transfer request signal is cleared.

<6> Once transfer has been performed, the transfer request signal is cleared. Therefore, transfer is not performed at the next transfer timing.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

10.4.5 TMPn tuning operation for A/D conversion start trigger signal output

This section explains the tuning operation of TMPn and TMQn in the 6-phase PWM output mode.

In the 6-phase PWM output mode, the tuning operation is performed with TMQn serving as the master and TMPn as a slave. The conversion start trigger signal of A/D converters 0 and 1 can be set as the A/D conversion start trigger source by the INTTPnCC0 and INTTPnCC1 signals of TMPn and the INTTQnOV and INTTQnCC0 signals of TMQn.

Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

(1) Tuning operation starting procedure

The TMPn and TMQn registers should be set using the following procedure to perform the tuning operation.

(a) Setting of TMPn register (stop the operations of TMQn and TMPn (by clearing the TQnCTL0.TQnCE bit and TPnCTL0.TPnCE bit to 0))

- Set the TPnCTL1 register to 85H (set the tuning operation slave mode and free-running timer mode).
- Clear the TP0IOC0 to TP0IOC2 registers to 00H (the I/O function for TMP0 is not used)[Note].
- Clear the TP0OPT0 register to 00H (select the compare register).
- Set an appropriate value to the TPnCCR0 and TPnCCR1 registers (set the default value for comparison for starting the operation).

Note Valid only for TMP0 (not available for TMP1).

(b) Setting of TMQn register

- Set the TQnCTL1 register to 07H (set the master mode and 6-phase PWM output mode).
- Set an appropriate value to the TQnIOC0 register (set the output mode of TOQnT1 to TOQnT3). However, clear the TQnOL0 bit to 0 and set the TQnOE0 bit to 1 (enable positive phase output). Unless this setting is made, the crest interrupt (INTTQnCC0) and valley interrupt (INTTQnOV) do not occur. Consequently, the conversion start trigger signal of A/D converters 0 and 1 is not correctly generated.
- Clear the TQ0IOC1 and TQ0IOC2 registers to 00H (TIQ00 to TIQ03, the EVTQ0, and TRGQ0 pins of TMQ0 are not used)[Note].
- Clear the TQnOPT0 register to 00H (select the compare register).
- Set an appropriate value to the TQnCCR0 to TQnCCR3 registers (set the default value for comparison for starting the operation).
- Set the TQnCTL0 register to 0xH (clear the TQnCE bit to 0 and set the operating clock of TMQn). The operating clock of TMQn set by the TQnCTL0 register is also supplied to TMPn, and the count operation is performed at the same timing. The operating clock of TMPn set by the TPnCTL0 register is ignored.

Note Valid only for TMQ0 (not available for TMQ1).

(c) Setting of TMQOPn (TMQn option) register

- Set an appropriate value to the TQnOPT1 and TQnOPT2 registers.
- Set an appropriate value to the TQnIOC3 register (set TOQnB1 to TOQnB3 in the output mode).
- Set an appropriate value to the TQnDTC register (set the default value for comparison for starting the operation).

(d) Setting of alternate function

- Select the alternate function of the port by setting the port to the port control mode.

(e) Set the TPnCE bit to 1 and set the TQnCE bit to 1 immediately after that to start the 6-phase PWM output operation.

Rewriting the TQnCTL0, TQnCTL1, TQ0IOC1[Note 1], TQ0IOC2[Note 1], TPnCTL0, TPnCTL1, and TP0IOC0[Note 2], TP0IOC1[Note 2], TP0IOC2[Note2] registers is prohibited during operation. The operation and the PWM output waveform are not guaranteed if any of these registers is rewritten during operation. However, rewriting the TQnCTL0.TQnCE bit to clear it is permitted. Manipulating (reading/writing) the other TMQn, TMPn, and TMQn option registers is prohibited until the TPnCTL0.TPnCE bit is set to 1 and then the TQnCE bit is set to 1.

Notes 1. Valid only for TMQ0 (not available for TMQ1).

2. Valid only for TMP0 (not available for TMP1).

Caution When tuning TMPn in the 6-phase PWM output mode, output of the TOP00 and TOP01 pins is disabled.
Clear the TP0IOC0.TP0OE0 and TP0IOC0.TP0OE1 bits to 0.

(2) Tuning operation clearing procedure

To clear the tuning operation and exit the 6-phase PWM output mode, set the TMPn and TMQn registers using the following procedure.

<1> Clear the TQnCTL0.TQnCE bit to 0 and stop the timer operation.
<2> Clear the TPnCTL0.TPnCE bit to 0 so that TMPn can be separated.
<3> Stop the timer output by using the TQ0IOC0[Note 1] and TP0IOC0[Note 2] registers.
<4> Clear the TPnCTL1.TPnSYE bit to 0 to clear the tuning operation.

Notes 1. Valid only for TMQ0 (not available for TMQ1).

2. Valid only for TMP0 (not available for TMP1).

Caution Manipulating (reading/writing) the other TMQn, TMPn, and TMQn option registers is prohibited until the TQnCE bit is set to 0 and then the TPnCE bit is set to 0.

(3) When not tuning TMPn

When the match interrupt signal of TMPn is not necessary as the conversion trigger source that starts A/D converters 0 and 1, TMPn can be used independently as a separate timer without being tuned. In this case, the match interrupt signal of TMPn cannot be used as a trigger source to start A/D conversion in the 6-phase PWM output mode. Therefore, fix the TQnOPT2.TQnAT00 to TQnOPT2.TQnAT03 bits and the TQnOPT3.TQnAT10 to TQnOPT3.TQnAT13 bits to 0.

The other control bits can be used in the same manner as when TMPn is tuned.

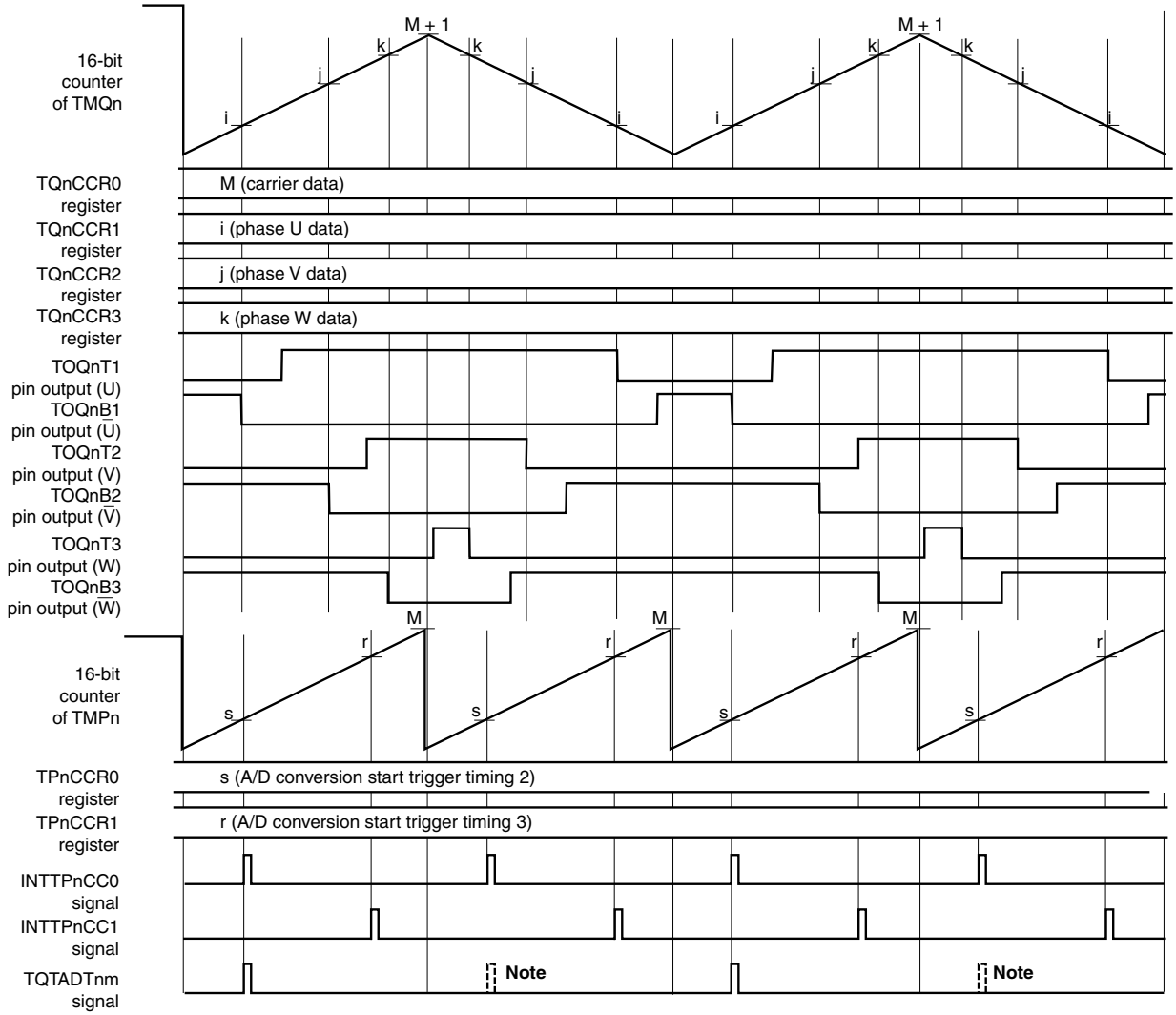
If TMPn is not tuned, the compare registers (TPnCCR0 and TPnCCR1) of TMPn are not affected by the settings of the TQnOPT0.TQnCMS and TQnOPT2.TQnRDE bits. For the initialization procedure when TMPn is not tuned, see (b) to (e) in **10.4.5 (1) Tuning operation starting procedure**. (a) is not necessary because it is a step used to set TMPn for the tuning operation.

(4) Basic operation of TMPn during tuning operation

The 16-bit counter of TMPn only counts up. The 16-bit counter is cleared by the set cycle value of the TQnCCR0 register and starts counting from 0000H again. The count value of this counter is the same as the value of the 16-bit counter of TMPn when it counts up. However, it is not the same when the 16-bit counter of TMQn counts down.

- When TMQn counts up (same value)
  16-bit counter of TMQn: 0000H → M (counting up)
  16-bit counter of TMPn: 0000H → M (counting up)
- When TMQn counts down (not same value)
  16-bit counter of TMQn: M + 1 → 0001H (counting down)
  16-bit counter of TMPn: 0000H → M (counting up)

Figure 10-37. TMPn During Tuning Operation



Note The TQTADTn0 signal is masked by the TQnOPT2.TQnATM02 and TQnOPT2.TQnATM03 bits. The TQTADTn1 signal is masked by the TQnOPT3.TQnATM12 and TQnOPT3.TQnATM13 bits.

Remark V850E/IA3: n = 0, m = 0, 1
V850E/IA4: n = 0, 1, m = 0, 1

10.4.6 A/D conversion start trigger output function

The V850E/IA3 and V850E/IA4 have a function to select four trigger sources (INTTQnOV, INTTQnCC0, INTTPnCC0, INTTPnCC1) to generate the A/D conversion start trigger signal (TQTADTn0, TQTADTn1) of A/D converters 0 and 1.

The trigger sources are specified by the TQnOPT2.TQnAT00 to TQnOPT2.TQnAT03 and TQnOPT3.TQnAT10 to TQnOPT3.TQnAT13 bits.

- TQnAT00, TQnAT10 bits = 1:
  A/D conversion start trigger signal generated when INTTQnOV (counter underflow) occurs.
- TQnAT01, TQnAT11 bits = 1:
  A/D conversion start trigger signal generated when INTTQnCC0 (cycle match) occurs.
- TQnAT02, TQnAT12 bits = 1:
  A/D conversion start trigger signal generated when INTTPnCC0 (match of TPnCCR0 register of TMPn during tuning operation) occurs.
- TQnAT03, TQnAT13 bits = 1:
  A/D conversion start trigger signal generated when INTTPnCC1 (match of TPnCCR1 register of TMPn during tuning operation) occurs.

The A/D conversion start trigger signals selected by the TQnAT00 to TQnAT03 and TQnAT10 to TQnAT13 bits are ORed and output. Therefore, two or more trigger sources can be specified at the same time.

The INTTQnOV and INTTQnCC0 signals selected by the TQnAT00, TQnAT01, TQnAT10, and TQnAT11 bits are culled interrupt signals.

Therefore, these signals are output after the interrupts have been culled and, unless interrupt output is enabled (TQnOPT1.TQnICE, TQnOPT1.TQnIOE bits), the A/D conversion start trigger is not output.

The trigger sources (INTTPnCC0 and INTTPnCC1) from TMPn have a function to mask the A/D conversion start trigger signal depending on the status of the count-up/count-down of the 16-bit counter, if so set by the TQnAT02, TQnAT03, TQnAT12, and TQnAT13 bits.

- TQnATM02, TQnATM12 bits:
  Correspond to the TQnAT02 and TQnAT12 bits and control INTTPnCC0 (match interrupt signal) of TMPn.
  - TQnATM02, TQnATM12 bits = 0
    The A/D conversion start trigger signal is output when the 16-bit counter counts up (TQnOPT0.TQnCUF bit = 0), and the A/D conversion start trigger signal is not output when the 16-bit counter counts down (TQnOPT0.TQnCUF bit = 1).
  - TQnATM02, TQnATM12 bits = 1
    The A/D conversion start trigger signal is output when the 16-bit counter counts down (TQnOPT0.TQnCUF bit = 1), and the A/D conversion start trigger signal is not output when the 16-bit counter counts up (TQnOPT0.TQnCUF bit = 0).

- TQnATM03, TQnATM13 bits:
  Correspond to the TQnAT03 and TQnAT13 bits and control INTTPnCC1 (match interrupt signal) of TMPn.
  - TQnATM03, TQnATM13 bits = 0
    The A/D conversion start trigger signal is output when the 16-bit counter counts up (TQnOPT0.TQnCUF bit = 0), and the A/D conversion start trigger signal is not output when the 16-bit counter counts down (TQnOPT0.TQnCUF bit = 1).

- TQnATM03, TQnATM13 bits = 1

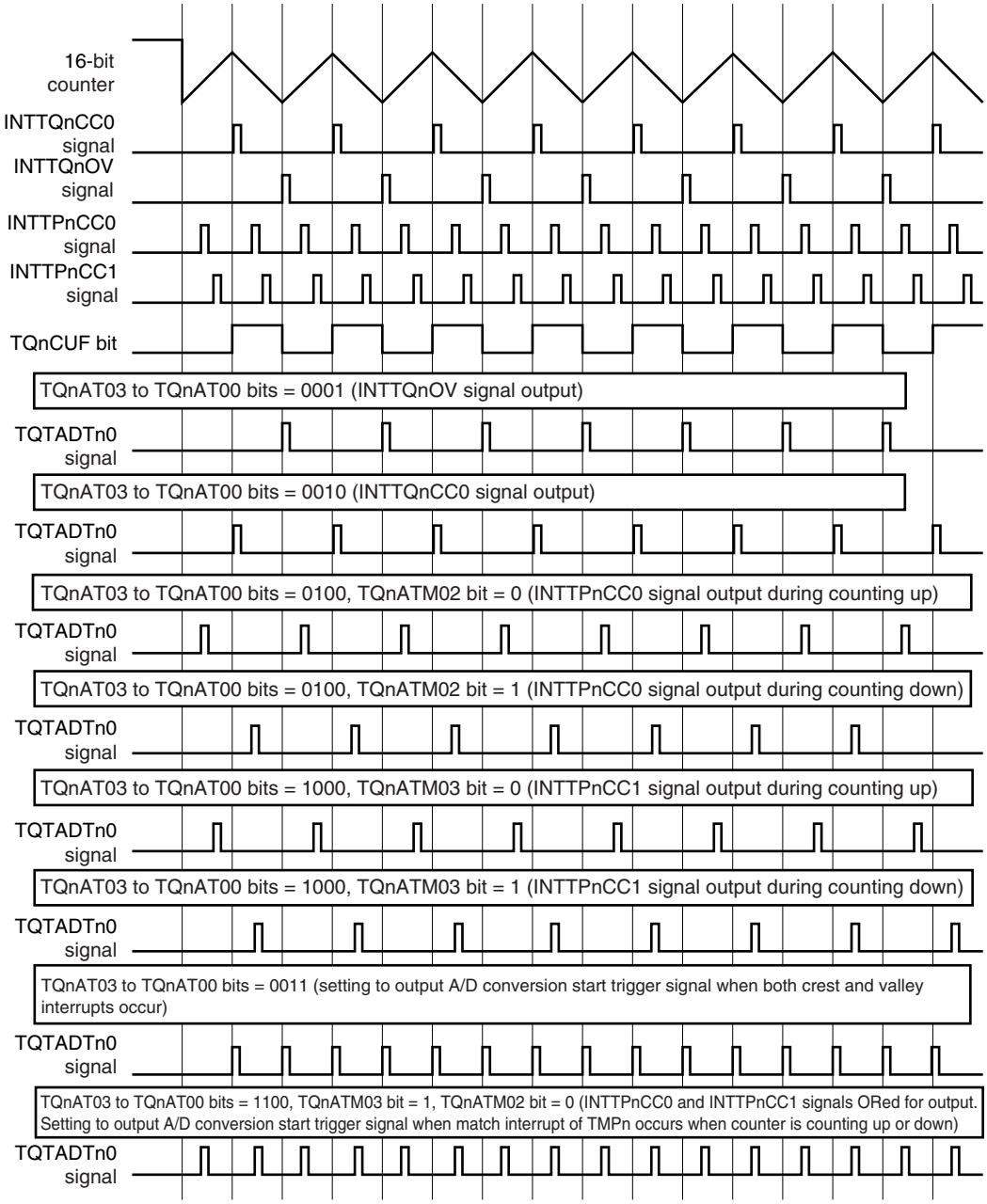
  The A/D conversion start trigger signal is output when the 16-bit counter counts down (TQnOPT0.TQnCUF bit = 1), and the A/D conversion start trigger signal is not output when the 16-bit counter counts up (TQnOPT0.TQnCUF bit = 0).

The TQnATM03, TQnATM02, TQnAT03 to TQnAT00, TQnATM13, TQnATM12, and TQnAT13 to TQnAT10 bits can be rewritten while the timer is operating. If the bit that sets the A/D conversion start trigger signal is rewritten while the timer is operating, the new setting is immediately reflected on the output status of the A/D conversion start trigger. These control bits do not have a transfer function and can be used only in the anytime rewriting mode.

Cautions 1. The A/D conversion start trigger signal output that is set by the TQnAT02, TQnAT03, TQnAT12, and TQnAT13 bits can be used only when TMPn is performing a tuning operation as the slave timer of TMQn. If TMQn and TMPn are not performing a tuning operation, or if a mode other than the 6-phase PWM output mode is used, the output cannot be guaranteed.

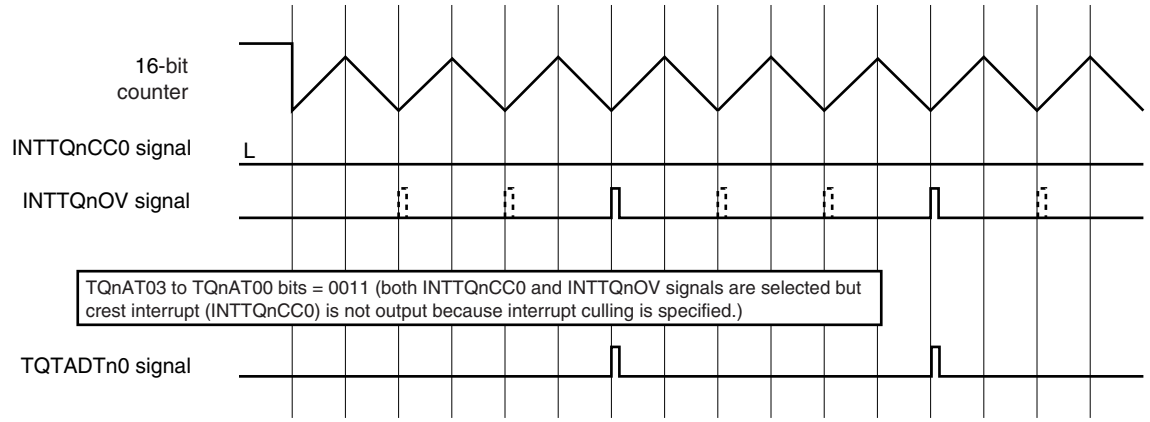
2. The TOQn0 signal output is internally used to identify whether the 16-bit counter is counting up or down. Therefore, enable TOQn0 pin output by clearing the TQnIOC0.TQnOL0 bit to 0 and setting the TQnOE0 bit to 1.

Figure 10-38. Example of A/D Conversion Start Trigger (TQTADTn0) Signal Output (TQnOPT1.TQnICE Bit = 1, TQnOPT1.TQnIOE Bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 Bits = 00000: Without Interrupt Culling)



Remark V850E/IA3: n = 0
V850E/IA4: n = 0, 1

Figure 10-39. Example of A/D Conversion Start Trigger (TQTADTn0) Signal Output (TQnOPT1.TQnICE Bit = 0, TQnOPT1.TQnIOE Bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 Bits = 00010: With Interrupt Culling) (1)

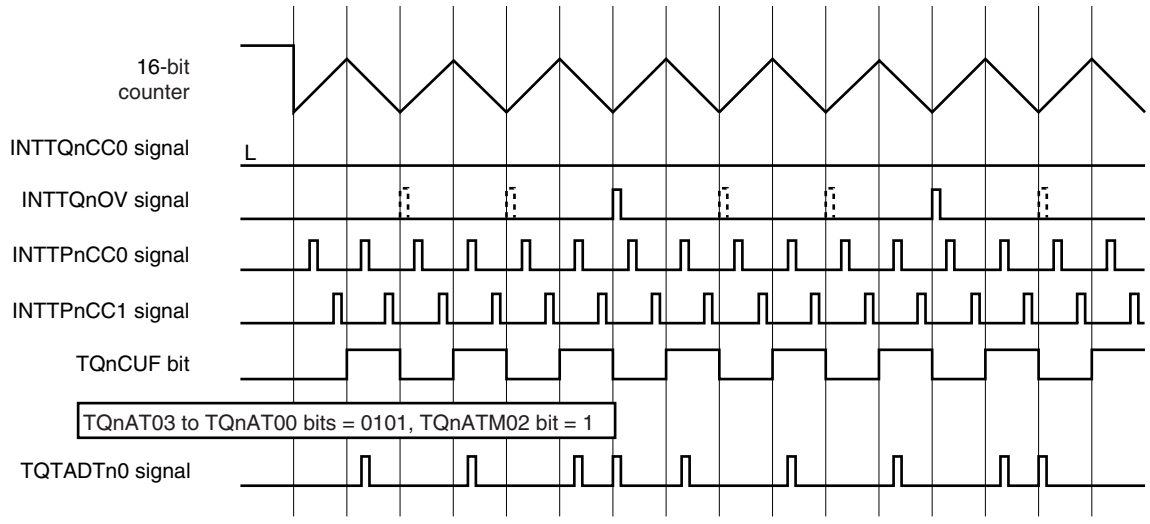


Remarks 1. ¦¯¦: Culled interrupt

2. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

Figure 10-40. Example of A/D Conversion Start Trigger (TQTADTn0) Signal Output (TQnOPT1.TQnICE Bit = 0, TQnOPT1.TQnIOE Bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 Bits = 00010: With Interrupt Culling) (2)



Caution The INTTQnCC0 signal is culled but the INTTPnCC0 signal is not.

Remarks 1. ¦¯¦: Culled interrupt

2. V850E/IA3: n = 0

V850E/IA4: n = 0, 1

(1) Operation under boundary condition (operation when 16-bit counter matches INTTPnCC0 signal)

Table 10-3. Operation When TQnCCR0 Register = M, TQnATm2 Bit = 1, TQnATMm2 Bit = 0 (Counting Up Period Selected)

Value of TPnCCR0 Register	Value of 16-bit Counter of TMQn	Value of 16-bit Counter of TMPn	Status of 16-bit Counter of TMQn	Output of INTTPnCC0 Signal from TQTADTnm Signal
0000H	0000H	0000H	–	Output
0000H	M + 1	0000H	–	Not output
0001H	0001H	0001H	Count up	Output
0001H	M	0001H	Count down	Not output
M	M	M	Count up	Output
M	0001H	M	Count down	Not output

Table 10-4. Operation When TQnCCR0 Register = M, TQnATm2 Bit = 1, TQnATMm2 Bit = 1 (Counting Down Period Selected)

Value of TPnCCR0 Register	Value of 16-bit Counter of TMQn	Value of 16-bit Counter of TMPn	Status of 16-bit Counter of TMQn	Output of INTTPnCC0 Signal from TQTADTnm Signal
0000H	0000H	0000H	–	Not output
0000H	M + 1	0000H	–	Output
0001H	0001H	0001H	Count up	Not output
0001H	M	0001H	Count down	Output
M	M	M	Count up	Not output
M	0001H	M	Count down	Output

Caution The TPnCCRm register enables setting of "0" to "M" when the TQnCCR0 register = M. Setting of a value of "M + 1" or higher is prohibited.
If a value higher than "M + 1" is set, the 16-bit counter of TMPn is cleared by "M". Therefore, the TQTADTnm signal is not output.

Remark V850E/IA3: n = 0, m = 0, 1
V850E/IA4: n = 0, 1, m = 0, 1

CHAPTER 11 WATCHDOG TIMER FUNCTIONS

11.1 Functions

The watchdog timer has the following functions.

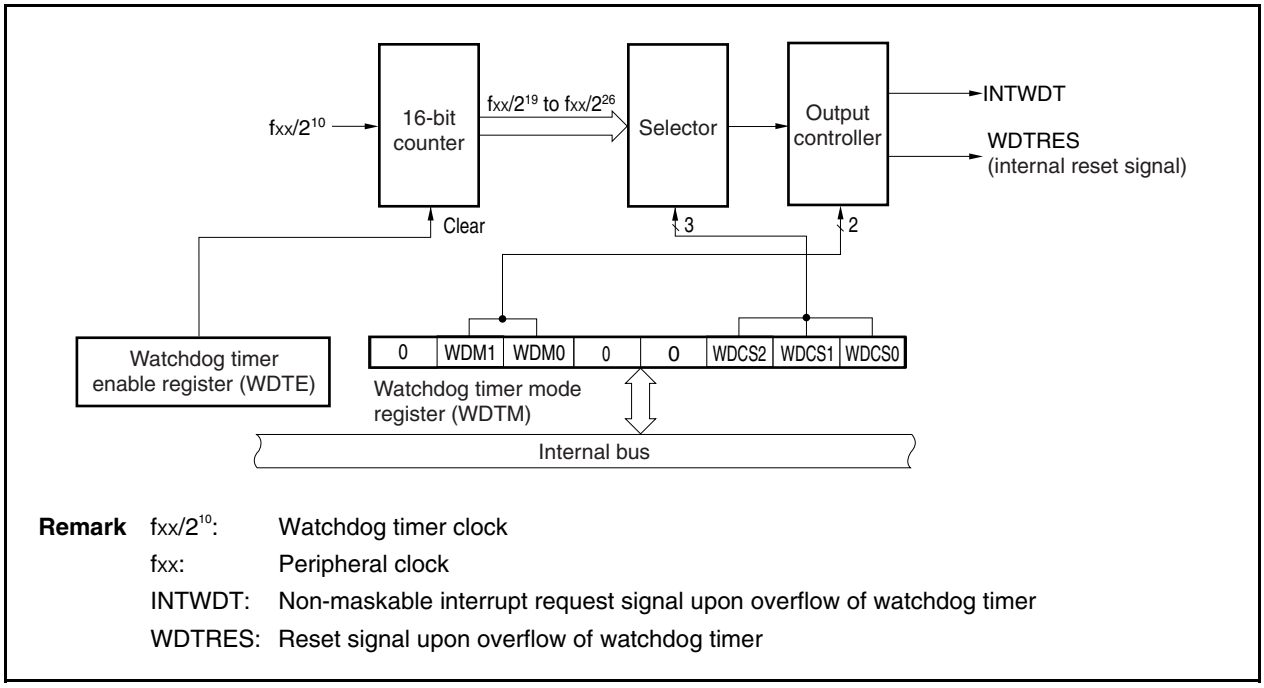
- Reset mode: Reset operation upon overflow of the watchdog timer (generation of WDTRES signal)
- Non-maskable interrupt request mode:
  Non-maskable interrupt operation upon overflow of the watchdog timer (generation of INTWDT signal)

Caution The watchdog timer is stopped after reset is released.
It starts operating when "ACH" is written to the WDTE register. Also, write to the WDTM register for verification purposes only once, even if the default settings (reset mode, interval time: 2^{26}/fxx) do not need to be changed.

11.2 Configuration

The block diagram of the watchdog timer is shown below.

Figure 11-1. Block Diagram of Watchdog Timer



Remark fxx/2^{10}: Watchdog timer clock
fxx: Peripheral clock
INTWDT: Non-maskable interrupt request signal upon overflow of watchdog timer
WDTRES: Reset signal upon overflow of watchdog timer

The watchdog timer consists of the following hardware.

Table 11-1. Configuration of Watchdog Timer

Item	Configuration
Control registers	Watchdog timer mode register (WDTM) Watchdog timer enable register (WDTE)

11.3 Control Registers

(1) Watchdog timer mode register (WDTM)

The WDTM register sets the overflow time and operation clock of the watchdog timer.

This register can be read or written in 8-bit units. This register can be read any number of times, but can be written only once following reset release; it cannot then be written a second or subsequent time.

Reset sets this register to 67H.

After reset: 67H R/W Address: FFFFF6D0H

	7	6	5	4	3	2	1	0
WDTM	0	WDM1	WDM0	0	0	WDCS2	WDCS1	WDCS0

WDM1	WDM0	Selection of operation mode of watchdog timer
0	0	Stop operation
0	1	Non-maskable interrupt request mode (generation of INTWDT signal)
1	×	Reset mode (generation of WDTRES signal)

Cautions 1. For details of the WDCS2 to WDCS0 bits, see Table 11-2 Overflow Time.

2. Be sure to clear bits 3, 4, and 7 to "0".

Table 11-2. Overflow Time

WDCS2	WDCS1	WDCS0	Overflow Time	f_{XX} = 64 MHz	f_{XX} = 32 MHz
0	0	0	$2^{19}/f_{XX}$	8.2 ms	16.4 ms
0	0	1	$2^{20}/f_{XX}$	16.4 ms	32.8 ms
0	1	0	$2^{21}/f_{XX}$	32.8 ms	65.5 ms
0	1	1	$2^{22}/f_{XX}$	65.5 ms	131.1 ms
1	0	0	$2^{23}/f_{XX}$	131.1 ms	262.1 ms
1	0	1	$2^{24}/f_{XX}$	262.1 ms	524.3 ms
1	1	0	$2^{25}/f_{XX}$	524.3 ms	1,048.5 ms
1	1	1	$2^{26}/f_{XX}$	1,048.5 ms	2,097.1 ms

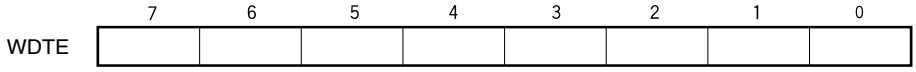
(2) Watchdog timer enable register (WDTE)

The counter of the watchdog timer is cleared and counting restarted by writing “ACH” to the WDTE register.

This register can be read or written in 8-bit units.

Reset sets this register to 1AH.

After reset: 1AH R/W Address: FFFFF6D1H



Cautions 1. If “ACH” is written to the WDTE register to enable the watchdog timer operation and then a value other than “ACH” is written to the WDTE register, a non-maskable interrupt request signal (INTWDT) or a reset signal (WDTRES) is generated due to watchdog timer overflow, depending on the specification of the WDTM.WDM1 and WDTM.WDM0 bits.

2. When the WDTE register is read or written in 1-bit units, an internal reset signal is output.

3. The read value of the WDTE register is “1AH” before the watchdog timer operates, and “9AH” after it operates. The value read from this register is different from the written value (ACH).

11.4 Operation

The watchdog timer is stopped after reset is released.

The WDTM register can be written only once after reset is released. If the register is written a second time after the watchdog timer has started operating, a non-maskable interrupt request signal (INTWDT) or a reset signal (WDTRES) is generated due to watchdog timer overflow, depending on the specification of the WDTM.WDM1 and WDTM.WDM0 bits. The INTWDT or WDTRES signal is also generated if the same value is written to the register. The operation is not guaranteed if the register is written three or more times.

To use the watchdog timer, write the operation mode and the interval time to the WDTM register in 8-bit units. After this, the operation of the watchdog timer cannot be stopped.

To not use the watchdog timer, write 00H to the WDTM register.

11.5 Caution

The cycle of the non-maskable interrupt request signal (INTWDT) that is generated due to watchdog timer overflow can be calculated from “Interval time set to WDTM register + 2^7 peripheral clock pulse width”, if INTWDT occurs successively without the watchdog timer being cleared.

Note that the pulse width until generation of the first interrupt request signal after the watchdog timer has been started is not included.

CHAPTER 12 A/D CONVERTERS 0 AND 1

12.1 Features

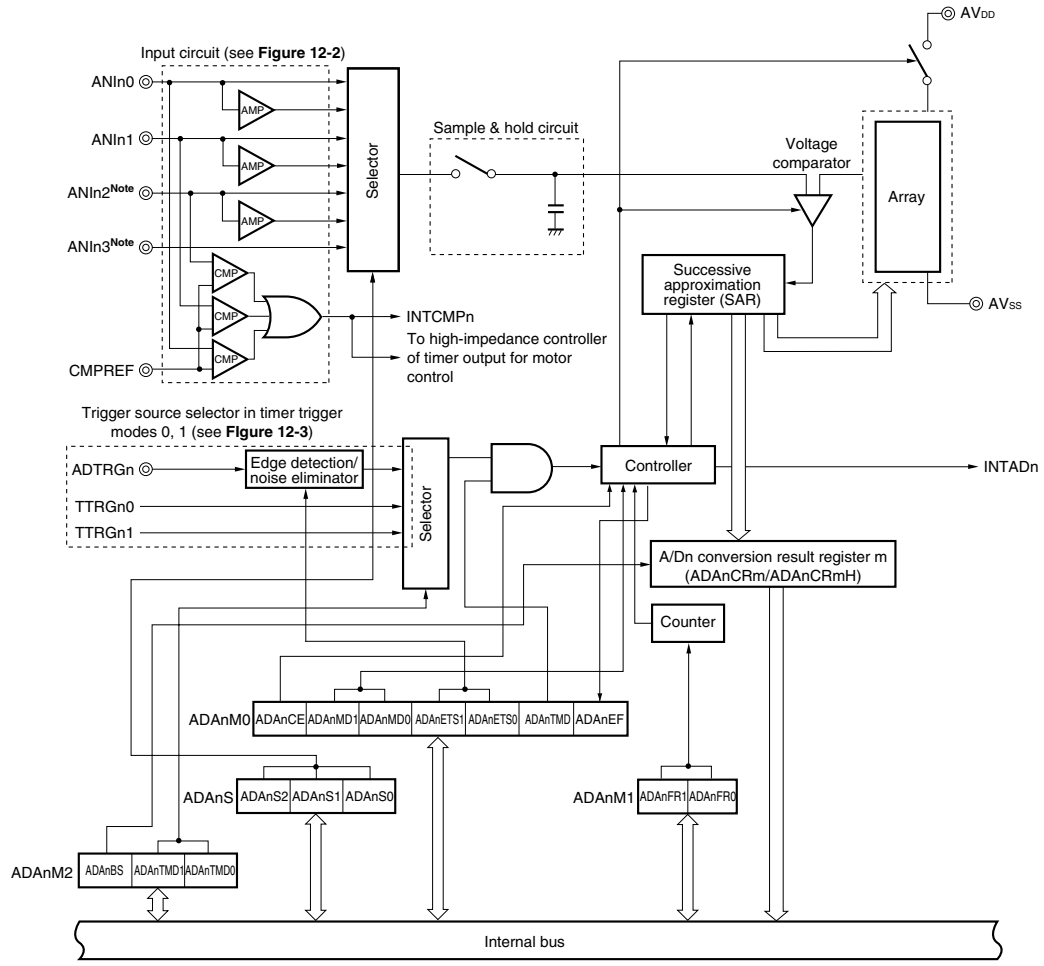
- Two 10-bit resolution A/D converter circuits (A/D converters 0 and 1)
  Simultaneous sampling of two circuits possible
- Analog input
  [V850E/IA3]
    Two circuits, total of six channels
    A/D converter 0: ANI00 and ANI01 (2 channels)
    A/D converter 1: ANI10 to ANI13 (4 channels)
  [V850E/IA4]
    Two circuits, total of eight channels
    A/D converter 0: ANI00 to ANI03 (4 channels)
    A/D converter 1: ANI10 to ANI13 (4 channels)
- A/D conversion result register
  10 bits × 4 + 10 bits × 4
  A/D converter 0: ADA0CR0 to ADA0CR3
  A/D converter 1: ADA1CR0 to ADA1CR3
- A/D conversion result registers for operational amplifier for input level amplification
  Can be used only when the operational amplifier for input level amplification is used.
  10 bits × 4 + 10 bits × 4
  A/D converter 0: ADA0CR4 to ADA0CR7
  A/D converter 1: ADA1CR4 to ADA1CR7
- A/D conversion trigger mode
  - Software trigger mode
  - Hardware trigger mode
      External trigger mode
      Timer trigger mode 0
      Timer trigger mode 1
- A/D conversion operation mode
    Continuous select mode
    Continuous scan mode
    One-shot select mode
    One-shot scan mode
- Buffer mode
    1-buffer mode
    4-buffer mode

- Operational amplifiers for input level amplification (×2.5 or ×5)
  These channels can be used only when the operational amplifier for input level amplification is used.
  [V850E/IA3]
    Two circuits, total of five channels
    A/D converter 0: ANI00 and ANI01
    A/D converter 1: ANI10 to ANI12
  [V850E/IA4]
    Two circuits, total of six channels
    A/D converter 0: ANI00 to ANI02
    A/D converter 1: ANI10 to ANI12
- Overvoltage detection comparator
  - These channels can be used only when the overvoltage detection comparator is used.
  - [V850E/IA3]
      Two circuits, total of five channels
      A/D converter 0: Two channels
      A/D converter 1: Three channels
    [V850E/IA4]
      Two circuits, total of six channels
      A/D converter 0: Three channels
      A/D converter 1: Three channels
  - Reference voltage input from CMPREF pin (input voltage range = $0.1AV_{DD}$ to $0.5AV_{DD}$)
  - An interrupt occurs when an overvoltage is detected. The interrupt request is output as a result of ORing (logical sum) the three channels.
  - The output of a timer for motor control can be set to a high-impedance state.
- Successive approximation method
- Operating voltage range
  $EV_{DD} = AV_{DD} = 4.5$ to 5.5 V

12.2 Configuration

The block diagram is shown below.

Figure 12-1. Block Diagram of A/D Converters 0 and 1



Note In the V850E/IA3, only ANI12 and ANI13 are valid.

Remark n = 0, 1

m = 0 to 3 (when using the operational amplifier for input level amplification: m = 4 to 7)

Cautions 1. If there is noise at the analog input pins (ANIn0 to ANIn3 (only ANI00 and ANI01 for A/D converter 0 of the V850E/IA3)) or at the A/D converter power supply voltage pin (AV_{DD}), that noise may generate an illegal conversion result (n = 0, 1).

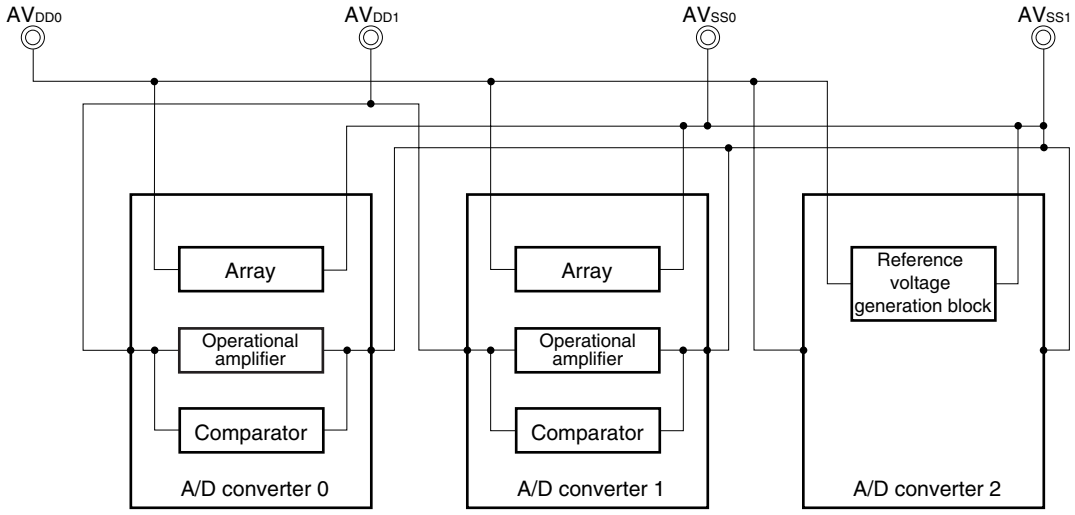
Software processing will be needed to avoid a negative effect on the system from this illegal conversion result.

An example of this software processing is shown below.

- **Take the average result of a number of A/D conversions and use that as the A/D conversion result.**
- **Execute a number of A/D conversions consecutively and use those results, omitting any exceptional results that may have been obtained.**
- **If an A/D conversion result that is judged to have generated a system malfunction is obtained, be sure to recheck the system malfunction before performing malfunction processing.**

2. Do not apply a voltage outside the AV_{SS} to AV_{DD} range to the pins that are used as input pins of A/D converters 0 and 1.

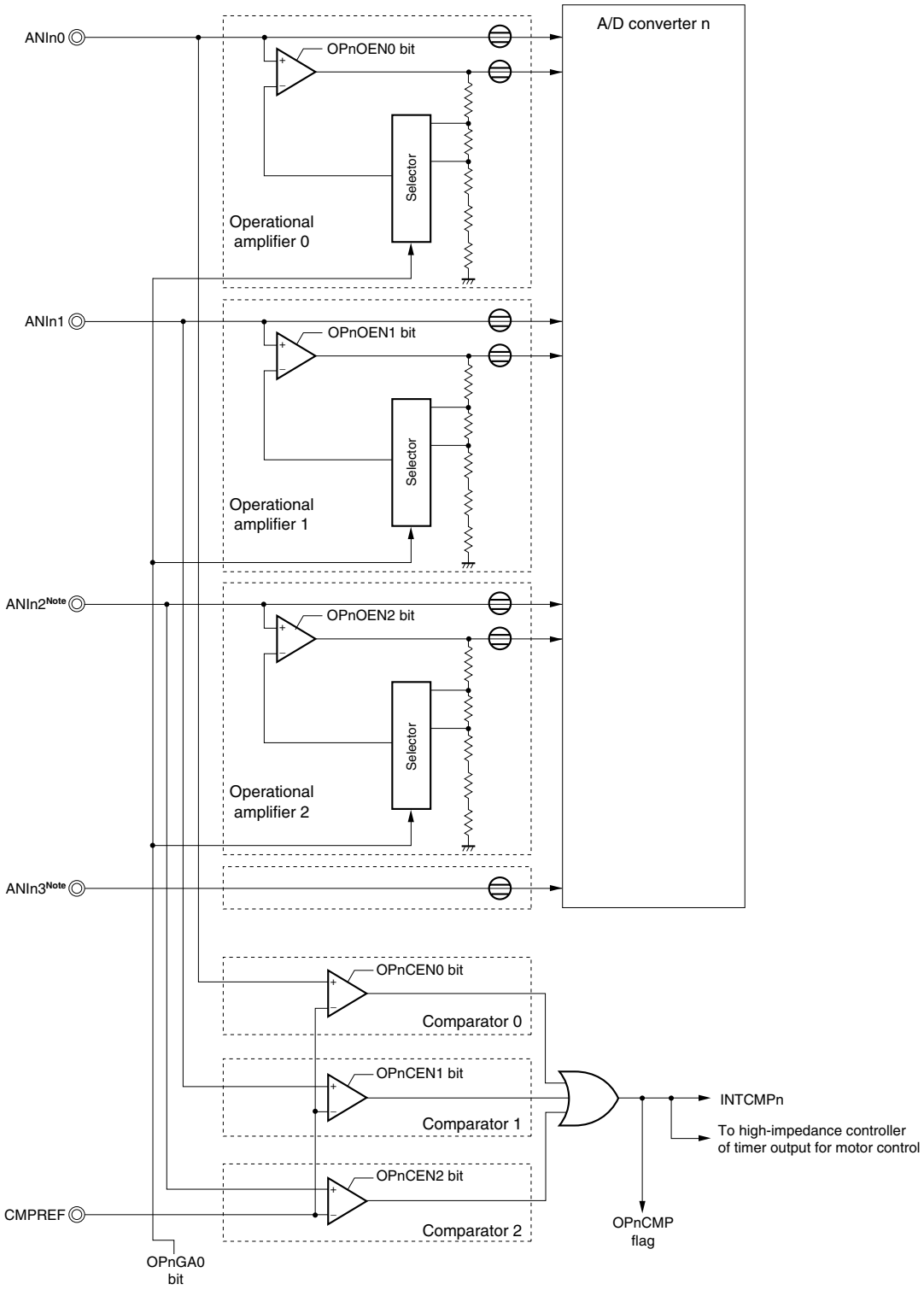
Remark The power supply connection specifications of A/D converters 0 to 2 are shown below.



Pin Name	Pin No.		
	IA3	IA4	
	GC	GC	GF
AV_{DD0}	7	9	37
AV_{DD1}	11	13	41
AV_{SS0}	8	10	38
AV_{SS1}	10	12	40

Remark IA3: V850E/IA3
IA4: V850E/IA4
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GC (V850E/IA4): 100-pin plastic QFP (14 × 20)

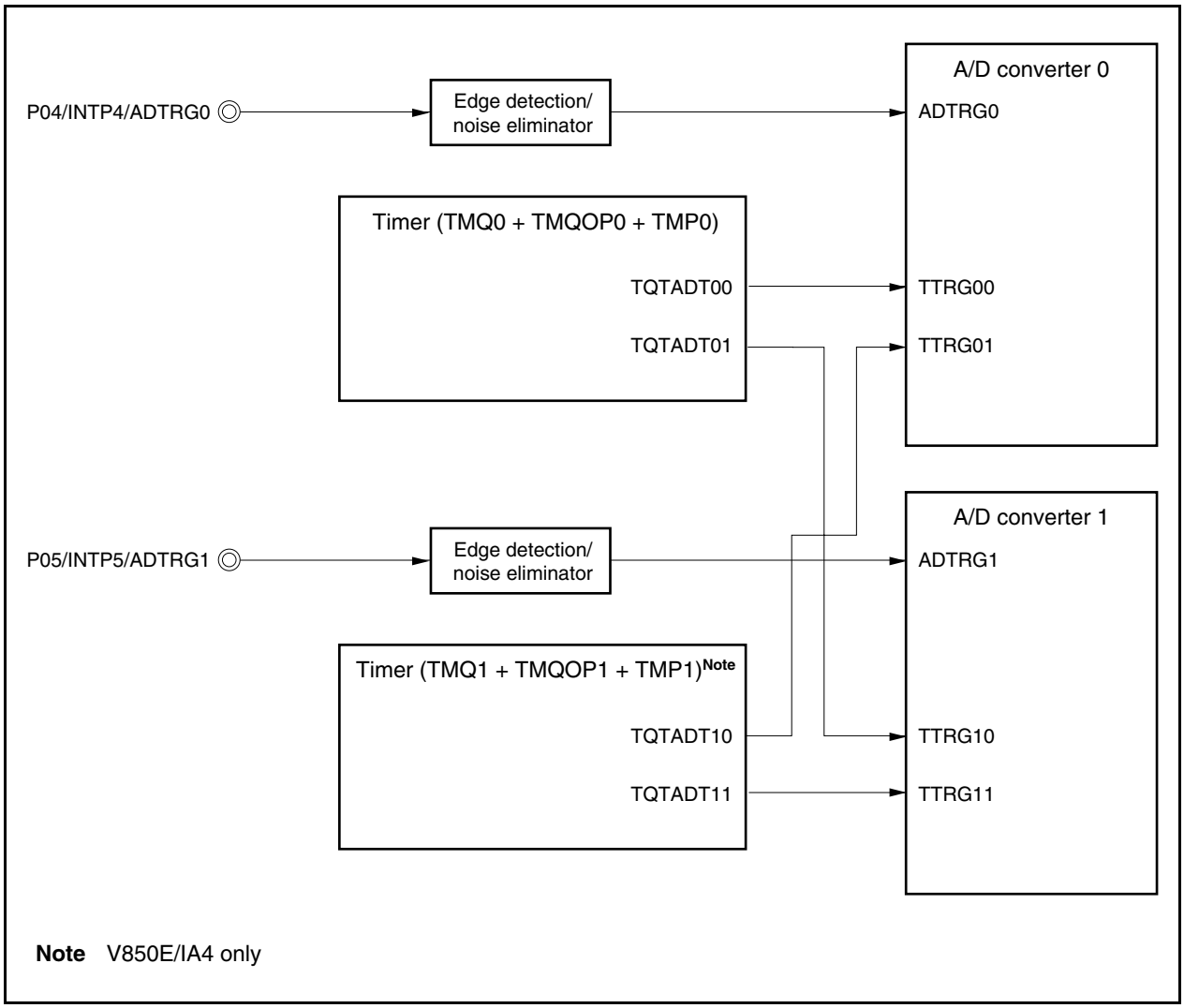
Figure 12-2. Block Diagram of Operational Amplifier for Input Level Amplification and Overvoltage Detection Comparator



Note In the V850E/IA3, only ANI12 and ANI13 are valid.

Remark n = 0, 1

Figure 12-3. Block Diagram of Trigger Source Selector in Timer Trigger Modes 0 and 1



A/D converters 0 and 1 consist of the following hardware.

Table 12-1. Configuration of A/D Converters 0 and 1

Item	Configuration
Analog input	V850E/IA3: ANI00, ANI01, ANI10 to ANI13 (two circuits, total of six channels) V850E/IA4: ANI00 to ANI03, ANI10 to ANI13 (two circuits, total of eight channels)
Registers	Successive approximation register (SAR) A/Dn conversion result registers 0 to 3 (ADAnCR0 to ADAnCR3) A/Dn conversion result registers 0H to 3H (ADAnCR0H to ADAnCR3H) A/Dn conversion result registers 4 to 7 (ADAnCR4 to ADAnCR7) (only when operational amplifier for input level amplification is used) A/Dn conversion result registers 4H to 7H (ADAnCR4H to ADAnCR7H) (only when operational amplifier for input level amplification is used)
Control registers	A/D converter n mode register 0 (ADAnM0) A/D converter n mode register 1 (ADAnM1) A/D converter n mode register 2 (ADAnM2) A/D converter n channel specification register 0 (ADAnS) Operational amplifier n control register 0 (OPnCTL0) Operational amplifier n control register 1 (OPnCTL1)

Remark n = 0, 1

(1) Selector

The input circuit selects the analog input pin (ANIn0 to ANIn3 (only ANI00 and ANI01 for A/D converter 0 of the V850E/IA3)) according to the mode set by the ADAnM0, ADAnM1, ADAnM2, and ADAnS registers and sends the input to the sample & hold circuit (n = 0, 1).

ANIn0 to ANIn2 (only ANI00 and ANI01 for A/D converter 0 of the V850E/IA3) are provided with an operational amplifier for input level amplification and an overvoltage detection comparator. The operational amplifier and comparator of each analog input pin can be specified to be on or off. The amplification (gain) of the operational amplifier can be selected from 2.5 or 5 times for ANIn0 to ANIn2 (only ANI00 and ANI01 for A/D converter 0 of the V850E/IA3).

(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 voltage comparator. When the operational amplifier for input level amplification is used, the gain specified by the OPnCTL0.OPnGA0 bit × the input voltage is sampled. This circuit also holds the sampled analog input voltage during A/D conversion.

(3) Voltage comparator

This comparator compares the voltage generated from the voltage tap of the array with the analog input voltage. If the analog input voltage is found to be greater than the reference voltage (1/2 AV_{DD}) 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 AV_{DD}), the MSB of the SAR is reset.

After that, bit 8 of the SAR is automatically set, and the next comparison is made. The voltage tap of the array is selected by the value of bit 9, to which the result has been already set.

Bit 9 = 0: (1/4 AV_{DD})

Bit 9 = 1: (3/4 AV_{DD})

The voltage tap of the array and the analog input voltage are compared and bit 8 of the SAR is manipulated according to the result of the comparison.

Analog input voltage ≥ Voltage tap of array: Bit 8 = 1
Analog input voltage ≤ Voltage tap of array: Bit 8 = 0

Comparison is continued like this to bit 0 of the SAR.

(4) Array

The array generates the comparison voltage input from an analog input pin (ANIn0 to ANIn3 (ANI00 and ANI01 only for A/D converter 0 of the V850E/IA3)) (n = 0, 1).

(5) Successive approximation register (SAR)

The SAR is a 10-bit register that sets voltage tap data whose values from the array 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 all the way to the least significant bit (LSB) (end of A/D conversion), the contents of the SAR (conversion results) are held in A/Dn conversion result registers 0 to 3 (ADAnCR0 to ADAnCR3) (n = 0, 1). If the operational amplifier for input level amplification is used, however, the conversion result is held by the ADAnCR4 to ADAnCR7 registers. When all the specified A/D conversion operations have ended, an A/Dn conversion end interrupt request signal (INTADn) is generated.

(6) A/Dn conversion result registers 0 to 3 (ADAnCR0 to ADAnCR3), A/Dn conversion result registers 0H to 3H (ADAnCR0H to ADAnCR3H) (n = 0, 1)

The ADAnCR0 to ADAnCR3 and ADAnCR0H to ADAnCR3H registers are registers that hold the A/D conversion results. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR) and stored in the higher 10 bits of the ADAnCR0 to ADAnCR3 registers. The lower 6 bits of these registers are always 0 when read.

The higher 8 bits of the result of A/D conversion are read from the ADAnCR0H to ADAnCR3H registers. To read the result of A/D conversion in 16-bit units, specify the ADAnCR0 to ADAnCR3 registers. To read the higher 8 bits, specify the ADAnCR0H to ADAnCR3H registers.

(7) A/Dn conversion result registers 4 to 7 (ADAnCR4 to ADAnCR7), A/Dn conversion result registers 4H to 7H (ADAnCR4H to ADAnCR7H) (n = 0, 1)

The ADAnCR4 to ADAnCR7 and ADAnCR4H to ADAnCR7H registers are registers that hold the A/D conversion results. These registers can be used only when the operational amplifier for input level amplification is used. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR) and stored in the higher 10 bits of the ADAnCR4 to ADAnCR7 registers. The lower 6 bits of these registers are always 0 when read.

The higher 8 bits of the result of A/D conversion are read from the ADAnCR4H to ADAnCR7H registers. To read the result of A/D conversion in 16-bit units, specify the ADAnCR4 to ADAnCR7 registers. To read the higher 8 bits, specify the ADAnCR4H to ADAnCR7H registers.

(8) A/D converter n mode register 0 (ADAnM0) (n = 0, 1)

This register is used to specify the operation mode and controls the conversion operation.

(9) A/D converter n mode register 1 (ADAnM1) (n = 0, 1)

This register is used to set the number of conversion clocks of the analog input to be A/D converted.

(10) A/D converter n channel specification register (ADAnS) (n = 0, 1)

This register is used to specify the analog input pin to be A/D converted.

(11) A/D converter n mode register 2 (ADAnM2) (n = 0, 1)

This register is used to specify the buffer mode and specify the mode in the hardware trigger mode.

(12) Operational amplifier n control register 0 (OPnCLT0) (n = 0, 1)

This register is used to control the operation of the operational amplifier for input level amplification and specify the gain.

(13) Operational amplifier n control register 1 (OPnCLT1) (n = 0, 1)

This register is used to control and indicate the status of the operation of the comparator for overvoltage detection.

(14) ANIn0 to ANIn3 pins (n = 0, 1)

The ANIn0 to ANIn3 pins (only ANI00 and ANI01 pins for A/D converter 0 of the V850E/IA3) are analog input pins for A/D converters 0 and 1. They input the analog signals to be A/D converted.

Caution Make sure that the voltages input to ANIn0 to ANIn3 do not exceed the rated values. If a voltage higher than or equal to AV_{DD} or lower than or equal to AV_{SS} (even within the range of the absolute maximum ratings) is input to a channel, the conversion value of the channel is undefined, and the conversion values of the other channels may also be affected.

(15) AV_{DD} pin

This alternates as the pin for inputting the positive power supply and reference voltage of A/D converters 0 to 2. It converts signals input to the ANIn0 to ANIn3 pins (only ANI00 and ANI01 pins for A/D converter 0 of the V850E/IA3) to digital signals based on the voltage applied between AV_{DD} and AV_{SS} (n = 0, 1).

Always make the potential at this pin the same as that at the EV_{DD} pin even when A/D converters 0 and 1 are not used.

The operating voltage range of the AV_{DD} pin is $EV_{DD} = AV_{DD} = 4.5$ to 5.5 V.

(16) AV_{SS} pin

This is the ground pin of A/D converters 0 to 2. Always make the potential at this pin the same as that at the EV_{SS} pin even when A/D converters 0 and 1 are not used.

(17) CMPREF pin

This pin supplies reference voltage to the overvoltage detection comparator (input voltage range = $0.1AV_{DD}$ to $0.5AV_{DD}$).

12.3 Control Registers

A/D converters 0 and 1 are controlled by the following registers.

- A/D converter n mode registers 0 to 2 (ADAnM0 to ADAnM2)
- A/D converter n channel specification register (ADAnS)
- Operational amplifier n control registers 0, 1 (OPnCTL0, OPnCTL1)

The following registers are also used.

- A/Dn conversion result registers 0 to 7 (ADAnCR0 to ADAnCR7)
- A/Dn conversion result registers 0H to 7H (ADAnCR0H to ADAnCR7H)

(1) A/D converter n mode register 0 (ADAnM0) (n = 0, 1)

The ADAnM0 register is an 8-bit register that specifies the operation mode and controls conversion operations. This register can be read or written in 8-bit or 1-bit units. However, bit 0 is read-only. Writing executed to bit 0 is ignored.

Reset sets this register to 00H.

After reset: 00H R/W Address: ADA0M0 FFFFF200H, ADA1M0 FFFFF220H

ADAnM0 (n = 0, 1)

<7>	6	5	4	3	2	1	0
ADAnCE	0	ADAnMD1	ADAnMD0	ADAnETS1	ADAnETS0	ADAnTMD	ADAnEF

ADAnCE	A/D conversion operation control
0	Stop conversion operation
1	Start conversion operation

ADAnMD1	ADAnMD0	Operation mode specification
0	0	Continuous select mode
0	1	Continuous scan mode
1	0	One-shot select mode
1	1	One-shot scan mode

ADAnETS1	ADAnETS0	Specification of external trigger (ADTRGn) valid edge
0	0	No edge detection (external trigger invalid)
0	1	Falling edge
1	0	Rising edge
1	1	Both rising and falling edges

ADAnTMD	Trigger mode specification
0	Software trigger mode
1	Hardware trigger mode[Note]

ADAnEF	Status of A/D converter n
0	A/D conversion stopped
1	A/D conversion operating

Note When the hardware trigger mode is selected, select a trigger mode by using the ADAnM2.ADAnTMD1 and ADAnM2.ADAnTMD0 bits.

Cautions 1. In the software trigger mode, conversion is triggered when 1 is written to the ADAnCE bit. In the hardware trigger mode (external trigger mode or timer trigger mode 0, 1), the trigger signal stands by when 1 is written to the ADAnCE bit.
The ADAnCE bit is not cleared to 0 even after the A/Dn conversion end interrupt request signal (INTADn) is generated in all modes. To stop the A/D conversion operation, therefore, write 0 to the ADAnCE bit.

2. If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion (ADAnEF bit = 1), the operation is performed as follows in each mode.

- **In software trigger mode**
  A/D conversion is stopped and executed again from the beginning.
- **In hardware trigger mode**
  A/D conversion is stopped and the trigger standby state is restored again.

(2) A/D converter n mode register 1 (ADAnM1) (n = 0, 1)

The ADAnM1 register is an 8-bit register that specifies the number of conversion clocks.

The number of conversion clocks includes the number of sampling clocks.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: ADA0M1 FFFFF201H, ADA1M1 FFFFF221H

	7	6	5	4	3	2	1	0
ADAnM1 (n = 0, 1)	0	0	0	0	0	0	ADAnFR1	ADAnFR0

Cautions 1. See Table 12-2 Number of Conversion Clocks for the ADAnFR1 and ADAnFR0 bits.

2. When ADAnM0.ADAnCE bit = 1 (conversion enabled), changing the ADAnFR1 and ADAnFR0 bits is prohibited.

3. Be sure to clear bits 2 to 7 to "0".

Table 12-2. Number of Conversion Clocks

ADAnFR1	ADAnFR0	Number of Conversion Clocks (f_{XX} = 64 MHz)	Number of Stabilization Clocks (f_{XX} = 64 MHz)	Number of Trigger Acknowledge Clocks (f_{XX} = 64 MHz)
0	0	Setting prohibited	–	–
0	1	124 (1.94 μs)	66	12
1	0	186 (2.91 μs)	100	14
1	1	248 (3.88 μs)	108	16

- Total number of A/D conversion clocks

Trigger Mode		Operation Mode		Total Number of A/D Conversion Clocks by First Trigger After Setting ADAnCE Bit = 1	Total Number of A/D Conversion Clocks by Second or Subsequent Trigger After Setting ADAnCE Bit = 1
Software trigger		Continuous select	1 buffer	Number of stabilized clocks + number of trigger acknowledgment clocks + number of conversion clocks	Number of conversion clocks
			4 buffers		
		Continuous scan	1 buffer		
		One-shot select	1 buffer		– (Conversion end with one conversion)
			4 buffers		Number of conversion clocks
		One-shot scan	1 buffer		Number of conversion clocks[Note 1]
Hardware trigger	Timer trigger 0, 1	Continuous select/ one-shot select	1 buffer	Number of trigger acknowledgment clocks + number of conversion clocks[Note 2]	– (Conversion end with one conversion)
			4 buffers		Number of conversion clocks
		Continuous scan/ one-shot scan	1 buffer		Number of conversion clocks[Note 1]
	External trigger	Continuous select/ one-shot select	1 buffer	Noise elimination time + number of trigger acknowledgment clocks + number of conversion clocks[Note 2]	– (Conversion end with one conversion)
			4 buffers		Number of conversion clocks
		Continuous scan/ one-shot scan	1 buffer		Number of conversion clocks[Note 1]

Notes 1. When two or more channels are scanned (ADAnS register ≠ 00H)

2. The stabilization time of the number of stabilized clocks elapses after the ADAnCE bit is set from 0 to 1. If the trigger is input during this time, the trigger is acknowledged after the lapse of the stabilization time. As a result, the maximum total number of A/D conversion clocks is the above stabilization time plus the number of stabilized clocks.

(3) A/D converter n channel specification register (ADAnS) (n = 0, 1)

The ADAnS register is an 8-bit register that specifies the analog input pin.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: ADA0S FFFFF202H, ADA1S FFFFF222H

	7	6	5	4	3	2	1	0
ADAnS (n = 0, 1)	0	0	0	0	0	ADAnS2	ADAnS1	ADAnS0

ADAnS2	ADAnS1	ADAnS0	Select mode	Scan mode
0	0	0	ANIn0	ANIn0
0	0	1	ANIn1	ANIn0, ANIn1
0	1	0	ANIn2Note	ANIn0 to ANIn2Note
0	1	1	ANIn3Note	ANIn0 to ANIn3Note
1	0	0	ANIn0 + operational amplifier operation	ANIn0 → (ANIn0 + operational amplifier operation)
1	0	1	ANIn1 + operational amplifier operation	ANIn0 → (ANIn1 + operational amplifier operation)
1	1	0	ANIn2 + operational amplifier operationNote	ANIn0 → (ANIn2 + operational amplifier operation)
1	1	1	Setting prohibited	Setting prohibited

Remark A/D conversion of the analog input pin is performed as follows in the scan mode.

- When ADAnS2 to ADAnS0 bits = 100
  ANIn0 → ANIn1 → ANIn2Note → ANIn3Note → ANIn0 operational amplifier operation
- When ADAnS2 to ADAnS0 bits = 101
  ANIn0 → ANIn1 → ANIn2Note → ANIn3Note → ANIn0 operational amplifier operation → ANIn1 operational amplifier operation
- When ADAnS2 to ADAnS0 bits = 110
  ANIn0 → ANIn1 → ANIn2Note → ANIn3Note → ANIn0 operational amplifier operation → ANIn1 operational amplifier operation → ANIn2 operational amplifier operationNote

Note In the V850E/IA3, these can be set only when A/D converter 1 is used. They must not be set when A/D converter 0 is used.

Cautions 1. If the ADAnS register is written during A/D conversion (ADAnM0.ADAnEF bit = 1), the operation is performed as follows in each mode.

- **In software trigger mode**
  A/D conversion is stopped and executed again from the beginning.
- **In hardware trigger mode**
  A/D conversion is stopped and the trigger standby state is restored again.

2. Be sure to clear bits 3 to 7 to "0".

(4) A/D converter n mode register 2 (ADAnM2) (n = 0, 1)

The ADAnM2 register is an 8-bit register that specifies the buffer mode and hardware trigger mode.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

(1/2)

After reset: 00H R/W Address: ADA0M2 FFFFF203H, ADA1M2 FFFFF223H

ADAnM2 (n = 0, 1)

7	6	5	4	3	2	1	0
ADAnBS	0	0	0	0	0	ADAnTMD1	ADAnTMD0

ADAnBS	Buffer mode specification
0	1-buffer mode
1	4-buffer mode[Note]

ADAnTMD1	ADAnTMD0	Hardware trigger mode specification
0	0	External trigger mode
0	1	Timer trigger mode 0
1	0	Timer trigger mode 1
1	1	Setting prohibited

Note In 4-buffer mode, the A/D conversion results are stored in the order of ADAnCR0 → ADAnCR1 → ADAnCR2 → ADAnCR3 (in 4-buffer mode when using the operational amplifier for input-level amplification: ADAnCR4 → ADAnCR5 → ADAnCR6 → ADAnCR7) regardless of the selected analog input pin.

Setting of the 4-buffer mode is prohibited in the continuous scan mode and one-shot scan mode.

Cautions 1. The external triggers of A/D converters 0 and 1 are respectively input from the P04/INTP4/ADTRG0 and P05/INTP5/ADTRG1 pins. To use the external trigger mode, therefore, be sure to set the PMC0.PMC04 and PMC0.PMC05 bits to 1.

2. The timer trigger of A/D converter n is the A/D conversion start trigger signal (TQTADTa0, TQTADTa1) of the timer (motor control function). The TQTADTa0 and TQTADTa1 signals are connected to the TTRG0a and TTRG1a signals of A/D converter n (see Figure 12-3) (V850E/IA3: a = 0, V850E/IA4: a = 0, 1).

- **Timer trigger of A/D converter 0**
  In timer trigger mode 0: TQTADT00
  In timer trigger mode 1: TQTADT10 (V850E/IA4 only)
- **Timer trigger of A/D converter 1**
  In timer trigger mode 0: TQTADT01
  In timer trigger mode 1: TQTADT11 (V850E/IA4 only)

The TQTADTa0 and TQTADTa1 signals are set by using the TQaAT00 to TQaAT03 bits of TMQa option register 2 (TQaOPT2) and the TQaAT10 to TQaAT13 bits of TMQa option register 3 (TQaOPT3). The trigger sources of the motor control function that can be selected as an A/D conversion start trigger (timer trigger) are the INTTPaCC0, INTTPaCC1, INTTQaCC0, and INTTQaOV signals (two or more signals can be selected).

(2/2)

Caution 3. If the ADAnM2 register is written during A/D conversion (ADAnM0.ADAnEF bit = 1), the operation is performed as follows in each mode.

- **In software trigger mode**
  A/D conversion is stopped and executed again from the beginning.
- **In hardware trigger mode**
  A/D conversion is stopped and the trigger standby state is restored again.

(5) Operational amplifier n control register 0 (OPnCTL0) (n = 0, 1)

The OPnCTL0 register is an 8-bit register that controls the operation of the operational amplifier for input level amplification and specifies the gain.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: OP0CTL0 FFFFF260H, OP1CTL0 FFFFF268H

	7	6	5	4	3	2	1	0
OPnCTL0	0	OPmOEN2[Note]	OPnOEN1	OPnOEN0	0	0	0	OPnGA0

V850E/IA3 (n = 0, 1, m = 1) / V850E/IA4 (n = 0, 1, m = 0, 1)

OPmOEN2[Note]	Operation control of operational amplifier 2
0	Disable operation
1	Enable operation

OPnOEN1	Operation control of operational amplifier 1
0	Disable operation
1	Enable operation

OPnOEN0	Operation control of operational amplifier 0
0	Disable operation
1	Enable operation

OPnGA0	Gain specification of operational amplifier
0	×2.5
1	×5

Note In the V850E/IA3, this bit can be set only when A/D converter 1 is used. This bit must be cleared to 0 when A/D converter 0 is used.

Cautions 1. Be sure to clear bits 1 to 3 and 7 to "0".

2. A stabilization time of 50 μs is required after operational amplifier operation is enabled.

(6) Operational amplifier n control register 1 (OPnCTL1) (n = 0, 1)

The OPnCTL1 register is an 8-bit register that controls the operation of the overvoltage detection comparator and indicates the output status of the overvoltage detection comparator.

This register can be read or written in 8-bit or 1-bit units. However, bit 0 is read-only.

Reset sets this register to 00H.

After reset: 00H R/W Address: OP0CTL1 FFFFF261H, OP1CTL1 FFFFF269H

	7	6	5	4	3	2	1	0
OPnCTL1	0	OPmCEN2[Note 1]	OPnCEN1	OPnCEN0	0	0	0	OPnCMP

V850E/IA3 (n = 0, 1, m = 1) / V850E/IA4 (n = 0, 1, m = 0, 1)

OPmCEN2[Note 1]	Operation control of comparator 2
0	Disable operation
1	Enable operation

OPnCEN1	Operation control of comparator 1
0	Disable operation
1	Enable operation

OPnCEN0	Operation control of comparator 0
0	Disable operation
1	Enable operation

OPnCMP[Note 2]	Comparator output status
0	Comparator output = 0 (no overvoltage detection)
1	Comparator output = 1 (overvoltage detection)

Notes 1. In the V850E/IA3, this bit can be set only when A/D converter 1 is used. This bit must be cleared to 0 when A/D converter 0 is used.

2. The OPnCMP bit is cleared to 0 when the input voltage is lowered to the level where overvoltage is not detected.

Cautions 1. A stabilization time of 50 μs after comparator operation is enabled.

2. The input voltage range for the comparator is $0.1AV_{DD}$ to $0.5AV_{DD}$. When the comparator is used with the input voltage range exceeding $0.5AV_{DD}$, it is recommended to use the on-chip operational amplifier. For the input voltage range of the on-chip operational amplifier, refer to CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3) and CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4).

(7) A/Dn conversion result registers 0 to 3, 0H to 3H (ADAnCR0 to ADAnCR3, ADAnCR0H to ADAnCR3H) (n = 0, 1)

The ADAnCRm and ADAnCRmH registers are registers that hold the A/D conversion results. Four of these registers are provided per circuit, and two circuits are available. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR) and stored in the higher 10 bits of the ADAnCRm register. The lower 6 bits of these registers are always 0 when read.

The higher 8 bits of A/D conversion result are read to the ADAnCRmH register.

These registers can only be read in 16-bit or 8-bit units. When the A/D conversion results are read in 16-bit units, the ADAnCRm register is specified, and when the higher 8 bits are read, the ADAnCRmH register is specified.

Reset makes these registers undefined.

After reset: Undefined R Address: ADA0CR0 FFFFF210H, ADA0CR1 FFFFF212H, ADA0CR2 FFFFF214H, ADA0CR3 FFFFF216H, ADA1CR0 FFFFF230H, ADA1CR1 FFFFF232H, ADA1CR2 FFFFF234H, ADA1CR3 FFFFF236H

ADAnCRm (n = 0, 1) (m = 0 to 3)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADnm9	ADnm8	ADnm7	ADnm6	ADnm5	ADnm4	ADnm3	ADnm2	ADnm1	ADnm0	0	0	0	0	0	0

After reset: Undefined R Address: ADA0CR0H FFFFF211H, ADA0CR1H FFFFF213H, ADA0CR2H FFFFF215H, ADA0CR3H FFFFF217H, ADA1CR0H FFFFF231H, ADA1CR1H FFFFF233H, ADA1CR2H FFFFF235H, ADA1CR3H FFFFF237H

ADAnCRmH (n = 0, 1) (m = 0 to 3)

7	6	5	4	3	2	1	0
ADnm9	ADnm8	ADnm7	ADnm6	ADnm5	ADnm4	ADnm3	ADnm2

The correspondence between the analog input pins and the ADAnCRm and ADAnCRmH registers is shown below.

Table 12-3. Correspondence Between Analog Input Pins and ADAnCRm and ADAnCRmH Registers

A/D Converter	Analog Input Pin	A/D Conversion Result Register
A/D converter 0	ANI00	ADA0CR0, ADA0CR0H
	ANI01	ADA0CR1, ADA0CR1H
	ANI02[Note]	ADA0CR2, ADA0CR2H
	ANI03[Note]	ADA0CR3, ADA0CR3H
A/D converter 1	ANI10	ADA1CR0, ADA1CR0H
	ANI11	ADA1CR1, ADA1CR1H
	ANI12	ADA1CR2, ADA1CR2H
	ANI13	ADA1CR3, ADA1CR3H

Note V850E/IA4 only

(8) A/Dn conversion result registers 4 to 7, 4H to 7H (ADAnCR4 to ADAnCR7, ADAnCR4H to ADAnCR7H) (n = 0, 1)

The ADAnCRm and ADAnCRmH registers are registers that hold the A/D conversion results. These registers can be used only when the operational amplifier for input level amplification is used. Three of these registers are provided per circuit, and two circuits are available. Each time A/D conversion ends, the conversion result is loaded from the successive approximation register (SAR) and stored in the higher 10 bits of the ADAnCRm register. The lower 6 bits of these registers are always 0 when read.

The higher 8 bits of A/D conversion result are read to the ADAnCRmH register.

These registers can only be read in 16-bit or 8-bit units. When the A/D conversion results are read in 16-bit units, the ADAnCRm register is specified, and when the higher 8 bits are read, the ADAnCRmH register is specified.

Reset makes these registers undefined.

After reset: Undefined R Address: ADA0CR4 FFFFF218H, ADA0CR5 FFFFF21AH, ADA0CR6 FFFFF21CH, ADA0CR7 FFFFF21EH, ADA1CR4 FFFFF238H, ADA1CR5 FFFFF23AH, ADA1CR6 FFFFF23CH, ADA1CR7 FFFFF23EH

ADAnCRm (n = 0, 1) (m = 4 to 7)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADn m9	ADn m8	ADn m7	ADn m6	ADn m5	ADn m4	ADn m3	ADn m2	ADn m1	ADn m0	0	0	0	0	0	0

After reset: Undefined R Address: ADA0CR4H FFFFF219H, ADA0CR5H FFFFF21BH, ADA0CR6H FFFFF21DH, ADA0CR7H FFFFF21FH, ADA1CR4H FFFFF239H, ADA1CR5H FFFFF23BH, ADA1CR6H FFFFF23DH, ADA1CR7H FFFFF23FH

ADAnCRmH (n = 0, 1) (m = 4 to 7)

7	6	5	4	3	2	1	0
ADnm9	ADnm8	ADnm7	ADnm6	ADnm5	ADnm4	ADnm3	ADnm2

The correspondence between the analog input pins and the ADAnCRm and ADAnCRmH registers is shown below.

Table 12-4. Correspondence Between Analog Input Pins and ADAnCRm and ADAnCRmH Registers (When Using Operational Amplifier for Input Level Amplification)

A/D Converter	Analog Input Pin	A/D Conversion Result Register
A/D converter 0	ANI00	ADA0CR4, ADA0CR4H
	ANI01	ADA0CR5, ADA0CR5H
	ANI02[Note]	ADA0CR6, ADA0CR6H
A/D converter 1	ANI10	ADA1CR4, ADA1CR4H
	ANI11	ADA1CR5, ADA1CR5H
	ANI12	ADA1CR6, ADA1CR6H

Note V850E/IA4 only

Caution ADAnCR7 and ADAnCR7H registers can be used only in the 4-buffer mode (ADAnM2.ADAnBS bit = 1).

The relationship between the analog voltage input to the analog input pin (ANInm) and the A/D conversion result (of A/Dn conversion result register m (ADAnCRm)) is as follows:

$$SAR = INT\left(\frac{V_{IN}}{AV_{DD}} \times 1{,}024 + 0.5\right)$$

$$ADCR^{Note} = SAR \times 64$$

or,

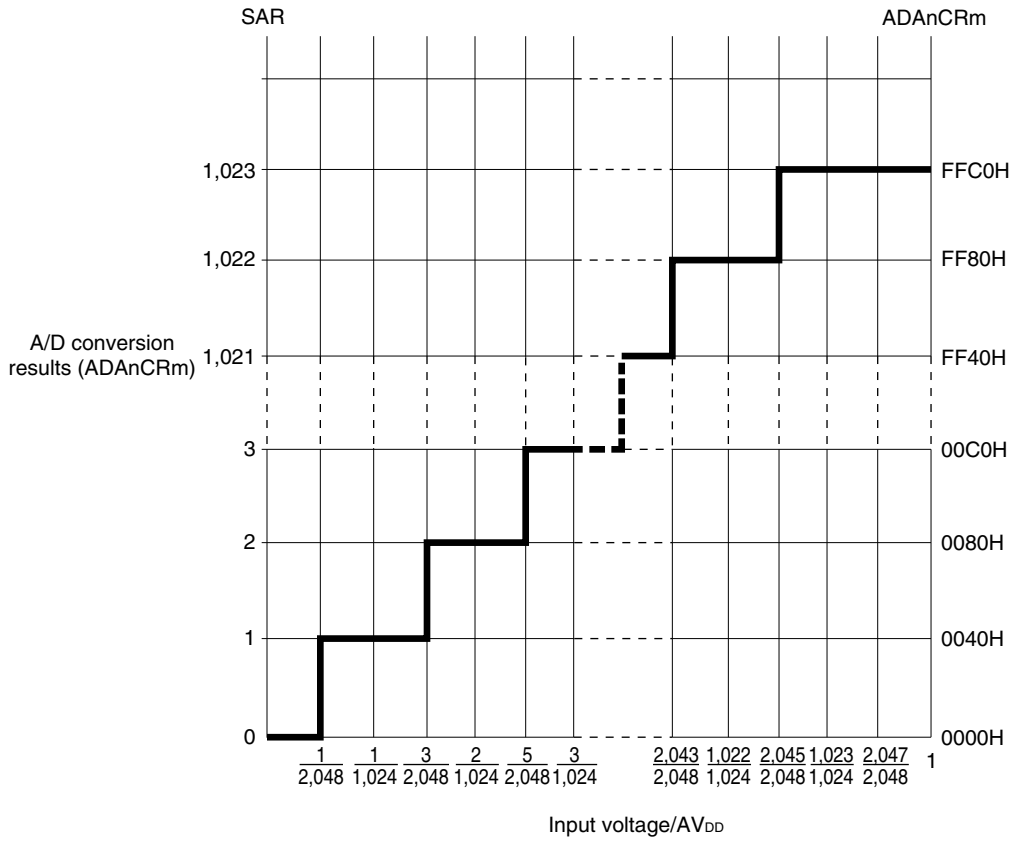
$$(SAR - 0.5) \times \frac{AV_{DD}}{1{,}024} \leq V_{IN} < (SAR + 0.5) \times \frac{AV_{DD}}{1{,}024}$$

INT( ): Function that returns the integer of the value in ( )
V_{IN}: Analog input voltage
AV_{DD}: AV_{DD} pin voltage
ADCR: Value of A/Dn conversion result register m (ADAnCRm)

Note The lower 6 bits of the ADAnCRm register are fixed to 0.

The relationship between the analog input voltage and the A/D conversion results is shown below.

Figure 12-4. Relationship Between Analog Input Voltage and A/D Conversion Results



Remark n = 0, 1
m = 0 to 3 (m = 4 to 7 when using operational amplifier for input level amplification)

12.4 Operation

Cautions 1. A/D converters 0 and 1 are capable of simultaneous sampling of two circuits.

2. For operation when using the operational amplifier for input level amplification, refer to 12.3 (3) A/D converter n channel specification register (ADAnS) (n = 0, 1).
For the relationship between the analog input pins and A/D conversion result registers, see Table 12-4.

12.4.1 Basic operation

A/D conversion is executed by the following procedure.

(1) Select an analog input pin, operation mode, and trigger mode, by using the ADAnM0, ADAnM1, ADAnM2, and ADAnS registers[Note] (n = 0, 1).
The setting of the number of stabilization clocks immediately after A/D conversion is enabled is determined by the specification of the ADAnM1.ADAnFR0 and ADAnM1.ADAnFR1 bits.

<R> **Note** If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, or if a valid trigger is input, the conversion result is not correctly stored in the ADAnCRm register and the conversion operation before the change is initialized and performed from the beginning again (m = 0 to 3).

(2) In the software trigger mode, setting the ADAnM0.ADAnCE bit to 1 starts A/D conversion after the lapse of the number of stabilization clocks (n = 0, 1). If the ADAnCE bit is set to 1 in the hardware trigger mode (external trigger mode, timer trigger modes 0, 1), the A/D converter enters the trigger wait status. For details, see **12.3 (2) A/D converter n mode register 1 (ADAnM1) (n = 0, 1)**.

(3) When A/D conversion is started, the voltage input to the selected analog input channel is sampled by the sample & hold circuit. When the operational amplifier for input level amplification is used, the gain specified by the OPnCTL0.OPnGA0 bit × the input voltage is sampled.

(4) When sampling has been performed for a specific time, the sample & hold circuit enters the hold status, and holds the input analog voltage until A/D conversion ends.

(5) Set bit 9 of the successive approximation register (SAR). The tap selector changes the level of the voltage tap of the array to the reference voltage (1/2 AV_{DD}).

(6) The voltage generated by the voltage tap of the array is compared with the analog input voltage by a comparator. If the analog input voltage is found to be greater than the reference voltage (1/2 AV_{DD}) as a result of comparison, the most significant bit (MSB) of the successive approximation register (SAR) remains set. If the analog input voltage is less than the reference voltage (1/2 AV_{DD}), the MSB of the SAR is reset.

(7) Next, bit 8 of the successive approximation register (SAR) is automatically set, and the next comparison is started. The voltage tap of the array is selected according to the value of bit 9, to which the result has been already set.

Bit 9 = 0: (1/4 AV_{DD})
Bit 9 = 1: (3/4 AV_{DD})

The voltage tap of the array and the analog input voltage are compared and bit 8 of the SAR is manipulated according to the result of the comparison.

Analog input voltage ≥ Voltage tap of array: Bit 8 = 1
Analog input voltage ≤ Voltage tap of array: Bit 8 = 0

Comparison is continued like this to bit 0 of the SAR.

(8) When comparison of 10 bits has been completed, the valid digital value result remains in the successive approximation register (SAR). This value is transferred to A/Dn conversion result register m (ADAnCRm) and the conversion result is stored in this register (n = 0, 1, m = 0 to 3). When A/D conversion has ended the specified number of times, an A/Dn conversion end interrupt request signal (INTADn) is generated.

12.4.2 Operation mode and trigger mode

Various conversion operations can be specified for the A/D converter by specifying the operation mode and trigger mode. The operation mode and trigger mode are set by the ADAnM0, ADAnM1, ADAnM2, and ADAnS registers.

The following shows the relationship between the operation mode and trigger mode.

Remark n = 0, 1

Trigger Mode		Operation Mode		Setting Value			
				ADAnM0	ADAnM1	ADAnM2	ADAnS
Software trigger		Continuous select	1 buffer	X000XX0XB	000000XXB	00000000B	00000XXXB
			4 buffers	X000XX0XB	000000XXB	10000000B	00000XXXB
		Continuous scan		X001XX0XB	000000XXB	00000000B	00000XXXB
		One-shot select	1 buffer	X010XX0XB	000000XXB	00000000B	00000XXXB
			4 buffers	X010XX0XB	000000XXB	10000000B	00000XXXB
		One-shot scan		X011XX0XB	000000XXB	00000000B	00000XXXB
Hardware trigger	External trigger	Continuous select	1 buffer[Note 1]	X000XX1XB	000000XXB	00000000B	00000XXXB
			4 buffers[Note 2]	X000XX1XB	000000XXB	10000000B	00000XXXB
		Continuous scan[Note 3]		X001XX1XB	000000XXB	00000000B	00000XXXB
		One-shot select	1 buffer[Note 1]	X010XX1XB	000000XXB	00000000B	00000XXXB
			4 buffers[Note 2]	X010XX1XB	000000XXB	10000000B	00000XXXB
		One-shot scan[Note 3]		X011XX1XB	000000XXB	00000000B	00000XXXB
	Timer trigger 0	Continuous select	1 buffer[Note 1]	X000XX1XB	000000XXB	00000001B	00000XXXB
			4 buffers[Note 2]	X000XX1XB	000000XXB	10000001B	00000XXXB
		Continuous scan[Note 3]		X001XX1XB	000000XXB	00000001B	00000XXXB
		One-shot select	1 buffer[Note 1]	X010XX1XB	000000XXB	00000001B	00000XXXB
			4 buffers[Note 2]	X010XX1XB	000000XXB	10000001B	00000XXXB
		One-shot scan[Note 3]		X011XX1XB	000000XXB	00000001B	00000XXXB
	Timer trigger 1	Continuous select	1 buffer[Note 1]	X000XX1XB	000000XXB	00000010B	00000XXXB
			4 buffers[Note 2]	X000XX1XB	000000XXB	10000010B	00000XXXB
		Continuous scan[Note 3]		X001XX1XB	000000XXB	00000010B	00000XXXB
		One-shot select	1 buffer[Note 1]	X010XX1XB	000000XXB	00000010B	00000XXXB
			4 buffers[Note 2]	X010XX1XB	000000XXB	10000010B	00000XXXB
		One-shot scan[Note 3]		X011XX1XB	000000XXB	00000010B	00000XXXB

Notes 1. The same operation is performed regardless of the trigger type.

2. The same operation is performed regardless of the trigger type.

3. The same operation is performed regardless of the trigger type.

(1) Trigger mode

There are two types of trigger modes that serve as the start timing of an A/D conversion operation: software trigger mode and hardware trigger mode. There are three types of hardware trigger modes: external trigger mode, timer trigger mode 0, and timer trigger mode 1.

These trigger modes are set by the ADAnM0 and ADAnM2 registers.

Remark n = 0, 1

(a) Software trigger mode

Of the ANIn0 to ANIn3 pins (only ANI00 and ANI01 pins for A/D converter 0 of the V850E/IA3), the analog input pin specified by the ADAnS.ADAnS2 to ADAnS.ADAnS0 bits is used for the A/D conversion start timing when the ADAnM0.ADAnCE bit is set to 1 in this mode.

After end of A/D conversion, the conversion result is stored in A/Dn conversion result register m (ADAnCRm). At the same time, an A/Dn conversion end interrupt request signal (INTADn) is generated.

If the operation mode set by the ADAnM0.ADAnMD1 and ADAnM0.ADAnMD0 bits is the continuous select mode or continuous scan mode, the conversion operation is repeated unless the ADAnM0.ADAnCE bit is cleared to 0. In the one-shot select mode or one-shot scan mode, the conversion operation is stopped after A/D conversion ends.

When conversion is started, the ADAnM0.ADAnEF bit is set to 1 (conversion in progress).

If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and executed again from the beginning.

Remark n = 0, 1
m = 0 to 3

(b) Timer trigger modes 0 and 1

Of the ANIn0 to ANIn3 pins (only ANI00 and ANI01 pins for A/D converter 0 of V850E/IA3), the analog input pin specified by the ADAnS.ADAnS2 to ADAnS.ADAnS0 bits is used for A/D conversion in this mode. The timer (motor control function) is used for the A/D conversion start timing.

The timer trigger signal of A/D converter n is the timer interrupt request signal (TQTADTa0, TQTADTa1) of the timer (motor control function). The TQTADTa0 and TQTADTa1 signals are connected to the TTRG0a and TTRG1a signals of A/D converter n (see **Figure 12-3**).

- Timer trigger of A/D converter 0
  In timer trigger mode 0: TQTADT00
  In timer trigger mode 1: TQTADT10
- Timer trigger of A/D converter 1 (V850E/IA4 only)
  In timer trigger mode 0: TQTADT01
  In timer trigger mode 1: TQTADT11

The TQTADTa0 and TQTADTa1 signals are set by using the TQaAT00 to TQaAT03 bits of TMQa option register 2 (TQaOPT2) and the TQaAT10 to TQaAT13 bits of TMQa option register 3 (TQaOPT3). The interrupt request signals of the motor control function that can be selected as a timer trigger signal are the INTTPaCC0, INTTPaCC1, INTTQaCC0, and INTTQaOV signals (two or more signals can be selected).

When the ADAnM2.ADAnTMD1 and ADAnM2.ADAnTMD0 bits are set to 01 or 10, A/D conversion is started at the rising edge of the timer interrupt request signal (TQTADTa0 or TQTADTa1) set for the motor control function.

When the ADAnM0.ADAnCE bit is set to 1, the A/D converter waits for a trigger and, when the timer interrupt request signal is input, starts A/D conversion.

After the end of A/D conversion, the conversion result is stored in A/Dn conversion result register m (ADAnCRm) and, at the same time, an A/Dn conversion end interrupt request signal (INTADn) is generated.

If the operation mode set by the ADAnM0.ADAnMD1 and ADAnM0.ADAnMD0 bits is the continuous select mode or continuous scan mode, the conversion operation is repeated, with the next timer interrupt request signal as the trigger, unless the ADAnM0.ADAnCE bit is cleared to 0. In the one-shot select mode or one-shot scan mode, the A/D converter waits for a trigger.

When conversion is started, the ADAnM0.ADAnEF bit is set to 1 (conversion in progress). While the converter waits for a trigger, however, the ADAnM0.ADAnEF bit = 0 (conversion stopped).

If the valid trigger is input during A/D conversion, the conversion is stopped and is executed again from the beginning. If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and the converter waits for a trigger again.

Caution In timer trigger modes 0 and 1, make sure that the timer interrupt request signal (A/D conversions start timing) is not generated at an interval shorter than the minimum number of conversion clocks that can be specified by the ADAnM1.ADAnFR1 and ADAnM1.ADAnFR0 bits. If the interrupt request signal is generated at an interval shorter than the minimum number of conversion clocks, the last trigger is valid.

Remark n = 0, 1
m = 0 to 3

(c) External trigger mode

Of the ANIn0 to ANIn3 pins (only ANI00 and ANI01 pins for A/D converter 0 of the V850E/IA3), the analog input pin specified by the ADAnS.ADAnS2 to ADAnS.ADAnS0 bits is used for A/D conversion in this mode (n = 0, 1). The ADTRGn pin is used for the A/D conversion start timing.

The ADTRG0 pin alternates as the P04/INTP4 pin, and the ADTRG1 pin as the P05/INTP5 pin. To set the external trigger mode, set the PMC04 and PMC05 bits of port mode control register 0 (PMC0) to 1, and the ADAnM2.ADAnTMD1 and ADAnM2.ADAnTMD0 bits to 00.

The valid edge of the external input signal in the external trigger mode can be selected from the rising edge, falling edge, or both the rising and falling edges, according to the setting of the ADAnM0.ADAnETS1 and ADAnM0.ADAnETS0 bits.

When the ADAnM0.ADAnCE bit is set to 1, the A/D converter waits for a trigger and starts conversion when the trigger is input from the ADTRGn pin.

After the end of conversion, the conversion result is stored in A/Dn conversion result register m (ADAnCRm) and, at the same time, an A/Dn conversion end interrupt request signal (INTADn) is generated (m = 0 to 3).

If the operation mode set by the ADAnM0.ADAnMD1 and ADAnM0.ADAnMD0 bits is the continuous select mode or continuous scan mode, the conversion operation is repeated, with the next ADTRGn signal as the trigger, unless the ADAnM0.ADAnCE bit is cleared to 0. In the one-shot select mode or one-shot scan mode, the A/D converter waits for a trigger.

When conversion is started, the ADAnM0.ADAnEF bit is set to 1 (conversion in progress). While the converter waits for a trigger, however, the ADAnEF bit = 0 (conversion stopped).

If the valid trigger is input during A/D conversion, the conversion is stopped and is executed again from the beginning. If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and the converter waits for a trigger again.

Caution In the external trigger mode, make sure that the ADTRGn signal (A/D conversion start timing) is not generated at an interval shorter than the minimum number of conversion clocks that can be specified by the ADAnM1.ADAnFR1 and ADAnM1.ADAnFR0 bits. If the ADTRGn signal is generated at an interval shorter than the minimum number of conversion clocks, the last trigger is valid.

Remark n = 0, 1
m = 0 to 3

(2) Operation mode

There are four operation modes in which the ANIn0 to ANIn3 pins are set (only ANI00 and ANI01 pins for A/D converter 0 of the V850E/IA3): continuous select mode, continuous scan mode, one-shot select mode, and one-shot scan mode. The continuous select mode and one-shot select mode have sub-modes that consist of 1-buffer mode and 4-buffer mode. These modes are set by the ADAnM0 and ADAnM2 registers.

Remark n = 0, 1

(a) Continuous select mode

In this mode, the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted continuously. The conversion results are stored in the A/Dn conversion result register (ADAnCRm) corresponding to the ANInm pin[Note]. In this mode, the 1-buffer mode and 4-buffer mode are provided for storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

- **1-buffer mode**

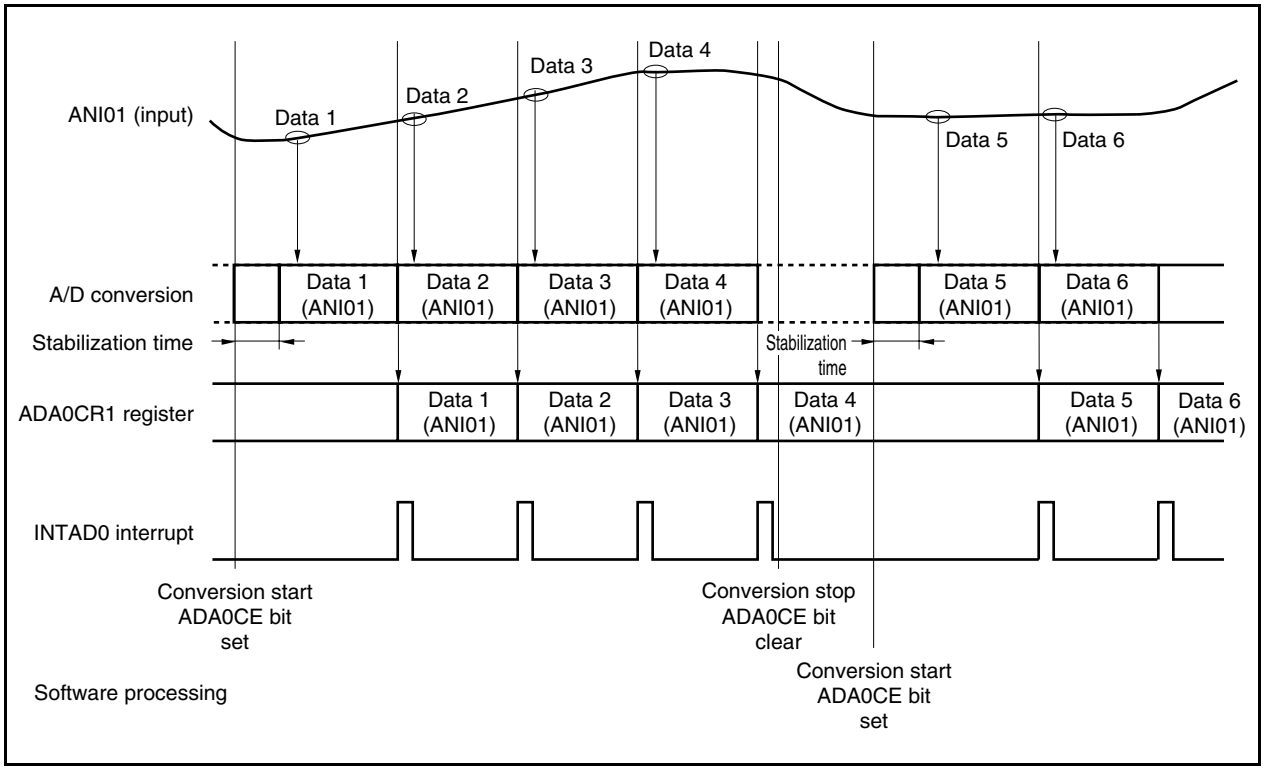
  In this mode, the voltage of the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted. The conversion results are stored in the ADAnCRm register corresponding to the ANInm pin[Note]. The ANInm pin[Note] and the ADAnCRm register correspond one to one, and an A/Dn conversion end interrupt request signal (INTADn) is generated each time one A/D conversion ends.

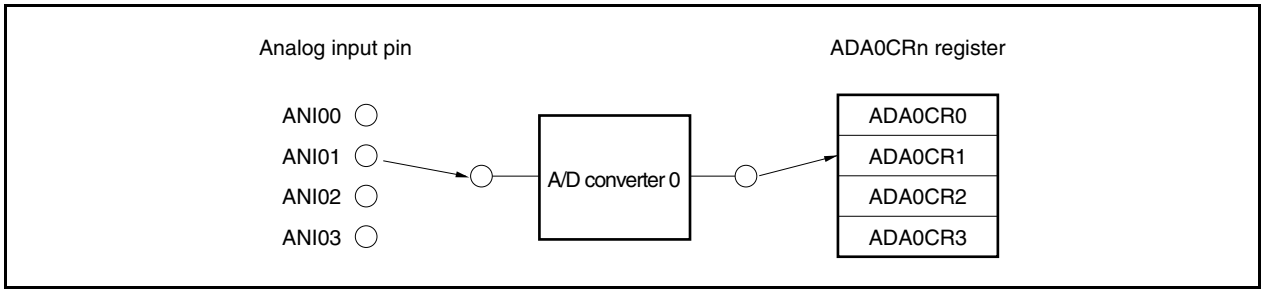
  After the end of A/D conversion, the conversion is repeated again unless the ADAnM0.ADAnCE bit is cleared to 0.

  Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

  Remark n = 0, 1
  m = 0 to 3

Figure 12-5. Continuous Select 1-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bits = 00, ADA0M2.ADA0BS Bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 001): V850E/IA4





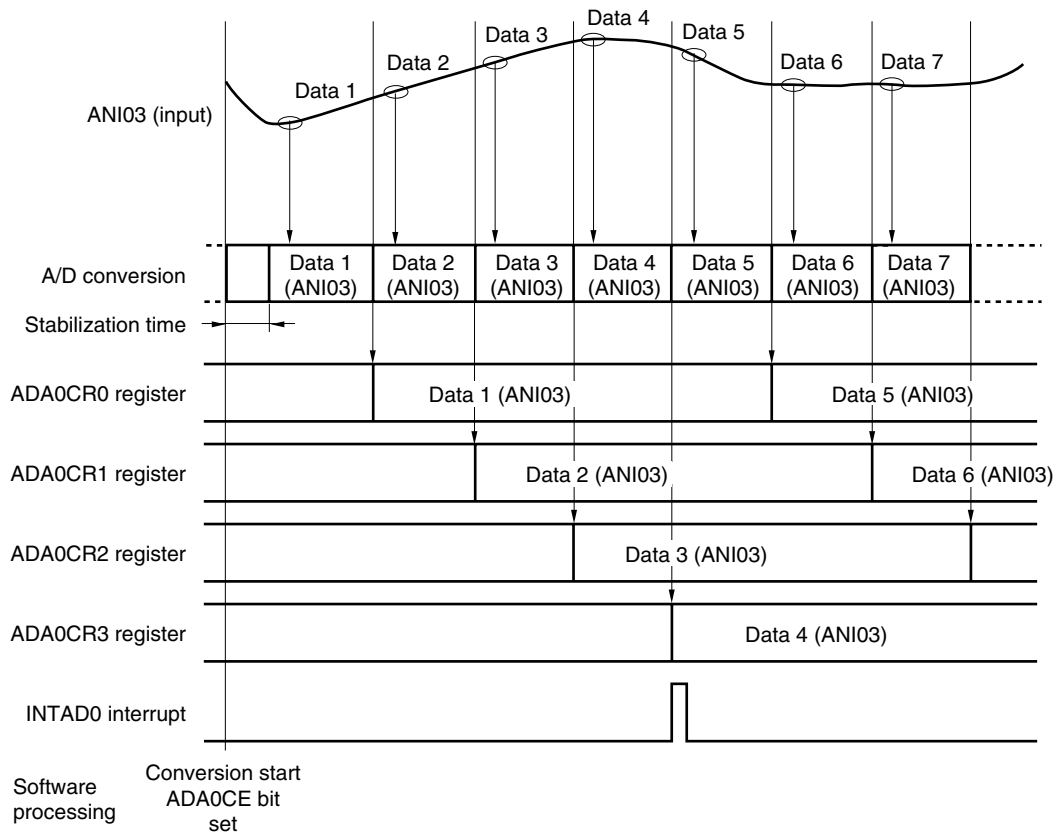
- **4-buffer mode**

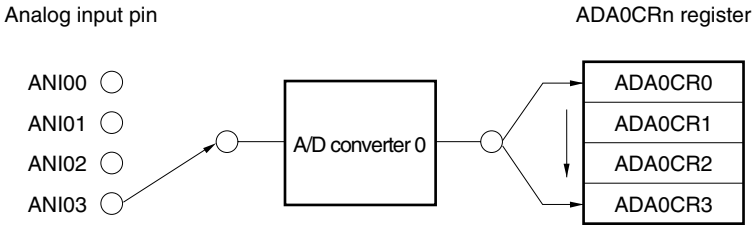
  In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted four times and the results are stored in the ADAnCRm register. The A/Dn conversion end interrupt request signal (INTADn) is generated when the four A/D conversions end. After end of A/D conversion, the conversion is started again from the beginning, unless the ADAnM0.ADAnCE bit is cleared to 0.

  Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

  Remark n = 0, 1, m = 0 to 3

Figure 12-6. Continuous Select 4-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bits = 00, ADA0M2.ADA0BS Bit = 1, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 011): V850E/IA4





Caution The A/D conversion results are stored in the order of ADA0CR0 → ADA0CR1 → ADA0CR2 → ADA0CR3 regardless of the selected analog input pin.

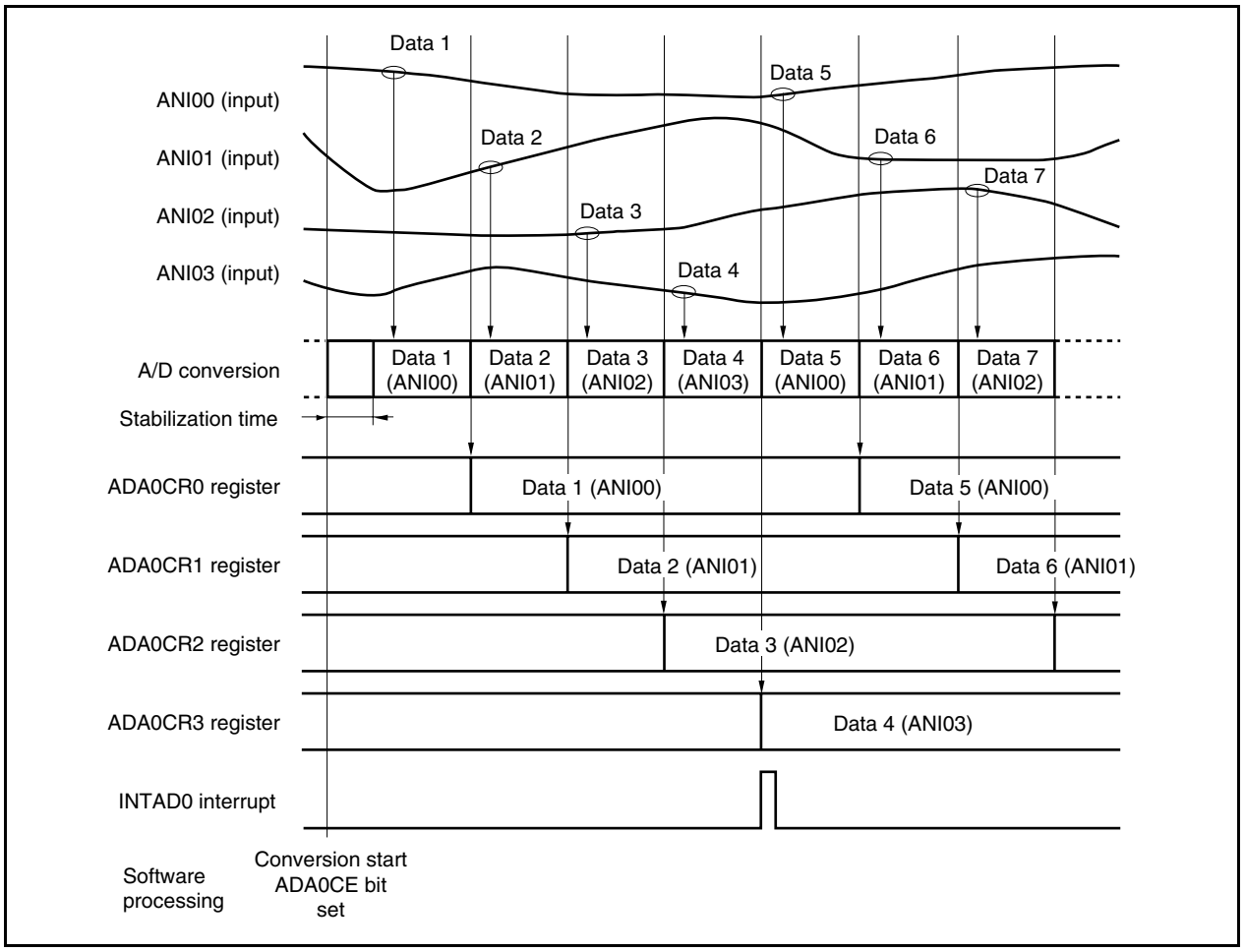
(b) Continuous scan mode

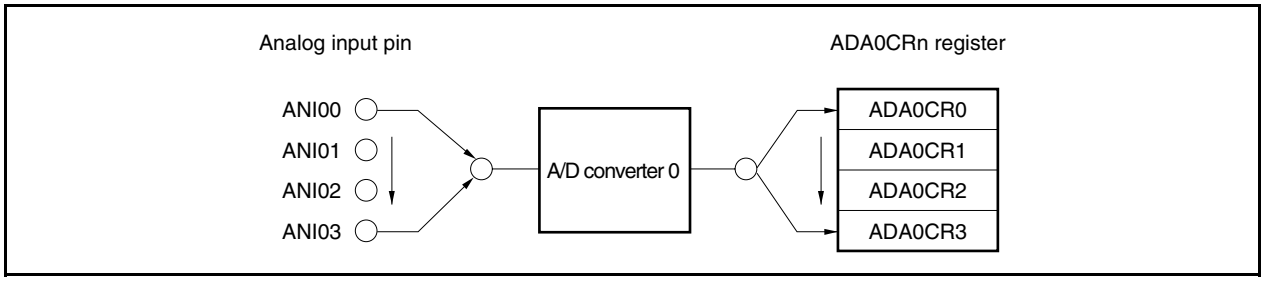
In this mode, the analog input pins (ANInm)[Note] specified by the ADAnS register are selected sequentially from the ANIn0 pin, and A/D conversion is executed continuously. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin[Note]. When conversion of all the specified analog input pin[Note] ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the conversion is started again from the ANIn0 pin, unless the ADAnM0.ADAnCE bit is cleared to 0.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1, m = 0 to 3

Figure 12-7. Continuous Scan Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bits = 01, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 011): V850E/IA4





(c) One-shot select mode

In this mode, the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted once. The conversion results are stored in the A/Dn conversion result register (ADAnCRm) corresponding to the ANInm pin[Note]. In this mode, the 1-buffer mode and 4-buffer mode are provided for storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

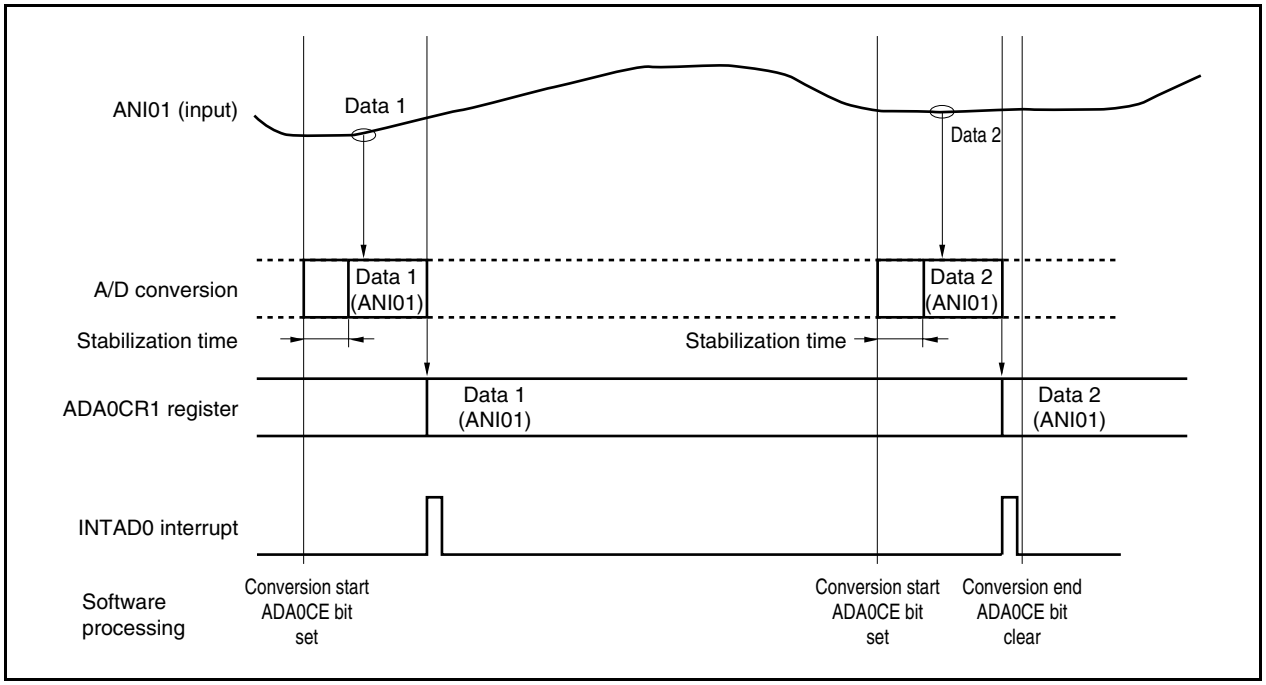
- **1-buffer mode**

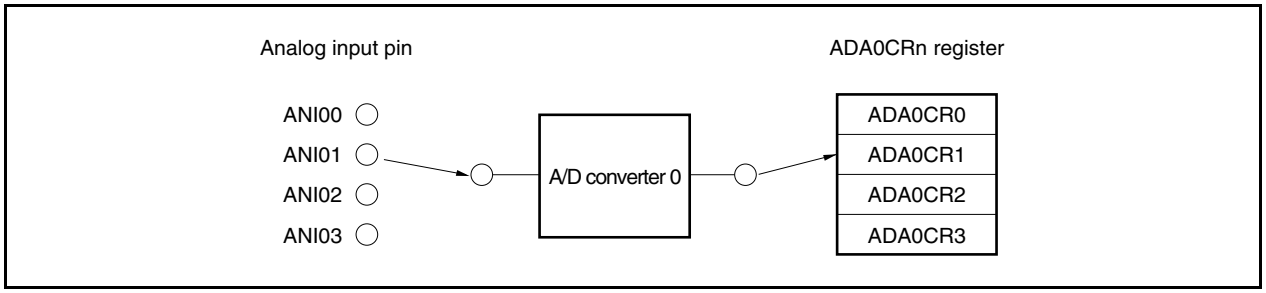
  In this mode, the voltage of the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted. The conversion results are stored in the ADAnCRm register corresponding to the ANInm pin[Note]. The ANInm pin[Note] and the ADAnCRm register correspond one to one, and an A/Dn conversion end interrupt request signal (INTADn) is generated each time one A/D conversion ends.
  After the end of A/D conversion, the conversion operation is stopped.

  Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

  Remark n = 0, 1, m = 0 to 3

Figure 12-8. One-Shot Select 1-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bits = 10, ADA0M2.ADA0BS Bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 001): V850E/IA4





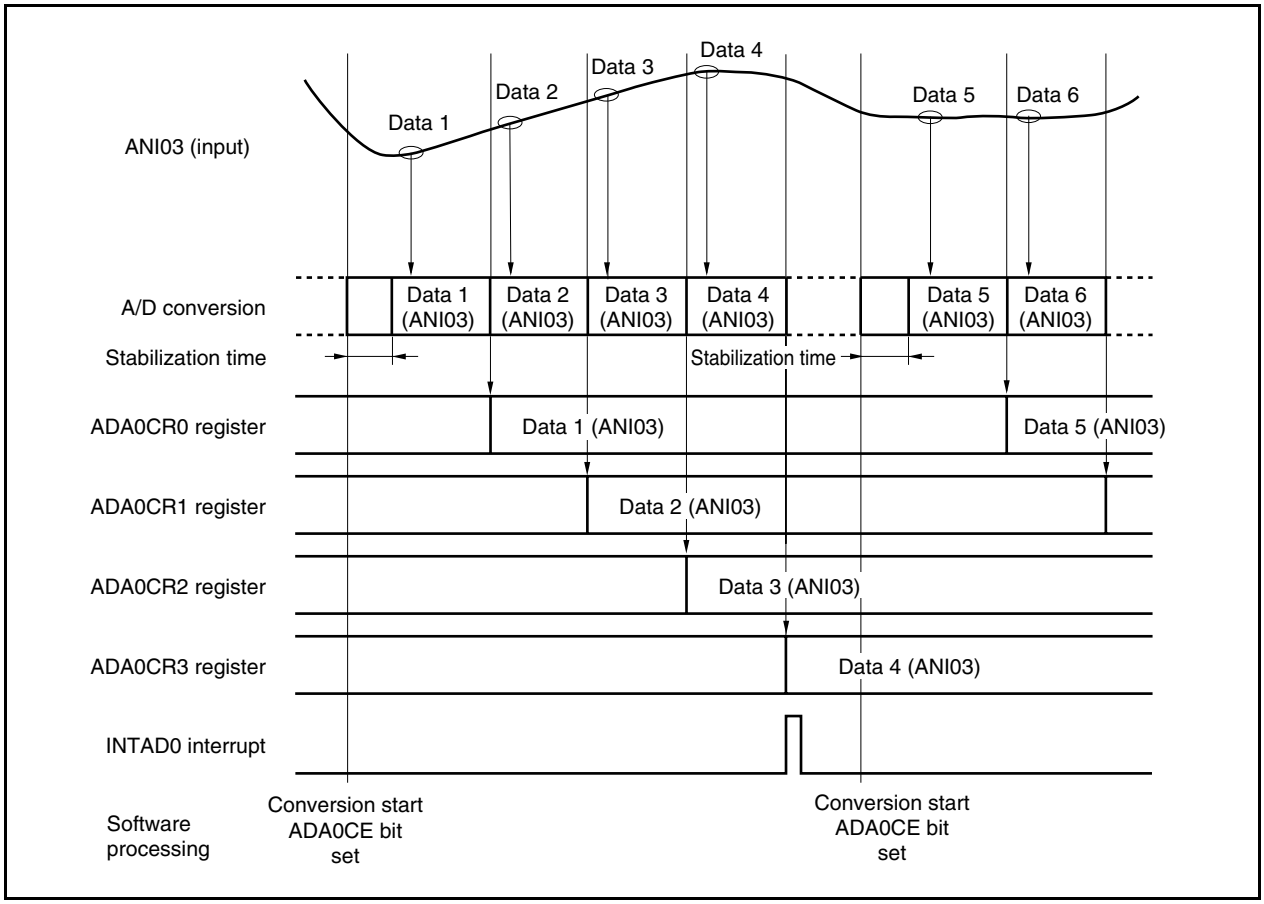
- **4-buffer mode**

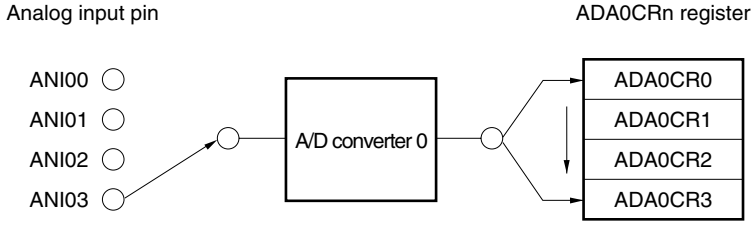
  In this mode the voltage of one analog input pin (ANInm)[Note] is A/D converted four times and the results are stored in the ADAnCRm register. The A/Dn conversion end interrupt request signal (INTADn) is generated when the four A/D conversions end. After end of A/D conversion, the conversion operation is stopped.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1, m = 0 to 3

Figure 12-9. One-Shot Select 4-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bit = 10, A0M2.ADA0BS Bit = 1, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 011): V850E/IA4





Caution The A/D conversion results are stored in the order of ADA0CR0 → ADA0CR1 → ADA0CR2 → ADA0CR3 regardless of the selected analog input pin.

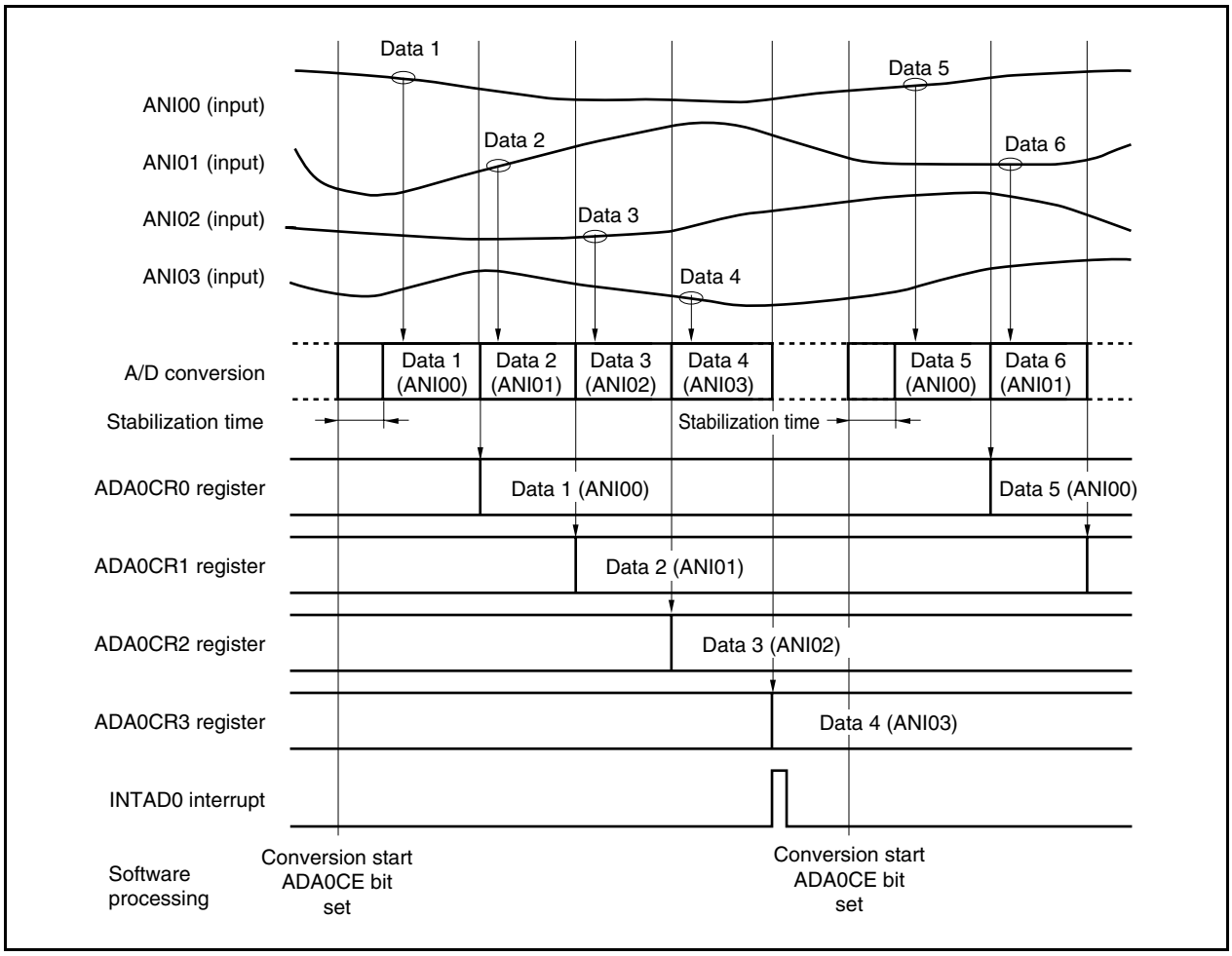
(d) One-shot scan mode

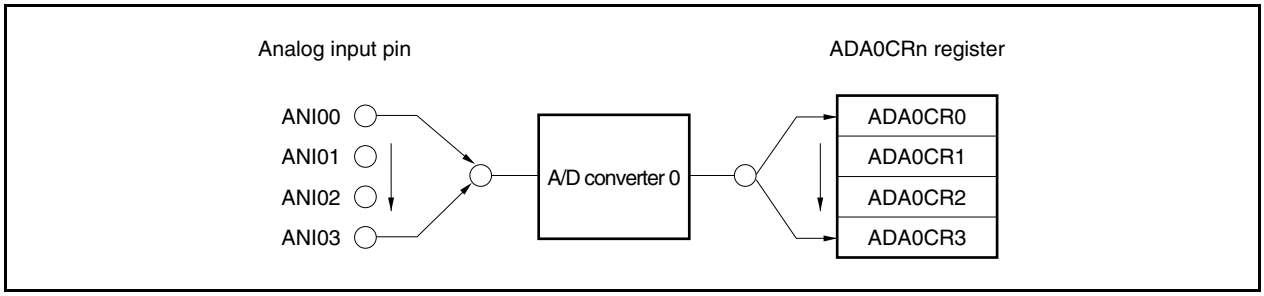
In this mode, the analog input pins (ANInm)[Note] specified by the ADAnS register are selected sequentially from the ANIn0 pin, and A/D conversion is executed. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin[Note]. When conversion of all the specified analog input pin[Note] ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After end of A/D conversion, the conversion operation is stopped.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-10. One-Shot Scan Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 Bits = 11, ADA0S.ADA0S2 to ADA0S.ADA0S0 Bits = 011): V850E/IA4





12.5 Operation in Software Trigger Mode

When the ADAnM0.ADAnCE bit is set (1), A/D conversion is started.

When A/D conversion is started, the ADAnM0.ADAnEF bit = 1 (conversion in progress).

If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and executed again from the beginning.

Remark n = 0, 1

12.5.1 Continuous select mode operations

In this mode, the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted continuously. The conversion results are stored in the ADAnCRm register. In the continuous select mode, the 1-buffer mode and 4-buffer mode are supported according to the method of storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

(1) 1-buffer mode (1 buffer of software trigger continuous select)

In this mode, the voltage of one analog input pin (ANInm)[Note 1] is A/D converted once. The conversion results are stored in one ADAnCRm register. The ANInm pin[Note 1] and ADAnCRm register correspond one to one.

Each time an A/D conversion is executed, an A/Dn conversion end interrupt request signal (INTADn) is generated and A/D conversion ends. After the end of A/D conversion, the conversion is repeated again unless the ADAnM0.ADAnCE bit is cleared to 0.

It is not necessary to set (1) the ADAnM0.ADAnCE bit to restart A/D conversion[Note 2].

Notes 1. Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

2. In the software trigger continuous select 1-buffer mode, the A/D conversion operation is not stopped unless the ADAnM0.ADAnCE bit is cleared to 0. If the ADAnCRm register is not read before the next A/D conversion ends, it is overwritten.

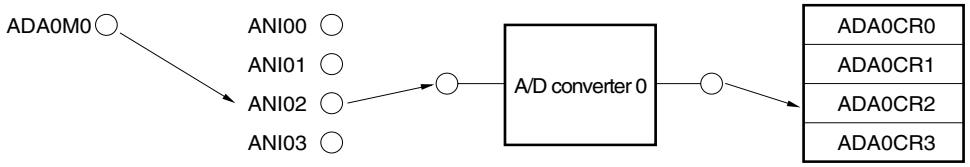
This mode is suitable for applications in which the A/D conversion value of one analog input pin is read.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCRm

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-11. Example of 1-Buffer Mode Operation (1 Buffer of Software Trigger Continuous Select): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI02 pin is A/D converted
(3) The conversion result is stored in the ADA0CR2 register
(4) The ADA0M0.ADA0EF bit = 0
(5) The INTAD0 interrupt request signal is generated
(6) Return to (2)
(7) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

(2) 4-buffer mode (4 buffers of software trigger continuous select)

In this mode, the voltage of one analog input pin (ANInm)[Note 1] is A/D converted four times and the results are stored in the ADAnCRm register.

When the 4th A/D conversion ends, an A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the conversion is started again from the beginning, unless the ADAnM0.ADAnCE bit is cleared to 0. It is not necessary to set (1) the ADAnM0.ADAnCE bit to restart A/D conversion[Note 2].

Notes 1. Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

2. In the software trigger continuous select 4-buffer mode, the A/D conversion operation is not stopped unless the ADAnM0.ADAnCE bit is cleared to 0. If the ADAnCRm register is not read before the next A/D conversion ends, it is overwritten.

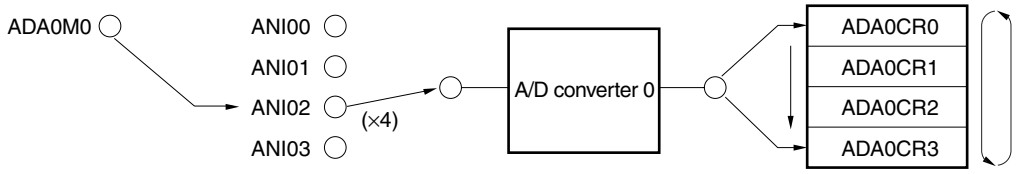
This mode is suitable for applications in which the average of the A/D conversion results of one analog input pin is calculated.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCR0
ANInm[Note]	ADAnCR1
ANInm[Note]	ADAnCR2
ANInm[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-12. Example of 4-Buffer Mode Operation (4 Buffers of Software Trigger Continuous Select): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI02 pin is A/D converted
(3) The conversion result is stored in the ADA0CR0 register
(4) The ANI02 pin is A/D converted
(5) The conversion result is stored in the ADA0CR1 register
(6) The ANI02 pin is A/D converted
(7) The conversion result is stored in the ADA0CR2 register
(8) The ANI02 pin is A/D converted
(9) The conversion result is stored in the ADA0CR3 register
(10) The INTAD0 interrupt request signal is generated
(11) Return to (2)
(12) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 1, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.5.2 Continuous scan mode operations

In this mode, the analog input pins (ANInm)[Note 1] specified by the ADAnS register are selected sequentially from the ANIn0 pin, and A/D conversion is executed continuously. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin.

When conversion of all the specified analog input pins ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the conversion is started again from the ANIn0 pin, unless the ADAnM0.ADAnCE bit is cleared to 0. It is not necessary to set (1) the ADAnM0.ADAnCE bit to restart A/D conversion[Note 2].

In the continuous scan mode, only the 1-buffer mode is supported.

Notes 1. Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

2. In the software trigger continuous scan mode, the A/D conversion operation is not stopped unless the ADAnM0.ADAnCE bit is cleared to 0. If the ADAnCRm register is not read before the next A/D conversion ends, it is overwritten.

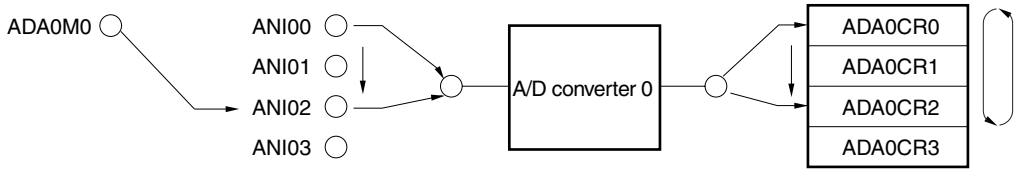
This mode is suitable for applications in which multiple analog inputs are constantly monitored.

Analog Input Pin	A/D Conversion Result Register
ANIn0	ADAnCR0
⋮	⋮
ANInm[Note]	ADAnCRm

Note Set by the ADAnS.ADAnS0 to ADAnS.ADAnS2 bits.
Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-13. Example of Continuous Scan Mode Operation (Software Trigger Continuous Scan): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI00 pin is A/D converted
(3) The conversion result is stored in the ADA0CR0 register
(4) The ANI01 pin is A/D converted
(5) The conversion result is stored in the ADA0CR1 register
(6) The ANI02 pin is A/D converted
(7) The conversion result is stored in the ADA0CR2 register
(8) The INTAD0 interrupt request signal is generated
(9) Return to (2)
(10) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 01, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.5.3 One-shot select mode operations

In this mode, the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted continuously. The conversion results are stored in the ADAnCRm register. In the one-shot select mode, the 1-buffer mode and 4-buffer mode are supported according to the method of storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

(1) 1-buffer mode (1 buffer of software trigger one-shot select)

In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted once. The conversion results are stored in one ADAnCRm register. The ANInm pin[Note] and the ADAnCRm register correspond one to one.
Each time an A/D conversion is executed, an A/Dn conversion end interrupt request signal (INTADn) is generated and A/D conversion ends. After the end of A/D conversion, the conversion operation is stopped.
If the ADAnM0.ADAnCE bit is set (1), A/D conversion can be restarted.
This mode is suitable for applications in which the results of each first-time A/D conversion are read.

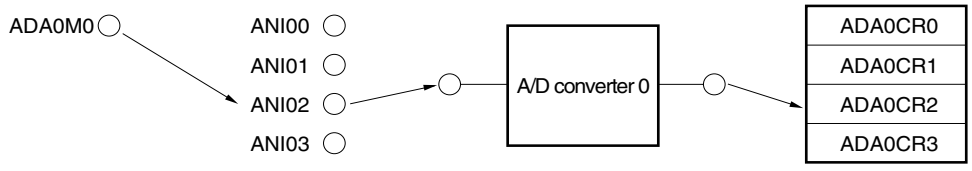
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCRm

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-14. Example of 1-Buffer Mode Operation (1 Buffer of Software Trigger One-Shot Select): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI02 pin is A/D converted
(3) The conversion result is stored in the ADA0CR2 register
(4) The ADA0M0.ADA0EF bit = 0
(5) The INTAD0 interrupt request signal is generated

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 10, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

(2) 4-buffer mode (4 buffers of software trigger one-shot select)

In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted four times and the results are stored in the ADAnCRm register.

When the 4th A/D conversion ends, an A/Dn conversion end interrupt request signal (INTADn) is generated.

After the end of A/D conversion, the conversion operation is stopped.

If the ADAnM0.ADAnCE bit is set (1), A/D conversion can be restarted.

This mode is suitable for applications in which the average of the A/D conversion results is calculated.

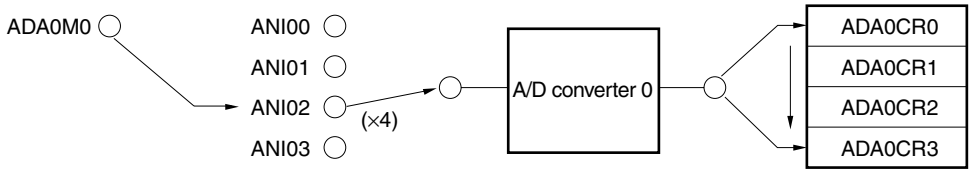
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCR0
ANInm[Note]	ADAnCR1
ANInm[Note]	ADAnCR2
ANInm[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-15. Example of 4-Buffer Mode Operation (4 Buffers of Software Trigger One-Shot Select): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI02 pin is A/D converted
(3) The conversion result is stored in the ADA0CR0 register
(4) The ANI02 pin is A/D converted
(5) The conversion result is stored in the ADA0CR1 register
(6) The ANI02 pin is A/D converted
(7) The conversion result is stored in the ADA0CR2 register
(8) The ANI02 pin is A/D converted
(9) The conversion result is stored in the ADA0CR3 register
(10) The ADA0M0.ADA0EF bit = 0
(11) The INTAD0 interrupt request signal is generated

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 10, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 1, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.5.4 One-shot scan mode operations

In this mode, the analog input pins (ANInm)[Note] specified by the ADAnS register are selected sequentially from the ANIn0 pin, and A/D conversion is executed continuously. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin.

When conversion of all the specified analog input pin ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the conversion operation is stopped.

If the ADAnM0.ADAnCE bit is set (1), A/D conversion can be restarted.

In the one-shot scan mode, only the 1-buffer mode is supported.

This mode is suitable for applications in which multiple analog inputs are constantly monitored.

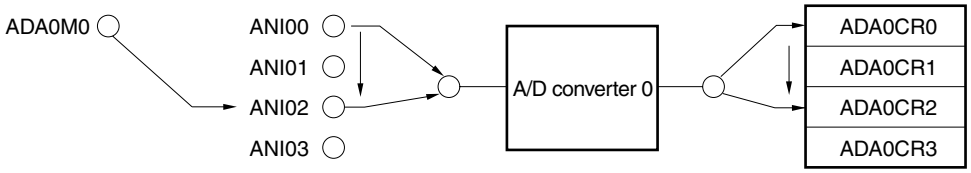
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANIn0	ADAnCR0
⋮	⋮
ANInm[Note]	ADAnCRm

Note Set by the ADAnS.ADAnS0 to ADAnS.ADAnS2 bits.
Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-16. Example of One-Shot Scan Mode Operation (Software Trigger One-Shot Scan): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The ANI00 pin is A/D converted
(3) The conversion result is stored in the ADA0CR0 register
(4) The ANI01 pin is A/D converted
(5) The conversion result is stored in the ADA0CR1 register
(6) The ANI02 pin is A/D converted
(7) The conversion result is stored in the ADA0CR2 register
(8) The ADA0M0.ADA0EF bit = 0
(9) The INTAD0 interrupt request signal is generated

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 11, ADA0M0.ADA0TMD bit = 0,
ADA0M2.ADA0BS bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.6 Operation in Timer Trigger Modes 0 and 1

With A/D converter n, the conversion timing is specified by using the A/D conversion start trigger signal (TQTADTn0, TQTADTn1) from the timer (motor control function) (see **Figure 12-3**).

- Timer trigger of A/D converter 0
  In timer trigger mode 0: TQTADT00
  In timer trigger mode 1: TQTADT10 (V850E/IA4 only)
- Timer trigger of A/D converter 1
  In timer trigger mode 0: TQTADT01
  In timer trigger mode 1: TQTADT11 (V850E/IA4 only)

The TQTADTa0 and TQTADTa1 signals are set by using the TQaAT00 to TQaAT03 bits of TMQa option register 2 (TQaOPT2) and the TQaAT10 to TQaAT13 bits of TMQa option register 3 (TQaOPT3). The trigger sources of the motor control function that can be selected as the A/D conversion start trigger, which is a timer trigger, are the INTTPaCC0, INTTPaCC1, INTTQaCC0, and INTTQaOV signals (two or more signals can be selected).

When the ADAnM2.ADAnTMD1 and ADAnM2.ADAnTMD0 bits are set to 01 or 10, A/D conversion is started at the rising edge of the A/D conversion start trigger signal (TQTADTa0, TQTADTa1) selected for the motor control function.

When the ADAnM0.ADAnCE bit is set to 1, the A/D converter waits for a trigger and, when the A/D conversion start trigger signal is input, it starts A/D conversion.

After the end of A/D conversion, the conversion result is stored in A/Dn conversion result register m (ADAnCRm) and, at the same time, the A/Dn conversion end interrupt request signal (INTADn) is generated.

After the end of A/D conversion, the A/D converter waits for a trigger regardless of the operation mode set by the ADAnM0.ADAnMD1 and ADAnM0.ADAnMD0 bits.

When conversion is started, the ADAnM0.ADAnEF bit is set to 1 (conversion in progress). However, while the A/D converter is waiting for a trigger, the ADAnM0.ADAnEF bit = 0 (conversion stopped).

If a valid trigger is input during A/D conversion, the conversion operation is stopped and started again from the beginning. If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and the A/D converter waits for a trigger again.

Caution In timer trigger modes 0 and 1, make sure that the A/D conversion start trigger signal (A/D conversions start timing) is not generated at an interval shorter than the minimum number of conversion clocks that can be specified by the ADAnM1.ADAnFR1 and ADAnM1.ADAnFR0 bits. If the A/D conversion start trigger signal is generated at an interval shorter than the minimum number of conversion clocks, the last trigger is valid.

Remark n = 0, 1
m = 0 to 3
V850E/IA3: a = 0
V850E/IA4: a = 0, 1

12.6.1 Continuous select mode/one-shot select mode operations

In these modes, the voltage of the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted. The conversion results are stored in the ADAnCRm register. In the continuous select mode or one-shot select mode, the 1-buffer mode and 4-buffer mode are supported according to the method of storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

(1) 1-buffer mode operation (1 buffer of continuous select/one-shot select by timer trigger)

In these modes, the voltage of one analog input pin (ANInm)[Note] is A/D converted once using the A/D conversion start trigger signal from the timer (motor control function) as a trigger, and the results are stored in one ADAnCRm register. The ANInm pin[Note] and the ADAnCRm register correspond one to one.

An A/Dn conversion end interrupt request signal (INTADn) is generated for each A/D conversion. After the end of A/D conversion, the A/D converter waits for a trigger.

This mode is suitable for applications in which the results of each first-time A/D conversion are read.

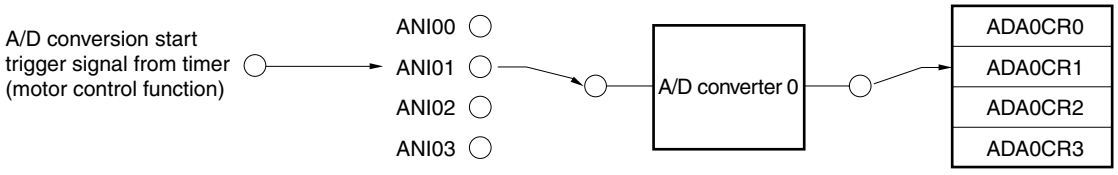
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCRm

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-17. Example of 1-Buffer Mode Operation (1 Buffer of Continuous Select/ One-Shot Select by Timer Trigger): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The A/D conversion start trigger signal from the timer (motor control function) is generated
(3) The ANI01 pin is A/D converted
(4) The conversion result is stored in the ADA0CR1 register
(5) The ADA0M0.ADA0EF bit = 0
(6) The INTAD0 interrupt request signal is generated
(7) If next trigger is input, return to (3)
(8) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00 or 10, ADA0M0.ADA0TMD bit = 1, ADA0M2.ADA0BS bit = 0, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 01 or 10, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 001

(2) 4-buffer mode operation (4 buffers of continuous select/one-shot select by timer trigger)

In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted four times using the A/D conversion start trigger signal from the timer (motor control function) as a trigger, and the results are stored in the ADAnCRm register.

The A/Dn conversion end interrupt request signal (INTADn) is generated when the four A/D conversions end. After the end of A/D conversion, the A/D converter waits for a trigger.

This mode is suitable for applications in which the average of the A/D conversion results is calculated.

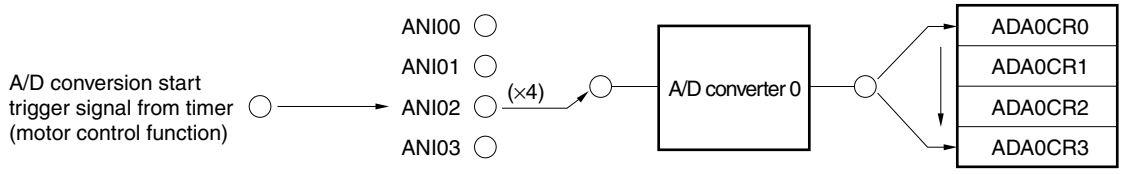
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCR0
ANInm[Note]	ADAnCR1
ANInm[Note]	ADAnCR2
ANInm[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-18. Example of 4-Buffer Mode Operation (4 Buffers of Continuous Select/ One-Shot Select by Timer Trigger): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The A/D conversion start trigger signal from the timer (motor control function) is generated
(3) The ANI02 pin is A/D converted
(4) The conversion result is stored in the ADA0CR0 register
(5) The ANI02 pin is A/D converted
(6) The conversion result is stored in the ADA0CR1 register
(7) The ANI02 pin is A/D converted
(8) The conversion result is stored in the ADA0CR2 register
(9) The ANI02 pin is A/D converted
(10) The conversion result is stored in the ADA0CR3 register
(11) The ADA0M0.ADA0EF bit = 0
(12) The INTAD0 interrupt request signal is generated
(13) If next trigger is input, return to (3)
(14) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00 or 10, ADA0M0.ADA0TMD bit = 1, ADA0M2.ADA0BS bit = 1, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 01 or 10, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.6.2 Continuous scan mode/one-shot scan mode operations

In these modes, the analog input pins (ANInm)[Note] specified by the ADAnS register are selected sequentially from the ANIn0 pin using the A/D conversion start trigger from the timer (motor control function) as a trigger and A/D conversion is performed continuously. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin[Note].

When conversion of all the specified analog input pins[Note] ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the A/D converter waits for a trigger.

This mode is suitable for applications in which multiple analog input pins are constantly monitored.

In the continuous scan mode or one-shot scan mode, only the 1-buffer mode is supported.

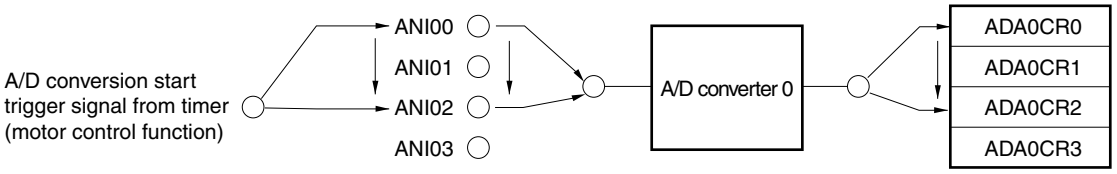
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANIn0	ADAnCR0
ANIn1	ADAnCR1
ANIn2[Note]	ADAnCR2
ANIn3[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-19. Example of Scan Mode Operation (Continuous Scan/ One-Shot Scan by Timer Trigger): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The A/D conversion start trigger signal from the timer (motor control function)
(3) The ANI00 pin is A/D converted
(4) The conversion result is stored in the ADA0CR0 register
(5) The ANI01 pin is A/D converted
(6) The conversion result is stored in the ADA0CR1 register
(7) The ANI02 pin is A/D converted
(8) The conversion result is stored in the ADA0CR2 register
(9) The ADA0M0.ADA0EF bit = 0
(10) The INTAD0 interrupt request signal is generated
(11) If next trigger is input, return to (3)
(12) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 01 or 11, ADA0M0.ADA0TMD bit = 1, ADA0M2.ADA0BS bit = 0, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 01 or 10, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.7 Operation in External Trigger Mode

In the external trigger mode, the analog input pins (ANIn0 to ANIn3)[Note] are A/D converted at the ADTRGn pin input timing.

The ADTRG0 pin has an alternate function as the P04/INTP4 pin and the ADTRG1 pin has an alternate function as the P05/INTP5 pin. To set the external trigger mode, set the PMC04 bit of port mode control register 0 (PMC0) to 1 and the ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits to 00B with A/D converter 0. With A/D converter 1, set the PMC05 bit of port mode control register 0 (PMC0) to 1 and the ADA1M2.ADA1TMD1 and ADA1M2.ADA1TMD0 bits to 00B.

For the valid edge of the external input signal in the external trigger mode, the rising edge, falling edge, or both rising and falling edges can be specified by setting the ADAnM0.ADAnETS1 and ADAnM0.ADAnETS0 bits.

When the ADAnM0.ADAnCE bit is set (1), the A/D converter waits for a trigger and, when the trigger is input from the ADTRGn pin, starts A/D conversion.

After the end of A/D conversion, the conversion result is stored in A/Dn conversion result register m (ADAnCRm) and, at the same time, the A/Dn conversion end interrupt request signal (INTADn) is generated.

After the end of A/D conversion, the A/D converter waits for a trigger regardless of the operation mode set by the ADAnM0.ADAnMD1 and ADAnM0.ADAnMD0 bits.

When conversion is started, the ADAnM0.ADAnEF bit is set to 1 (conversion in progress). However, while the A/D converter is waiting for a trigger, the ADAnEF bit = 0 (conversion stopped).

If a valid trigger is input during A/D conversion, the conversion operation is stopped and started again from the beginning. If the ADAnM0, ADAnM2, and ADAnS registers are written during A/D conversion, the conversion is stopped and the A/D converter waits for a trigger again.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

12.7.1 Continuous select mode/one-shot select mode operations

In these modes, the analog input pin (ANInm)[Note] specified by the ADAnS register is A/D converted. The conversion results are stored in the ADAnCRm register. In the continuous select mode or one-shot select mode, there are two select modes: 1-buffer mode and 4-buffer mode, according to the method of storing the A/D conversion results.

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

(1) 1-buffer mode (1 buffer of continuous select/one-shot select by external trigger)

In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted once using the ADTRGn signal as a trigger. The conversion results are stored in one ADAnCRm register. The ANInm pin[Note] and the ADAnCRm register correspond one to one.

The A/Dn conversion end interrupt request signal (INTADn) is generated for each A/D conversion. After the end of A/D conversion, the A/D converter waits for a trigger

This mode is suitable for applications in which the results of each first-time A/D conversion are read.

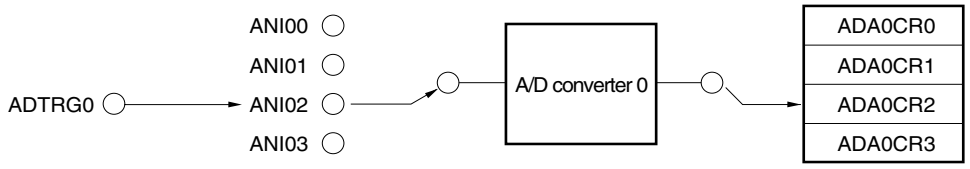
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCRm

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-20. Example of 1-Buffer Mode Operation (1 Buffer of Continuous Select/ One-Shot Select by External Trigger): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The external trigger signal (ADTRG0) is generated
(3) The ANI02 pin is A/D converted
(4) The conversion result is stored in the ADA0CR2 register
(5) The ADA0M0.ADA0EF bit = 0
(6) The INTAD0 interrupt request signal is generated
(7) If next trigger is input, return to (3)
(8) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00 or 10, ADA0M0.ADA0TMD bit = 1,
ADA0M2.ADA0BS bit = 0, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 00,
ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

(2) 4-buffer mode (4 buffers of continuous select/one-shot select by external trigger)

In this mode, the voltage of one analog input pin (ANInm)[Note] is A/D converted four times using the ADTRGn signal as a trigger and the results are stored in the ADAnCRm register.

The A/Dn conversion end interrupt request signal (INTADn) is generated when the four A/D conversions end. After the end of A/D conversion, the A/D converter waits for a trigger.

This mode is suitable for applications in which the average of the A/D conversion results is calculated.

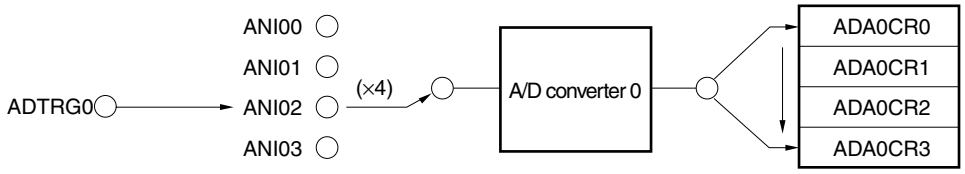
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANInm[Note]	ADAnCR0
ANInm[Note]	ADAnCR1
ANInm[Note]	ADAnCR2
ANInm[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-21. Example of 4-Buffer Mode Operation (4 Buffers of Continuous Select/ One-Shot Select by External Trigger): V850E/IA4



(1) The ADA0CE bit = 1 (enable)
(2) The external trigger signal (ADTRG0) is generated
(3) The ANI02 pin is A/D converted
(4) The conversion result is stored in the ADA0CR0 register
(5) The ANI02 pin is A/D converted
(6) The conversion result is stored in the ADA0CR1 register
(7) The ANI02 pin is A/D converted
(8) The conversion result is stored in the ADA0CR2 register
(9) The ANI02 pin is A/D converted
(10) The conversion result is stored in the ADA0CR3 register
(11) The ADA0M0.ADA0EF bit = 0
(12) The INTAD0 interrupt request signal is generated
(13) If next trigger is input, return to (3)
(14) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 00 or 10, ADA0M0.ADA0TMD bit = 1, ADA0M2.ADA0BS bit = 1, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 00, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 010

12.7.2 Continuous scan mode/one-shot scan mode operations

In this mode, the analog input pins (ANInm)[Note] specified by the ADAnS register are selected sequentially from the ANIn0 pin using the ADTRGn signal as a trigger, and A/D converted continuously. The A/D conversion results are stored in the ADAnCRm register corresponding to the analog input pin[Note].

When conversion of all the specified analog input pins[Note] ends, the A/Dn conversion end interrupt request signal (INTADn) is generated. After the end of A/D conversion, the A/D converter waits for a trigger.

This is most suitable for applications in which multiple analog inputs are constantly monitored.

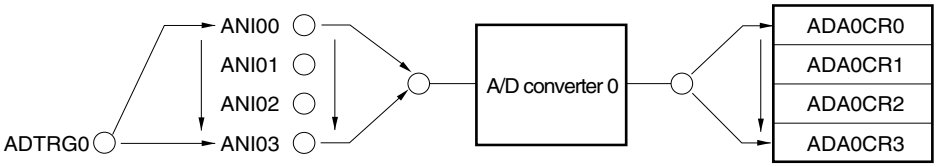
Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Analog Input Pin	A/D Conversion Result Register
ANIn0	ADAnCR0
ANIn1	ADAnCR1
ANIn2[Note]	ADAnCR2
ANIn3[Note]	ADAnCR3

Note Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Remark n = 0, 1
m = 0 to 3

Figure 12-22. Example of Scan Mode Operation (Continuous Scan/ One-Shot Scan by External Trigger): V850E/IA4



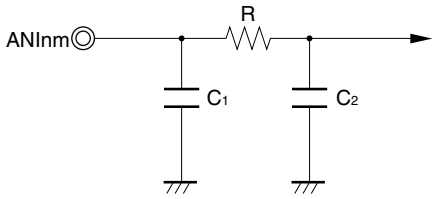
(1) The ADA0CE bit = 1 (enable)
(2) The external trigger signal (ADTRG0) is generated
(3) The ANI00 pin is A/D converted
(4) The conversion result is stored in the ADA0CR0 register
(5) The ANI01 pin is A/D converted
(6) The conversion result is stored in the ADA0CR1 register
(7) The ANI02 pin is A/D converted
(8) The conversion result is stored in the ADA0CR2 register
(9) The ANI03 pin is A/D converted
(10) The conversion result is stored in the ADA0CR3 register
(11) The ADA0M0.ADA0EF bit = 0
(12) The INTAD0 interrupt request signal is generated
(13) If next trigger is input, return to (3)
(14) To end the conversion, the ADA0CE bit = 0 (stop)

Remark This is an operation example with the following setting.
ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 01 or 11, ADA0M0.ADA0TMD bit = 1, ADA0M2.ADA0BS bit = 0, ADA0M2.ADA0TMD1 and ADA0M2.ADA0TMD0 bits = 00, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 011

12.8 Internal Equivalent Circuit

The following figure shows the equivalent circuit of the analog input block.

Figure 12-23. ANInm Pin Internal Equivalent Circuit



R	C_1	C_2
4.0 kΩ	15 pF	5.7 pF

Remarks 1. The maximum values are shown (reference values).

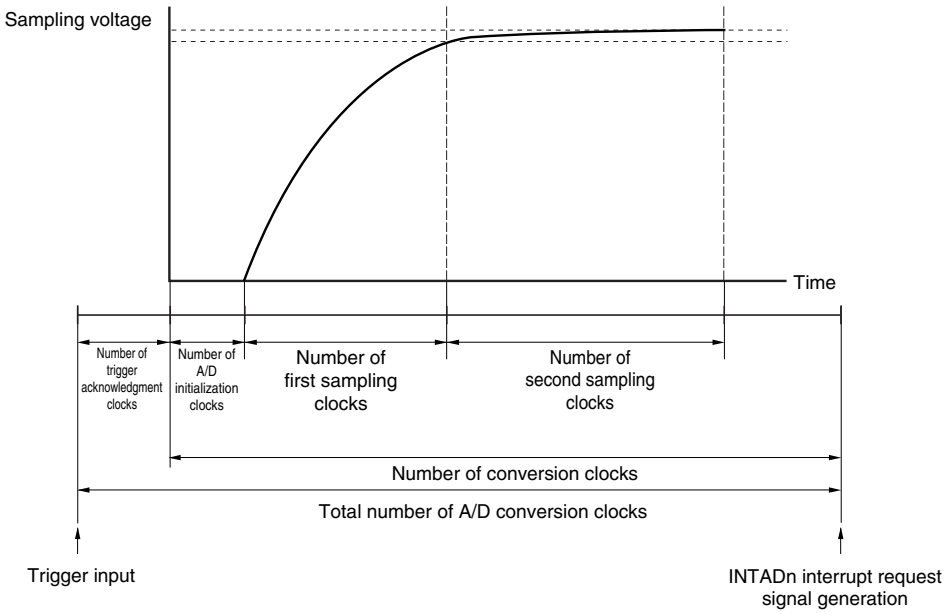
2. n = 0, 1

m = 0 to 3

Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

Caution A/D converters 0 and 1 perform the first sampling when A/D conversion starts after A/D initialization. Sampling power is almost fully charged during this period. Therefore, the sampling error can be calculated under this condition. After that, the converters perform the second sampling to correct the error.

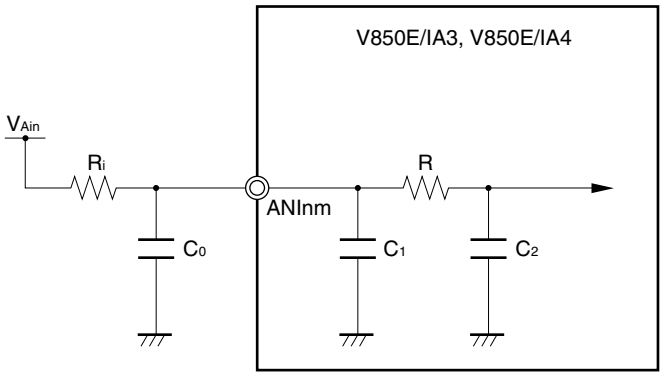
The sampling period of A/D converters 0 and 1 is from the start of conversion to the end of the second sampling period. Avoid noise during this period.



ADAnM1 Register		Number of A/D Initialization Clocks	Number of First Sampling Clocks	Number of Second Sampling Clocks
ADAnFR1 Bit	ADAnFR0 Bit			
0	1	4	36	36
1	0	6	54	54
1	1	8	72	72

An example of calculating an overall error of A/D converters 0 and 1 is shown below.

Figure 12-24. Example of Calculating Overall Error of A/D Converters 0 and 1



fxx (MHz)	Conversion Time (μs)	A/D Initialization Clock (μs)	First Sampling (μs)	Second Sampling (μs)	R (kΩ)	C1 (pF)	C2 (pF)	C0 (pF)	Ri (kΩ)	Sampling Error (LSB)[Note]
64	1.95 (124 clocks)	0.0625 (4 clocks)	0.5 (32 clocks)	0.5625 (36 clocks)	4.0	15	5.7	100	1.0	20.8
								100	0.5	0.4
								100	0.25	0.1 or lower
								100	0.1	0.1 or lower
								50	1.0	1.4
								50	0.5	0.1 or lower
								50	0.25	0.1 or lower
								50	0.125	0.1 or lower

Note The error when considering the signal source impedance is “sampling error + overall error”.

Remarks 1. These values are reference values calculated by simulating what happens to C_2 voltage by R_i and C_0 when V_{Ain} is applied from 0 V to 5 V at the same time as sampling start.

2. n = 0, 1
m = 0 to 3
Only the ANI00 and ANI01 pins can be used when A/D converter 0 of the V850E/IA3 is used.

3. fxx: System clock frequency

12.9 Notes on Operation

Caution For operation when using the operational amplifier for input level amplification, refer to 12.3 (3) A/D converter n channel specification register (ADAnS) (n = 0, 1).
For the relationship between the analog input pins and A/D conversion result registers, see Table 12-4.

12.9.1 Stopping conversion operation

When the ADAnM0.ADAnCE bit is cleared to 0 during a conversion operation, the conversion operation stops and the conversion results are not stored in A/Dn conversion result register m (ADAnCRm).

The ADAnCE bit is not cleared to 0 even after the A/Dn conversion end interrupt request signal (INTADn) has been generated in all modes.

Remark n = 0, 1
m = 0 to 3

12.9.2 Timer/external trigger interval

Make sure that the occurrence interval of the trigger in timer trigger mode 0, 1, or external trigger mode is longer than the total number of conversion clocks specified by the ADAnM1.ADAnFR1 and ADAnM1.ADAnFR0 bits (see **Table 12-2 Number of Conversion Clocks**).

(1) When 0 < trigger occurrence interval < total number of A/D conversion clocks

When the timer/external trigger is input during a conversion operation, the conversion operation is aborted and the conversion starts according to the last timer/external trigger input.

When conversion operations are aborted, the conversion results from the conversion operation immediately before are not stored in the ADAnCRm register. Note, therefore, that the generation of the INTADn signal and storing of the result in the ADAnCRm register are not guaranteed.

Remark n = 0, 1
m = 0 to 3

(2) When trigger occurrence interval ≥ total number of A/D conversion clocks

The INTADn signal is generated, and the value at the end of conversion is correctly stored in the ADAnCRm register. Design so that the trigger occurrence interval is equal or greater than the total number of A/D conversion clocks.

Remark n = 0, 1
m = 0 to 3

12.9.3 Operation in standby mode

(1) HALT mode

In this mode, A/D conversion continues.

(2) IDLE mode, STOP mode

As clock supply to A/D converters 0 and 1 is stopped, no conversion operations are performed.

When these modes are released by the maskable interrupt request signal input pin[Note] and the overvoltage detection interrupt request signal (INTCMPn), the ADAnM0, ADAnM1, ADAnM2, ADAnS, OPnCTL0, and OPnCTL1 registers and A/Dn conversion result register m (ADAnCRm) hold their values. However, when the IDLE or STOP mode is set during a conversion operation, the conversion operation is suspended. At this time, if the mode released by the maskable interrupt request signal input pin[Note] and the overvoltage detection interrupt request signal (INTCMPn), the conversion operation resumes. At this time, the A/Dn conversion end interrupt request signal (INTADn) may be generated, but the conversion result written to the ADAnCRm register will be undefined.

Note V850E/IA3: INTP0, INTP2 to INTP5, INTP7 pins
V850E/IA4: INTP0 to INTP5, INTP7 pins

Remark n = 0, 1
m = 0 to 3

12.9.4 Timer interrupt request signal in timer trigger modes 0 and 1

The timer interrupt request signal (TQTADTn0, TQTADTn1) becomes an A/D conversion start trigger and starts the conversion operation. When this happens, the timer interrupt request signal also functions as an interrupt for the CPU. In order to prevent the generation of interrupts for the CPU, disable interrupts using the mask bits of the interrupt control register.

Remark n = 0, 1

12.9.5 Re-conversion start trigger input during stabilization time

If the stabilization time end timing and the register writing conflict or the stabilization time end timing and the trigger input conflict, the stabilization time is inserted again.

<R> ### 12.9.6 Variation of A/D conversion results

The results of the A/D conversion may vary depending on the fluctuation of the supply voltage, or may be affected by noise. To reduce the variation, take counteractive measures with the program, such as by averaging the A/D conversion results.

<R> ### 12.9.7 A/D conversion result hysteresis characteristics

Successive comparison type A/D converters hold an analog input voltage in an internal sample & hold capacitor and then perform A/D conversion. After the A/D conversion has finished, the analog input voltage remains in the internal sample & hold capacitor. As a result, the following phenomena may occur if the output impedance from the analog input source is too high.

- When the same channel is used for A/D conversions, if the voltage is higher or lower than the previous A/D conversion, then hysteresis characteristics may appear where the conversion result is affected by the previous value. Even if the conversion were to be performed at the same potential, the results may thus vary.
- When switching the analog input channel, hysteresis characteristics may appear where the conversion result is affected by the previous channel value. This is because one A/D converter is used for the A/D conversions. Even if the conversion were to be performed at the same potential, the results may thus vary.

To obtain more accurate conversion results, lower the output impedance from the analog input source or execute A/D conversion twice consecutively on the same channel, and discard the first conversion result.

<R> **12.9.8 Restrictions on setting one-shot mode and software trigger mode**

If the A/D converters 0 and 1 are set in the one-shot select mode and software trigger mode (ADAnM0 register = 1010XX0XB) or one-shot scan mode and software trigger mode (ADAnM0 register = 1011XX0XB), a re-conversion operation should be performed in a new condition when data is written to any of the ADAnM0, ADAnM2, and ADAnS registers upon completion of an A/D conversion operation. However, the re-conversion operation is not performed but the conversion operation is enabled (ADAnM0.ADAnCE bit = 1) and stopped (ADAnM0.ADAnEF bit = 0). The A/Dn conversion end interrupt request signal (INTADn) is not generated, nor is the last A/D conversion result stored. However, the data is correctly written to any of the ADAnM0, ADAnM2, and ADAnS registers.

If this happens, normal operation can be restored by setting the ADAnM0.ADAnCE bit to 1.

For example, if the ANIn0 and ANIn1 pins are set in the scan mode (ADAnS register = 00000001B) and data is written to the ADAnM0 register upon completion of an A/D conversion operation in the one-shot scan mode and software trigger mode (ADAnM0 register = 1011XX0XB), the signal of the ANIn0 pin is correctly converted and the conversion result is correctly stored in the ADAnCR0 register. However, the result of converting the signal of the ANIn1 pin which has been performed immediately before the completion of the A/D conversion is not stored in the ADAnCR1 register, nor is the INTADn interrupt request signal generated.

[Countermeasure]

The above restriction can be avoided by performing any of steps <1> to <3>, below.

<1> Before writing to any of the ADAnM0, ADAnM2, and ADAnS registers, confirm that A/D conversion is stopped (ADAnM0.ADAnEF bit = 0).

<2> After disabling the interrupt (PSW.ID bit = 1) or disabling the DMA transfer (DCHCn.Enn bit = 0), execute an instruction that writes data to any of the ADAnM0, ADAnM2, and ADAnS registers and an instruction that sets the ADAnM0.ADAnCE bit to 1 consecutively, and then enable the interrupt (PSW.ID bit = 0) or enable the DMA transfer (DCHCn.Enn bit = 1).
This action is to avoid coincidence between the completion of the A/D conversion operation and writing to the ADAnM0, ADAnM2, or ADAnS register. If, for example, executing a write instruction and the completion of the A/D conversion operation coincide and thus the A/D conversion is stopped, the A/D conversion can be started by setting of the ADAnCE bit to 1. If the ADAnM0.ADAnCE bit = 1, the ADAnCE bit is set to 1 again consecutively.

<3> Disable the A/D conversion operation by clearing the ADAnCE bit to 0, write data to any of the ADAnM0, ADAnM2, and ADAnS registers, enable the A/D conversion operation by setting the ADAnCE bit to 1, and start the A/D conversion.

12.10 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 1 LSB (Least Significant Bit). The percentage of 1 LSB with respect to the full scale is expressed by %FSR (Full Scale Range). %FSR indicates the ratio of analog input voltage that can be converted as a percentage, and is always represented by the following formula regardless of the resolution.

1%FSR = (Max. value of analog input voltage that can be converted − Min. value of analog input voltage that can be converted)/100
= (AV_{DD} – 0)/100
= AV_{DD}/100

1 LSB is as follows when the resolution is 10 bits.

1 LSB = $1/2^{10}$ = 1/1,024
= 0.098%FSR

Accuracy has no relation to resolution, but is determined by overall error.

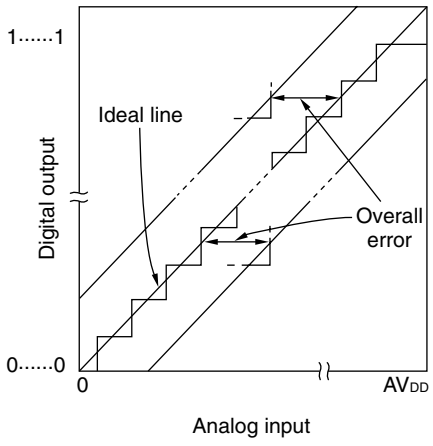
(2) Overall error

This shows the maximum error value between the actual measured value and the theoretical value.

Zero-scale error, full-scale error, linearity error and 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.

Figure 12-25. Overall Error

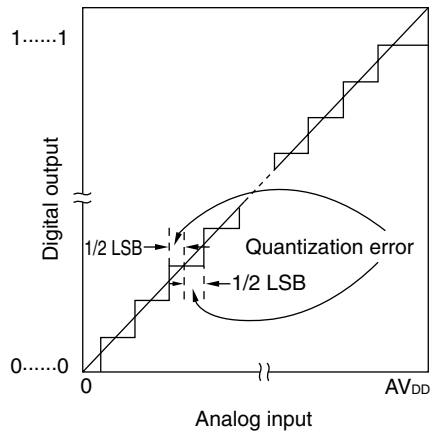


(3) Quantization error

When analog values are converted to digital values, a ±1/2 LSB error naturally occurs. In an A/D converter, an analog input voltage in a range of ±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.

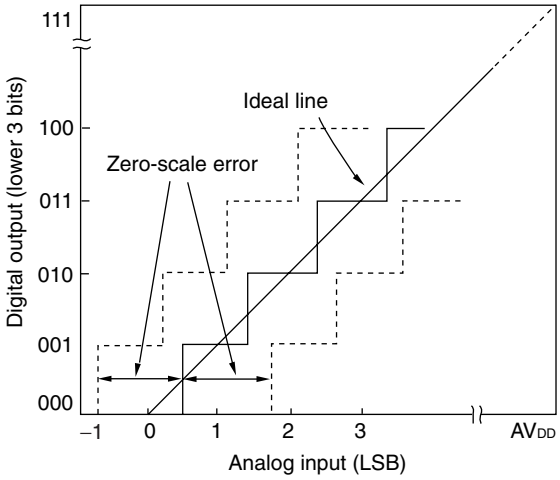
Figure 12-26. Quantization Error



(4) Zero-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (1/2 LSB) when the digital output changes from 0……000 to 0……001.

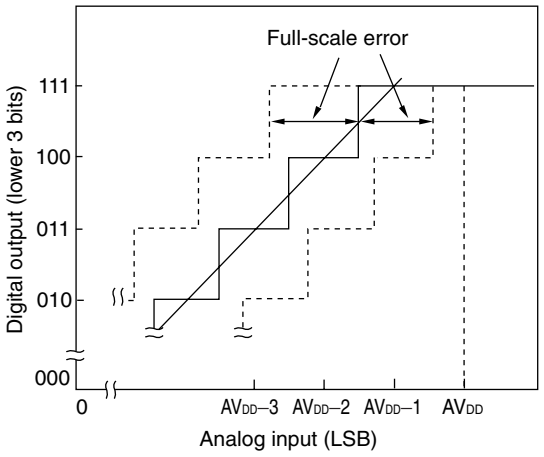
Figure 12-27. Zero-Scale Error



(5) Full-scale error

This shows the difference between the actual measurement value of the analog input voltage and the theoretical value (full-scale value − 3/2 LSB) when the digital output changes from 1……110 to 1……111.

Figure 12-28. Full-Scale Error

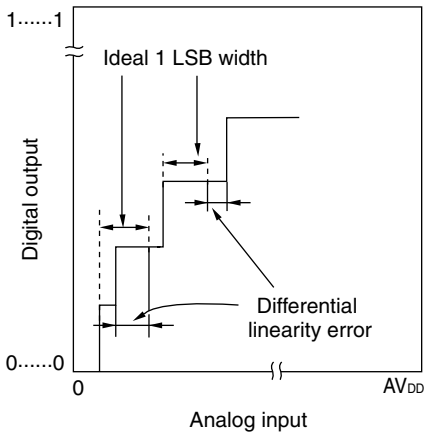


(6) Differential linearity error

While the ideal width of code output is 1 LSB, this indicates the difference between the actual measurement value and the ideal value.

<R> This indicates the basic characteristics of the A/D conversion when the voltage applied to the analog input pins of the same channel is consistently increased bit by bit from AV_{SS} to AV_{DD}. See **12.10 (2) Overall error** for when the input voltage is increased or decreased, or when two or more channels are used.

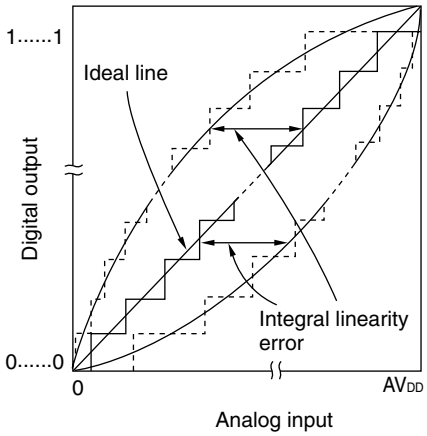
Figure 12-29. Differential Linearity Error



(7) 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.

Figure 12-30. Integral Linearity Error



(8) Conversion time

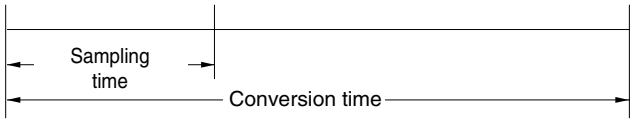
This expresses the time from when the trigger is generated to when the digital output is obtained.

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.

Figure 12-31. Sampling Time



CHAPTER 13 A/D CONVERTER 2

The V850E/IA3 and V850E/IA4 are provided with the A/D converter of the first-order $\Delta\Sigma$ conversion method for which 8-bit or 10-bit resolution can be selected.

13.1 Features

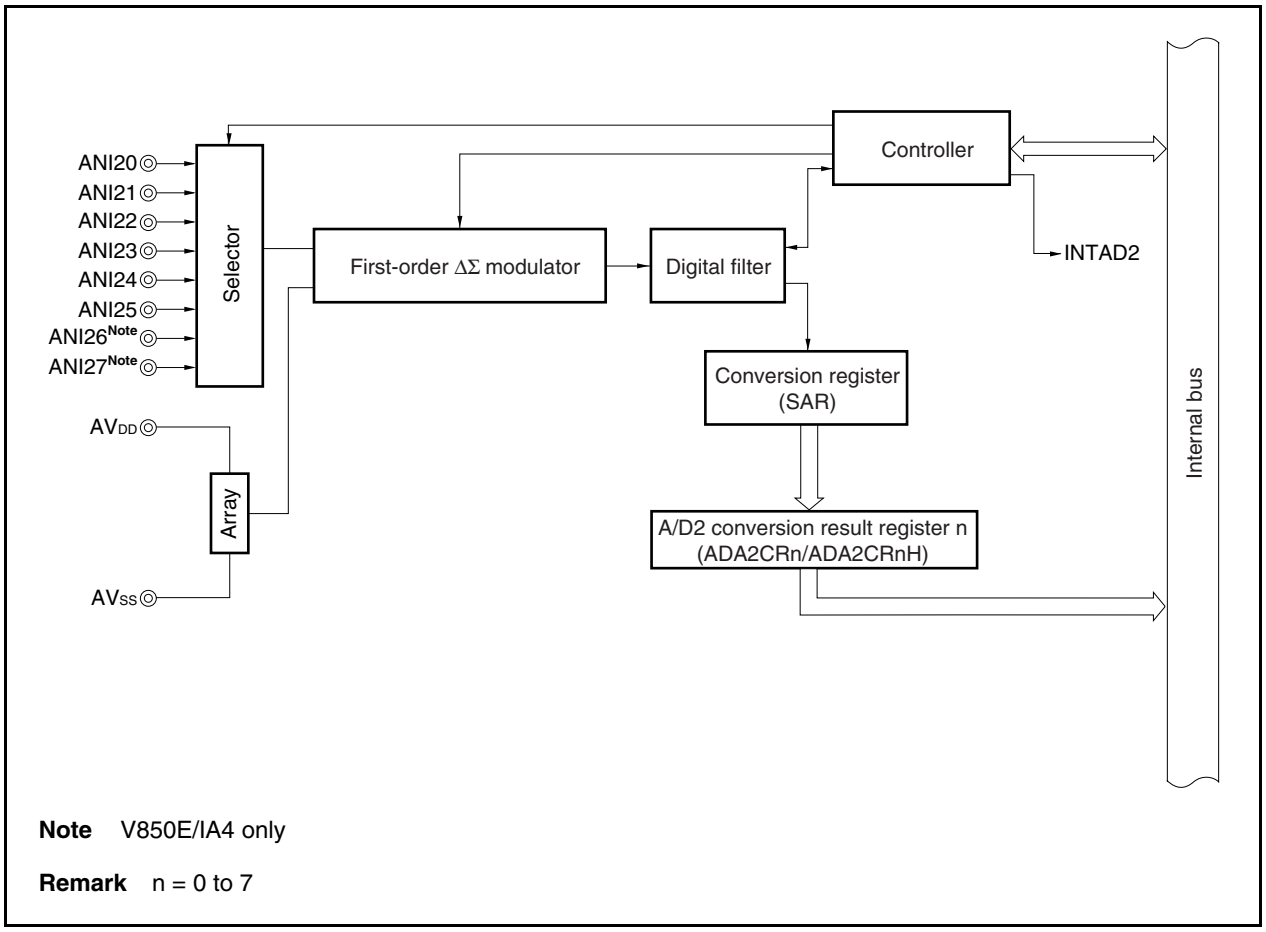
- On-chip 8-/10-bit resolution A/D converter using first-order $\Delta\Sigma$ conversion method
- Analog input
  V850E/IA3: ANI20 to ANI25 (6 channels)
  V850E/IA4: ANI20 to ANI27 (8 channels)
- On-chip A/D2 conversion result register n (ADA2CRn)
  8/10 bits $\times$ 8
- A/D conversion trigger mode
  Software trigger mode
- A/D conversion operation mode
  Serial mode
  Parallel mode
- Buffer mode
  1-buffer mode
  4-buffer mode

Remark n = 0 to 7

13.2 Configuration

The block diagram is shown below.

Figure 13-1. Block Diagram of A/D Converter 2



Note V850E/IA4 only

Remark n = 0 to 7

Cautions 1. If there is noise at the analog input pins (ANI2n) or at the A/D converter power supply voltage pin (AV_{DD}), that noise may generate an illegal conversion result (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7).
Software processing will be needed to avoid a negative effect on the system from this illegal conversion result.
An example of this software processing is shown below.

- **Take the average result of a number of A/D conversions and use that as the A/D conversion result.**
- **Execute a number of A/D conversions consecutively and use those results, omitting any exceptional results that may have been obtained.**
- **If an A/D conversion result that is judged to have generated a system malfunction is obtained, be sure to recheck the system malfunction before performing malfunction processing.**

2. Do not apply a voltage outside the AV_{SS} to AV_{DD} range to the pins that are used as A/D converter 2 input pins.

Remark For details on the power supply connection specifications of A/D converter 2, see **12.2 Configuration**.

A/D converter 2 consists of the following hardware.

Table 13-1. Configuration of A/D Converter 2

Item	Configuration
Analog input	V850E/IA3: ANI20 to ANI25 (6 channels) V850E/IA4: ANI20 to ANI27 (8 channels)
Registers	Conversion register (SAR) A/D2 conversion result registers 0 to 7 (ADA2CR0 to ADA2CR7) A/D2 conversion result registers 0H to 7H (ADA2CR0H to ADA2CR7H)
Control registers	A/D converter 2 control register 0 (ADA2CTL0) A/D converter 2 control register 1 (ADA2CTL1) A/D converter 2 control register 2 (ADA2CTL2) A/D converter 2 control register 3 (ADA2CTL3) A/D converter 2 status register (ADA2STR)

(1) Selector

The input circuit selects the analog input pins (V850E/IA3: ANI20 to ANI25, V850E/IA4: ANI20 to ANI27) according to the mode set by the ADA2CTL0, ADA2CTL1, ADA2CTL2, and ADA2CTL3 registers and performs A/D conversion.

(2) First-order ΔΣ modulator

This circuit samples the analog input signal selected by the input circuit, converts it into 1-bit data, and sends it to a digital filter.

It samples the output of the operational amplifier 512 times when 8-bit resolution is selected and 2,048 times when 10-bit resolution is selected.

The sampling clock is selected by the ADA2CTL1 register.

(3) Digital filter

This filter converts the 1-bit data converted by the first-order ΔΣ modulator into 8 or 10 bits.

(4) Conversion register (SAR)

SAR stores the 8-bit or 10-bit A/D conversion result of the digital filter. When the result has been stored in this register (end of A/D conversion), the contents (conversion result) of the SAR register are held by A/D2 conversion result registers 0 to 7 (ADA2CR0 to ADA2CR7). When A/D conversion has ended under the specified conditions, an A/D2 conversion end interrupt request signal (INTAD2) is generated.

(5) A/D2 conversion result registers 0 to 7 (ADA2CR0 to ADA2CR7) and A/D2 conversion result registers 0H to 7H (ADA2CR0H to ADA2CR7H)

The ADA2CRn and ADA2CRnH registers are registers that hold a result of A/D conversion. Each time A/D conversion ends, the conversion result is loaded to this register from the conversion register (SAR), and the conversion result is stored in the higher 8 or 10 bits of the ADA2CRn register at the resolution set by the ADA2CTL1 register (8 or 10 bits). The lower 7 bits or 5 bits are always 0 when read.

For the ADA2CRnH register, the higher 8 bits of the A/D conversion result are read.

During the A/D conversion result is read in 16-bit units, the ADA2CRn register is specified and during the higher 8 bits are read, the ADA2CRnH register is specified.

(6) A/D converter 2 control register 0 (ADA2CTL0)

This register is used to control the analog power supply and conversion operation.

(7) A/D converter 2 control register 1 (ADA2CTL1)

This register is used to specify the sampling clock and resolution (number of samplings) of the analog input to be A/D converted.

(8) A/D converter 2 control register 2 (ADA2CTL2)

This register is used to specify the analog input pin to be A/D converted.

(9) A/D converter 2 control register 3 (ADA2CTL3)

This register is used to specify the buffer mode and operation mode of the A/D conversion.

(10) A/D converter 2 status register (ADA2STR)

This register is used to check the A/D-converted analog input pin.

(11) Controller

The controller selects the analog input pin, generates the operation timing of the first-order $\Delta\Sigma$ modulator and digital filter, and controls the conversion trigger according to the mode set by the ADA2CTL0, ADA2CTL1, ADA2CTL2, and ADA2CTL3 registers.

(12) ANI20 to ANI27 pins

The ANI2n pin is an analog input pin for A/D converter 2 (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7). These pins input the analog signals to be A/D converted.

Caution Make sure that the voltages input to ANI2n do not exceed the rated values. If a voltage higher than or equal to AV_{DD} or lower than or equal to AV_{SS} (even within the range of the absolute maximum ratings) is input to a channel, the conversion value of the channel is undefined, and the conversion values of the other channels may also be affected.

(13) AV_{DD} pin

This alternates as the pin for inputting the positive power supply and reference voltage of A/D converters 0 to 2. With A/D converter 2, it converts signals input to the ANI2n pin to digital signals based on the voltage applied between AV_{DD} and AV_{SS} (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7).

Always make the potential at this pin the same as that at the EV_{DD} pin even when A/D converter 2 is not used.

(14) AV_{SS} pin

This is the ground pin of A/D converters 0 to 2. Always make the potential at this pin the same as that at the EV_{SS} pin even when A/D converter 2 is not used.

13.3 Control Registers

A/D converter 2 is controlled by the following registers.

- A/D converter 2 control registers 0 to 3 (ADA2CTL0 to ADA2CTL3)
- A/D converter 2 status register (ADA2STR)

The following registers are also used.

- A/D2 conversion result registers 0 to 7 (ADA2CR0 to ADA2CR7)
- A/D2 conversion result registers 0H to 7H (ADA2CR0H to ADA2CR7H)

(1) A/D converter 2 control register 0 (ADA2CTL0)

The ADA2CTL0 register is an 8-bit register that controls the analog power supply and conversion operations.

This register can be read or written in 8-bit or 1-bit units. However, bits 6 and 7 are write-only in 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF240H

	<7>	<6>	5	4	3	2	1	0
ADA2CTL0	ADA2PON	ADA2CE	0	0	0	0	0	0

ADA2PON	Analog power supply control
0	Turns off analog power supply
1	Turns on analog power supply

ADA2CE	A/D conversion operation control
0	Stops conversion operation
1	Starts conversion operation

Caution Do not set the analog power supply control bit (ADA2PON) and A/D conversion operation bit (ADA2CE) to 1 at the same time. Set the ADA2CE bit to 1 at least 5 μs after the ADA2PON bit was set to 1. If the ADA2CE bit is set to 1 before the lapse of 5 μs, A/D conversion is executed but the accuracy of the result of the first conversion cannot be guaranteed.

(2) A/D converter 2 control register 1 (ADA2CTL1)

The ADA2CTL1 register is an 8-bit register that specifies the sampling clock for analog input and the resolution (the number of sampling times).

This register can be read or written in 8-bit or 1-bit units. However, writing the ADA2CTL1 register during A/D conversion, including writing the same value, is prohibited. If data is written to the register during A/D conversion, the conversion result of A/D2 conversion result register n (ADA2CRn) and the conversion operation cannot be guaranteed. To write data to the ADA2CTL1 register, be sure to stop the conversion once (ADA2CTL0.ADA2CE bit = 0), and then re-set the conversion conditions.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF241H

	7	6	5	4	3	2	1	0
ADA2CTL1	ADA2FS1	ADA2FS0	0	0	0	0	ADA2N1	ADA2N0

ADA2FS1	ADA2FS0	Sampling clock specification[Note 1]
0	0	f_{XX}/16 (4 MHz, f_{XX} = 64 MHz)
0	1	f_{XX}/32
1	0	f_{XX}/64
1	1	Setting prohibited

ADA2N1	ADA2N0	Specification of resolution (number of sampling times)[Note 2]
0	0	Setting prohibited
0	1	8 bits (512 times)
1	0	Setting prohibited
1	1	10 bits (2,048 times)

Notes 1. Supply the sampling clock in a range of 1 MHz to 4 MHz.

2. The conversion time can be calculated by sampling clock × number of sampling times.
However, the conversion time immediately after the ADA2CTL0.ADA2CE bit is set (1) or the first conversion time after the analog input pin (ANI2n) is converted can be calculated from “sampling clock × (sampling counts + 6)” (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7).

Caution Be sure to clear bits 2 to 5 to “0”.

Remark f_{XX}: Peripheral clock

(3) A/D converter 2 control register 2 (ADA2CTL2)

The ADA2CTL2 register is an 8-bit register that specifies the analog input pin (ANI2n)[Note].

This register can be read or written in 8-bit or 1-bit units. However, if the ADA2CTL2 register is written (if the ANI2n pin[Note] is changed) during A/D conversion, the conversion under execution is aborted, and conversion of the signal at the newly specified ANI2n pin[Note] is started.

Reset sets this register to 00H.

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

After reset: 00H R/W Address: FFFFF242H

	7	6	5	4	3	2	1	0
ADA2CTL2	0	0	0	0	0	ADA2S2	ADA2S1	ADA2S0

ADA2S2	ADA2S1	ADA2S0	Analog input pin
0	0	0	ANI20
0	0	1	ANI21
0	1	0	ANI22
0	1	1	ANI23
1	0	0	ANI24
1	0	1	ANI25
1	1	0	ANI26[Note]
1	1	1	ANI27[Note]

Note V850E/IA4 only

Cautions 1. If there is a chance that end of the A/D conversion under execution will conflict with the specification of an analog input pin (ANI2n), use the ADA2STR register to identify which ANI2n pin has already finished being converted.

2. If data is written to the ADA2CTL2 register (if the ANI2n pin has been changed) during A/D conversion, the conversion result is not stored in A/D2 conversion result register n (ADA2CRn). The A/D conversion in progress will be initialized and conversion of the signal at the newly specified ANI2n pin will be started.

(4) A/D converter 2 control register 3 (ADA2CTL3)

The ADA2CTL3 register is an 8-bit register that specifies the A/D conversion buffer mode and operation mode.

This register can be read or written in 8-bit or 1-bit units. However, writing the ADA2CTL3 register during A/D conversion, including writing the same value, is prohibited. If data is written to the register during A/D conversion, the conversion result of A/D2 conversion result register n (ADA2CRn) and the conversion operation cannot be guaranteed. To write data to the ADA2CTL3 register, be sure to stop the conversion once (ADA2CTL0.ADA2CE bit = 0), and then re-set the conversion conditions.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF243H

	7	6	5	4	3	2	1	0
ADA2CTL3	ADA2BS	ADA2TS	0	0	0	0	0	0

ADA2BS	Buffer mode specification
0	1-buffer mode
1	4-buffer mode

ADA2TS	Operation mode specification
0	Serial mode
1	Parallel mode

Caution The relationship between the analog input pin (ANI2n) and A/D2 conversion result register n (ADA2CRn) in each buffer mode is as follows (n = 0 to 7).

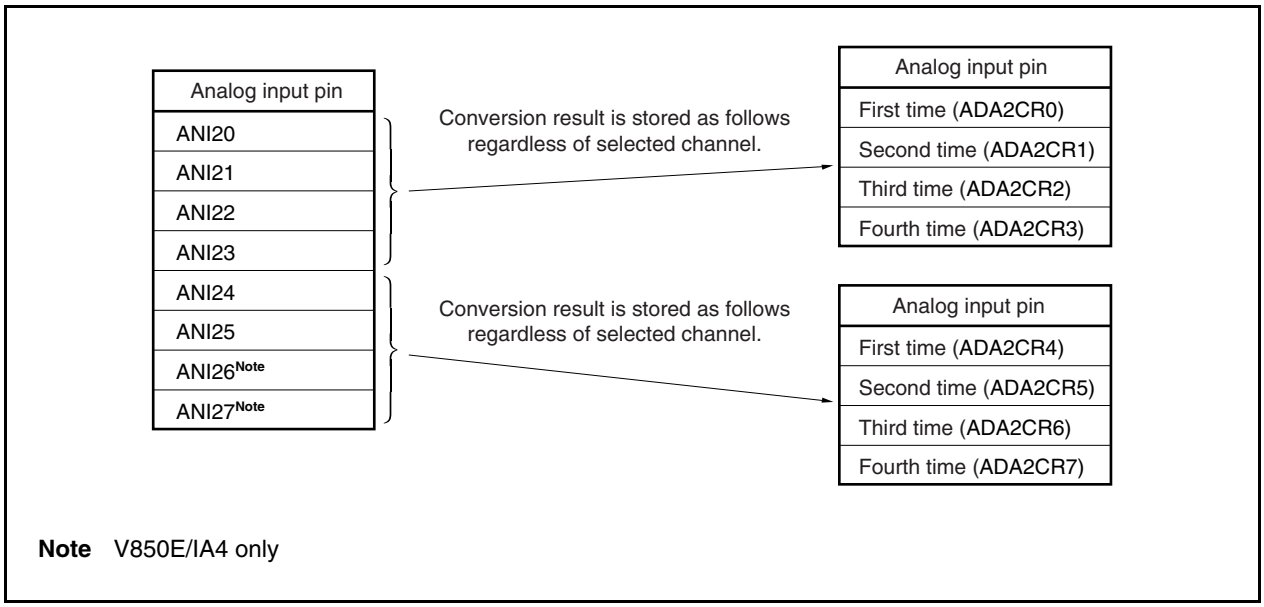
- **In 1-buffer mode**
  One analog input pin (ANI2n) specified by the ADA2CTL2 register corresponds to one ADA2CRn register (see Table 13-2).
- **In 4-buffer mode**
  The conversion result of one analog input pin (ANI2n) of the ANI20 to ANI23 pins specified by the ADA2CTL2 register is stored in the ADA2CR0 to ADA2CR3 registers, and the result of one of the ANI24 to ANI247 pins is stored in the ADA2CR4 to ADA2CR7 registers (see Figure 13-2).

Table 13-2. Correspondence Between Analog Input Pin and ADA2CRn Register in 1-Buffer Mode

Analog Input Pin	ADA2CRn Register
ANI20	ADA2CR0
ANI21	ADA2CR1
ANI22	ADA2CR2
ANI23	ADA2CR3
ANI24	ADA2CR4
ANI25	ADA2CR5
ANI26Note	ADA2CR6
ANI27Note	ADA2CR7

Note V850E/IA4 only

Figure 13-2. Correspondence Between Analog Input Pin and ADA2CRn Register in 4-Buffer Mode



Note V850E/IA4 only

(5) A/D converter 2 status register (ADA2STR)

The ADA2STR register is an 8-bit register that is used to confirm the analog input pin (ANI2n) whose signal has been converted (V850E/IA3: n = 0 to 5, V850E/IA4: n = 0 to 7).

If selecting an ANI2n pin and the end of the previous A/D conversion conflict, or if there is such a possibility, it can be checked which ANI2n pin has finished being converted. Each time the A/D2 conversion end interrupt request signal (INTAD2) is generated, information on the ANI2n pin whose conversion has finished ended is updated.

This register is read-only in 8-bit units.

Reset sets this register to 00H.

After reset: 00H R Address: FFFFF246H

	7	6	5	4	3	2	1	0
ADA2STR	0	0	0	0	0	ADA2IT2	ADA2IT1	ADA2IT0

ADA2IT2	ADA2IT1	ADA2IT0	Analog input pin which has finished being A/D converted
0	0	0	ANI20
0	0	1	ANI21
0	1	0	ANI22
0	1	1	ANI23
1	0	0	ANI24
1	0	1	ANI25
1	1	0	ANI26Note
1	1	1	ANI27Note

Note V850E/IA4 only

(6) A/D2 conversion result registers 0 to 7, 0H to 7H (ADA2CR0 to ADA2CR7, ADA2CR0H to ADA2CR7H)

The ADA2CRn and ADA2CRnH registers are registers that hold the A/D conversion results. Each time A/D conversion ends, the conversion result is loaded from the conversion register (SAR) to this register. The conversion result is stored in the higher 8 bits or 10 bits of the ADA2CRn register at the resolution (8 bits or 10 bits) specified by the ADA2CTL1 register. The lower 7 bits or 5 bits of the ADA2CRn register are always 0 when read.

For the ADA2CRnH register, the higher 8 bits of the A/D conversion result are read.

These registers are read-only, in 16-bit or 8-bit units. During this register is read in 16-bit units, the ADA2CRn register is specified, and during higher 8 bits are read, the ADA2CRnH register is specified.

Reset sets these registers to 0000H.

After reset: 0000H R Address: ADA2CR0 FFFFF250H, ADA2CR1 FFFFF252H, ADA2CR2 FFFFF254H, ADA2CR3 FFFFF256H, ADA2CR4 FFFFF258H, ADA2CR5 FFFFF25AH, ADA2CR6 FFFFF25CH, ADA2CR7 FFFFF25EH

[With resolution of 10 bits]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADA2CRn (n = 0 to 7)	AD2n9	AD2n8	AD2n7	AD2n6	AD2n5	AD2n4	AD2n3	AD2n2	AD2n1	AD2n0	Undefined	0	0	0	0	0

[With resolution of 8 bits]

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADA2CRn (n = 0 to 7)	AD2n7	AD2n6	AD2n5	AD2n4	AD2n3	AD2n2	AD2n1	AD2n0	Undefined	0	0	0	0	0	0	0

After reset: 00H R Address: ADA2CR0H FFFFF251H, ADA2CR1H FFFFF253H, ADA2CR2H FFFFF255H, ADA2CR3H FFFFF257H, ADA2CR4H FFFFF259H, ADA2CR5H FFFFF25BH, ADA2CR6H FFFFF25DH, ADA2CR7H FFFFF25FH

[With resolution of 10 bits]

	7	6	5	4	3	2	1	0
ADA2CRnH (n = 0 to 7)	AD2n9	AD2n8	AD2n7	AD2n6	AD2n5	AD2n4	AD2n3	AD2n2

[With resolution of 8 bits]

	7	6	5	4	3	2	1	0
ADA2CRnH (n = 0 to 7)	AD2n7	AD2n6	AD2n5	AD2n4	AD2n3	AD2n2	AD2n1	AD2n0

The correspondence between the analog input pins and the ADA2CRn and ADA2CRnH registers is shown below.

Table 13-3. Correspondence Between Analog Input Pins and ADA2CRn and ADA2CRnH Registers

Analog Input Pin	A/D Conversion Result Register
ANI20	ADA2CR0, ADA2CR0H
ANI21	ADA2CR1, ADA2CR1H
ANI22	ADA2CR2, ADA2CR2H
ANI23	ADA2CR3, ADA2CR3H
ANI24	ADA2CR4, ADA2CR4H
ANI25	ADA2CR5, ADA2CR5H
ANI26[Note]	ADA2CR6, ADA2CR6H
ANI27[Note]	ADA2CR7, ADA2CR7H

Note V850E/IA4 only

The relationship between the analog voltage input to the analog input pin (ANI2n) and the A/D conversion result (of A/D2 conversion result register n (ADA2CRn)) is as follows (when 10-bit resolution setting):

$$SAR = INT\left(\frac{V_{IN}}{AV_{DD}} \times 1{,}024 + 0.5\right)$$

$$ADCR^{Note} = SAR \times 64$$

or,

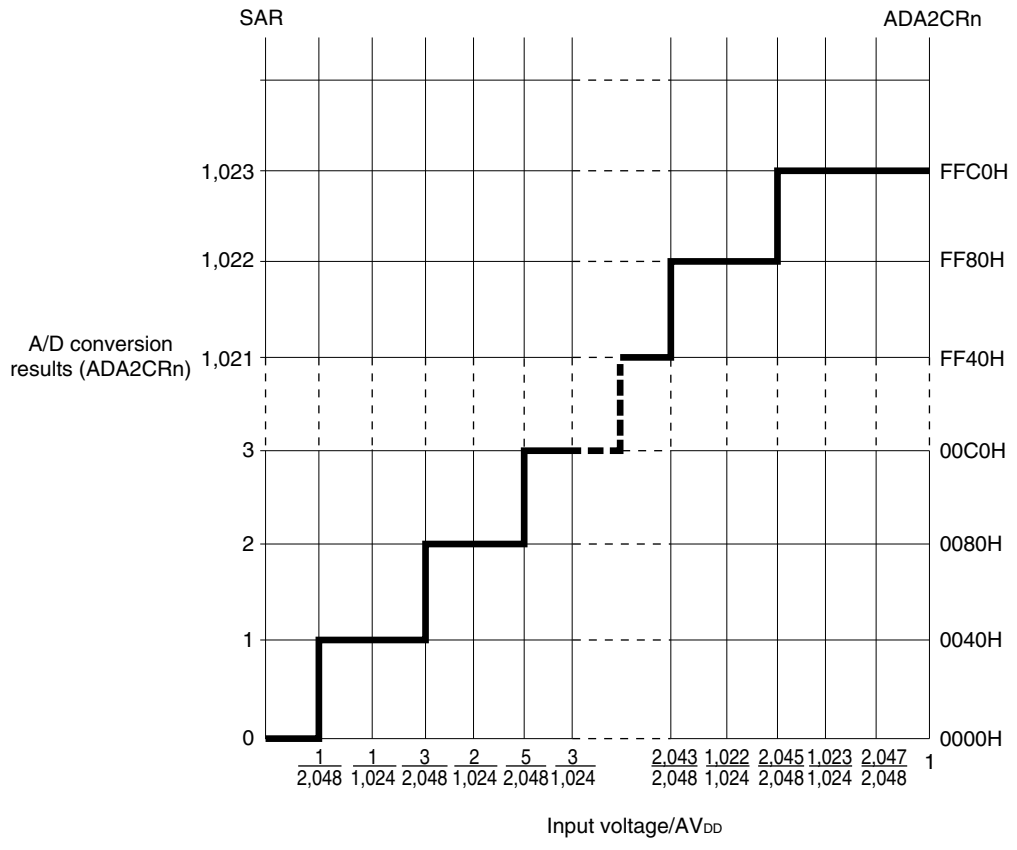
$$(SAR - 0.5) \times \frac{AV_{DD}}{1{,}024} \leq V_{IN} < (SAR + 0.5) \times \frac{AV_{DD}}{1{,}024}$$

INT( ): Function that returns the integer of the value in ( )
V_{IN}: Analog input voltage
AV_{DD}: AV_{DD} pin voltage
ADCR: Value of A/D2 conversion result register n (ADA2CRn)

Note The lower 5 bits of the ADA2CRn register are fixed to 0.

The relationship between the analog input voltage and the A/D conversion results is shown below.

Figure 13-3. Relationship Between Analog Input Voltage and A/D Conversion Results (When 10-Bit Resolution Setting)



Remark n = 0 to 7

13.4 Operation

13.4.1 Basic operation

Once started, A/D converter 2 performs conversion until it is stopped. Each time conversion ends, an A/D2 conversion end interrupt request signal (INTAD2) is generated. A/D conversion is performed in the following sequence.

(1) Turn on the analog power supply by setting the ADA2CTL0.ADA2PON bit to 1.

(2) Specify the clock for sampling the analog input and the resolution using the ADA2CTL1 register, select an analog input pin using the ADA2CTL2 register, and specify the buffer mode and operation mode using the ADA2CTL3 register.

(3) A/D conversion is started when the ADA2CTL0.ADA2CE bit is set to 5 μs after the ADA2CTL0.ADA2PON bit is set to 1. If A/D conversion is started by setting the ADA2CE bit to 1 before the lapse of 5 μs, conversion ends at the set time and the A/D2 conversion end interrupt request signal (INTAD2) is generated, but the accuracy of the first conversion result is not guaranteed.

(4) The voltage on the analog input pin (ANI2n)[Note 1] is sampled by the sampling clock set by the ADA2CTL1.ADA2FS1 and ADA2CTL1.ADA2FS0 bits for the number of times set by the ADA2N1 and ADA2N0 bits (n = 0 to 7). After sampling, the voltage is converted into the bit length of the specified resolution by a digital filter and the conversion result is stored in A/D2 conversion result register n (ADA2CRn). At the same time, the A/D2 conversion end interrupt request signal (INTAD2) is generated. After the first A/D conversion, A/D conversion of the same ANI2n pin is repeated, unless the ADA2CTL0.ADA2CE bit is cleared to 0 (conversion stopped). If data is written to the ADA2CTL2 register during A/D conversion (if the ANI2n pin is changed) [Note 2], the conversion under execution is aborted, and the conversion of the newly specified ANI2n pin is started. ANI2n pins whose conversion has finished can be identified by using the ADA2STR register.

(5) A/D conversion is stopped when the ADA2CTL0.ADA2CE bit is cleared to 0.

(6) To lower the operating current when not using A/D converter 2, clear the ADA2CTL0.ADA2PON bit to 0.

Notes 1. The ANI26 and ANI27 pins are only available in the V850E/IA4.

2. If data is written to the ADA2CTL2 register (if the ANI2n pin has been changed) during A/D conversion, the conversion result is not stored in A/D2 conversion result register n (ADA2CRn) (n = 0 to 7). The A/D conversion in progress will be initialized and conversion of the signal at the newly specified ANI2n pin will be started.

13.4.2 Buffer mode and operation mode

With A/D converter 2, continuous conversion and time difference conversion can be specified for one analog input pin (ANI2n)[Note] specified by the ADA2CTL2 register (n = 0 to 7).

The buffer mode and operation mode are specified by the ADA2CTL3 register.

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

Buffer Mode	Operation Mode	Analog Input Pin	A/D Conversion Result Register	Interrupt Occurrence Interval[Note 1]	Conversion Result Read Pending Period[Note 1]
1-buffer	Serial mode	ANI2n[Note 2]	ADA2CRn	128 μs	128 μs
	Parallel mode	ANI2n[Note 2]	ADA2CRn	32 μs	32 μs
4-buffer	Serial mode	One of ANI20 to ANI23	ADA2CR0 to ADA2CR3	512 μs	128 μs[Note 3]
		One of ANI24 to ANI27[Note 2]	ADA2CR4 to ADA2CR7		
	Parallel mode	One of ANI20 to ANI23	ADA2CR0 to ADA2CR3	128 μs	32 μs[Note 3]
		One of ANI24 to ANI27[Note 2]	ADA2CR4 to ADA2CR7		

Notes 1. With 8-bit resolution, fxx = 64 MHz

2. The ANI26 and ANI27 pins are only available in to the V850E/IA4.

3. Time until the ADA2CR0 or ADA2CR4 register is updated as a result of the first A/D conversion after generation of the A/D2 conversion end interrupt request signal (INTAD2)

(1) Buffer mode

The result of A/D conversion of one analog input pin (ANI2n)[Note] can be stored in two buffer modes: 1-buffer mode and 4-buffer mode (n = 0 to 7).

These buffer modes are selected by the ADA2CTL3.ADA2BS bit.

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

(a) 1-buffer mode

In this mode, the analog input pin (ANI2n)[Note] specified by the ADA2CTL2 register and A/D2 conversion result register n (ADA2CRn) correspond one to one (see **Table 13-2**) (n = 0 to 7).

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

(b) In 4-buffer mode

The voltage of the analog input pin (ANI2n)[Note] specified by the ADA2CTL2 register is A/D converted four times, and the results are stored in four A/D2 conversion result registers (ADA2CRn) (n = 0 to 7).

The conversion result of one of the ANI20 to ANI23 pins is stored in the ADA2CR0 to ADA2CR3 registers, and the result of one of the ANI24 to ANI27 pins[Note] is stored in the ADA2CR4 to ADA2CR7 registers (see **Figure 13-2**).

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

(2) Operation mode

Two operation modes are available: serial mode in which A/D conversion of one analog input pin (ANI2n)[Note] is continuously executed, and parallel mode in which conversion operations are executed in parallel with the time difference between operations set on starting the first conversion (n = 0 to 7).

These operation modes are selected by the ADA2CTL3.ADA2TS bit.

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

(a) Serial mode

In this mode, the next conversion is started when A/D conversion of the analog input pin (ANI2n)[Note] specified by the ADA2CTL2 register ends (n = 0 to 7).

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

- **In 1-buffer mode**
  Each time A/D conversion ends, the A/D2 conversion end interrupt request signal (INTAD2) is generated.
- **In 4-buffer mode**
  The A/D2 conversion end interrupt request signal (INTAD2) is generated when A/D conversion has ended four times.

(b) Parallel mode

In this mode, conversion of the analog input pin (ANI2n)[Note] specified by the ADA2CTL2 register is started with a time difference of 1/4 the conversion time (n = 0 to 7).

Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

- **In 1-buffer mode**
  Conversion is started at a time difference that is 1/4 of the conversion time specified by the ADA2CTL1 register, and the conversion result is stored in the ADA2CRn register corresponding to the ANI2n pin[Note] (n = 0 to 7). The A/D2 conversion end interrupt request signal (INTAD2) is generated each time the conversion result is stored in the ADA2CRn register (each 1/4 conversion time).

  Note The ANI26 and ANI27 pins are only available in the V850E/IA4.

- **In 4-buffer mode**
  Conversion is started at a time difference that is 1/4 of the conversion time specified by the ADA2CTL1 register, and the conversion result is stored in four ADA2CRn registers (n = 0 to 7). The A/D2 conversion end interrupt request signal (INTAD2) is generated after the conversion result is stored in the four ADA2CRn registers.

13.4.3 Operation timing

The operation timing in each buffer mode and operation mode is shown below.

(1) 1-buffer serial mode

A/D conversion of the analog input pin (ANI2n)[Note 1] specified by the ADA2CTL2 register is executed, and the conversion result is continuously stored in A/D2 conversion result register n (ADA2CRn) corresponding to the ANI2n pin (n = 0 to 7).

The ANI2n pin and the ADA2CRn register correspond one to one (see **Table 13-2**).

Each time A/D conversion ends, the A/D2 conversion end interrupt request signal (INTAD2) is generated. After conversion has ended once, it is repeated with the same ANI2n pin[Note 1], unless the ADA2CTL0.ADA2CE bit is cleared to 0 (conversion stopped).

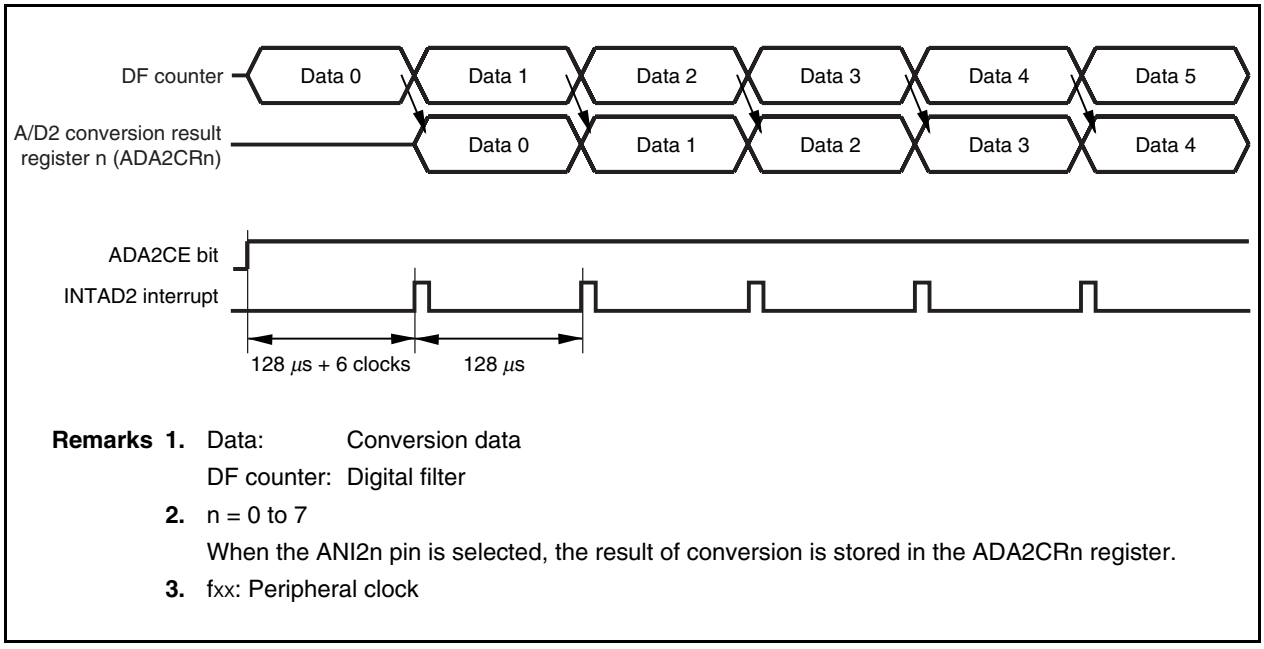
It is not required to set (1) the ADA2CTL0.ADA2CE bit to restart A/D conversion[Note 2].

Conversion can be stopped by clearing the ADA2CE bit to 0.

Notes 1. The ANI26 and ANI27 pins are only available in the V850E/IA4.

2. In 1-buffer serial mode, unless the ADA2CTL0.ADA2CE bit is cleared to 0, A/D conversion is not stopped. Therefore, the contents of the ADA2CRn register must be read before A/D conversion ends, or the register is overwritten (n = 0 to 7).

Figure 13-4. Example of Operation Timing in 1-Buffer Serial Mode (with 8-Bit Resolution, f_{XX} = 64 MHz)



Remarks 1. Data: Conversion data
DF counter: Digital filter

2. n = 0 to 7
When the ANI2n pin is selected, the result of conversion is stored in the ADA2CRn register.

3. f_{XX}: Peripheral clock

(2) 1-buffer parallel mode

A/D conversion of the analog input pin (ANI2n)[Note 1] specified by the ADA2CTL2 register is performed four times in parallel, with a time difference. The conversion results are continuously stored in A/D2 conversion result register n (ADA2CRn) corresponding to the ANI2n pin[Note 1] (n = 0 to 7).

The ANI2n pin[Note 1] and the ADA2CRn register correspond one to one (see **Table 13-2**).

A/D conversion is performed with a time difference that is 1/4 of the conversion time set by the ADA2CTL1 register.

After conversion, the A/D2 conversion end interrupt request signal (INTAD2) is generated each time the conversion result is stored in the ADA2CRn register corresponding to the ANI2n pin[Note 1] (each 1/4 conversion time). After the end of the first conversion, conversion is repeated unless the ADA2CTL0.ADA2CE bit is cleared to 0 (conversion stopped).

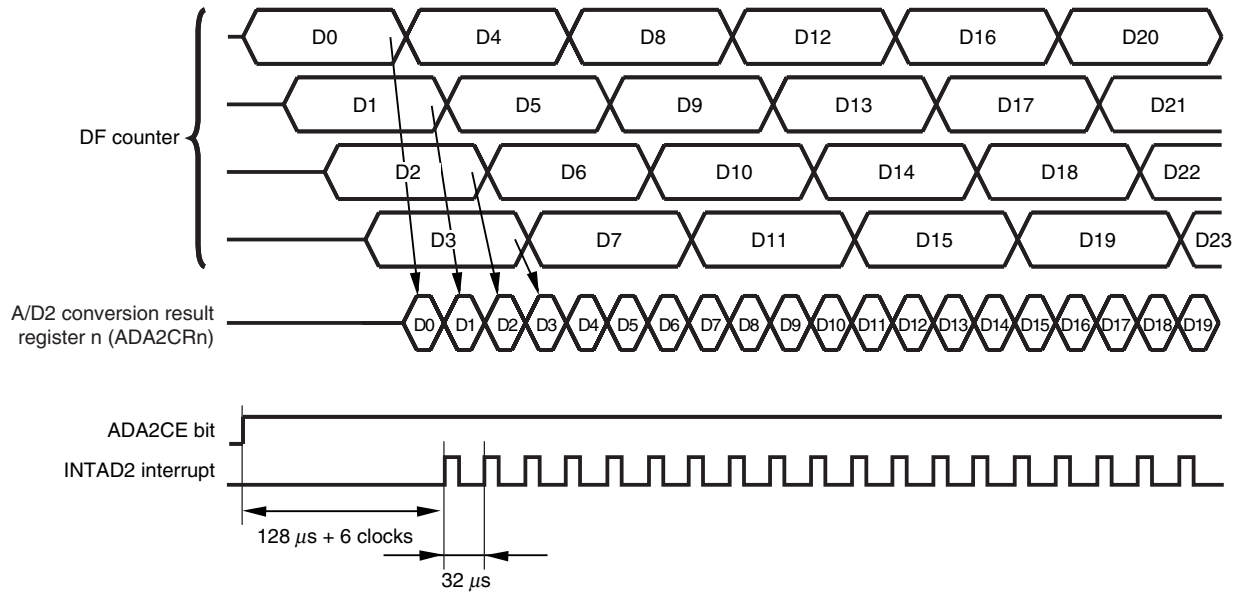
It is not required to set (1) the ADA2CTL0.ADA2CE bit to restart A/D conversion[Note 2].

A/D conversion can be stopped by clearing the ADA2CE bit to 0.

Notes 1. The ANI26 and ANI27 pins are only available in the V850E/IA4.

2. In 1-buffer parallel mode, unless the ADA2CTL0.ADA2CE bit is cleared to 0, A/D conversion is not stopped. Therefore, the contents of the ADA2CRn register must be read before A/D conversion ends, or the register is overwritten (n = 0 to 7).

Figure 13-5. Example of Operation Timing in 1-Buffer Parallel Mode (with 8-Bit Resolution, fxx = 64 MHz)



Remarks 1. D: Conversion data
DF counter: Digital filter

2. n = 0 to 7
When the ANI2n pin is selected, the result of conversion is stored in the ADA2CRn register.

3. fxx: Peripheral clock

(3) 4-buffer serial mode

A/D conversion of the analog input pin (ANI2n)[Note 1] specified by the ADA2CTL2 register is performed four times, and the four conversion results are stored in four A/D2 conversion result registers n (ADA2CRn) (n = 0 to 7).

The conversion results of one of the ANI20 to ANI23 pins are stored in the ADA2CR0 to ADA2CR3 registers, and those of one of the ANI24 to ANI27 pins[Note 1] are stored in the ADA2CR4 to ADA2CR7 registers (see **Figure 13-2**).

The A/D2 conversion end interrupt request signal (INTAD2) is generated when the A/D conversion has been ended four times. After end of the A/D conversion, it is repeated again with the same ANI2n pin[Note 1], unless the ADA2CTL0.ADA2CE bit is cleared to 0 (conversion stopped).

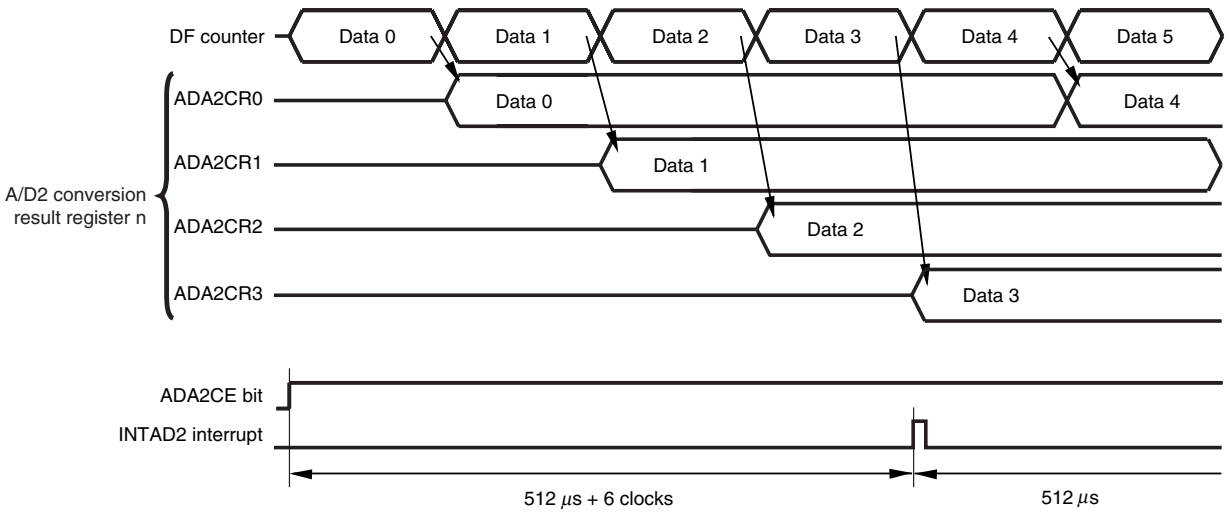
It is not required to set (1) the ADA2CTL0.ADA2CE bit to restart A/D conversion[Note 2].

A/D conversion can be stopped by clearing the ADA2CE bit to 0.

Notes 1. The ANI26 and ANI27 pins are only available in the V850E/IA4.

2. In 4-buffer serial mode, unless the ADA2CTL0.ADA2CE bit is cleared to 0, A/D conversion is not stopped. Therefore, the contents of the ADA2CRn register must be read before A/D conversion ends, or the register is overwritten (n = 0 to 7).

Figure 13-6. Example of Operation Timing in 4-Buffer Serial Mode (with 8-Bit Resolution, fxx = 64 MHz)



Remarks 1. Data: Conversion data
DF counter: Digital filter

2. n = 0 to 7
The A/D conversion results of one of the ANI20 to ANI23 pins are stored in the ADA2CR0 to ADA2CR3 registers.
The A/D conversion results of one of the ANI24 to ANI27 pins are stored in the ADA2CR4 to ADA2CR7 registers.
The above chart shows an example of the operation timing when one of the ANI20 to ANI23 pins is selected.

3. fxx: Peripheral clock

(4) 4-buffer parallel mode

A/D conversion of the analog input pin (ANI2n)[Note 1] specified by the ADA2CTL2 register is performed four times in parallel, with a time difference. The conversion results are continuously stored in four A/D2 conversion result registers n (ADA2CRn) corresponding to the ANI2n pin[Note 1] (n = 0 to 7).

The conversion results of one of the ANI20 to ANI23 pins are stored in the ADA2CR0 to ADA2CR3 registers, and those of one of the ANI24 to ANI27 pins[Note 1] are stored in the ADA2CR4 to ADA2CR7 registers (see **Figure 13-2**).

A/D conversion is performed with a time difference that is 1/4 of the conversion time set by the ADA2CTL1 register.

After the four the A/D conversion, the A/D2 conversion end interrupt request signal (INTAD2) is generated when the conversion results are stored in the four ADA2CRn registers. After end of A/D conversion, it is repeated again with the same ANI2n pin, unless the ADA2CTL0.ADA2CE bit is cleared to 0 (conversion stopped).

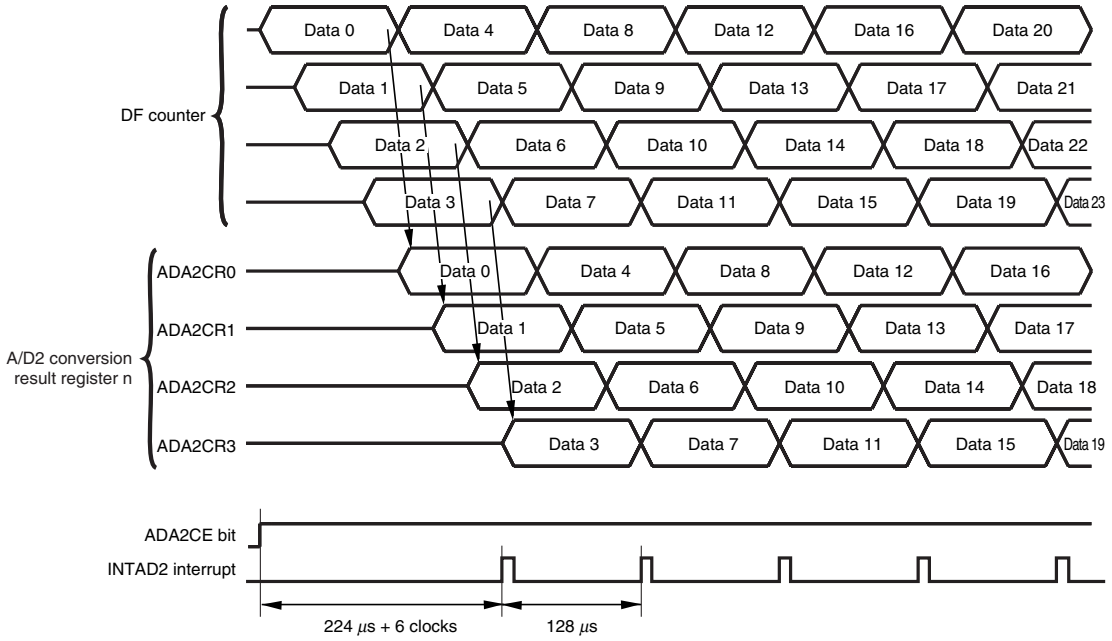
It is not required to set (1) the ADA2CTL0.ADA2CE bit to restart A/D conversion[Note 2].

A/D conversion can be stopped by clearing the ADA2CE bit to 0.

Notes 1. The ANI26 and ANI27 pins are only available in the V850E/IA4.

2. In 4-buffer parallel mode, unless the ADA2CTL0.ADA2CE bit is cleared to 0, A/D conversion is not stopped. Therefore, the contents of the ADA2CRn register must be read before A/D conversion ends, or the register is overwritten (n = 0 to 7).

Figure 13-7. Example of Operation Timing in 4-Buffer Parallel Mode (with 8-Bit Resolution, fxx = 64 MHz)



Remarks 1. Data: Conversion data
DF counter: Digital filter

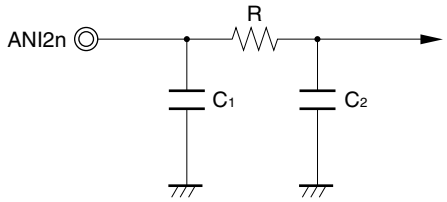
2. n = 0 to 7
The A/D conversion results of one of the ANI20 to ANI23 pins are stored in the ADA2CR0 to ADA2CR3 registers.
The A/D conversion results of one of the ANI24 to ANI27 pins are stored in the ADA2CR4 to ADA2CR7 registers.
The above chart shows an example of the operation timing when one of the ANI20 to ANI23 pins is selected.

3. fxx: Peripheral clock

13.5 Internal Equivalent Circuit

The following figure shows the equivalent circuit of the analog input block.

Figure 13-8. ANI2n Pin Internal Equivalent Circuit



R	C_1	C_2
30 kΩ	15 pF	0.25 pF

Remarks 1. The maximum values are shown (reference values).

2. V850E/IA3: n = 0 to 5
V850E/IA4: n = 0 to 7

13.6 How to Read A/D Converter Characteristics Table

For details, see **12.10 How to Read A/D Converter Characteristics Table**.

<R> ## 13.7 Cautions

13.7.1 Writing to the ADA2CTL1 and ADA2CTL3 registers during conversion

Writing the ADA2CTL1 and ADA2CTL3 registers (including writing the same value) during A/D conversion is prohibited. If data is written to the registers during A/D conversion, the conversion result of A/D2 conversion result register n (ADA2CRn) and the conversion operation cannot be guaranteed (n = 0 to 7). To write data to the ADA2CTL1 and ADA2CTL3 registers, be sure to stop the conversion once (ADA2CTL0.ADA2CE bit = 0), and then re-set the conversion conditions.

13.7.2 Conflict with timing of storing data in the conversion result register

Cautions when the timing of storing the A/D conversion result from the successive approximation register (SAR) to A/D2 conversion result register n (ADA2CRn) on A/D conversion end conflicts with the operations (1) to (4) are given below.

(1) Conflict with timing of reading the ADA2CRn register

If the timing of storing the A/D conversion result from the SAR register to the ADA2CRn register conflicts with the timing of reading the ADA2CRn register, an undefined value is read from the ADA2CRn register.
Nevertheless, the conversion result is correctly stored in the ADA2CRn register.

[Countermeasures]

Take either of the following measures to prevent an undefined value from being read from the ADA2CRn register.

- Immediately read the ADA2CRn register during the A/D2 conversion end interrupt (INTAD2) servicing. Even if that reading cannot immediately be performed in accordance with the interrupt priority or multiple interrupts, read the ADA2CRn register before the next storing timing.
- Start DMA with the A/D2 conversion end interrupt (INTAD2) and read the ADA2CRn register with DMA.
- Continuously read the ADA2CRn register twice and compare those values. If those values are the same, it is judged as the correct value. If those values differ, read the ADA2CRn register once more (for a third time), and that read value is handled as the correct value. This processing must be performed before the next conversion ends. However, if an A/D2 conversion end interrupt (INTAD2) occurs during the series of software operations due to the third reading, repeat reading twice consecutively to check if the values are the same.

(2) Conflicts with A/D conversion end timing

If the timing of storing the A/D conversion result from the SAR register to the ADA2CRn register conflicts with the timing of stopping the A/D conversion (ADA2CTL0.ADA2CE bit = 0), an undefined value is written to the ADA2CRn register.

[Countermeasures]

Take either of the following measures to prevent an undefined value from being written to the ADA2CRn register.

- Immediately set the ADA2CTL0 register to 80H or 00H during the A/D2 conversion end interrupt (INTAD2) servicing. Even if that writing cannot immediately be performed in accordance with the interrupt priority or multiple interrupts, stop the A/D conversion (ADA2CTL0 register = 80H or 00H) before the next storing timing.
- Start DMA with the A/D2 conversion end interrupt (INTAD2) and set the ADA2CTL0 register to 80H or 00H with DMA.

(3) Conflicts with ADA2STR register read timing

If the timing of storing the A/D conversion result from the SAR register to the ADA2CRn register conflicts with the timing of reading the ADA2STR register, an undefined value is read from the ADA2STR register.
The ADA2STR register value itself, though, is correctly updated.

[Countermeasures]

Take either of the following measures to prevent an undefined value from being read from the ADA2STR register.

- Immediately read the ADA2STR register during the A/D2 conversion end interrupt (INTAD2) servicing. Even if that reading cannot immediately be performed in accordance with the interrupt priority or multiple interrupts, read the ADA2STR register before the next storing timing.
- Start DMA with the A/D2 conversion end interrupt (INTAD2) and read the ADA2STR register with DMA.
- Continuously read the ADA2STR register twice and compare those values. If those values are the same, it is judged as the correct value. If those values differ, read the ADA2STR register once more (for a third time), and that read value is handled as the correct value. This processing must be performed before the next conversion ends. However, if an A/D2 conversion end interrupt (INTAD2) occurs during the series of software operations due to the third reading, repeat reading twice consecutively to check if the values are the same.

(4) Conflicts with switching the analog input pin

If the timing of storing the A/D conversion result from the SAR register to the ADA2CRn register conflicts with the timing of switching the analog input pin (rewriting the ADA2CTL2 register), an undefined value is written to the ADA2CRn register.
Nevertheless, the analog input pin is correctly switched (rewriting the ADA2CTL2 register is correctly performed).

[Countermeasures]

Switch the analog input pin after reading the ADA2CRn register to prevent an undefined value from being written to the ADA2CRn register.

14.1 Mode Switching Between UARTA1 and CSIB1

In the V850E/IA3 and V850E/IA4, UARTA1 and CSIB1 function alternately, and these pins cannot be used at the same time. To switch between UARTA1 and CSIB1, the PMC3 and PFC3 registers must be set in advance.

Caution The operations related to transmission and reception of UARTA1 or CSIB1 are not guaranteed if the mode is switched during transmission or reception. Be sure to disable the unit that is not used.

Figure 14-1. Mode Switch Settings of UARTA1 and CSIB1

After reset: 00H R/W Address: FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

After reset: 00H R/W Address: FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	0	0	PFC35	0	PFC33	PFC32	0	0

PMC34	Specification of alternate function of P34 pin
0	I/O port
1	$\overline{\text{SCKB1}}$ I/O

PMC33	PFC33	Specification of alternate function of P33 pin
0	×	I/O port
1	0	SOB1 output
1	1	TXDA1 output

PMC32	PFC32	Specification of alternate function of P32 pin
0	×	I/O port
1	0	SIB1 input
1	1	RXDA1 input

Remark x = don't care

14.2 Features

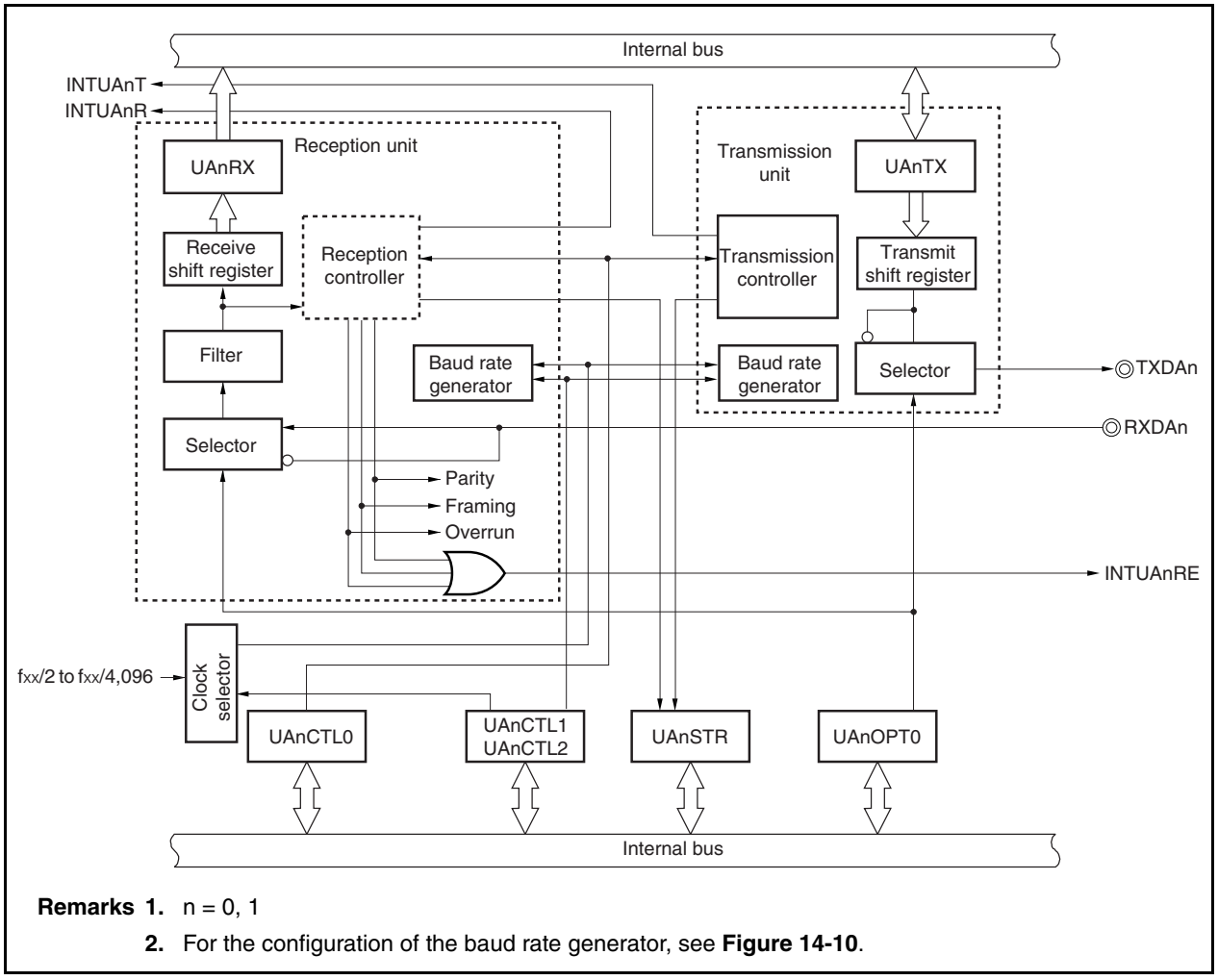
- Transfer rate: 300 bps to 1.25 Mbps (using peripheral clock (fxx) of 64 MHz and dedicated baud rate generator)
- Full-duplex communication: Internal UARTA receive data register n (UAnRX)
  Internal UARTA transmit data register n (UAnTX)
- 2-pin configuration: TXDAn: Transmit data output pin
  RXDAn: Receive data input pin
- Reception error output function
  - Parity error
  - Framing error
  - Overrun error
- Interrupt sources: 3
  - Reception error interrupt (INTUAnRE): This interrupt is generated by ORing the three types of reception errors
  - Reception end interrupt (INTUAnR): This interrupt occurs upon transfer of receive data from the shift register to the UAnRX register after serial transfer end, in the reception enabled status.
  - Transmission enable interrupt (INTUAnT): This interrupt occurs upon transfer of transmit data from the UAnTX register to the shift register in the transmission enabled status.
- Character length: 7, 8 bits
- Parity function: Odd, even, 0, none
- Transmission stop bit: 1, 2 bits
- On-chip dedicated baud rate generator
- MSB-/LSB-first transfer selectable
- Transmit/receive data inverted input/output possible

Remark n = 0, 1

14.3 Configuration

The block diagram of the UARTAn is shown below.

<R> **Figure 14-2. Block Diagram of UARTAn**



Remarks 1. n = 0, 1

2. For the configuration of the baud rate generator, see **Figure 14-10**.

UARTAn consists of the following hardware units.

Table 14-1. Configuration of UARTAn

Item	Configuration
Registers	UARTAn control register 0 (UAnCTL0) UARTAn control register 1 (UAnCTL1) UARTAn control register 2 (UAnCTL2) UARTAn option control register 0 (UAnOPT0) UARTAn status register (UAnSTR) UARTAn receive shift register UARTAn receive data register (UAnRX) UARTAn transmit shift register UARTAn transmit data register (UAnTX)

(1) UARTAn control register 0 (UAnCTL0)

The UAnCTL0 register is an 8-bit register used to specify the UARTAn operation.

(2) UARTAn control register 1 (UAnCTL1)

The UAnCTL1 register is an 8-bit register used to select the base clock (f_{UCLK}) for the UARTAn.

(3) UARTAn control register 2 (UAnCTL2)

The UAnCTL2 register is an 8-bit register used to control the baud rate for the UARTAn.

(4) UARTAn option control register 0 (UAnOPT0)

The UAnOPT0 register is an 8-bit register used to control serial transfer for the UARTAn.

<R> **(5) UARTAn status register (UAnSTR)**

The UAnSTRn register consists of flags indicating the error contents when a reception error occurs. Each one of the reception error flags is set (to 1) upon occurrence of a reception error.

(6) UARTAn receive shift register

This is a shift register used to convert the serial data input to the RXDAn pin into parallel data. Upon reception of 1 byte of data and detection of the stop bit, the receive data is transferred to the UAnRX register.

This register cannot be manipulated directly.

(7) UARTAn receive data register (UAnRX)

The UAnRX register is an 8-bit register that holds receive data. When 7 characters are received, 0 is stored in the highest bit (when data is received LSB first).

In the reception enabled status, receive data is transferred from the UARTAn receive shift register to the UAnRX register in synchronization with the completion of shift-in processing of 1 frame.

Transfer to the UAnRX register also causes the reception end interrupt request signal (INTUAnR) to be output.

(8) UARTAn transmit shift register

The UARTAn transmit shift register is a shift register used to convert the parallel data transferred from the UAnTX register into serial data.

When 1 byte of data is transferred from the UAnTX register, the UARTAn transmit shift register data is output from the TXDAn pin.

This register cannot be manipulated directly.

(9) UARTAn transmit data register (UAnTX)

The UAnTX register is an 8-bit transmit data buffer. Transmission starts when transmit data is written to the UAnTX register. When data can be written to the UAnTX register (when data of one frame is transferred from the UAnTX register to the UARTAn transmit shift register), the transmission enable interrupt request signal (INTUAnT) is generated.

14.4 Control Registers

(1) UARTAn control register 0 (UAnCTL0)

The UAnCTL0 register is an 8-bit register that controls the UARTAn serial transfer operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 10H.

After reset: 10H R/W Address: UA0CTL0 FFFFFA00H, UA1CTL0 FFFFFA10H

	<7>	<6>	<5>	<4>	3	2	1	0
UAnCTL0 (n = 0, 1)	UAnPWR	UAnTXE	UAnRXE	UAnDIR	UAnPS1	UAnPS0	UAnCL	UAnSL

UAnPWR	UARTAn operation control
0	Disable UARTAn operation (UARTAn reset asynchronously)
1	Enable UARTAn operation
The UARTAn operation is controlled by the UAnPWR bit. The TXDAn pin output is fixed to high level by clearing the UAnPWR bit to 0 (fixed to low level if UAnOPT0.UAnTDL bit = 1).	

UAnTXE	Transmission operation enable
0	Disable transmission operation
1	Enable transmission operation

<R>
- To start transmission, set the UAnPWR bit to 1 and then set the UAnTXE bit to 1.
- To initialize the transmission unit, clear the UAnTXE bit to 0, wait for two cycles of the base clock (f_{UCLK}), and then set the UAnTXE bit to 1 again. Otherwise, initialization may not be executed (for the base clock, see **14.7 (1) (a) Base clock**).
- When the operation is enabled (UAnPWR bit = 1), the transmission operation is enabled after two or more cycles of the base clock (f_{UCLK}) have elapsed since UAnTXE = 1.
- When the UAnPWR bit is cleared to 0, the status of the internal circuit becomes the same status as UAnTXE bit = 0 by the UAnPWR bit even if the UAnTXE bit is 1. The transmission operation is enabled when the UAnPWR bit is set to 1 again.

UAnRXE	Reception operation enable
0	Disable reception operation
1	Enable reception operation

<R>
- To start reception, set the UAnPWR bit to 1 and then set the UAnRXE bit to 1.
- To initialize the reception unit, clear the UAnRXE bit to 0, wait for two cycles of the base clock, and then set the UAnRXE bit to 1 again. Otherwise, initialization may not be executed (for the base clock, see **14.7 (1) (a) Base clock**).

<R>
- When the operation is enabled (UAnPWR bit = 1), the reception operation is enabled after two or more cycles of the base clock (f_{UCLK}) have elapsed since UAnRXE = 1. When a start bit is received before data reception is enabled, the start bit is ignored.
- When the UAnPWR bit is cleared to 0, the status of the internal circuit becomes the same status as UAnRXE bit = 0 by the UAnPWR bit even if the UAnRXE bit is 1. The reception operation is enabled when the UAnPWR bit is set to 1 again.

(2/2)

UAnDIR[Note]	Transfer direction selection
0	MSB-first transfer
1	LSB-first transfer

UAnPS1[Note]	UAnPS0[Note]	Parity selection during transmission	Parity selection during reception
0	0	No parity output	Reception with no parity
0	1	0 parity output	Reception with 0 parity
1	0	Odd parity output	Odd parity check
1	1	Even parity output	Even parity check
If “Reception with 0 parity” is selected during reception, a parity check is not performed. Therefore, since the UAnSTR.UAnPE bit is not set, no error interrupt due to a parity error is output.			

UAnCL[Note]	Specification of data character length of 1 frame of transmit/receive data
0	7 bits
1	8 bits

UAnSL[Note]	Specification of length of stop bit for transmit data
0	1 bit
1	2 bits
Only the first bit of the receive data stop bits is checked, regardless of the value of the UAnSL bit.	

Note This register can be rewritten only when the UAnPWR bit = 0 or the UAnTXE bit = UAnRXE bit = 0. However, setting any or all of the UAnPWR, UAnTXE, and UAnRXE bits to 1 at the same time is possible.

Remark For details of parity, see **14.6.6 Parity types and operations**.

(2) UARTAn control register 1 (UAnCTL1)

For details, see **14.7 (2) UARTAn control register 1 (UAnCTL1)**.

(3) UARTAn control register 2 (UAnCTL2)

For details, see **14.7 (3) UARTAn control register 2 (UAnCTL2)**.

(4) UARTAn option control register 0 (UAnOPT0)

The UAnOPT0 register is an 8-bit register that controls the serial transfer operation of UARTAn.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 14H.

After reset: 14H R/W Address: UA0OPT0 FFFFFA03H, UA1OPT0 FFFFFA13H

UAnOPT0 (n = 0, 1)

7	6	5	4	3	2	1	0
0	0	0	1	0	1	UAnTDL	UAnRDL

UAnTDL	Transmit data level bit
0	Normal output of transfer data
1	Inverted output of transfer data

- The output level of the TXDAn pin can be inverted using the UAnTDL bit.
- This register can be set when the UAnCTL0.UAnPWR bit = 0 or when the UAnCTL0.UAnTXE bit = 0.

UAnRDL	Receive data level bit
0	Normal input of transfer data
1	Inverted input of transfer data

- The input level of the RXDAn pin can be inverted using the UAnRDL bit.
- This register can be set when the UAnPWR bit = 0 or the UAnCTL0.UAnRXE bit = 0.
- When the UAnRDL bit is set to 1 (inverted input of receive data), reception must be enabled (UAnCTL0.UAnRXE bit = 1) after setting the data reception pin to the UART reception pin (RXDAn) when reception is started. When the pin mode is changed after reception is enabled, the start bit will be mistakenly detected if the pin level is high.

Caution Be sure to clear bits 3 and 5 to 7 to "0", and set bits 2 and 4 to "1". Operation with other settings is not guaranteed.

(5) UARTAn status register (UAnSTR)

The UAnSTR register is an 8-bit register that displays the UARTAn transfer status and reception error contents.

This register can be read or written in 8-bit or 1-bit units, but the UAnTSF bit is a read-only bit, while the UAnPE, UAnFE, and UAnOVE bits can both be read and written. However, these bits can only be cleared by writing 0; they cannot be set by writing 1 (even if 1 is written to them, the value is retained).

The initialization conditions are shown below.

Register/Bit	Initialization Conditions
UAnSTR register	• After reset • UAnCTL0.UAnPWR bit = 0
UAnTSF bit	• UAnCTL0.UAnTXE bit = 0
UAnPE, UAnFE, UAnOVE bits	• 0 write • UAnCTL0.UAnRXE bit = 0

Caution Be sure to read and check the error flags of the UAnPE, UAnFE, and UAnOVE bits, and clear the flags by writing "0" to them.

After reset: 00H R/W Address: UA0STR FFFFFA04H, UA1STR FFFFFA14H

	<7>	6	5	4	3	<2>	<1>	<0>
UAnSTR	UAnTSF	0	0	0	0	UAnPE	UAnFE	UAnOVE

(n = 0, 1)

UAnTSF	Transfer status flag
0	• When the UAnPWR bit = 0 or the UAnTXE bit = 0 has been set. • When, following transfer end, there was no next data transfer from UAnTX register
1	Write to UAnTX register
The UAnTSF bit is always 1 when performing continuous transmission. When initializing the transmission unit, check that the UAnTSF bit = 0 before performing initialization. The transmit data is not guaranteed when initialization is performed while the UAnTSF bit = 1.	

UAnPE	Parity error flag
0	• When the UAnPWR bit = 0 or the UAnRXE bit = 0 has been set. • When 0 has been written
1	When parity of data and parity bit do not match during reception.
• The operation of the UAnPE bit is controlled by the settings of the UAnCTL0.UAnPS1 and UAnCTL0.UAnPS0 bits. • The UAnPE bit can be read and written, but it can only be cleared by writing 0 to it, and it cannot be set by writing 1 to it. When 1 is written to this bit, the value is retained.	

UAnFE	Framing error flag
0	• When the UAnPWR bit = 0 or the UAnRXE bit = 0 has been set. • When 0 has been written
1	When no stop bit is detected during reception
• Only the first bit of the receive data stop bits is checked, regardless of the value of the UAnCTL0.UAnSL bit. • The UAnFE bit can be both read and written, but it can only be cleared by writing 0 to it, and it cannot be set by writing 1 to it. When 1 is written to this bit, the value is retained.	

UAnOVE	Overrun error flag
0	• When the UAnPWR bit = 0 or the UAnRXE bit = 0 has been set. • When 0 has been written
1	When receive data has been set to the UAnRX register and the next receive operation is ended before that receive data has been read.
• When an overrun error occurs, the data is discarded without the next receive data being written to the UAnRX register. • The UAnOVE bit can be both read and written, but it can only be cleared by writing 0 to it, and it cannot be set by writing 1 to it. When 1 is written to this bit, the value is retained.	

(6) UARTAn receive data register (UAnRX)

The UAnRX register is an 8-bit buffer register that stores parallel data converted by the UARTAn receive shift register.

The data stored in the UARTAn receive shift register is transferred to the UAnRX register upon end of reception of 1 byte of data. A reception end interrupt request signal (INTUAnR) is generated at this timing.

During LSB-first reception when the data length has been specified as 7 bits, the receive data is transferred to bits 6 to 0 of the UAnRX register and the MSB always becomes 0. During MSB-first reception, the receive data is transferred to bits 7 to 1 of the UAnRX register and the LSB always becomes 0.

When an overrun error occurs (UAnSTR.UAnOVE bit = 1), the receive data at this time is not transferred to the UAnRX register and is discarded.

This register is read-only, in 8-bit units.

In addition to reset, the UAnRX register can be set to FFH by clearing the UAnCTL0.UAnPWR bit to 0.

After reset: FFH R Address: UA0RX FFFFFA06H, UA1RX FFFFFA16H

	7	6	5	4	3	2	1	0
UAnRX								

(n = 0, 1)

(7) UARTAn transmit data register (UAnTX)

The UAnTX register is an 8-bit register used to set transmit data.

Transmission starts when transmit data is written to the UAnTX register in the transmission enabled status (UAnCTL0.UAnTXE bit = 1). Upon end of the transfer of the data of the UAnTX register to the UARTAn transmit shift register, the transmission enable interrupt request signal (INTUAnT) is generated.

This register can be read or written in 8-bit units.

Reset sets this register to FFH.

After reset: FFH R/W Address: UA0TX FFFFFA07H, UA1TX FFFFFA17H

	7	6	5	4	3	2	1	0
UAnTX								

(n = 0, 1)

14.5 Interrupt Request Signals

The following three interrupt request signals are generated from UARTAn.

- Reception error interrupt request signal (INTUAnRE)
- Reception end interrupt request signal (INTUAnR)
- Transmission enable interrupt request signal (INTUAnT)

Among these three interrupt signals, the reception error interrupt signal has the highest default priority, and the reception end interrupt request signal and transmission enable interrupt request signal follow in this order.

Table 14-2. Interrupts and Their Default Priorities

Interrupt	Priority
Reception error	High
Reception end	⇕
Transmission enable	Low

(1) Reception error interrupt request signal (INTUAnRE)

A reception error interrupt request signal is generated while reception is enabled by ORing the three types of reception errors (parity error, framing error, and overrun error) explained in the UAnSTR register section.

(2) Reception end interrupt request signal (INTUAnR)

A reception end interrupt request signal is output when data is shifted into the UARTAn receive shift register and transferred to the UAnRX register in the reception enabled status.

No reception end interrupt request signal is generated in the reception disabled status.

(3) Transmission enable interrupt request signal (INTUAnT)

If transmit data is transferred from the UAnTX register to the UARTAn transmit shift register with transmission enabled, the transmission enable interrupt request signal is generated.

14.6 Operation

14.6.1 Data format

Full-duplex serial data reception and transmission is performed.

As shown in Figure 14-3, one data frame of transmit/receive data consists of a start bit, character bits, parity bit, and stop bit(s).

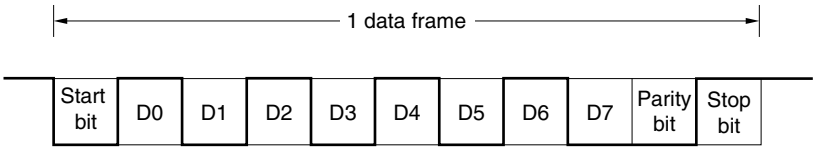
Specification of the character bit length within 1 data frame, parity selection, specification of the stop bit length, and specification of MSB-/LSB-first transfer are performed using the UAnCTL0 register.

Moreover, control of UARTAn output/inverted output for the TXDAn bit is performed using the UAnOPT0.UAnTDL bit.

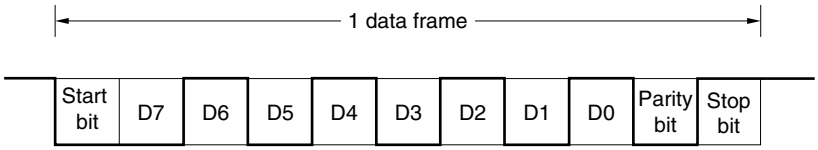
- Start bit ..... 1 bit
- Character bits..... 7 bits/8 bits
- Parity bit ..... Even parity/odd parity/0 parity/no parity
- Stop bit..... 1 bit/2 bits

Figure 14-3. UARTA Transmit/Receive Data Format

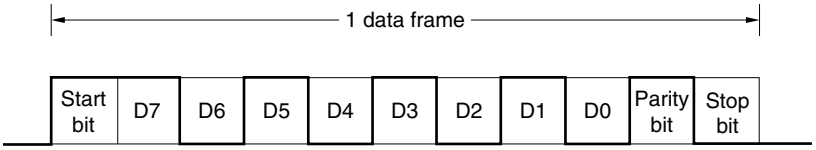
(a) 8-bit data length, LSB first, even parity, 1 stop bit, transfer data: 55H



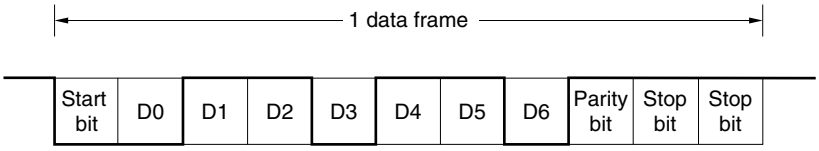
(b) 8-bit data length, MSB first, even parity, 1 stop bit, transfer data: 55H



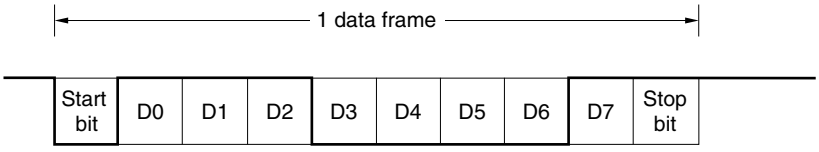
(c) 8-bit data length, MSB first, even parity, 1 stop bit, transfer data: 55H, TXDAn inversion



(d) 7-bit data length, LSB first, odd parity, 2 stop bits, transfer data: 36H



(e) 8-bit data length, LSB first, no parity, 1 stop bit, transfer data: 87H



14.6.2 UART transmission

A high level is output to the TXDAn pin by setting the UAnCTL0.UAnPWR bit to 1.

Next, the transmission enabled status is set by setting the UAnCTL0.UAnTXE bit to 1, and transmission is started by writing transmit data to the UAnTX register. The start bit, parity bit, and stop bit are automatically added.

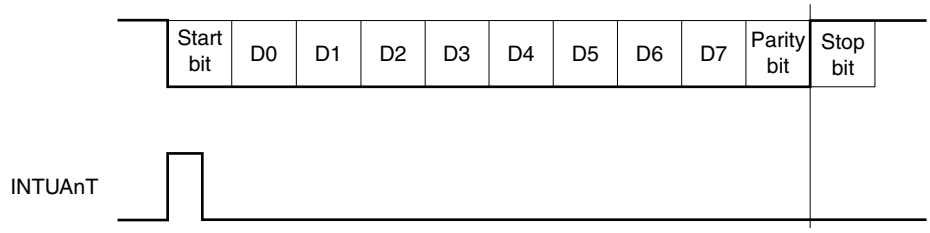
Since the CTS (transmit enable signal) input pin is not provided in UARTAn, use a port to check that reception is enabled at the transmit destination.

The data in the UAnTX register is transferred to the UARTAn transmit shift register upon the start of the transmit operation.

A transmission enable interrupt request signal (INTUAnT) is generated upon end of transmission of the data of the UAnTX register to the UARTAn transmit shift register, and thereafter the contents of the UARTAn transmit shift register are output to the TXDAn pin.

Write of the next transmit data to the UAnTX register is enabled by generating the INTUAnT signal.

Figure 14-4. UART Transmission



Remarks 1. LSB first

2. n = 0, 1

14.6.3 Continuous transmission procedure

UARTAn can write the next transmit data to the UAnTX register when the UARTAn transmit shift register starts the shift operation. The transmit timing of the UARTAn transmit shift register can be judged from the transmission enable interrupt request signal (INTUAnT). An efficient communication rate is realized by writing the data to be transmitted next to the UAnTX register during transfer.

Caution During continuous transmission execution, perform initialization after checking that the UAnSTR.UAnTSF bit is 0. The transmit data cannot be guaranteed when initialization is performed while the UAnTSF bit is 1.

Remark n = 0, 1

Figure 14-5. Continuous Transmission Processing Flow

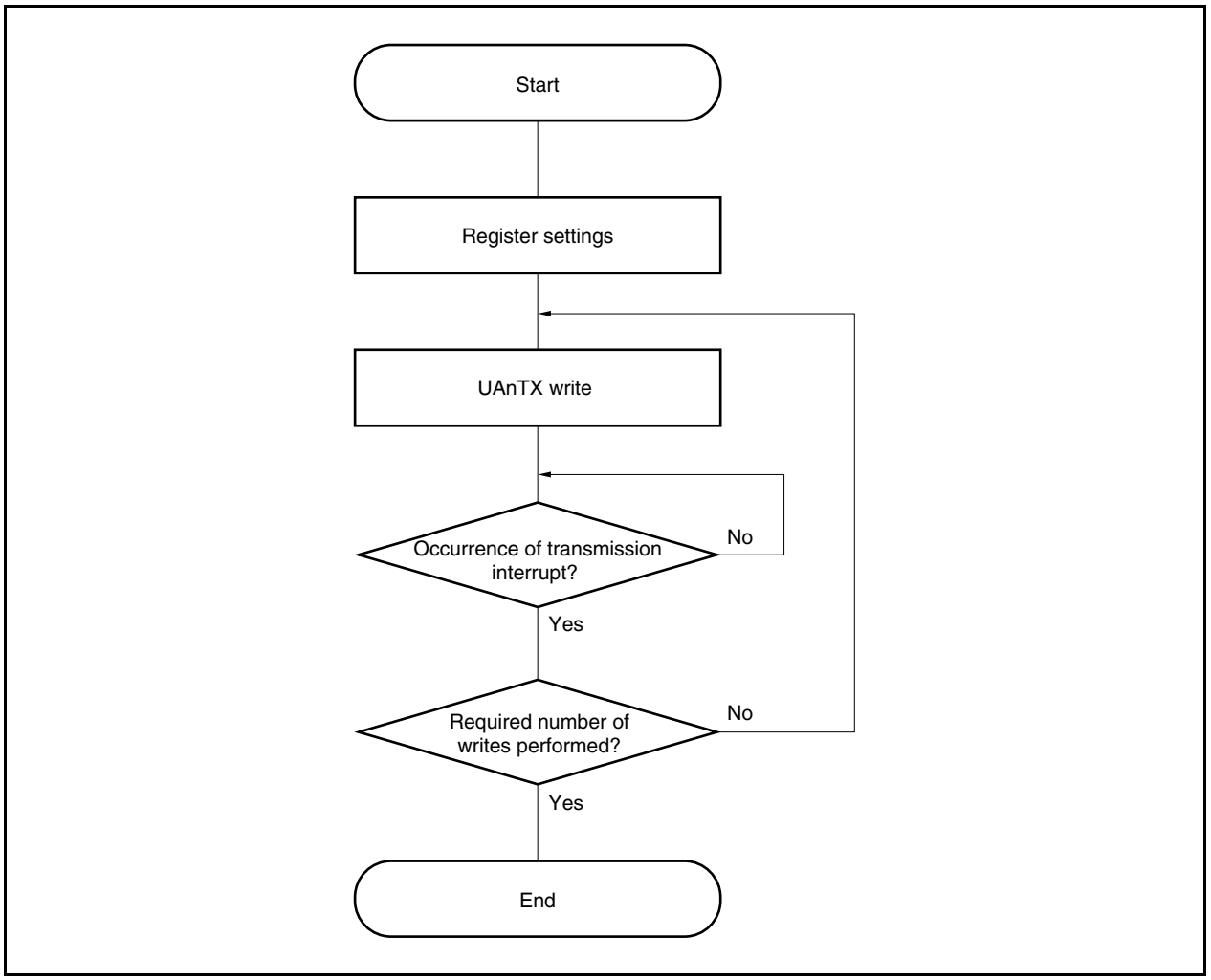
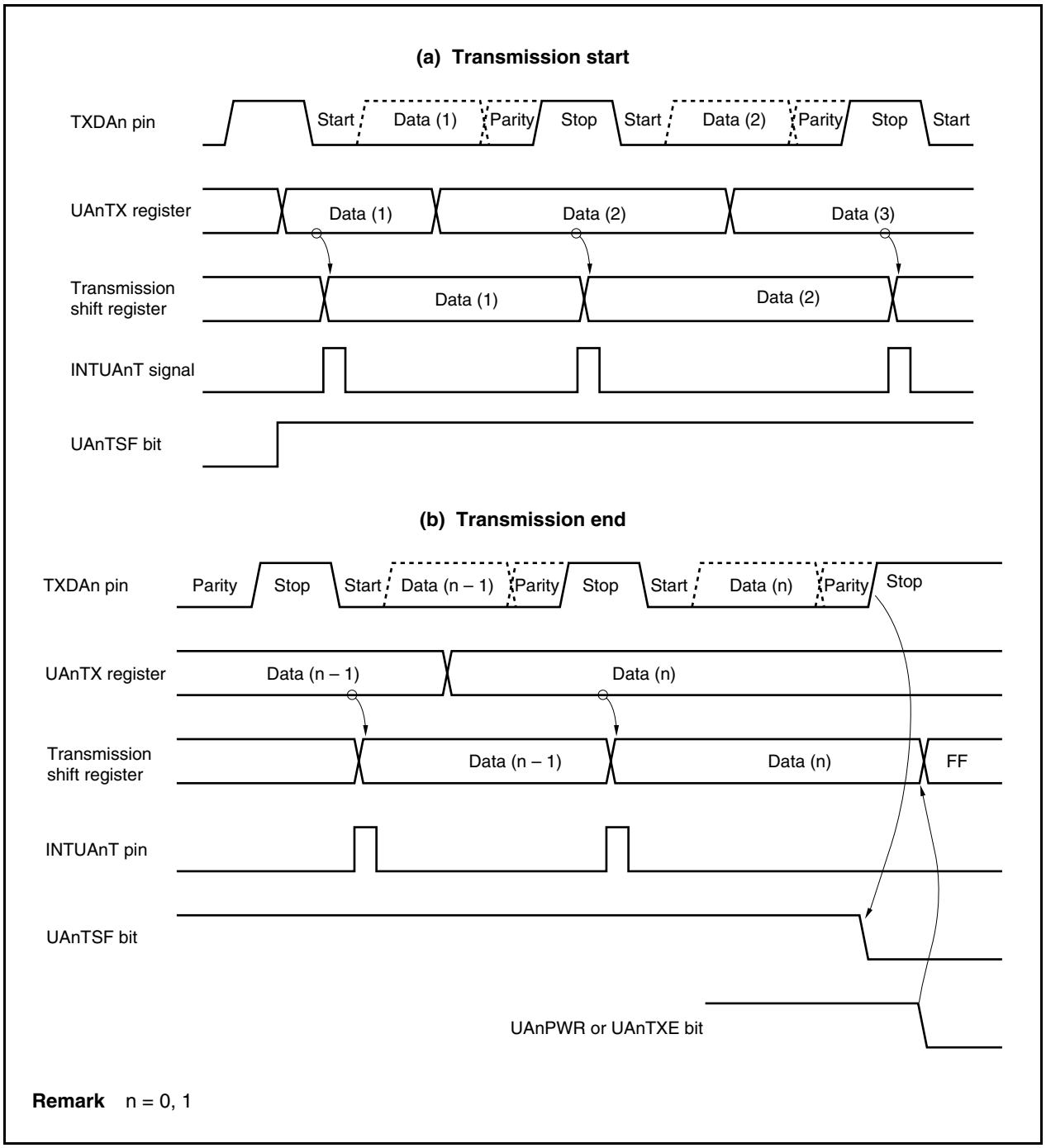


Figure 14-6. Continuous Transmission Operation Timing



Remark n = 0, 1

14.6.4 UART reception

The reception wait status is set by setting the UAnCTL0.UAnPWR bit to 1 and then setting the UAnCTL0.UAnRXE bit to 1. In the reception wait status, the RXDAn pin is monitored and start bit detection is performed.

Start bit detection is performed using a two-step detection routine.

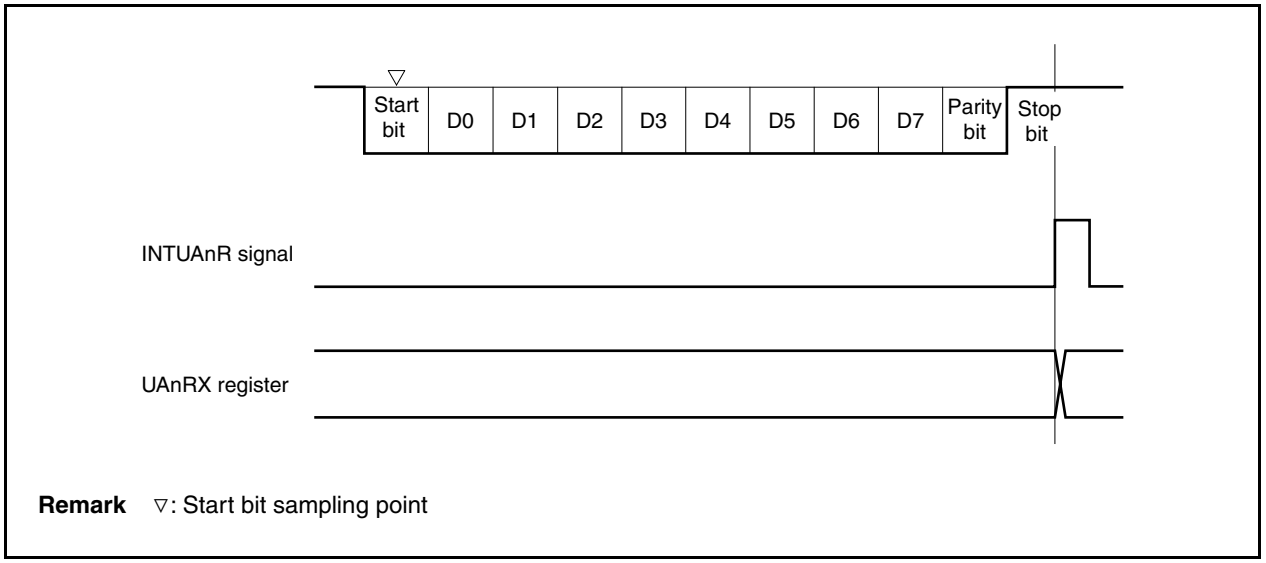
First the falling edge of the RXDAn pin is detected and sampling is started at the falling edge. The start bit is recognized if the RXDAn pin is low level at the start bit sampling point. After a start bit has been recognized, the receive operation starts, and serial data is saved to the UARTAn receive shift register according to the set baud rate.

When the reception end interrupt request signal (INTUAnR) is output upon reception of the stop bit, the data of the UARTAn receive shift register is written to the UAnRX register. However, if an overrun error occurs (UAnSTR.UAnOVE bit = 1), the receive data at this time is not written to the UAnRX register and is discarded.

Even if a parity error (UAnSTR.UAnPE bit= 1) or a framing error (UAnSTR.UAnFE bit = 1) occurs during reception, reception continues until the reception position of the first stop bit, and the INTUAnRE signal is output following reception end.

Remark n = 0, 1

Figure 14-7. UART Reception



Remark ▽: Start bit sampling point

Cautions 1. Be sure to read the UAnRX register even when a reception error occurs. If the UAnRX register is not read, an overrun error occurs during reception of the next data, and reception errors continue occurring indefinitely.

2. The operation during reception is performed assuming that there is only one stop bit. A second stop bit is ignored.

3. When reception is completed, read the UAnRX register after the reception end interrupt request signal (INTUAnR) has been generated, and clear the UAnPWR or UAnRXE bit to 0. If the UAnPWR or UAnRXE bit is cleared to 0 before the INTUAnR signal is generated, the read value of the UAnRX register cannot be guaranteed.

4. If receive end processing (INTUAnR signal generation) of UARTAn and the UAnPWR bit = 0 or UAnRXE bit = 0 conflict, the INTUAnR signal may be generated in spite of these being no data stored in the UAnRX register. To end reception without waiting INTUAnR signal generation, be sure to clear (0) the interrupt request flag (UAnRIC.UAnRIF), after setting (1) the interrupt mask flag (UAnRIC.UAnRMK) and then set (1) the UAnPWR bit = 0 or UAnRXE bit = 0.

14.6.5 Reception errors

Errors during a receive operation are of three types: parity errors, framing errors, and overrun errors. Data reception result error flags are set in the UAnSTR register and a reception error interrupt request signal (INTUAnRE) is output when an error occurs.

It is possible to ascertain which error occurred during reception by reading the contents of the UAnSTR register.

Clear the reception error flag by writing 0 to it after reading it.

Caution The reception end interrupt request signal (INTUAnR) and reception error interrupt request signal (INTUAnRE) are not generated simultaneously. The INTUAnR signal is generated when a reception ends normally. The INTUAnRE signal is generated and the INTUAnR signal is not generated when a reception error occurs.

Remark n = 0, 1

- Reception error causes

Error Flag	Reception Error	Cause
UAnPE	Parity error	Received parity bit does not match the setting
UAnFE	Framing error	Stop bit not detected
UAnOVE	Overrun error	Reception of next data ended before data was read from UAnRX register

14.6.6 Parity types and operations

The parity bit is used to detect bit errors in the communication data. Normally the same parity is used on the transmission side and the reception side.

In the case of even parity and odd parity, it is possible to detect odd-count bit errors. In the case of 0 parity and no parity, errors cannot be detected.

(a) Even parity

(i) During transmission

The number of bits whose value is “1” among the transmit data, including the parity bit, is controlled so as to be an even number. The parity bit values are as follows.

- Odd number of bits whose value is “1” among transmit data: 1
- Even number of bits whose value is “1” among transmit data: 0

(ii) During reception

The number of bits whose value is “1” among the reception data, including the parity bit, is counted, and if it is an odd number, a parity error is output.

(b) Odd parity

(i) During transmission

Opposite to even parity, the number of bits whose value is “1” among the transmit data, including the parity bit, is controlled so that it is an odd number. The parity bit values are as follows.

- Odd number of bits whose value is “1” among transmit data: 0
- Even number of bits whose value is “1” among transmit data: 1

(ii) During reception

The number of bits whose value is “1” among the receive data, including the parity bit, is counted, and if it is an even number, a parity error is output.

(c) 0 parity

During transmission, the parity bit is always made 0, regardless of the transmit data.

During reception, parity bit check is not performed. Therefore, no parity error occurs, regardless of whether the parity bit is 0 or 1.

(d) No parity

No parity bit is added to the transmit data.

Reception is performed assuming that there is no parity bit. No parity error occurs since there is no parity bit.

14.6.7 Receive data noise filter

This filter samples the RXDAn pin using the base clock (f_{UCLK}) of the prescaler output.

When the same sampling value is read twice, the match detector output changes and the RXDAn signal is sampled as the input data. Therefore, data not exceeding 2 clock width is judged to be noise and is not delivered to the internal circuit (see **Figure 14-9**). See **14.7 (1) (a) Base clock** regarding the base clock.

Moreover, since the circuit is as shown in Figure 14-8, the processing that goes on within the receive operation is delayed by 3 clocks in relation to the external signal status.

Remark n = 0, 1

Figure 14-8. Noise Filter Circuit

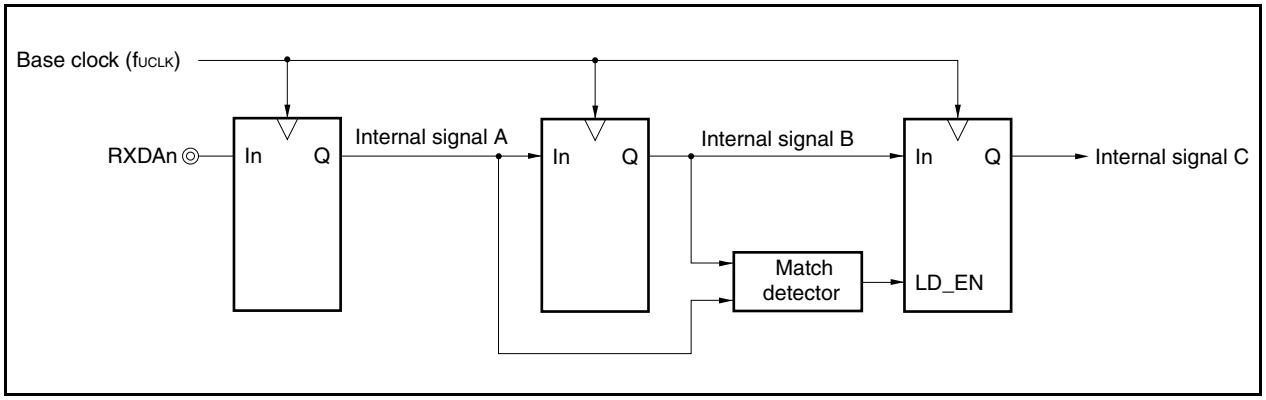
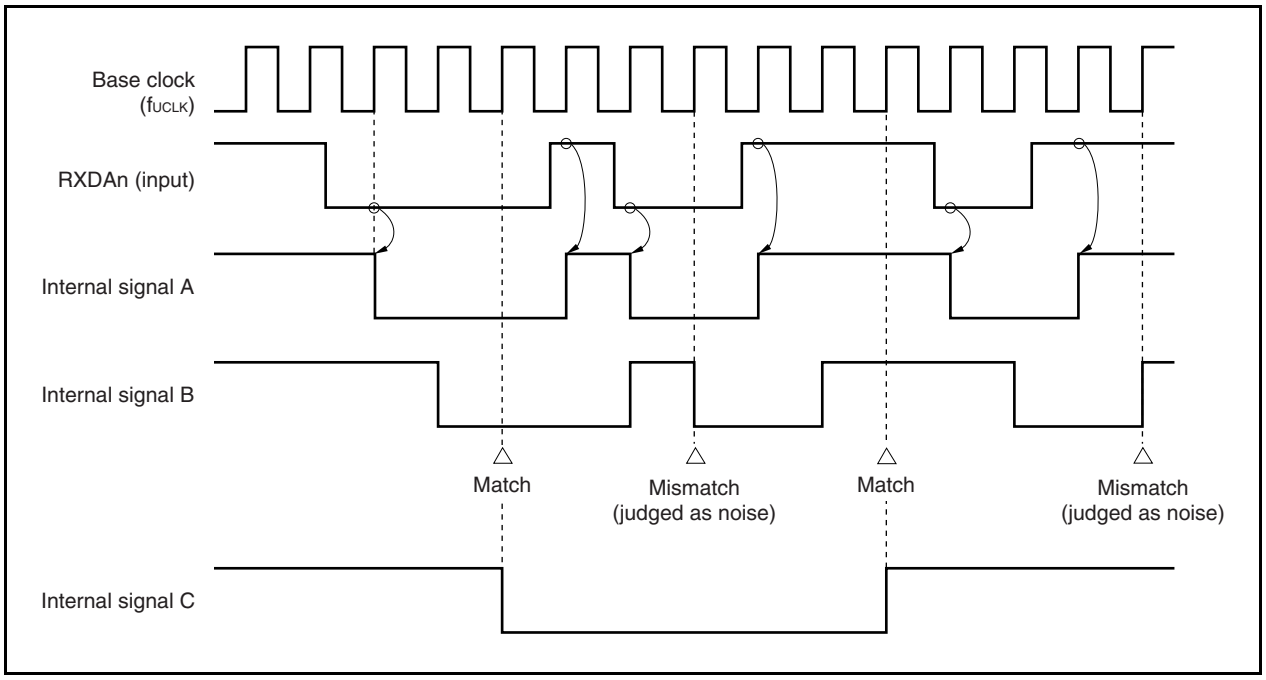


Figure 14-9. Timing of RXDAn Signal Judged as Noise



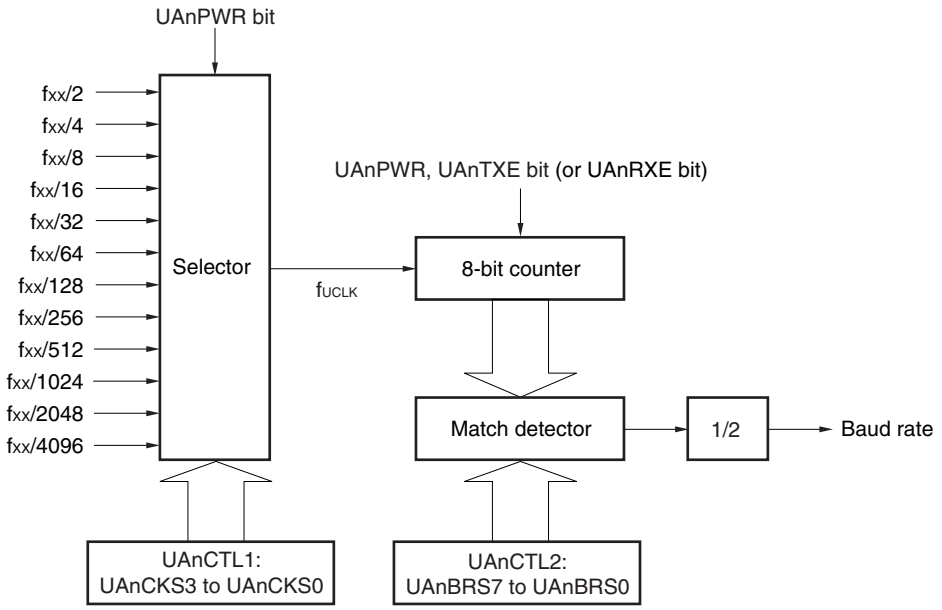
14.7 Dedicated Baud Rate Generator

The dedicated baud rate generator consists of a source clock selector block and an 8-bit programmable counter, and generates a serial clock during transmission and reception with UARTAn. Regarding the serial clock, a dedicated baud rate generator output can be selected for each channel.

There is an 8-bit counter for transmission and another one for reception.

(1) Baud rate generator configuration

Figure 14-10. Configuration of Baud Rate Generator



Caution If the CPU clock (f_{CPU}) is slower than f_{UCLK}, UARTAn cannot be used.

Remarks 1. n = 0, 1

2. f_{XX}: Peripheral clock frequency

(a) Base clock

When the UAnCTL0.UAnPWR bit is 1, the clock selected by the UAnCTL1.UAnCKS3 to UAnCTL1.UAnCKS0 bits is supplied to the 8-bit counter. This clock is called the base clock (f_{UCLK}). When the UAnPWR bit = 0, f_{UCLK} is fixed to the low level.

(b) Serial clock generation

A serial clock can be generated by setting the UAnCTL1 register and the UAnCTL2 register.

The base clock (f_{UCLK}) is selected by the UAnCTL1.UAnCKS3 to UAnCTL1.UAnCKS0 bits.

The frequency division value for the 8-bit counter can be set using the UAnCTL2.UAnBRS7 to UAnCTL2.UAnBRS0 bits.

(2) UARTAn control register 1 (UAnCTL1)

The UAnCTL1 register is an 8-bit register that selects the UARTAn base clock.

<R> This register can be read or written in 8-bit unit.

Reset sets this register to 00H.

Caution Clear the UAnCTL0.UAnPWR bit to 0 before rewriting the UAnCTL1 register.

After reset: 00H R/W Address: UA0CTL1 FFFFFA01H, UA1CTL1 FFFFFA11H

	7	6	5	4	3	2	1	0
UAnCTL1	0	0	0	0	UAnCKS3	UAnCKS2	UAnCKS1	UAnCKS0

(n = 0, 1)

UAnCKS3	UAnCKS2	UAnCKS1	UAnCKS0	Base clock (f_{UCLK}) selection
0	0	0	0	fxx/2
0	0	0	1	fxx/4
0	0	1	0	fxx/8
0	0	1	1	fxx/16
0	1	0	0	fxx/32
0	1	0	1	fxx/64
0	1	1	0	fxx/128
0	1	1	1	fxx/256
1	0	0	0	fxx/512
1	0	0	1	fxx/1,024
1	0	1	0	fxx/2,048
1	0	1	1	fxx/4,096
Other than above				Setting prohibited

Remark fxx: Peripheral clock frequency

(3) UARTAn control register 2 (UAnCTL2)

The UAnCTL2 register is an 8-bit register that selects the baud rate (serial transfer speed) clock of UARTAn.

This register can be read or written in 8-bit units.

Reset sets this register to FFH.

Caution Clear the UAnCTL0.UAnPWR bit to 0 or clear the UAnTXE and UAnRXE bits to 00 before rewriting the UAnCTL2 register.

After reset: FFH R/W Address: UA0CTL2 FFFFFA02H, UA1CTL2 FFFFFA12H

	7	6	5	4	3	2	1	0
UAnCTL2 (n = 0, 1)	UAnBRS7	UAnBRS6	UAnBRS5	UAnBRS4	UAnBRS3	UAnBRS2	UAnBRS1	UAnBRS0

UAn BRS7	UAn BRS6	UAn BRS5	UAn BRS4	UAn BRS3	UAn BRS2	UAn BRS1	UAn BRS0	Default (k)	Serial clock
0	0	0	0	0	0	×	×	×	Setting prohibited
0	0	0	0	0	1	0	0	4	$f_{UCLK}/4$
0	0	0	0	0	1	0	1	5	$f_{UCLK}/5$
0	0	0	0	0	1	1	0	6	$f_{UCLK}/6$
:	:	:	:	:	:	:	:	:	:
1	1	1	1	1	1	0	0	252	$f_{UCLK}/252$
1	1	1	1	1	1	0	1	253	$f_{UCLK}/253$
1	1	1	1	1	1	1	0	254	$f_{UCLK}/254$
1	1	1	1	1	1	1	1	255	$f_{UCLK}/255$

Remark f_{UCLK}: Frequency of base clock selected by the UAnCTL1.UAnCKS3 to UAnCTL1.UAnCKS0 bits

(4) Baud rate

The baud rate is obtained by the following equation.

$$\text{Baud rate} = \frac{f_{UCLK}}{2 \times k} \text{ [bps]}$$

f_{UCLK}: Frequency of base clock selected by the UAnCTL1.UAnCKS3 to UAnCTL1.UAnCKS0 bits

k: Value set using the UAnCTL2.UAnBRS7 to UAnCTL2.UAnBRS0 bits (k = 4, 5, 6, ..., 255)

(5) Baud rate error

The baud rate error is obtained by the following equation.

$$\text{Error (\%)} = \left(\frac{\text{Actual baud rate (baud rate with error)}}{\text{Target baud rate (correct baud rate)}} - 1 \right) \times 100 \text{ [\%]}$$

Cautions 1. The baud rate error during transmission must be within the error tolerance on the receiving side.

2. The baud rate error during reception must satisfy the range indicated in section (7) Allowable baud rate range during reception.

Example Peripheral clock frequency = 32 MHz = 32,000,000 Hz

Setting value of UAnCTL1.UAnCKS3 to UAnCTL1.UAnCKS0 bits = 0000B (f_{UCLK} = 16,000,000 Hz)

Setting value of UAnCTL2.UAnBRS7 to UAnCTL2.UAnBRS0 bits = 00110100B (k = 52)

Target baud rate = 153,600

Baud rate = 16,000,000/ (2 × 52) = 153,846 [bps]

Error = (153,846/153,600 − 1) × 100

= 0.160 [%]

(6) Baud rate setting example

Table 14-3. Baud Rate Generator Setting Data

Baud Rate (bps)	f_{XX} = 64 MHz			f_{XX} = 32 MHz		
	UAnCTL1	UAnCTL2	ERR (%)	UAnCTL1	UAnCTL2	ERR (%)
300	08H	D0H	0.16	07H	D0H	0.16
600	07H	D0H	0.16	06H	D0H	0.16
1,200	06H	D0H	0.16	05H	D0H	0.16
2,400	05H	D0H	0.16	04H	D0H	0.16
4,800	04H	D0H	0.16	03H	D0H	0.16
9,600	03H	D0H	0.16	02H	D0H	0.16
19,200	02H	D0H	0.16	01H	D0H	0.16
31,250	02H	80H	0	00H	80H	0
38,400	01H	D0H	0.16	00H	D0H	0.16
76,800	00H	D0H	0.16	00H	68H	0.16
153,600	00H	68H	0.16	00H	34H	0.16
312,500	00H	33H	0.39	00H	1AH	−1.54
625,000	00H	1AH	−1.54	00H	0DH	−1.54
1,250,000	00H	0DH	−1.54	00H	06H	6.67

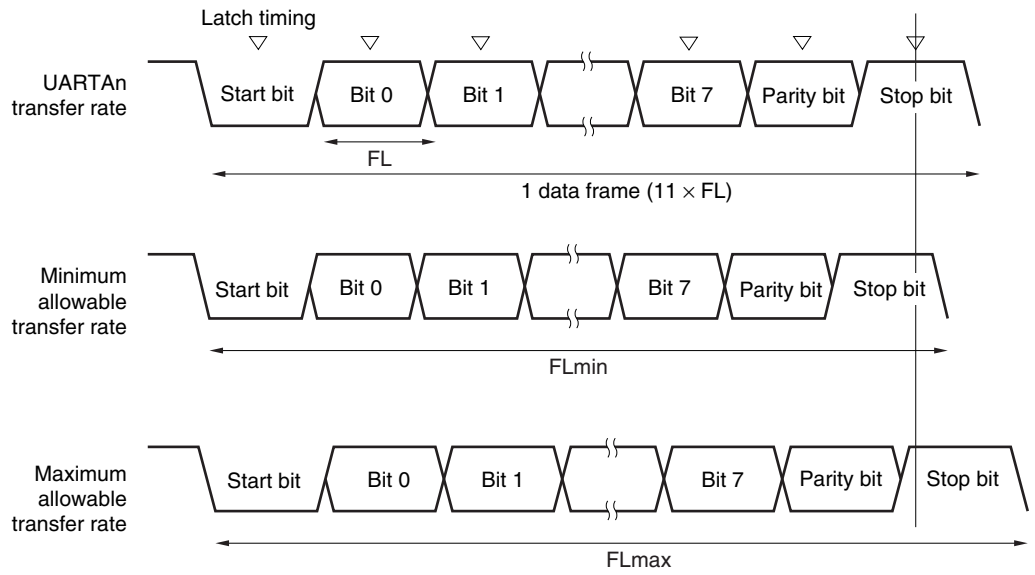
Remark f_{XX}: Peripheral clock frequency
ERR: Baud rate error (%)

(7) Allowable baud rate range during reception

The baud rate error range at the destination that is allowable during reception is shown below.

Caution The baud rate error during reception must be set within the allowable error range using the following equation.

Figure 14-11. Allowable Baud Rate Range During Reception



Remark n = 0, 1

As shown in Figure 14-11, the receive data latch timing is determined by the counter set using the UAnCTL2 register following start bit detection. The transmit data can be normally received if up to the last data (stop bit) can be received in time for this latch timing.

When this is applied to 11-bit reception, the following is the theoretical result.

$FL = (Brate)^{-1}$

Brate: UARTAn baud rate (n = 0, 1)

k: Setting value of UAnCTL2.UAnBRS7 to UAnCTL2.UAnBRS0 bits (n = 0, 1)

FL: 1-bit data length

Latch timing margin: 2 clocks

Minimum allowable transfer rate: $FLmin = 11 \times FL - \frac{k-2}{2k} \times FL = \frac{21k+2}{2k} FL$

Therefore, the maximum baud rate that can be received by the destination is as follows.

$$BRmax = (FLmin/11)^{-1} = \frac{22k}{21k + 2} \text{ Brate}$$

Similarly, obtaining the following maximum allowable transfer rate yields the following.

$$\frac{10}{11} \times FLmax = 11 \times FL - \frac{k + 2}{2 \times k} \times FL = \frac{21k - 2}{2 \times k} FL$$

$$FLmax = \frac{21k - 2}{20\ k} FL \times 11$$

Therefore, the minimum baud rate that can be received by the destination is as follows.

$$BRmin = (FLmax/11)^{-1} = \frac{20k}{21k - 2} \text{ Brate}$$

Obtaining the allowable baud rate error for UARTAn and the destination from the above-described equations for obtaining the minimum and maximum baud rate values yields the following.

Table 14-4. Maximum/Minimum Allowable Baud Rate Error

Division Ratio (k)	Maximum Allowable Baud Rate Error	Minimum Allowable Baud Rate Error
4	+2.32%	–2.43%
8	+3.52%	–3.61%
20	+4.26%	–4.30%
50	+4.56%	–4.58%
100	+4.66%	–4.67%
255	+4.72%	–4.72%

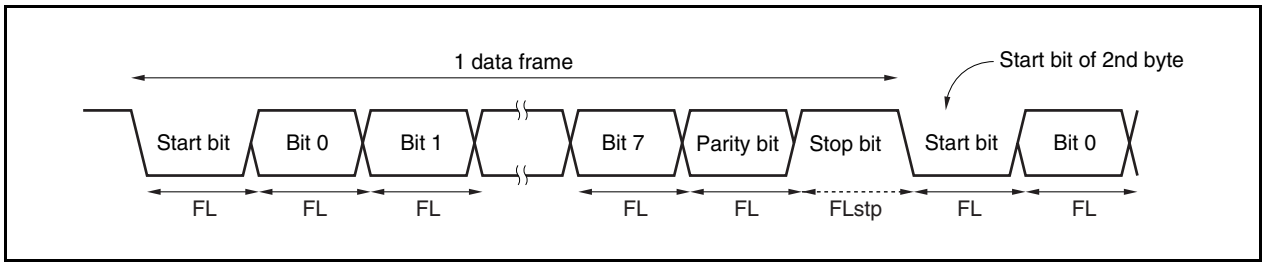
Remarks 1. The reception accuracy depends on the bit count in 1 frame, the input clock frequency, and the division ratio (k). The higher the input clock frequency and the larger the division ratio (k), the higher the accuracy.

2. k: Setting value of UAnCTL2.UAnBRS7 to UAnCTL2.UAnBRS0 bits (n = 0, 1)

(8) Transfer rate during continuous transmission

During continuous transmission, the transfer rate from the stop bit to the next start bit is usually 2 base clocks longer. However, timing initialization is performed via start bit detection by the receiving side, so this has no influence on the transfer result.

Figure 14-12. Transfer Rate During Continuous Transmission



Assuming 1 bit data length: FL; stop bit length: FLstp; and base clock frequency: f_{UCLK}, we obtain the following equation.

$$FLstp = FL + 2/f_{UCLK}$$

Therefore, the transfer rate during continuous transmission is as follows.

$$\text{Transfer rate} = 11 \times FL + (2/f_{UCLK})$$

14.8 Cautions

When the clock supply to UARTAn is stopped (for example, in IDLE or STOP mode), the operation stops with each register retaining the value it had immediately before the clock supply was stopped. The TXDAn pin output also holds and outputs the value it had immediately before the clock supply was stopped. However, the operation is not guaranteed after the clock supply is resumed. Therefore, after the clock supply is resumed, the circuits should be initialized by setting the UAnCTL0.UAnPWR, UAnCTL0.UAnRXE, and UAnCTL0.UAnTXE bits to 000.

Remark n = 0, 1

15.1 Mode Switching Between UARTA1 and CSIB1

In the V850E/IA3 and V850E/IA4, UARTA1 and CSIB1 function alternately, and these functions cannot be used at the same time. To use UARTA1 and CSIB1, the PMC3 and PFC3 registers must be set in advance.

Caution The operations related to transmission and reception of UARTA1 or CSIB1 are not guaranteed if the mode is switched during transmission or reception. Be sure to disable the unit that is not used.

Figure 15-1. Mode Switch Settings of UARTA1 and CSIB1

After reset: 00H R/W Address: FFFFF446H

	7	6	5	4	3	2	1	0
PMC3	PMC37	PMC36	PMC35	PMC34	PMC33	PMC32	PMC31	PMC30

After reset: 00H R/W Address: FFFFF466H

	7	6	5	4	3	2	1	0
PFC3	0	0	PFC35	0	PFC33	PFC32	0	0

PMC34	Specification of alternate function of P34 pin
0	I/O port
1	$\overline{\text{SCKB1}}$ I/O

PMC33	PFC33	Specification of alternate function of P33 pin
0	×	I/O port
1	0	SOB1 output
1	1	TXDA1 output

PMC32	PFC32	Specification of alternate function of P32 pin
0	×	I/O port
1	0	SIB1 input
1	1	RXDA1 input

Remark x = don't care

15.2 Features

- Transfer rate: 8 Mbps (using internal clock)
- Master mode and slave mode selectable
- 8-bit to 16-bit transfer, 3-wire serial interface
- Interrupt request signals (INTCBnRE, INTCBnT, INTCBnR)
- Serial clock and data phase switchable
- Transfer data length selectable in 1-bit units between 8 and 16 bits
- Transfer data MSB-first/LSB-first switchable
- 3-wire transfer SOBn: Serial data output
  SIBn: Serial data input
  $\overline{\text{SCKBn}}$: Serial clock I/O

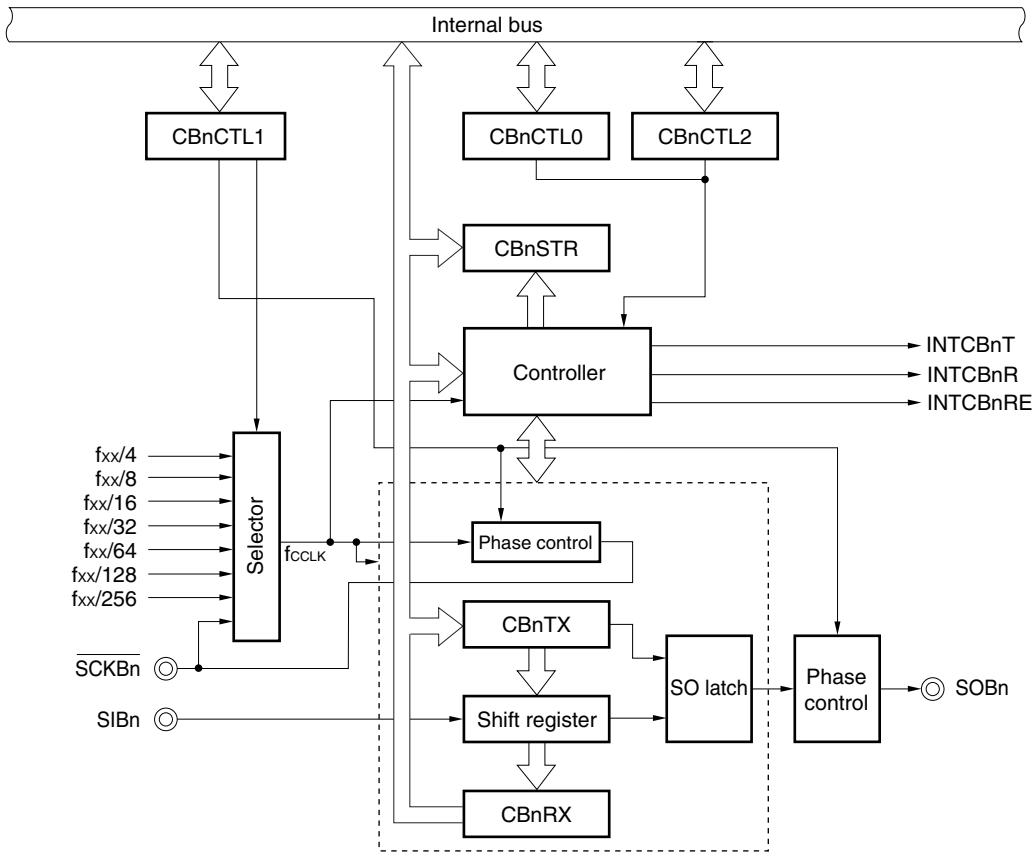
  Transmission mode, reception mode, and transmission/reception mode specifiable

Remark n = 0, 1

15.3 Configuration

The following shows the block diagram of CSIBn.

Figure 15-2. Block Diagram of CSIBn



Remarks 1. n = 0, 1

2. f_{CCLK}: Communication clock (8 MHz (max.))

CSIBn includes the following hardware.

Table 15-1. Configuration of CSIBn

Item	Configuration
Registers	CSIBn receive data register (CBnRX)
	CSIBn transmit data register (CBnTX)
Control registers	CSIBn control register 0 (CBnCTL0)
	CSIBn control register 1 (CBnCTL1)
	CSIBn control register 2 (CBnCTL2)
	CSIBn status register (CBnSTR)

(1) CSIBn receive data register (CBnRX)

The CBnRX register is a 16-bit buffer register that holds receive data.

This register is read-only, in 16-bit units.

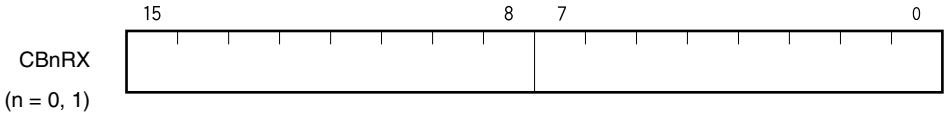
The receive operation is started by reading the CBnRX register in the reception enabled status.

If the transfer data length is 8 bits, the lower 8 bits of this register are read-only in 8-bit units as the CBnRXL register.

Reset sets this register to 0000H.

In addition to reset, the CBnRX register can be initialized by clearing (to 0) the CBnCTL0.CBnPWR bit.

After reset: 0000H R Address: CB0RX FFFFFD04H, CB0RXL FFFFFD04H
CB1RX FFFFFD14H, CB1RXL FFFFFD14H



(2) CSIBn transmit data register (CBnTX)

The CBnTX register is a 16-bit buffer register used to write the CSIBn transfer data.

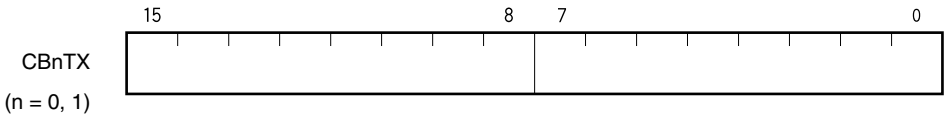
This register can be read or written in 16-bit units.

The transmit operation is started by writing data to the CBnTX register in the transmission enabled status.

If the transfer data length is 8 bits, the lower 8 bits of this register can be read or written in 8-bit units as the CBnTXL register.

Reset sets this register to 0000H.

After reset: 0000H R/W Address: CB0TX FFFFFD06H, CB0TXL FFFFFD06H
CB1TX FFFFFD16H, CB1TXL FFFFFD16H



Remark The communication start conditions are shown below.

Transmission mode (CBnTXE bit = 1, CBnRXE bit = 0):	Write to CBnTX register
Transmission/reception mode (CBnTXE bit = 1, CBnRXE bit = 1):	Write to CBnTX register
Reception mode (CBnTXE bit = 0, CBnRXE bit = 1):	Read from CBnRX register

15.4 Control Registers

The following registers are used to control CSIBn.

- CSIBn control register 0 (CBnCTL0)
- CSIBn control register 1 (CBnCTL1)
- CSIBn control register 2 (CBnCTL2)
- CSIBn status register (CBnSTR)

(1) CSIBn control register 0 (CBnCTL0)

CBnCTL0 is a register that controls the CSIBn serial transfer operation.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 01H.

(1/2)

After reset: 01H R/W Address: CB0CTL0 FFFFFD00H, CB1CTL0 FFFFFD10H

CBnCTL0 (n = 0, 1)

<7>	<6>	<5>	<4>	3	2	1	<0>
CBnPWR	CBnTXE[Note]	CBnRXE[Note]	CBnDIR[Note]	0	0	CBnTMS[Note]	CBnSCE

CBnPWR	Specification of CSIBn operation disable/enable
0	Disable CSIBn operation and reset the CBnSTR register
1	Enable CSIBn operation
• The CBnPWR bit controls the CSIBn operation and resets the internal circuit.	

CBnTXE[Note]	Specification of transmit operation disable/enable
0	Disable transmit operation
1	Enable transmit operation
• The SOBn output is low level when the CBnTXE bit is 0.	

CBnRXE[Note]	Specification of receive operation disable/enable
0	Disable receive operation
1	Enable receive operation
• When the CBnRXE bit is cleared to 0, no reception end interrupt is output even when the prescribed data is transferred in order to disable the receive operation, and the receive data (CBnRX register) is not updated.	

Note These bits can only be rewritten when the CBnPWR bit = 0. However, CBnPWR bit = 1 can also be set at the same time as rewriting these bits.

Caution Be sure to clear bits 3 and 2 to "0".

CBnDIR[Note 1]	Specification of transfer direction mode (MSB/LSB)
0	MSB first
1	LSB first

CBnTMS[Note 1]	Transfer mode specification
0	Single transfer mode
1	Continuous transfer mode

- When using single transmission or transmission/reception mode with communication type 2 or 4 (CBnCTL1.CBnDAP bit = 1), write the transfer data to the CBnTX register after checking that the CBnSTR.CBnTSF bit is 0.
- When using DMA, use the continuous transfer mode.

CBnSCE	Specification of start transfer disable/enable
0	Communication start trigger invalid
1	Communication start trigger valid

- In master mode
  This bit enables or disables the communication start trigger.
  (a) In single reception mode
  Clear the CBnSCE bit to 0 before reading the receive data (CBnRX register)[Note 2].
  (b) In continuous reception mode
  Clear the CBnSCE bit to 0 one communication clock before reception of the last data is ended[Note 3].
- In slave mode
  This bit enables or disables the communication start trigger.
  (a) In single reception mode or continuous reception mode
  Set the CBnSCE bit to 1[Note 4].
- In single transmission or transmission/reception mode, or continuous transmission or transmission/reception mode
  The function of the CBnSCE bit is invalid. It is recommended to set this bit to 1.

Notes 1. These bits can only be rewritten when the CBnPWR bit = 0. However, the CBnPWR can be set to 1 at the same time as these bits are rewritten.

2. If the CBnSCE bit is read while it is 1, the next communication operation is started.

3. The CBnSCE bit is not cleared to 0 one communication clock before the end of the last data reception, the next communication operation is automatically started.

To start communication operation again after reading the last data, set the CBnSCE bit to 1 and perform a dummy read of the CBnRX register.

4. To start the reception, a dummy read is necessary.

(a) How to use CBnSCE bit

(i) In single reception mode

<1> When the reception of the last data is completed with INTCBnR interrupt servicing, clear the CBnSCE bit to 0, and then read the CBnRX register.

<2> When the reception is disabled after the reception of the last data has been completed, check that the CBnSTR.CBnTSF bit is 0, and then clear the CBnPWR and CBnRXE bits to 0. To continue reception, set the CBnSCE bit to 1 and start the next receive operation by performing a dummy read of the CBnRX register.

(ii) In continuous reception mode

<1> Clear the CBnSCE bit to 0 during reception of the last data with INTCBnR interrupt servicing by the reception before the last reception, and then read the CBnRX register.

<2> After receiving the INTCBnR signal of the last reception, read the last data from the CBnRX register.

<3> When the reception is disabled after the reception of the last data has been completed, check that the CBnSTR.CBnTSF bit is 0, and then clear the CBnPWR and CBnRXE bits to 0. To continue reception, set the CBnSCE bit to 1 and start the next receive operation by performing a dummy read of the CBnRX register.

Caution In continuous reception mode, the serial clock is not stopped until the reception executed when the CBnSCE bit is cleared to 0 is completed after the reception is started by a dummy read.

(2) CSIBn control register 1 (CBnCTL1)

CBnCTL1 is an 8-bit register that controls the CSIBn serial transfer operation.

This register can be read or written in 8-bit or 1-bit units.

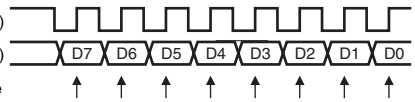
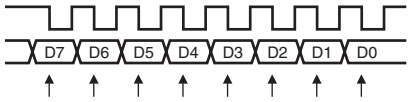
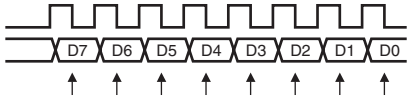
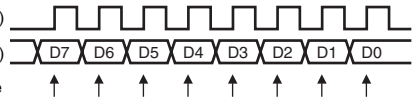
Reset sets this register to 00H.

Caution The CBnCTL1 register can be rewritten only when the CBnCTL0.CBnPWR bit = 0.

After reset: 00H R/W Address: CB0CTL1 FFFFFD01H, CB1CTL1 FFFFFD11H

	7	6	5	4	3	2	1	0
CBnCTL1	0	0	0	CBnCKP	CBnDAP	CBnCKS2	CBnCKS1	CBnCKS0

(n = 0, 1)

	CBnCKP	CBnDAP	Specification of data transmission/ reception timing in relation to $\overline{\text{SCKBn}}$
Communication type 1	0	0	
Communication type 2	0	1	
Communication type 3	1	0	
Communication type 4	1	1	

CBnCKS2	CBnCKS1	CBnCKS0	Communication clock (f_{CCLK})	Mode
0	0	0	$f_{XX}/4$	Master mode
0	0	1	$f_{XX}/8$	Master mode
0	1	0	$f_{XX}/16$	Master mode
0	1	1	$f_{XX}/32$	Master mode
1	0	0	$f_{XX}/64$	Master mode
1	0	1	$f_{XX}/128$	Master mode
1	1	0	$f_{XX}/256$	Master mode
1	1	1	External clock ($\overline{\text{SCKBn}}$)	Slave mode

<R> **Caution Set f_{CCLK} to 8 MHz or lower.**

(3) CSIBn control register 2 (CBnCTL2)

CBnCTL2 is an 8-bit register that controls the number of CSIBn serial transfer bits.

This register can be read or written in 8-bit units.

Reset sets register to 00H.

Caution The CBnCTL2 register can be rewritten only when the CBnCTL0.CBnPWR bit = 0 or when both the CBnTXE and CBnRXE bits = 0.

After reset: 00H R/W Address: CB0CTL2 FFFFFD02H, CB1CTL2 FFFFFD12H

	7	6	5	4	3	2	1	0
CBnCTL2 (n = 0, 1)	0	0	0	0	CBnCL3	CBnCL2	CBnCL1	CBnCL0

CBnCL3	CBnCL2	CBnCL1	CBnCL0	Serial register bit length
0	0	0	0	8 bits
0	0	0	1	9 bits
0	0	1	0	10 bits
0	0	1	1	11 bits
0	1	0	0	12 bits
0	1	0	1	13 bits
0	1	1	0	14 bits
0	1	1	1	15 bits
1	×	×	×	16 bits

Remark If the number of transfer bits is other than 8 or 16, prepare and use data stuffed from the LSB of the CBnTX and CBnRX registers.

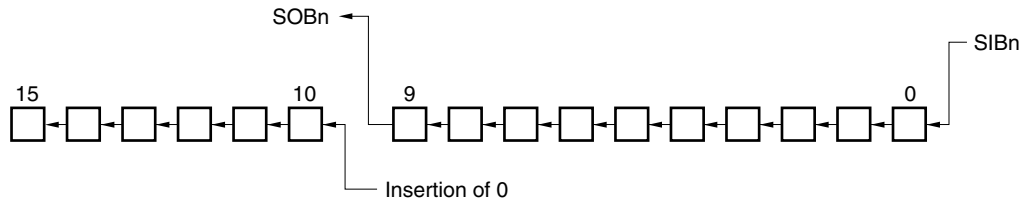
(a) Transfer data length change function

The CSIBn transfer data length can be set in 1-bit units between 8 and 16 bits using the CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits.

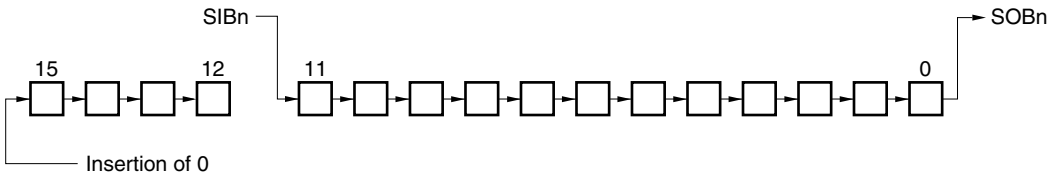
When the transfer bit length is set to a value other than 16 bits, set the data to the CBnTX or CBnRX register starting from the LSB, regardless of whether the transfer start bit is the MSB or LSB. Any data can be set for the higher bits that are not used, but the receive data becomes 0 following serial transfer.

Remark n = 0, 1

(i) Transfer bit length = 10 bits, MSB first



(ii) Transfer bit length = 12 bits, LSB first



(4) CSIBn status register (CBnSTR)

CBnSTR is an 8-bit register that displays the CSIBn status.

This register can be read or written in 8-bit or 1-bit units, but the CBnTSF flag is read-only.

Reset sets this register to 00H.

In addition to reset, the CBnSTR register can be initialized by clearing (0) the CBnCTL0.CBnPWR bit.

After reset: 00H R/W Address: CB0STR FFFFFD03H, CB1STR FFFFFD13H

	<7>	6	5	4	3	2	1	<0>
CBnSTR	CBnTSF	0	0	0	0	0	0	CBnOVE

(n = 0, 1)

CBnTSF	Communication status flag
0	Communication stopped
1	Communicating

- During transmission, this register is set when data is prepared in the CBnTX register, and during reception, it is set when a dummy read of the CBnRX register is performed.
  When transfer ends, this flag is cleared to 0 at the last edge of the clock.

CBnOVE	Overrun error flag
0	No overrun
1	Overrun

- An overrun error occurs when the next reception starts without performing a CPU read of the value of the CBnRX register, upon end of the receive operation.
  The CBnOVE flag displays the overrun error occurrence status in this case.
- The CBnOVE flag is cleared by writing 0 to it. It cannot be set even by writing 1 to it.

Caution In single transfer mode, writing to the CBnTX register with the CBnTSF bit set to 1 is ignored. This has no influence on the operation during transfer.

For example, if the next data is written to the CBnTX register when DMA is started by generating the INTCBnR signal, the written data is not transferred because the CBnTSF bit is set to 1.

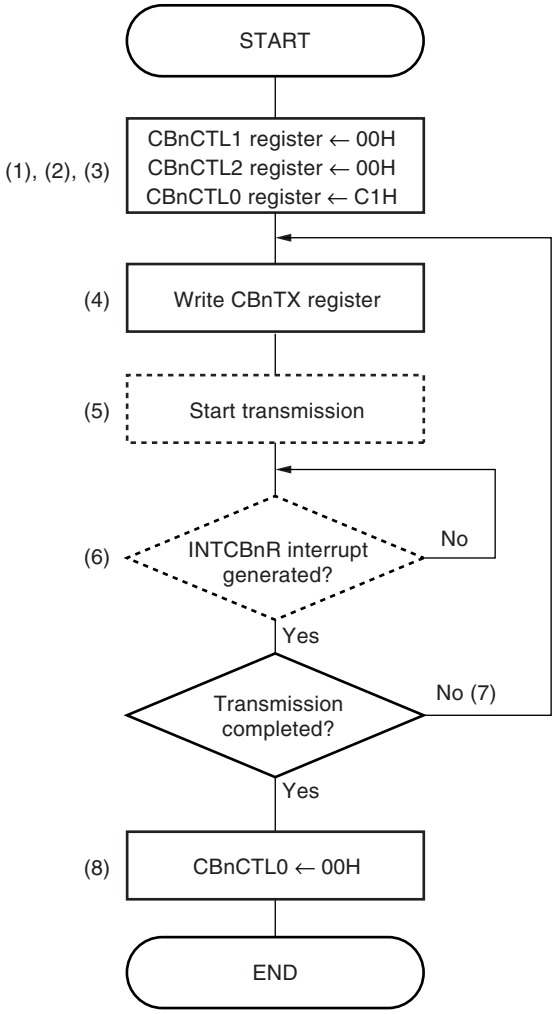
Use the continuous transfer mode, not the single transfer mode, for such applications.

15.5 Operation

15.5.1 Single transfer mode (master mode, transmission mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

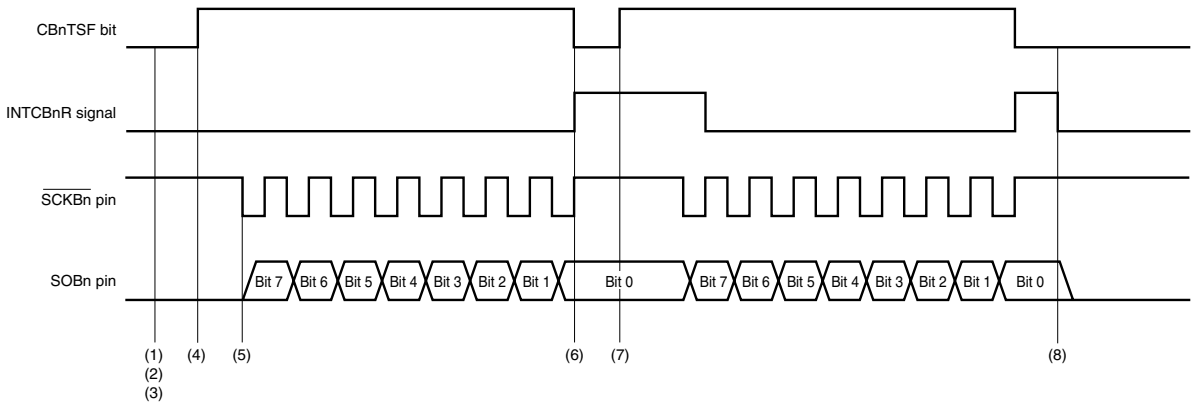


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write C1H to the CBnCTL0 register, and select the transmission mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and transmission is started.

(5) When transmission is started, output the serial clock to the $\overline{\text{SCKBn}}$ pin, and output the transmit data from the SOBn pin in synchronization with the serial clock.

(6) When transmission of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output and transmit data output, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.

(7) To continue transmission, start the next transmission by writing the transmit data to the CBnTX register again after the INTCBnR signal is generated.

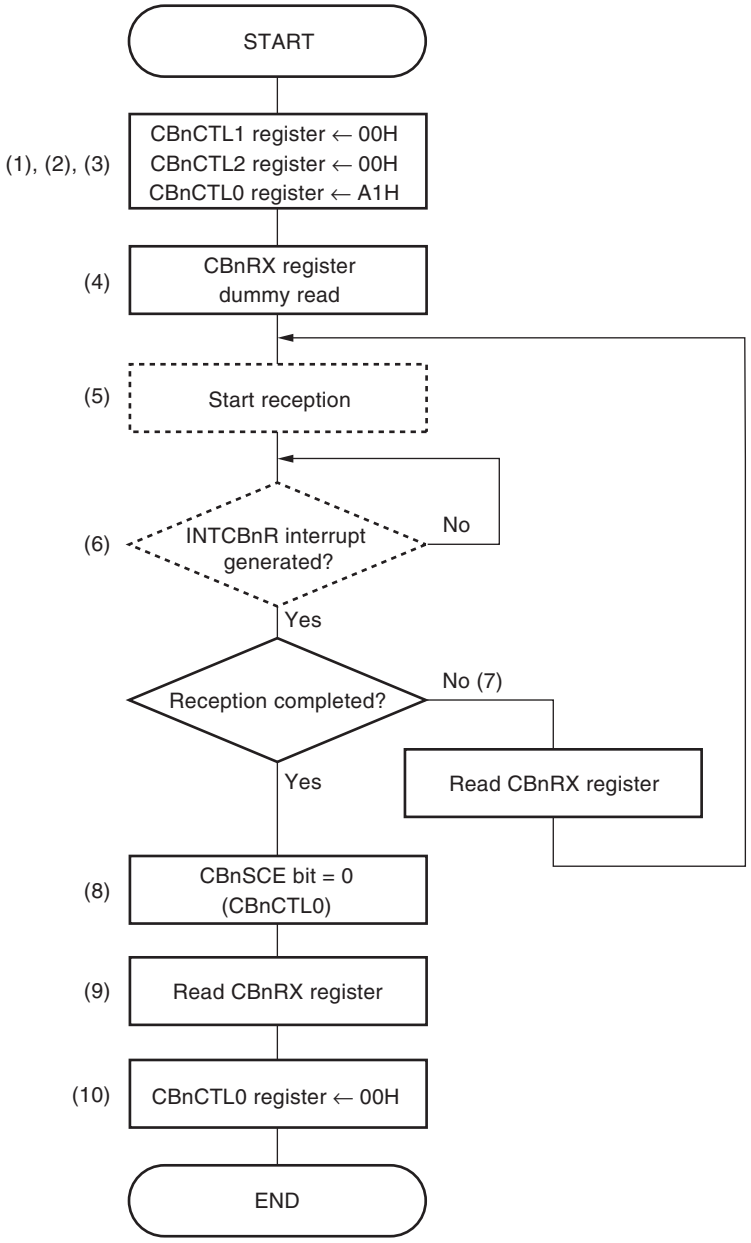
(8) To end transmission, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnTXE bit = 0.

Remark n = 0, 1

15.5.2 Single transfer mode (master mode, reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

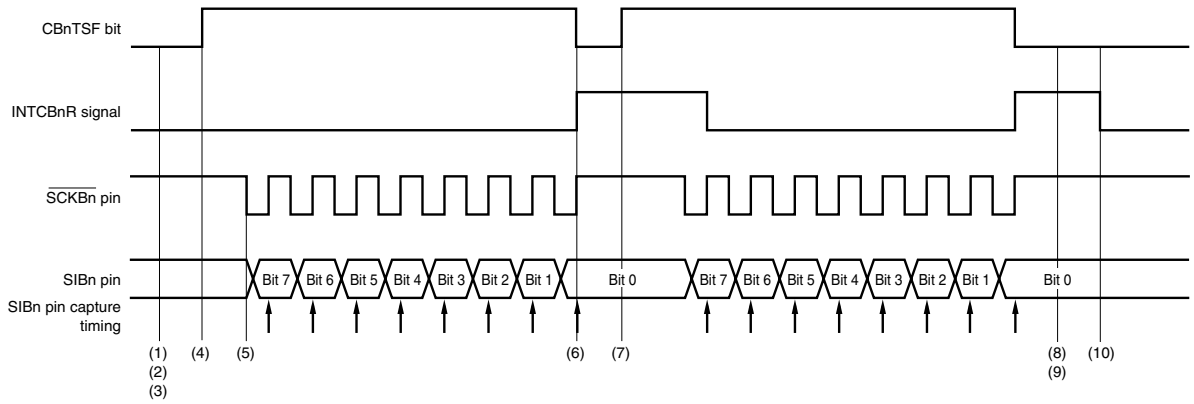


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write A1H to the CBnCTL0 register, and select the reception mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and reception is started.

(5) When reception is started, output the serial clock to the $\overline{\text{SCKBn}}$ pin, and capture the receive data of the SIBn pin in synchronization with the serial clock.

(6) When reception of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output and data capturing, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.

(7) To continue reception, read the CBnRX register with the CBnCTL0.CBnSCE bit = 1 remained after the INTCBnR signal is generated.

(8) To read the CBnRX register without starting the next reception, write the CBnSCE bit = 0.

(9) Read the CBnRX register.

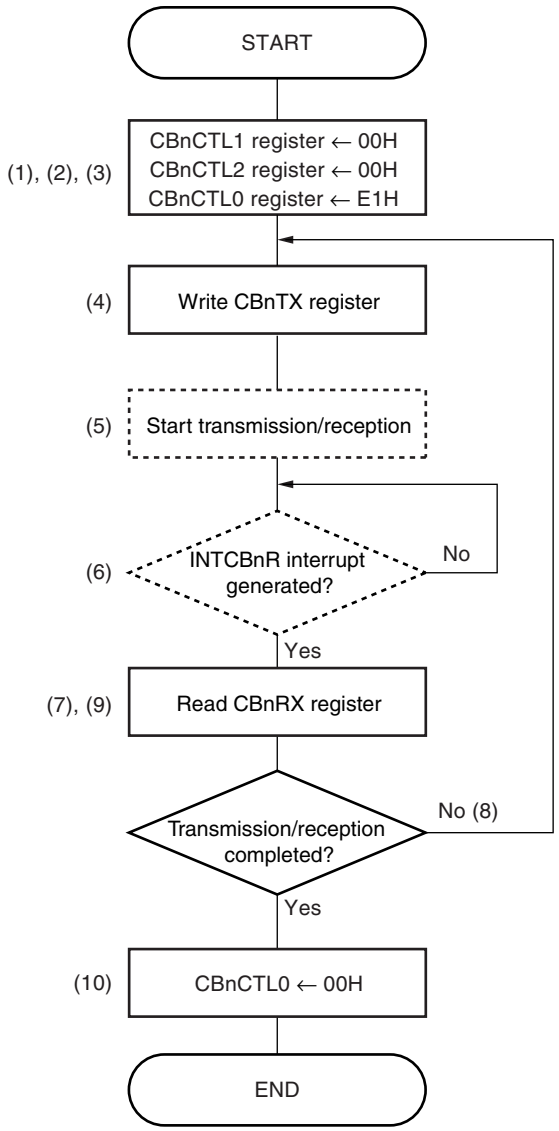
(10) To end reception, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnRXE bit = 0.

Remark n = 0, 1

15.5.3 Single transfer mode (master mode, transmission/reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

<R> **(1) Operation flow**

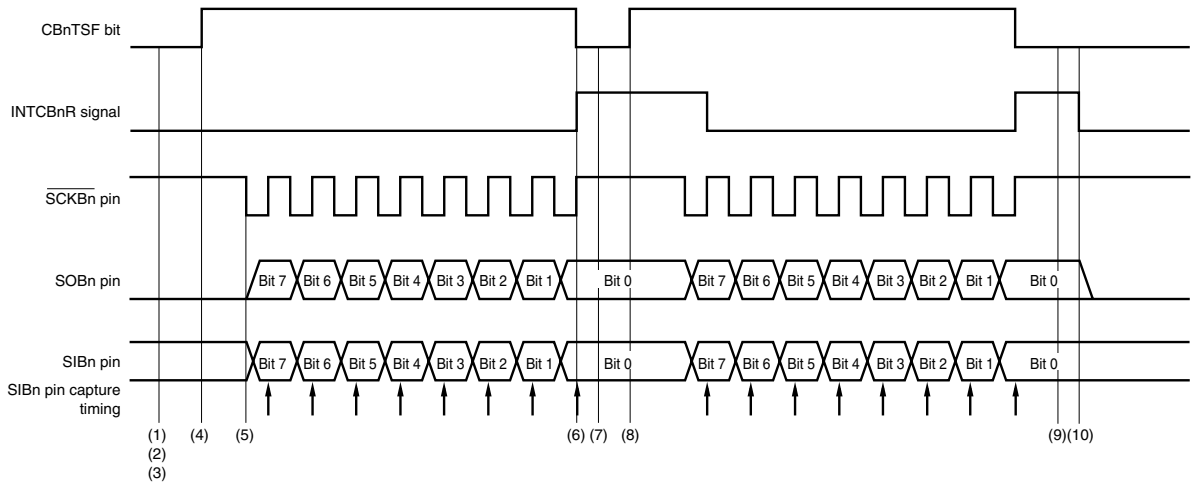


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write E1H to the CBnCTL0 register, and select the transmission/reception mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and transmission/reception is started.

(5) When transmission/reception is started, output the serial clock to the $\overline{SCKBn}$ pin, output the transmit data to the SOBn pin in synchronization with the serial clock, and capture the receive data of the SIBn pin.

(6) When transmission/reception of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output, transmit data output, and data capturing, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.

(7) Read the CBnRX register.

(8) To continue transmission/reception, write the transmit data to the CBnTX register again.

(9) Read the CBnRX register.

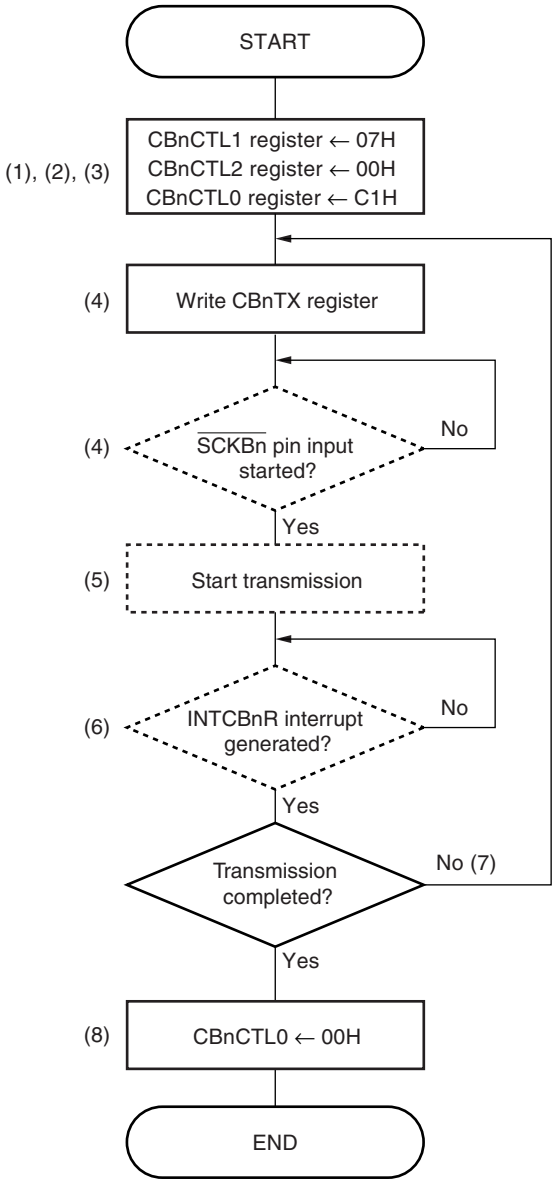
(10) To end transmission/reception, write the CBnCTL0.CBnPWR bit = 0, the CBnCTL0.CBnTXE bit = 0, and the CBnCTL0.CBnRXE bit = 0.

Remark n = 0, 1

15.5.4 Single transfer mode (slave mode, transmission mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

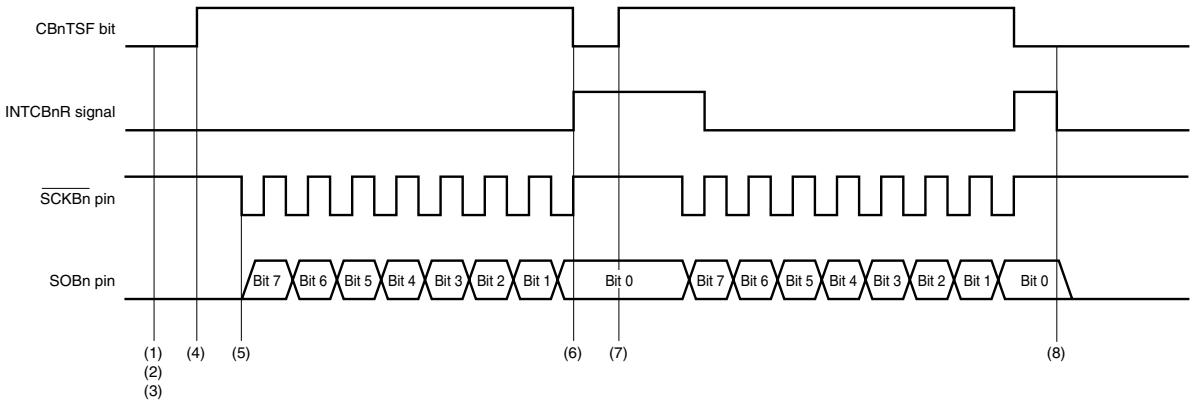


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



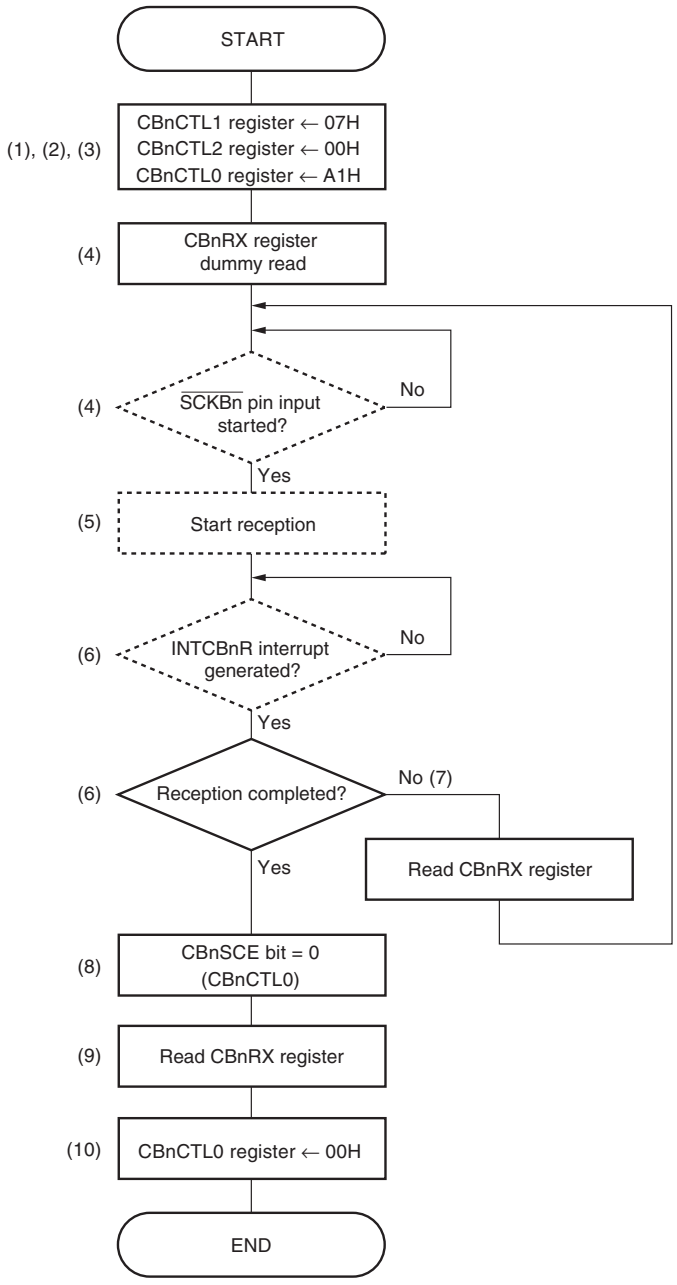
(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.
(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.
(3) Write C1H to the CBnCTL0 register, and select the transmission mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).
(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and the device waits for a serial clock input.
(5) When a serial clock is input, output the transmit data from the SOBn pin in synchronization with the serial clock.
(6) When transmission of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output and transmit data output, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.
(7) To continue transmission, write the transmit data to the CBnTX register again after the INTCBnR signal is generated, and wait for a serial clock input.
(8) To end transmission, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnTXE bit = 0.

Remark n = 0, 1

15.5.5 Single transfer mode (slave mode, reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

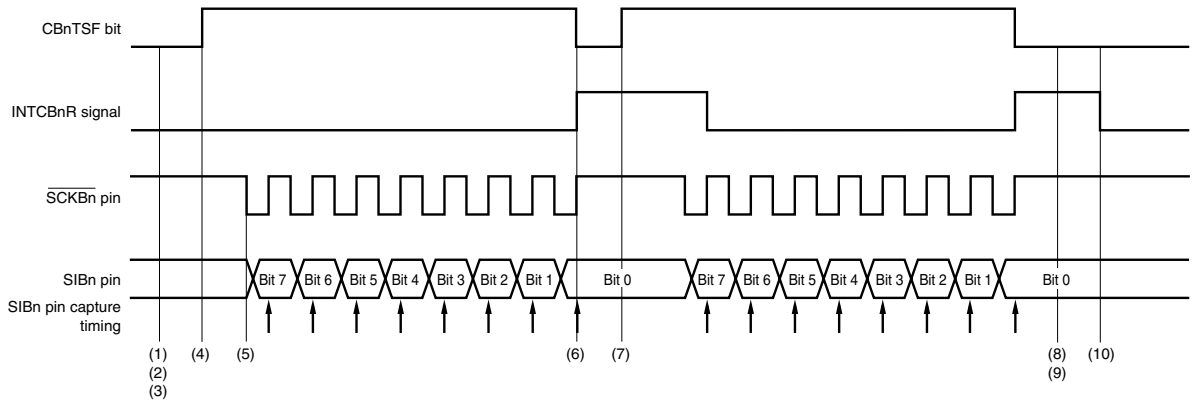


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



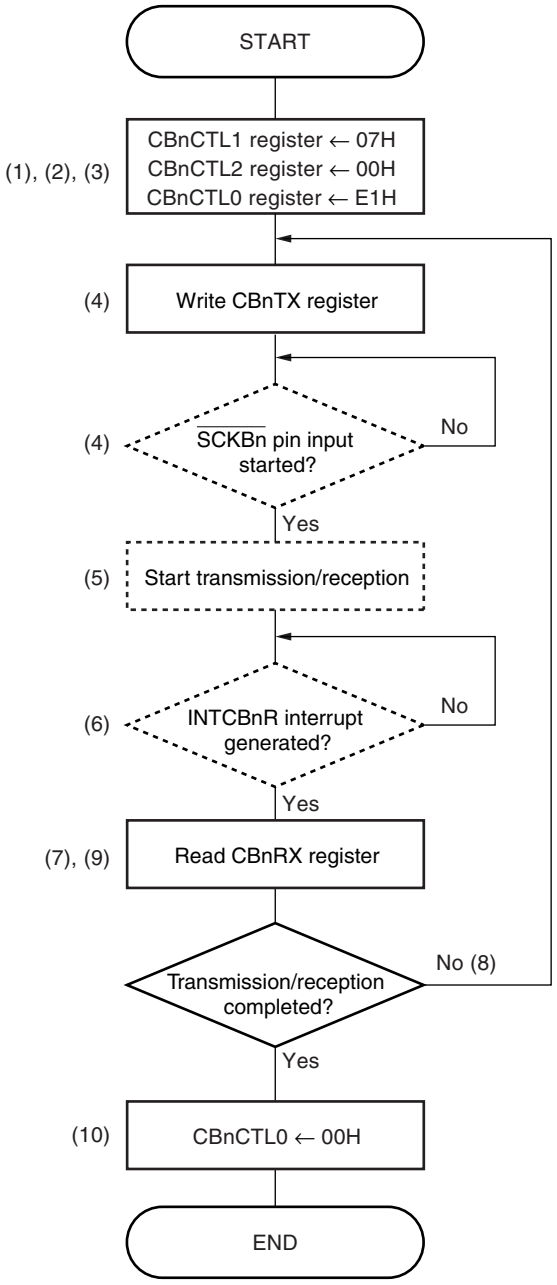
(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.
(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.
(3) Write A1H to the CBnCTL0 register, and select the reception mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).
(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and the device waits for a serial clock input.
(5) When a serial clock is input, capture the receive data of the SIBn pin in synchronization with the serial clock.
(6) When reception of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output and data capturing, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.
(7) To continue reception, read the CBnRX register with the CBnCTL0.CBnSCE bit = 1 remained after the INTCBnR signal is generated, and wait for a serial clock input.
(8) To end reception, write the CBnSCE bit = 0.
(9) Read the CBnRX register.
(10) To end reception, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnRXE bit = 0.

Remark n = 0, 1

15.5.6 Single transfer mode (slave mode, transmission/reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

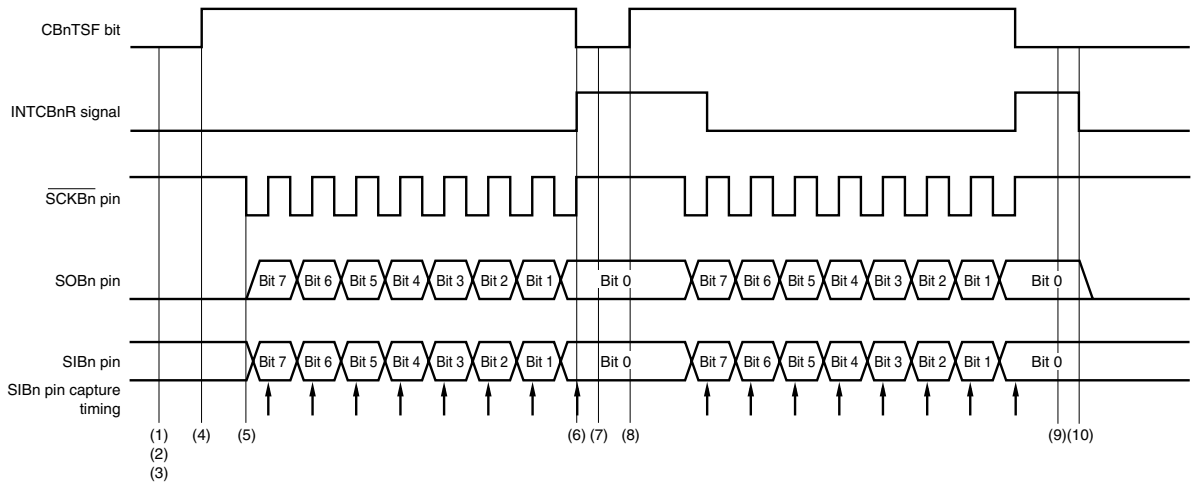


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write E1H to the CBnCTL0 register, and select the transmission/reception mode and MSB first at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and the device waits for a serial clock input.

(5) When a serial clock is input, output the transmit data to the SOBn pin in synchronization with the serial clock, and capture the receive data of the SIBn pin.

(6) When transmission/reception of the transfer data length set with the CBnCTL2 register is completed, stop the serial clock output, transmit data output, and data capturing, generate the reception end interrupt request signal (INTCBnR) at the last edge of the serial clock, and clear the CBnTSF bit to 0.

(7) Read the CBnRX register.

(8) To continue transmission/reception, write the transmit data to the CBnTX register again, and wait for a serial clock input.

(9) Read the CBnRX register.

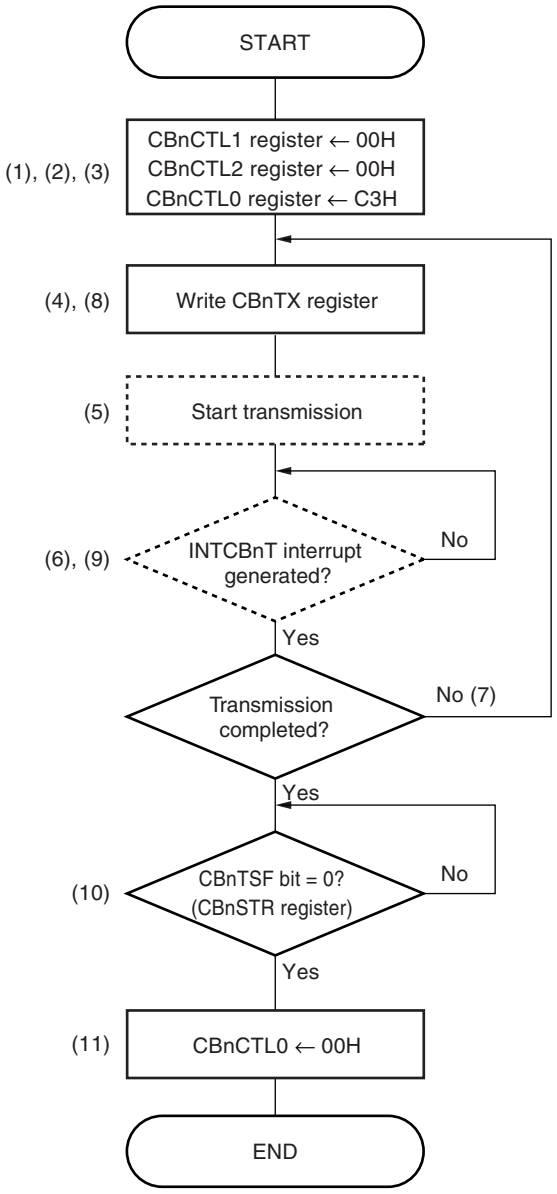
(10) To end transmission/reception, write the CBnCTL0.CBnPWR bit = 0, the CBnCTL0.CBnTXE bit = 0, and the CBnCTL0.CBnRXE bit = 0.

Remark n = 0, 1

15.5.7 Continuous transfer mode (master mode, transmission mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow



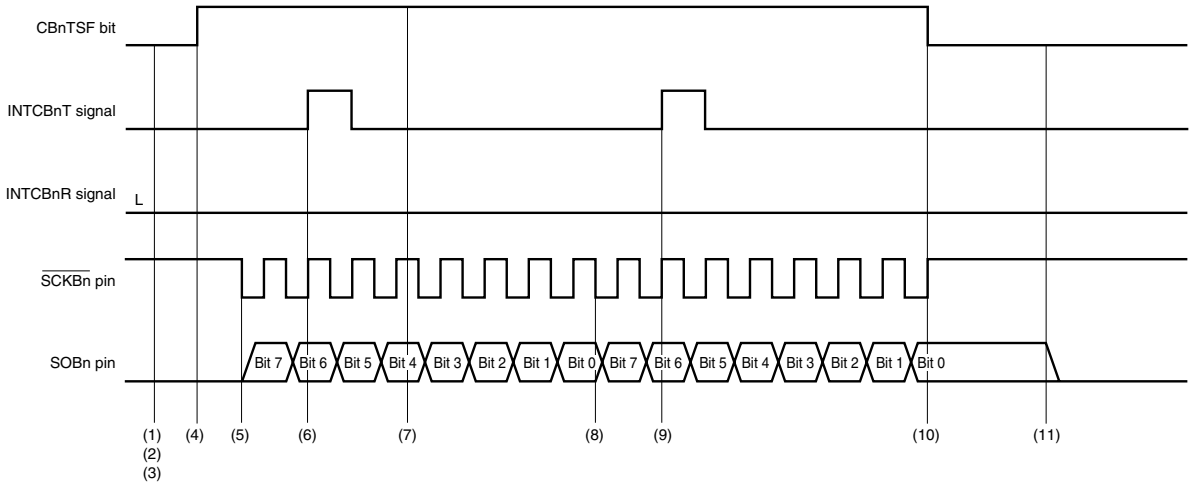
Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing

<R>



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write C3H to the CBnCTL0 register, and select the transmission mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and transmission is started.

(5) When transmission is started, output the serial clock to the $\overline{SCKBn}$ pin, and output the transmit data from the SOBn pin in synchronization with the serial clock.

(6) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the transmission enable interrupt request signal (INTCBnT) is generated.

(7) To continue transmission, write the transmit data to the CBnTX register again after the INTCBnT signal is generated.

(8) When a new transmit data is written to the CBnTX register before communication completion, the next communication is started following communication completion.

(9) The transfer of the transmit data from the CBnTX register to the shift register is completed and the INTCBnT signal is generated. To end continuous transmission with the current transmission, do not write to the CBnTX register.

(10) When the next transmit data is not written to the CBnTX register before transfer completion, stop the serial clock output to the $\overline{SCKBn}$ pin after transfer completion, and clear the CBnTSF bit to 0.

(11) To release the transmission enable status, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnTXE bit = 0 after checking that the CBnTSF bit = 0.

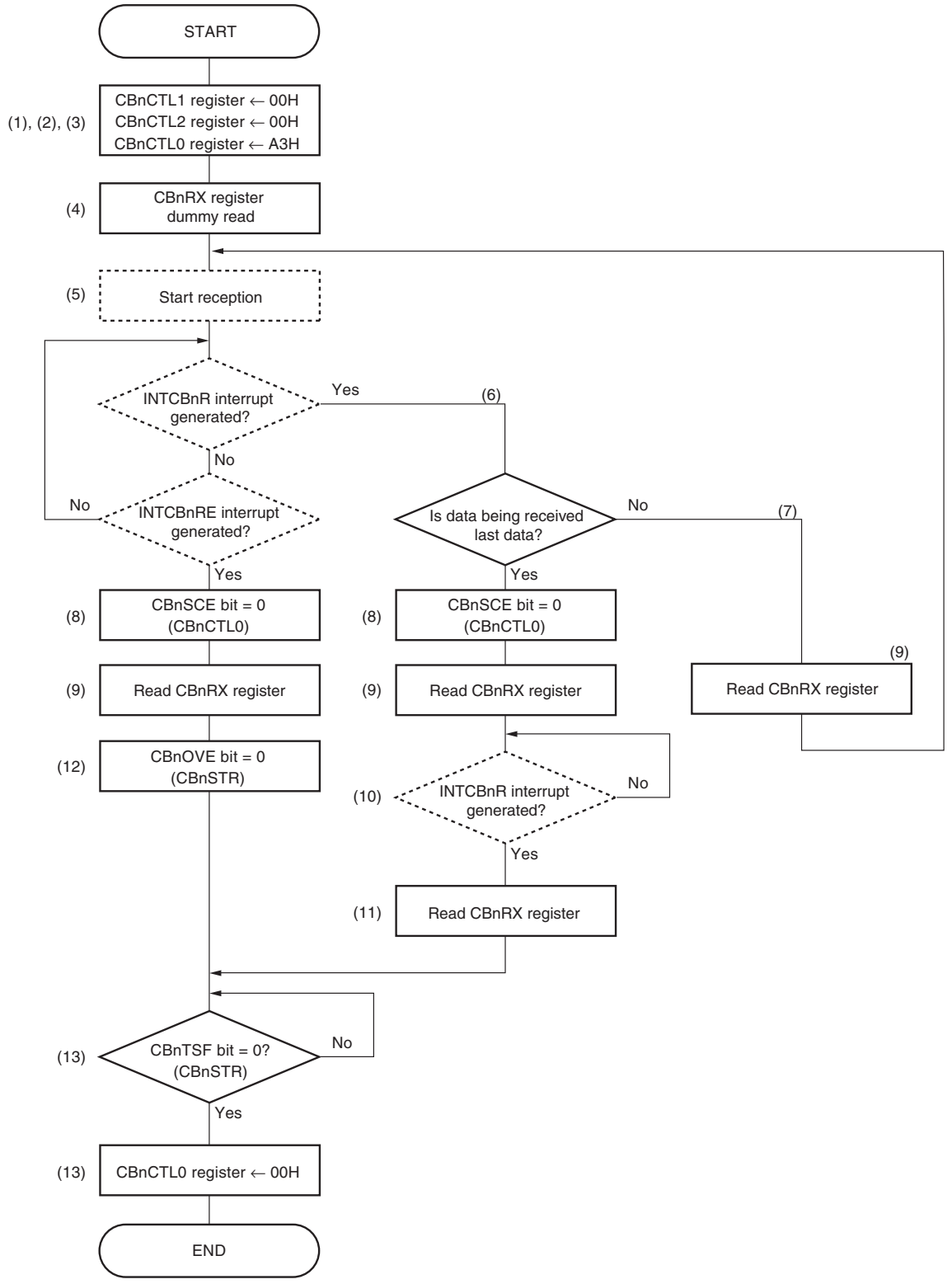
Caution In continuous transmission mode, the reception end interrupt request signal (INTCBnR) is not generated.

Remark n = 0, 1

15.5.8 Continuous transfer mode (master mode, reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

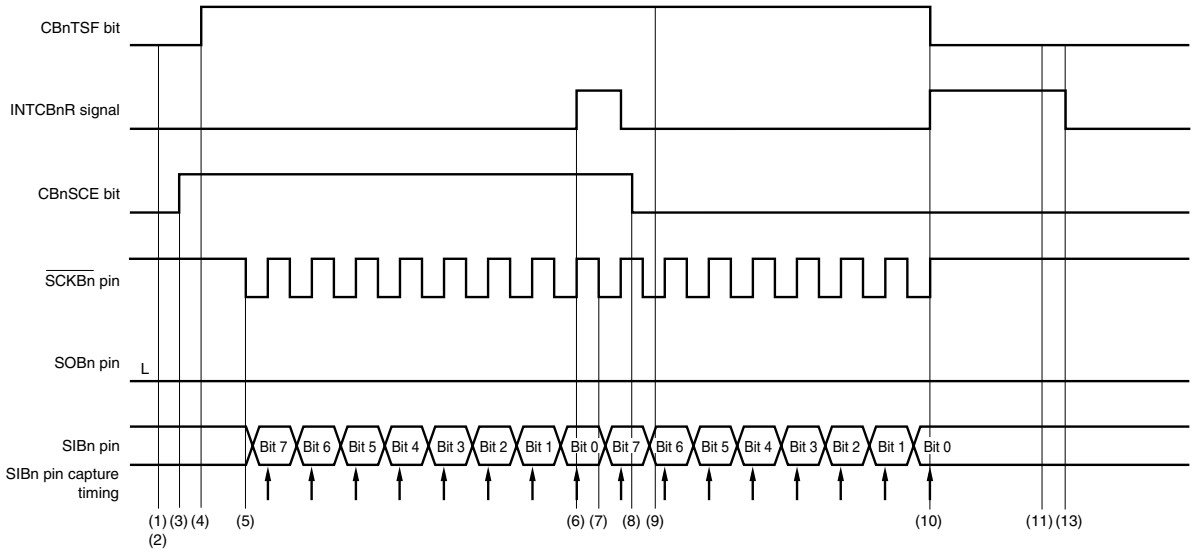


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write A3H to the CBnCTL0 register, and select the reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and reception is started.

(5) When reception is started, output the serial clock to the $\overline{\text{SCKBn}}$ pin, and capture the receive data of the SIBn pin in synchronization with the serial clock.

(6) When reception is completed, the reception end interrupt request signal (INTCBnR) is generated, and reading of the CBnRX register is enabled.

(7) When the CBnCTL0.CBnSCE bit = 1 upon communication completion, the next communication is started following communication completion.

(8) To end continuous reception with the current reception, write the CBnSCE bit = 0.

(9) Read the CBnRX register.

(10) When reception is completed, the INTCBnR signal is generated, and reading of the CBnRX register is enabled. When the CBnSCE bit = 0 is set before communication completion, stop the serial clock output to the $\overline{\text{SCKBn}}$ pin, and clear the CBnTSF bit to 0, to end the receive operation.

(11) Read the CBnRX register.

(12) If an overrun error occurs, write the CBnSTR.CBnOVE bit = 0, and clear the error flag.

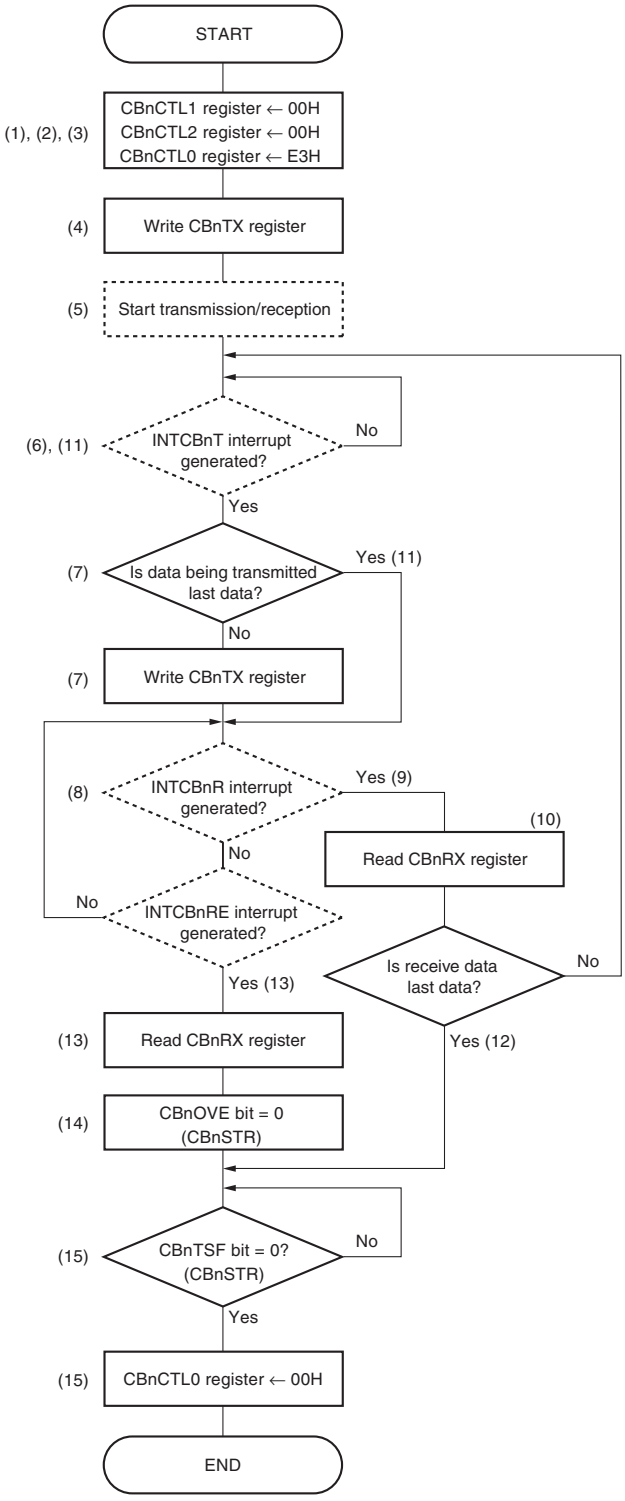
(13) To release the reception enable status, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnRXE bit = 0 after checking that the CBnTSF bit = 0.

Remark n = 0, 1

15.5.9 Continuous transfer mode (master mode, transmission/reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = f_{XX}/4 (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 000), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow



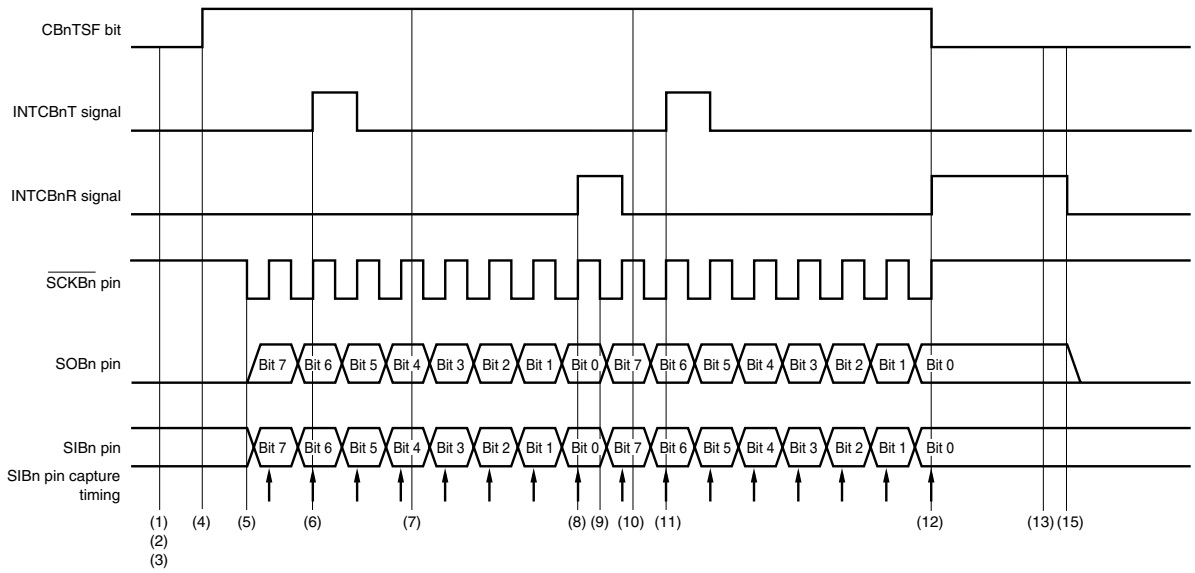
Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing

(1/2)



(1) Write 00H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = f_{XX}/4, and master mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write E3H to the CBnCTL0 register, and select the transmission/reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and transmission/reception is started.

(5) When transmission/reception is started, output the serial clock to the $\overline{\text{SCKBn}}$ pin, output the transmit data to the SOBn pin in synchronization with the serial clock, and capture the receive data of the SIBn pin.

(6) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the transmission enable interrupt request signal (INTCBnT) is generated.

(7) To continue transmission/reception, write the transmit data to the CBnTX register again after the INTCBnT signal is generated.

(8) When one transmission/reception is completed, the reception end interrupt request signal (INTCBnR) is generated, and reading of the CBnRX register is enabled.

(9) When a new transmit data is written to the CBnTX register before communication completion, the next communication is started following communication completion.

(10) Read the CBnRX register.

Remark n = 0, 1

(2/2)

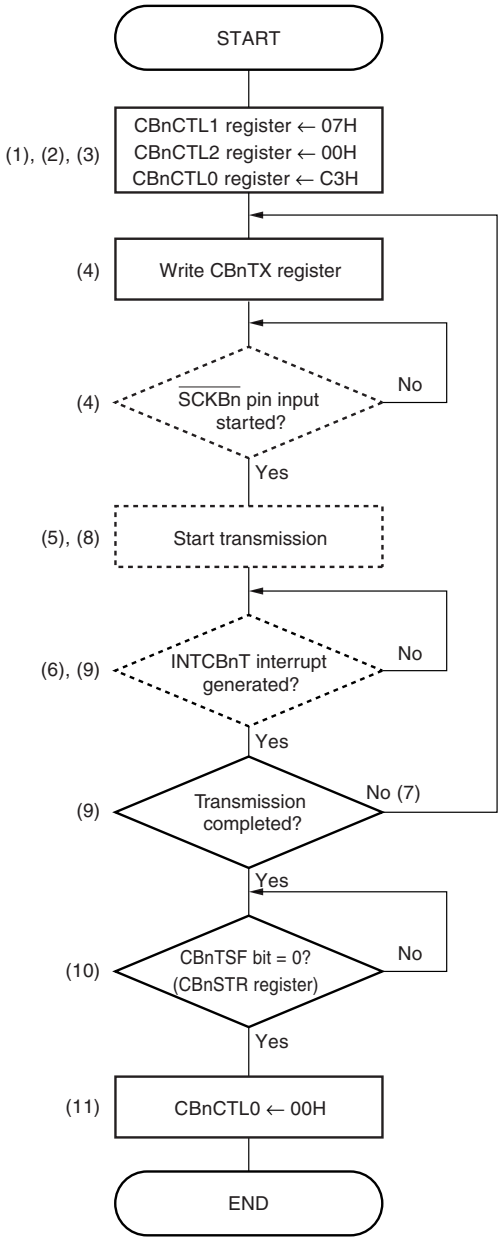
(11) The transfer of the transmit data from the CBnTX register to the shift register is completed and the INTCBnT signal is generated. To end continuous transmission/reception with the current transmission/reception, do not write to the CBnTX register.
(12) When the next transmit data is not written to the CBnTX register before transfer completion, stop the serial clock output to the $\overline{\text{SCKBn}}$ pin after transfer completion, and clear the CBnTSF bit to 0.
(13) When the reception error interrupt request signal (INTCBnRE) is generated, read the CBnRX register.
(14) If an overrun error occurs, write the CBnSTR.CBnOVE bit = 0, and clear the error flag.
(15) To release the transmission/reception enable status, write the CBnCTL0.CBnPWR bit = 0, the CBnCTL0.CBnTXE bit = 0, and the CBnCTL0.CBnRXE bit = 0 after checking that the CBnTSF bit = 0.

Remark n = 0, 1

15.5.10 Continuous transfer mode (slave mode, transmission mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

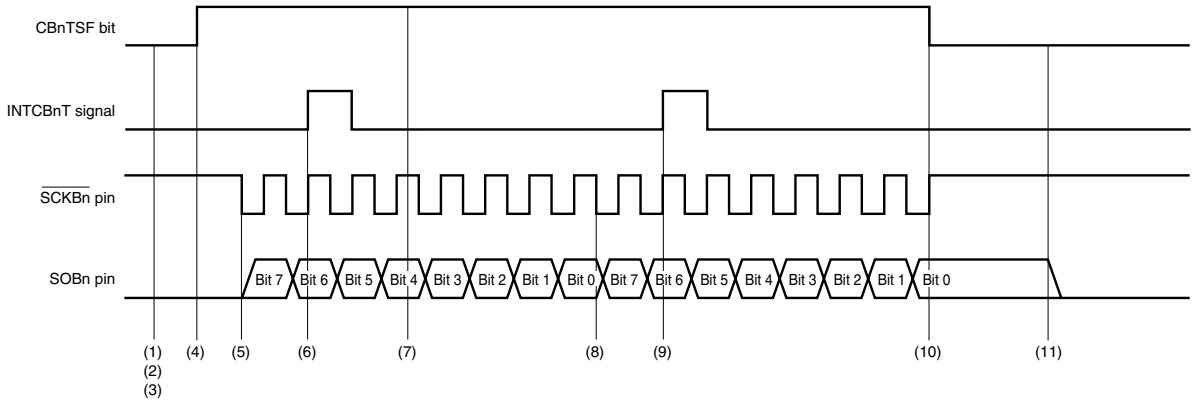


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write C3H to the CBnCTL0 register, and select the transmission mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and the device waits for a serial clock input.

(5) When a serial clock is input, output the transmit data from the SOBn pin in synchronization with the serial clock.

(6) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the transmission enable interrupt request signal (INTCBnT) is generated.

(7) To continue transmission, write the transmit data to the CBnTX register again after the INTCBnT signal is generated.

(8) When a serial clock is input following completion of the transmission of the transfer data length set with the CBnCTL2 register, continuous transmission is started.

(9) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the INTCBnT signal is generated. To end continuous transmission with the current transmission, do not write to the CBnTX register.

(10) When the clock of the transfer data length set with the CBnCTL2 register is input without writing to the CBnTX register, clear the CBnTSF bit to 0 to end transmission.

(11) To release the transmission enable status, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnTXE bit = 0 after checking that the CBnTSF bit = 0.

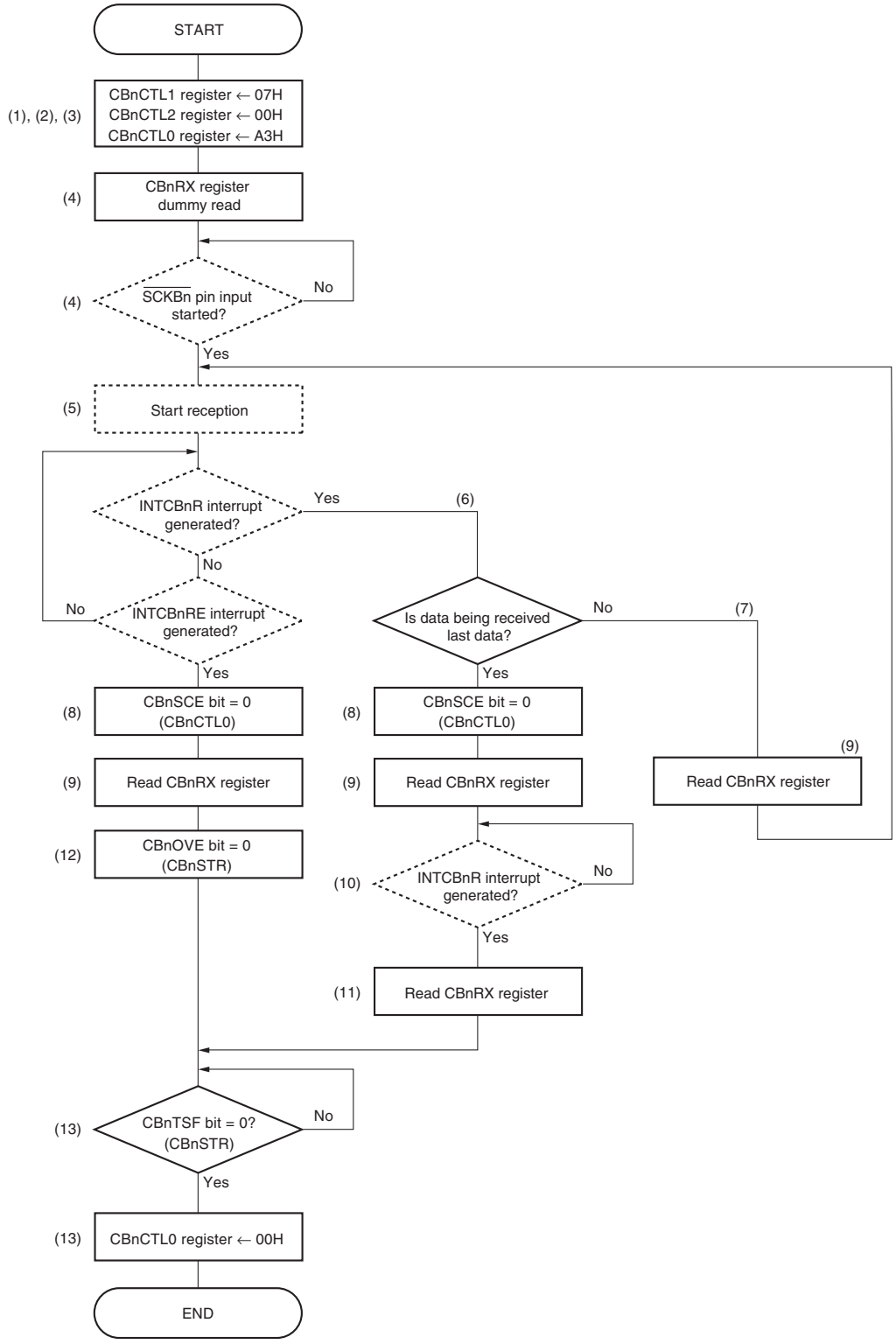
Caution In continuous transmission mode, the reception end interrupt request signal (INTCBnR) is not generated.

Remark n = 0, 1

15.5.11 Continuous transfer mode (slave mode, reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow

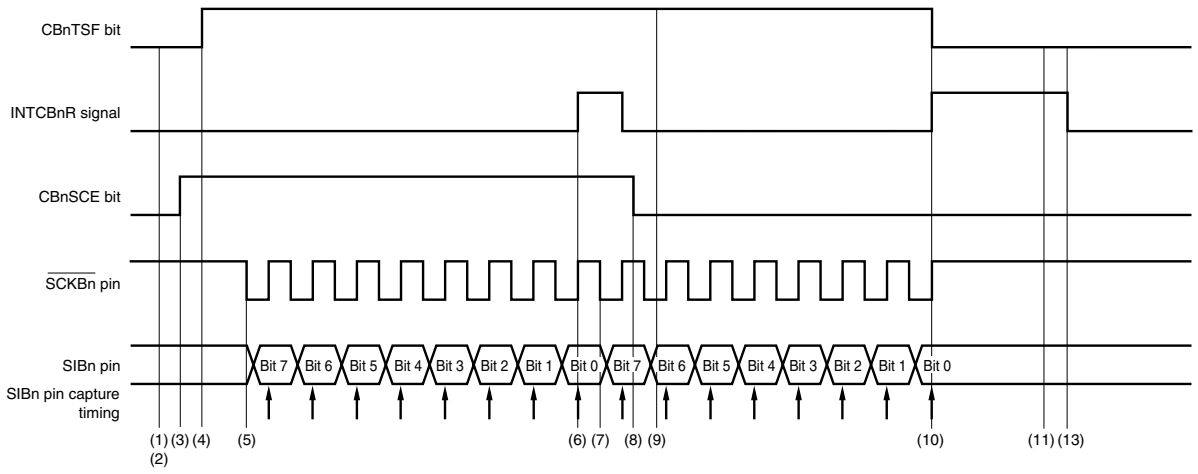


Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing



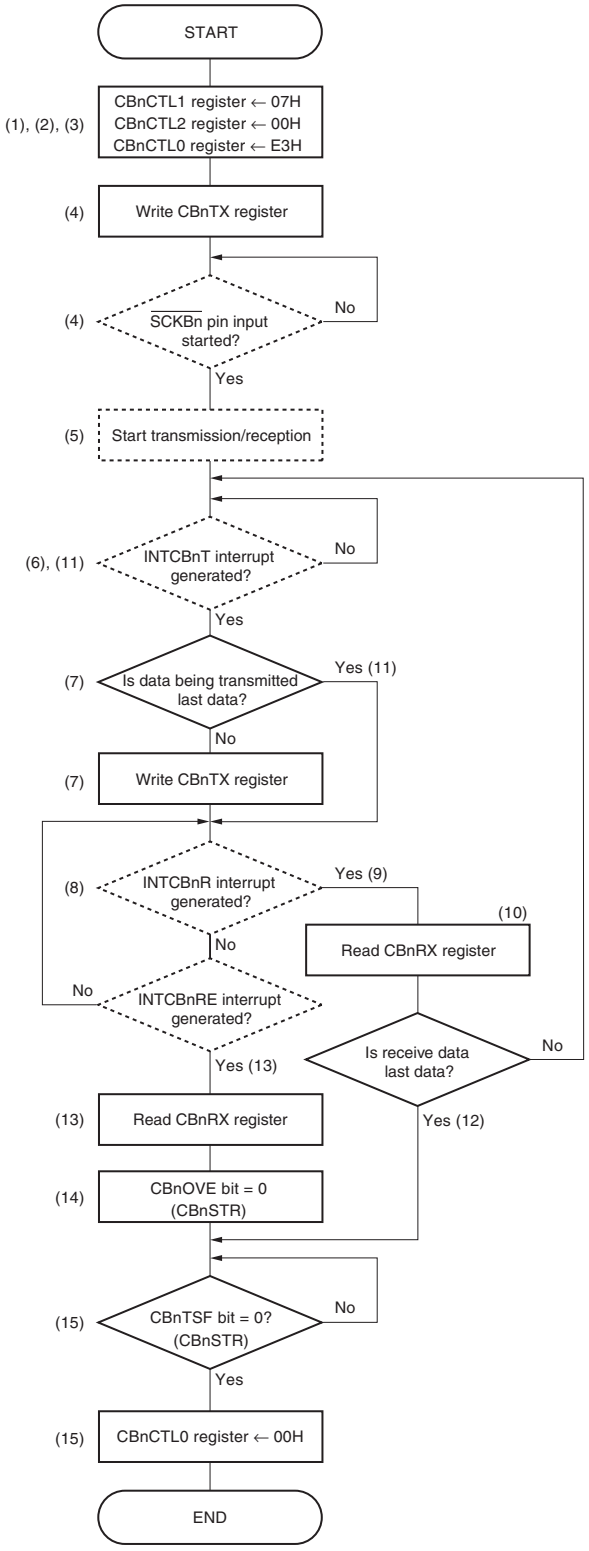
(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.
(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.
(3) Write A3H to the CBnCTL0 register, and select the reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).
(4) The CBnSTR.CBnTSF bit is set to 1 by performing a dummy read of the CBnRX register, and the device waits for a serial clock input.
(5) When a serial clock is input, capture the receive data of the SIBn pin in synchronization with the serial clock.
(6) When reception is completed, the reception end interrupt request signal (INTCBnR) is generated, and reading of the CBnRX register is enabled.
(7) When a serial clock is input in the CBnCTL0.CBnSCE bit = 1 status, continuous reception is started.
(8) To end continuous reception with the current reception, write the CBnSCE bit = 0.
(9) Read the CBnRX register.
(10) When reception is completed, the INTCBnR signal is generated, and reading of the CBnRX register is enabled. When the CBnSCE bit = 0 is set before communication completion, clear the CBnTSF bit to 0 to end the receive operation.
(11) Read the CBnRX register.
(12) If an overrun error occurs, write the CBnSTR.CBnOVE bit = 0, and clear the error flag.
(13) To release the reception enable status, write the CBnCTL0.CBnPWR bit = 0 and the CBnCTL0.CBnRXE bit = 0 after checking that the CBnTSF bit = 0.

Remark n = 0, 1

15.5.12 Continuous transfer mode (slave mode, transmission/reception mode)

MSB first (CBnCTL0.CBnDIR bit = 0), communication type 1 (CBnCTL1.CBnCKP and CBnCTL1.CBnDAP bits = 00), communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$) (CBnCTL1.CBnCKS2 to CBnCTL1.CBnCKS0 bits = 111), transfer data length = 8 bits (CBnCTL2.CBnCL3 to CBnCTL2.CBnCL0 bits = 0000)

(1) Operation flow



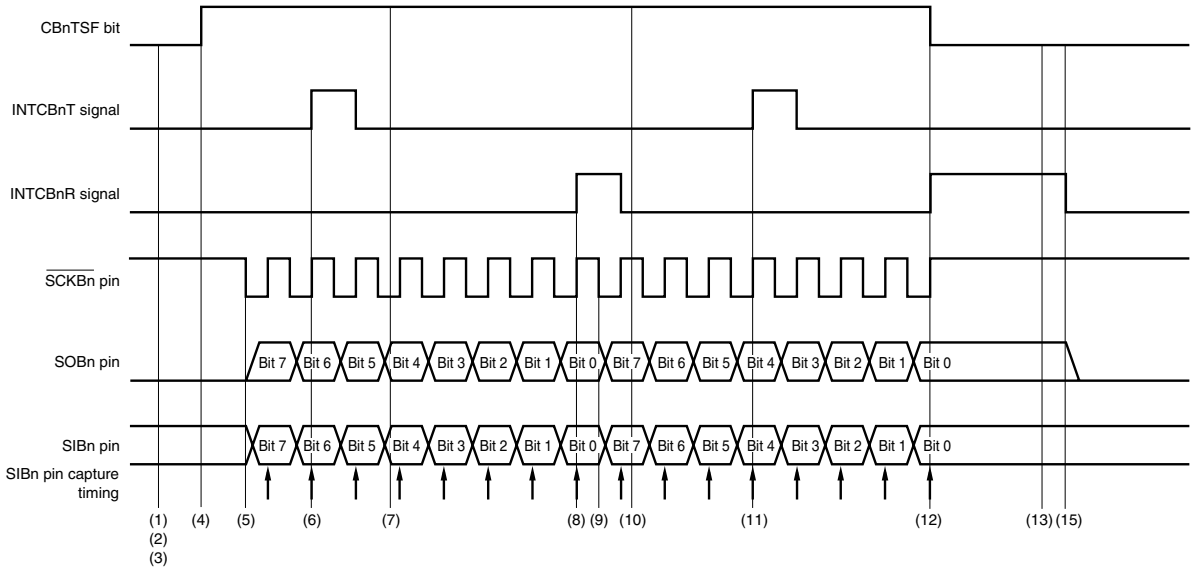
Remarks 1. The broken lines indicate the hardware processing.

2. The numbers in this figure correspond to the processing numbers in (2) Operation timing.

3. n = 0, 1

(2) Operation timing

(1/2)



(1) Write 07H to the CBnCTL1 register, and select communication type 1, communication clock (f_{CCLK}) = external clock ($\overline{SCKBn}$), and slave mode.

(2) Write 00H to the CBnCTL2 register, and set the transfer data length to 8 bits.

(3) Write E3H to the CBnCTL0 register, and select the transmission/reception mode, MSB first, and continuous transfer mode at the same time as enabling the operation of the communication clock (f_{CCLK}).

(4) The CBnSTR.CBnTSF bit is set to 1 by writing the transmit data to the CBnTX register, and the device waits for a serial clock input.

(5) When a serial clock is input, output the transmit data to the SOBn pin in synchronization with the serial clock, and capture the receive data of the SIBn pin.

(6) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the transmission enable interrupt request signal (INTCBnT) is generated.

(7) To continue transmission, write the transmit data to the CBnTX register again after the INTCBnT signal is generated.

(8) When reception of the transfer data length set with the CBnCTL2 register is completed, the reception end interrupt request signal (INTCBnR) is generated, and reading of the CBnRX register is enabled.

(9) When a serial clock is input continuously, continuous transmission/reception is started.

(10) Read the CBnRX register.

(11) When transfer of the transmit data from the CBnTX register to the shift register is completed and writing to the CBnTX register is enabled, the INTCBnT signal is generated. To end continuous transmission/reception with the current transmission/reception, do not write to the CBnTX register.

Remark n = 0, 1

(2/2)

(12) When the clock of the transfer data length set with the CBnCTL2 register is input without writing to the CBnTX register, the INTCBnR signal is generated. Clear the CBnTSF bit to 0 to end transmission/reception.
(13) When the reception error interrupt request signal (INTCBnRE) is generated, read the CBnRX register.
(14) If an overrun error occurs, write the CBnSTR.CBnOVE bit = 0, and clear the error flag.
(15) To release the transmission/reception enable status, write the CBnCTL0.CBnPWR bit = 0, the CBnCTL0.CBnTXE bit = 0, and the CBnCTL0.CBnRXE bit = 0 after checking that the CBnTSF bit = 0.

Remark n = 0, 1

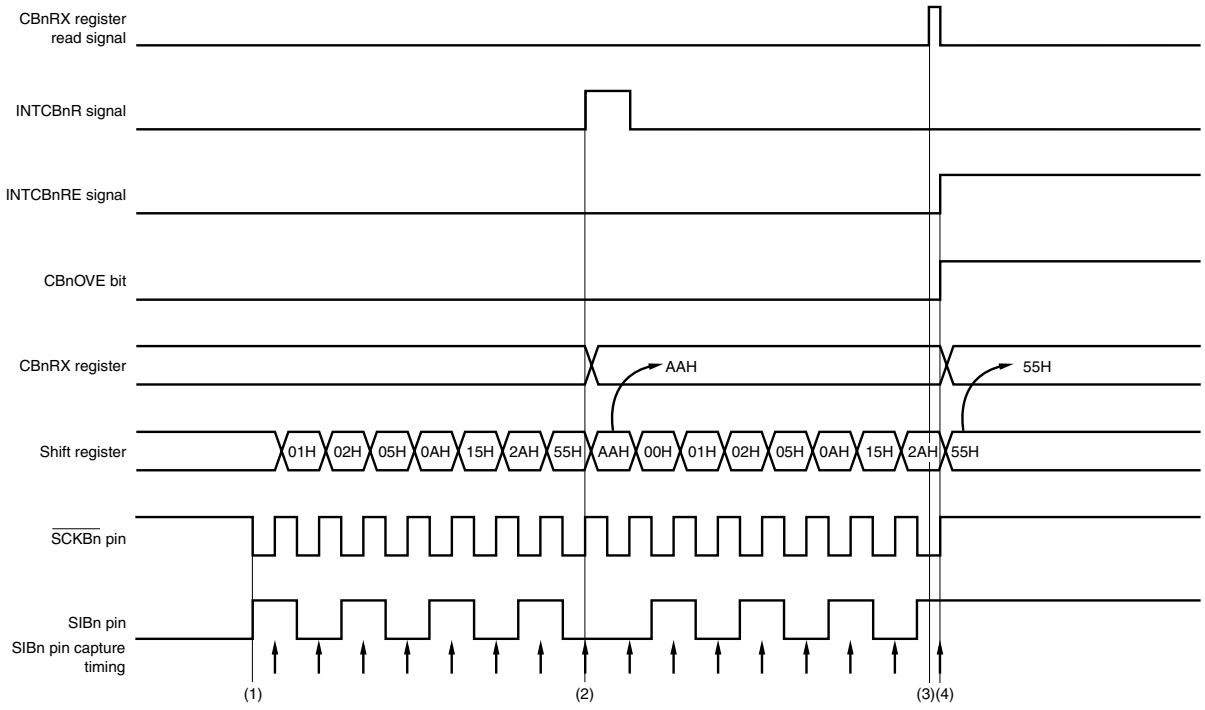
15.5.13 Reception error

When transfer is performed with reception enabled (CBnCTL0.CBnRXE bit = 1) in the continuous transfer mode, the reception error interrupt request signal (INTCBnRE) is generated when the next receive operation is completed before the CBnRX register is read after the reception end interrupt request signal (INTCBnR) is generated, and the overrun error flag (CBnSTR.CBnOVE) is set to 1.

Even if an overrun error has occurred, the previous receive data is lost since the CBnRX register is updated. Even if a reception error has occurred, the INTCBnRE signal is generated again upon the next reception completion if the CBnRX register is not read.

To avoid an overrun error, complete reading the CBnRX register until one half clock before sampling the last bit of the next receive data from the INTCBnR signal generation.

(1) Operation timing



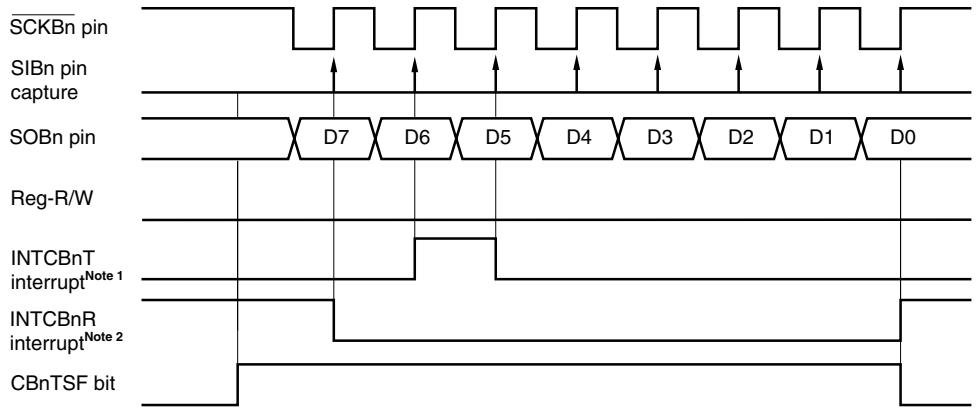
(1) Start continuous transfer.
(2) Completion of the first transfer
(3) The CBnRX register cannot be read until one half clock before the completion of the second transfer.
(4) An overrun error occurs, and the reception error interrupt request signal (INTCBnRE) is generated. The receive data is overwritten.

Remark n = 0, 1

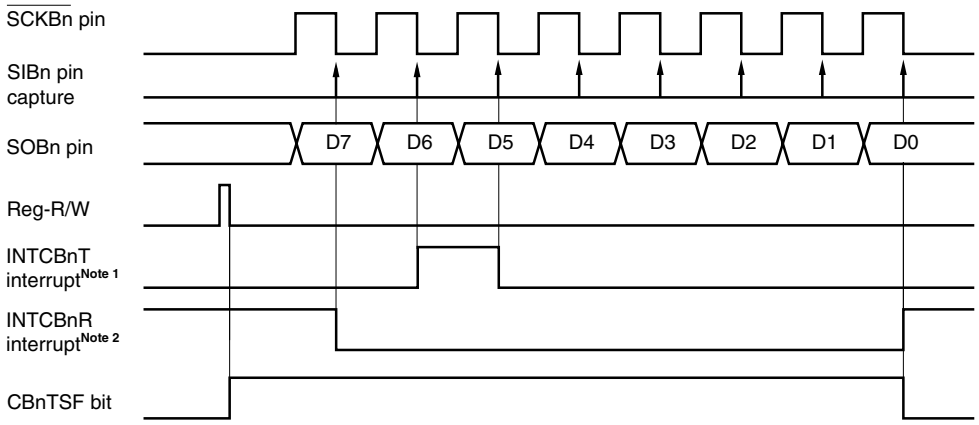
15.5.14 Clock timing

(1/2)

(i) Communication type 1 (CBnCKP and CBnDAP bits = 00)



(ii) Communication type 3 (CBnCKP and CBnDAP bits = 10)



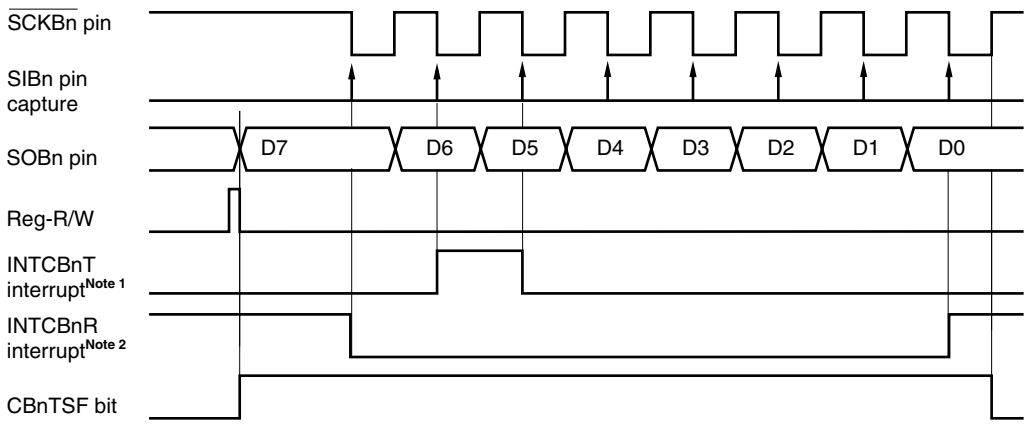
Notes 1. The INTCBnT interrupt is set when the data written to the CBnTX register is transferred to the data shift register in the continuous transmission or continuous transmission/reception mode. In the single transmission or single transmission/reception mode, the INTCBnT interrupt request signal is not generated, but the INTCBnR interrupt request signal is generated upon end of communication.

2. The INTCBnR interrupt occurs if reception is correctly ended and receive data is ready in the CBnRX register while reception is enabled. In the single mode, the INTCBnR interrupt request signal is generated even in the transmission mode, upon end of communication.

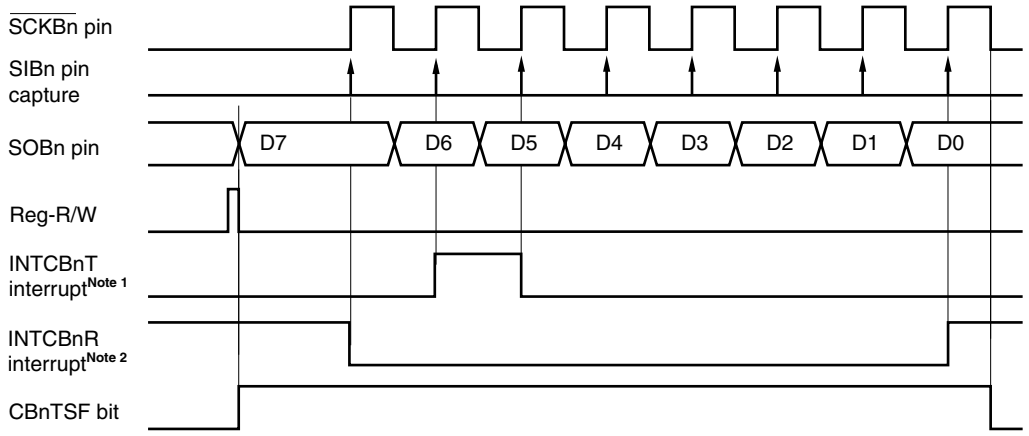
Caution In single transfer mode, writing to the CBnTX register with the CBnTSF bit set to 1 is ignored. This has no influence on the operation during transfer.
For example, if the next data is written to the CBnTX register when DMA is started by generating the INTCBnR signal, the written data is not transferred because the CBnTSF bit is set to 1.
Use the continuous transfer mode, not the single transfer mode, for such applications.

Remark n = 0, 1

(iii) Communication type 2 (CBnCKP and CBnDAP bits = 01)



(iv) Communication type 4 (CBnCKP and CBnDAP bits = 11)



Notes 1. The INTCBnT interrupt is set when the data written to the CBnTX register is transferred to the data shift register in the continuous transmission or continuous transmission/reception mode. In the single transmission or single transmission/reception mode, the INTCBnT interrupt request signal is not generated, but the INTCBnR interrupt request signal is generated upon end of communication.

2. The INTCBnR interrupt occurs if reception is correctly ended and receive data is ready in the CBnRX register while reception is enabled. In the single mode, the INTCBnR interrupt request signal is generated even in the transmission mode, upon end of communication.

Caution In single transfer mode, writing to the CBnTX register with the CBnTSF bit set to 1 is ignored. This has no influence on the operation during transfer.

For example, if the next data is written to the CBnTX register when DMA is started by generating the INTCBnR signal, the written data is not transferred because the CBnTSF bit is set to 1.

Use the continuous transfer mode, not the single transfer mode, for such applications.

Remark n = 0, 1

15.6 Output Pins

(1) $\overline{\text{SCKBn}}$ pin

When CSIBn operation is disabled (CBnCTL0.CBnPWR bit = 0), the $\overline{\text{SCKBn}}$ pin output status is as follows.

Remark n = 0, 1

CBnCKP	CBnCKS2	CBnCKS1	CBnCKS0	$\overline{\text{SCKBn}}$ Pin Output
0	1	1	1	High impedance
	Other than above			Fixed to high level
1	1	1	1	High impedance
	Other than above			Fixed to low level

Remark The output level of the $\overline{\text{SCKBn}}$ pin changes if any of the CBnCTL1.CBnCKP and CBnCKS2 to CBnCKS0 bits is rewritten.

(2) SOBn pin

When CSIBn operation is disabled (CBnPWR bit = 0), the SOBn pin output status is as follows.

Remark n = 0, 1

CBnTXE	CBnDAP	CBnDIR	SOBn Pin Output
0	×	×	Fixed to low level
1	0	×	SOBn latch value (low level)
	1	0	CBnTX value (MSB)
		1	CBnTX value (LSB)

Remarks 1. The SOBn pin output changes when any one of the CBnCTL0.CBnTXE, CBnCTL0.CBnDIR, or CBnCTL1.CBnDAP bits is rewritten.

2. ×: Don't care

CHAPTER 16 DMA FUNCTIONS (DMA CONTROLLER)

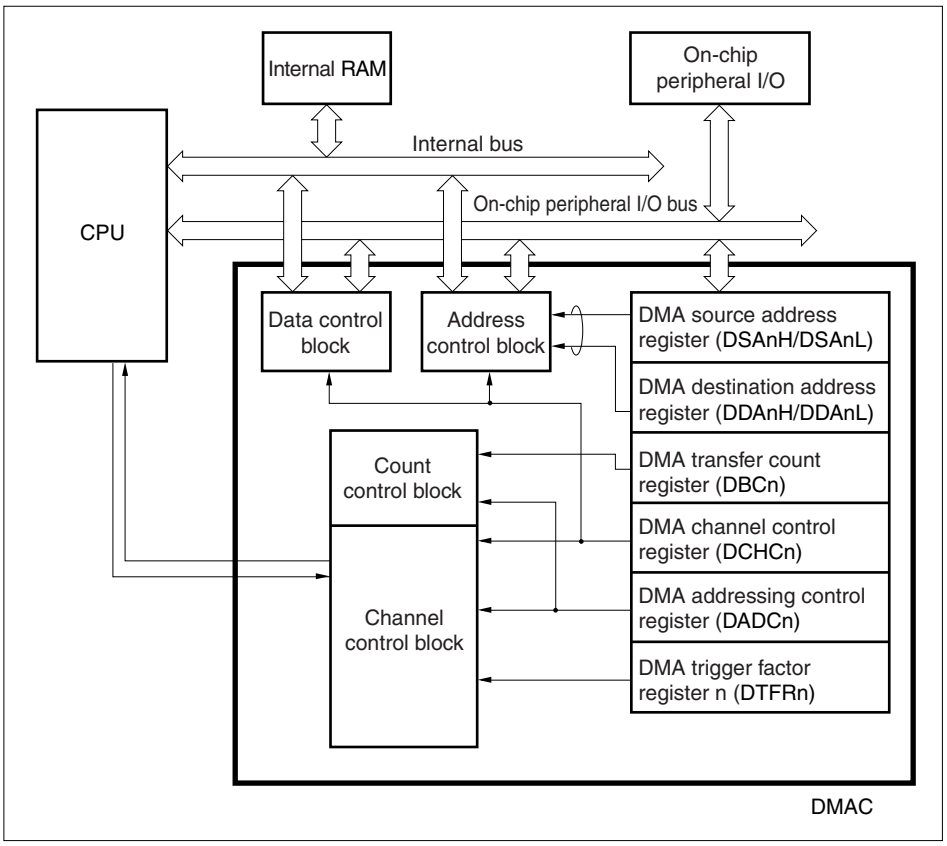
The V850E/IA3 and V850E/IA4 include a direct memory access (DMA) controller (DMAC) that executes and controls DMA transfer.

The DMAC controls data transfers between the internal memory and peripheral I/O, or between peripheral I/Os, based on requests by interrupts from the on-chip peripheral I/O (serial interface, timer, and A/D converter) or DMA requests issued by software triggers.

16.1 Features

- 4 independent DMA channels
- Transfer unit: 8/16 bits
- Maximum transfer count: 65,536 (2^{16})
- Transfer type: 2-cycle transfer
- Three transfer modes
  - Single transfer mode
  - Single-step transfer mode
  - Block transfer mode
- Transfer requests
  - Request by interrupts from on-chip peripheral I/O (serial interface, timer, A/D converter)
  - Requests by software trigger
- Transfer targets
  - Internal memory ↔ peripheral I/O
  - Peripheral I/O ↔ peripheral I/O
- Next address setting function

16.2 Configuration



V850E/IA3,
V850E/IA4

Remark n = 0 to 3

16.3 Control Registers

16.3.1 DMA source address registers 0 to 3 (DSA0 to DSA3)

The DSA0 to DSA3 registers set the DMA transfer source address (28 bits) for DMA channel n (n = 0 to 3). These registers are divided into two 16-bit registers, DSAnH and DSAnL.

Since these registers are configured as 2-stage FIFO buffer registers consisting of the master register and slave register, a new transfer source address for DMA transfer can be specified during DMA transfer (see **16.8 Next Address Setting Function**). When setting the next address, the newly set value of the DSAn register is transferred to the slave register and becomes valid only when DMA transfer has been completed normally and the DCHCn.TCn bit is set to 1, or when the DCHCn.INITn bit is set to 1 (n = 0 to 3). However, the set value of the DSAn register is invalid even when the DCHCn.Enn bit is cleared to 0 to disable DMA transfer and then the DSAn register is set.

(1) DMA source address registers 0H to 3H (DSA0H to DSA3H)

The DSA0H to DSA3H registers can be read or written in 16-bit units.

Reset makes these registers undefined.

Cautions 1. When setting an address of an on-chip peripheral I/O register for the source address, be sure to specify an address between FFFF000H and FFFFFFFH. An address of the on-chip peripheral I/O register image (3FFF000H to 3FFFFFFH) must not be specified.

2. Do not set the DSAnH register while DMA is suspended.

After reset: Undefined R/W Address: DSA0H FFFFF082H, DSA1H FFFFF08AH, DSA2H FFFFF092H, DSA3H FFFFF09AH

DSAnH (n = 0 to 3)

15	14	13	12	11	10	9	8
IRSn	0	0	0	SAn27	SAn26	SAn25	SAn24

7	6	5	4	3	2	1	0
SAn23	SAn22	SAn21	SAn20	SAn19	SAn18	SAn17	SAn16

IRSn	DMA transfer source specification
0	On-chip peripheral I/O
1	Internal RAM

SAn27 to SAn16	Set the DMA transfer source address (A27 to A16). During DMA transfer, these bits store the next DMA transfer source address.

Caution Be sure to clear bits 14 to 12 to "0". If they are set to 1, the operation is not guaranteed.

(2) DMA source address registers 0L to 3L (DSA0L to DSA3L)

The DSA0L to DSA3L registers can be read or written in 16-bit units.

Reset makes these registers undefined.

After reset: Undefined R/W Address: DSA0L FFFFF080H, DSA1L FFFFF088H, DSA2L FFFFF090H, DSA3L FFFFF098H

DSAnL (n = 0 to 3)

15	14	13	12	11	10	9	8
SAn15	SAn14	SAn13	SAn12	SAn11	SAn10	SAn9	SAn8

7	6	5	4	3	2	1	0
SAn7	SAn6	SAn5	SAn4	SAn3	SAn2	SAn1	SAn0

SAn15 to SAn0	Set the DMA transfer source address (A15 to A0). During DMA transfer, these bits store the next DMA transfer source address.

16.3.2 DMA destination address registers 0 to 3 (DDA0 to DDA3)

The DDA0 to DDA3 registers set the DMA transfer destination address (28 bits) for DMA channel n (n = 0 to 3). They are divided into two 16-bit registers, DDAnH and DDAnL.

Since these registers are configured as 2-stage FIFO buffer registers consisting of the master register and slave register, a new transfer destination address for DMA transfer can be specified during DMA transfer (see **16.8 Next Address Setting Function**). When setting the next address, the newly set value of the DDAn register is transferred to the slave register and becomes valid only when DMA transfer has been completed normally and the DCHCn.TCn bit is set to 1, or when the DCHCn.INITn bit is set to 1 (n = 0 to 3). However, the set value of the DDAn register is invalid even when the DCHCn.Enn bit is cleared to 0 to disable DMA transfer and then the DDAn register is set.

(1) DMA destination address registers 0H to 3H (DDA0H to DDA3H)

The DDA0H to DDA3H registers can be read or written in 16-bit units.

Reset makes these registers undefined.

Cautions 1. When setting an address of an on-chip peripheral I/O register for the destination address, be sure to specify an address between FFFF000H and FFFFFFFH. An address of the on-chip peripheral I/O register image (3FFF000H to 3FFFFFFH) must not be specified.

2. Do not set the DDAnH register while DMA is suspended.

After reset: Undefined R/W Address: DDA0H FFFFF086H, DDA1H FFFFF08EH, DDA2H FFFFF096H, DDA3H FFFFF09EH

DDAnH (n = 0 to 3)

15	14	13	12	11	10	9	8
IRAn	0	0	0	DAn27	DAn26	DAn25	DAn24

7	6	5	4	3	2	1	0
DAn23	DAn22	DAn21	DAn20	DAn19	DAn18	DAn17	DAn16

IRAn	DMA transfer destination specification
0	On-chip peripheral I/O
1	Internal RAM

DAn27 to DAn16	Set the DMA transfer destination address (A27 to A16). During DMA transfer, these bits store the next DMA transfer destination address.

Caution Be sure to clear bits 14 to 12 to "0". If they are set to 1, the operation is not guaranteed.

(2) DMA destination address registers 0L to 3L (DDA0L to DDA3L)

The DDA0L to DDA3L registers can be read or written in 16-bit units.

Reset makes these registers undefined.

After reset: Undefined R/W Address: DDA0L FFFFF084H, DDA1L FFFFF08CH, DDA2L FFFFF094H, DDA3L FFFFF09CH

DDAnL (n = 0 to 3)

15	14	13	12	11	10	9	8
DAn15	DAn14	DAn13	DAn12	DAn11	DAn10	DAn9	DAn8

7	6	5	4	3	2	1	0
DAn7	DAn6	DAn5	DAn4	DAn3	DAn2	DAn1	DAn0

DAn15 to DAn0	Set the DMA transfer destination address (A15 to A0). During DMA transfer, these bits store the next DMA transfer destination address.

16.3.3 DMA transfer count registers 0 to 3 (DBC0 to DBC3)

The DBC0 to DBC3 registers are 16-bit registers that set the byte transfer count for DMA channel n (n = 0 to 3). These registers store the remaining transfer count during DMA transfer.

Since these registers are configured as 2-stage FIFO buffer registers consisting of the master register and slave register, a new DMA byte transfer count for DMA transfer can be specified during DMA transfer (see **16.8 Next Address Setting Function**). When setting the next address, the newly set value of the DBCn register is transferred to the slave register and becomes valid only when DMA transfer has been completed normally and the DCHCn.TCn bit is set to 1, or when the DCHCn.INITn bit is set to 1 (n = 0 to 3). However, the set value of the DBCn register is invalid even when the DCHCn.Enn bit is cleared to 0 to disable DMA transfer and then the DBCn register is set.

These registers are decremented by 1 for each transfer, and transfer ends when a borrow occurs.

These registers can be read or written in 16-bit units.

Reset makes these registers undefined.

Caution Do not set the DBCn register while DMA is suspended.

Remark If the DBCn register is read during DMA transfer after a terminal count has occurred without the register being overwritten, the value set immediately before the DMA transfer will be read out (0000H will not be read, even if DMA transfer has ended).

After reset: Undefined R/W Address: DBC0 FFFFF0C0H, DBC1 FFFFF0C2H, DBC2 FFFFF0C4H, DBC3 FFFFF0C6H

DBCn (n = 0 to 3)

15	14	13	12	11	10	9	8
BCn15	BCn14	BCn13	BCn12	BCn11	BCn10	BCn9	BCn8

7	6	5	4	3	2	1	0
BCn7	BCn6	BCn5	BCn4	BCn3	BCn2	BCn1	BCn0

BCn15 to BCn0	Transfer count setting (store remaining transfer count during DMA transfer)
0000H	Transfer count 1 or remaining transfer count
0001H	Transfer count 2 or remaining transfer count
:	:
FFFFH	Transfer count 65,536 (2^{16}) or remaining transfer count

16.3.4 DMA addressing control registers 0 to 3 (DADC0 to DADC3)

The DADC0 to DADC3 registers are 16-bit registers that control the DMA transfer mode for DMA channel n (n = 0 to 3). These registers cannot be accessed during a DMA operation.

These registers can be read or written in 16-bit units.

Reset sets these registers to 0000H.

Cautions 1. The DSn0 bit sets how many bits of data is to be transferred.
If the transfer data size is set to 16 bits, transfer is always started from an address with the lowest bit of the address aligned to "0". In this case, transfer cannot be started from an odd address.

2. Set the DADCn register when the target channel is in one of the following periods (the operation is not guaranteed if the register is set at any other time).

- **Period from system reset to the generation of the first DMA transfer request**
- **Period from completion of DMA transfer (after terminal count) to the generation of the next DMA transfer request**
- **Period from forced termination of DMA transfer (after the DCHCn.INITn bit was set to 1) to the generation of the next DMA transfer request**

After reset: 0000H R/W Address: DADC0 FFFFF0D0H, DADC1 FFFFF0D2H, DADC2 FFFFF0D4H, DADC3 FFFFF0D6H

<R> DADCn (n = 0 to 3)

15	14	13	12	11	10	9	8
0	DSn0	0	0	0	0	0	0

7	6	5	4	3	2	1	0
SADn1	SADn0	DADn1	DADn0	TMn1	TMn0	0	0

DSn0	Setting of transfer data size for DMA transfer
0	8 bits
1	16 bits

SADn1	SADn0	Setting of count direction of transfer source address for DMA channel n
0	0	Increment
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited

DADn1	DADn0	Setting of count direction of transfer destination address for DMA channel n
0	0	Increment
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited

TMn1	TMn0	Setting of transfer mode during DMA transfer
0	0	Single transfer mode
0	1	Single-step transfer mode
1	0	Setting prohibited
1	1	Block transfer mode

<R> **Caution Be sure to clear bits 15, 13 to 8, 1, and 0 to “0”. If they are set to 1, the operation is not guaranteed.**

16.3.5 DMA channel control registers 0 to 3 (DCHC0 to DCHC3)

The DCHC0 to DCHC3 registers are 8-bit registers that control the DMA transfer operating mode for DMA channel n (n = 0 to 3).

These registers can be read or written in 8-bit or 1-bit units. (However, bit 7 is read-only.)

Reset sets these registers to 00H.

Cautions 1. If transfer has been completed with the MLEn bit set to 1 and if the next transfer request is made by DMA transfer (hardware DMA) that is started by an interrupt from an on-chip peripheral I/O, the next transfer is executed with the TCn bit set to 1 (not automatically cleared to 0).

2. Set the MLEn bit when the target channel is in one of the following periods (the operation is not guaranteed if the bit is set at any other time).

- **Period from system reset to the generation of the first DMA transfer request**
- **Period from completion of DMA transfer (after terminal count) to the generation of the next DMA transfer request**
- **Period from forced termination of DMA transfer (after the INITn bit was set to 1) to the generation of the next DMA transfer request**

3. If DMA transfer is forcibly terminated in the last transfer cycle with the MLEn bit set to 1, the operation is performed in the same manner as when transfer is completed (the TCn bit is set to 1). (The Enn bit is cleared to 0 upon forced termination, regardless of the value of the MLEn bit.)

In this case, the Enn bit must be set to 1 and the TCn bit must be read (cleared to 0) when the next DMA transfer request is made.

4. Upon completion of DMA transfer (during terminal count), each bit is updated with the Enn bit cleared to 0 and then the TCn bit set to 1. If the statuses of the TCn bit and Enn bit are polled and if the DCHCn register is read while each bit is updated, therefore, a value indicating the status "transfer not completed and prohibited" (TCn bit = 0 and Enn bit = 0) may be read (this is not abnormal).

<R> **5. Be sure to read (clear to 0) the TCn bit after end of DMA transfer (after terminal count). The TCn bit does not have to be read (cleared to 0) only if the following two conditions are satisfied.**

- **The MLEn bit is set to 1 upon end of DMA transfer (during terminal count).**
- **The next DMA transfer start factor is an interrupt from the on-chip peripheral I/O (hardware DMA)**

If even one of these conditions is not satisfied, be sure to read (clear to 0) the TCn bit before the next DMA transfer request is generated.

The operation cannot be guaranteed if the next DMA transfer request is generated while the TCn bit is set to 1.

6. Do not set the Enn and STGn bits while DMA is suspended. Otherwise, the operation is not guaranteed.

<R> **7. Do not end DMA transfer by clearing the Enn bit to 0.**

<R> **8. The relationship between the status of DMA transfer and the register value is as follows.**

- **DMA transfer is in progress: TCn bit = 0, Enn bit = 1**
- **DMA transfer is aborted: TCn bit = 0, Enn bit = 0**
- **DMA transfer is stopped (ended): TCn bit = 1**

After reset: 00H R/W Address: DCHC0 FFFFF0E0H, DCHC1 FFFFF0E2H, DCHC2 FFFFF0E4H, DCHC3 FFFFF0E6H

	<7>	6	5	4	<3>	<2>	<1>	<0>
DCHCn	TCn	0	0	0	MLEn	INITn	STGn	Enn

(n = 0 to 3)

TCn[Note 1]	Status bit that indicates whether DMA transfer via DMA channel n has ended or not
0	DMA transfer has not ended.
1	DMA transfer has ended.
This bit is set (1) at the last DMA transfer and cleared (0) when it is read. If DMA transfer is executed to transfer data from the internal RAM, this bit is set (1) 4 clocks after completion of the last transfer.	

MLEn	When this bits is set (1) at DMA transfer completion (at the terminal count output), the Enn bit is not cleared (0) and the DMA transfer enabled state is retained. If the next DMA transfer start factor is input from an on-chip peripheral I/O (hardware DMA), the DMA transfer request is acknowledged even if the TCn bit is not read. If the next DMA transfer start factor is input by setting the STGn bit to 1 (software DMA), the DMA transfer request is acknowledged if the TCn bit is read and cleared (0). When this bit is cleared (0) at DMA transfer completion (at the terminal count output), the Enn bit is cleared (0) and the DMA transfer disabled state is entered. At the next DMA transfer request, the TCn bit must be read and the Enn bit must be set (1).

INITn[Note 2]	If this bit is set (1) during DMA transfer or while DMA is suspended, DMA transfer is forcibly terminated.

STGn[Note 2]	If this bit is set (1) in the DMA transfer enabled state (TCn bit = 0, Enn bit = 1), DMA transfer is started.

Enn	Setting whether DMA transfer via DMA channel n is to be enabled or disabled
0	DMA transfer disabled
1	DMA transfer enabled
• This bit is cleared (0) when DMA transfer ends. It is also cleared (0) when DMA transfer is forcibly terminated by setting (1) the INITn bit. • If the Enn bit is set (1), do not set it until DMA transfer has been completed the number of times set by the DBCn register or DMA transfer is forcibly terminated by the INITn bit.	

Notes 1. TCn bit is read-only.

2. INITn and STGn bits are write-only. If these bits are read, 0 is read.

Caution Be sure to clear bits 6 to 4 to "0". If they are set to 1, the operation is not guaranteed.

16.3.6 DMA trigger factor registers 0 to 3 (DTFR0 to DTFR3)

The DTFR0 to DTFR3 registers are 8-bit registers that control the DMA transfer start trigger via interrupt requests from on-chip peripheral I/O.

The interrupt requests set by these registers serve as DMA transfer start factors.

These registers can be read or written in 8-bit or 1-bit units. However, only bit 7 (DFn) can be read or written in 1-bit units; bits 5 to 0 (IFCn5 to IFCn0) can only be read or written in 8-bit units.

Reset sets these registers to 00H.

<R> **Cautions 1. Be sure to follow the steps below when changing the DTFRn register settings.**

- **When the values to be set to the IFCn5 to IFCn0 bits are not set to the IFCm5 to IFCm0 bits of another channel (n = 0 to 3, m = 0 to 3, n ≠ m)**
  - **<1> Follow steps <3> to <5> when the DCHCn.Enn bit is cleared to 0, and follow steps <2> to <5> when the Enn bit is set to 1.**
  - **<2> Stop the DMAn operation of the channel to be rewritten (DCHCn.INITn bit = 1).**
  - **<3> Change the DTFRn register settings. (Be sure to set the DFn bit to 0 and change the settings in the 8-bit manipulation.)**
  - **<4> To clear a DMA transfer request, clear the DMA transfer request flag (DTFRn.DFn bit) to 0.**
  - **<5> Enable the DMAn operation (Enn bit = 1).**
- **When the values to be set to the IFCn5 to IFCn0 bits are set to the IFCm5 to IFCm0 bits of another channel (n = 0 to 3, m = 0 to 3, n ≠ m)**
  - **<1> Follow steps <4> to <6> when the DCHCn.Enn bit is cleared to 0, and follow steps <2> to <6> when the Enn bit is set to 1.**
  - **<2> Stop the DMAn operation of the channel to be rewritten (DCHCn.INITn bit = 1).**
  - **<3> Stop the DMAm operation of the channel where the same values are set to the IFCm5 to IFCm0 bits as the values to be used to rewrite the IFCn5 to IFCn0 bits (DCHCm.INITm bit = 0).**
  - **<4> Change the DTFRn register settings. (Be sure to set the DFn bit to 0 and change the settings in the 8-bit manipulation.)**
  - **<5> To clear a DMA transfer request, clear the DMA transfer request flag (DTFRn.DFn bit) to 0.**
  - **<6> Enable the DMAn operation (Enn and Emm bits = 1).**

2. An interrupt request from an on-chip peripheral I/O input in the standby mode (IDLE or STOP mode) is held pending as a DMA transfer start factor. The held DMA start factor is executed after restoring to the normal operation mode.

3. If the start factor of DMA transfer is changed using the IFCn5 to IFCn0 bits, be sure to clear (0) the DFn bit by instruction immediately after.

After reset: 00H R/W Address: DTFR0 FFFFF810H, DTFR1 FFFFF812H, DTFR2 FFFFF814H, DTFR3 FFFFF816H

	<7>	6	5	4	3	2	1	0
DTFRn	DFn	0	IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0

(n = 0 to 3)

DFn[Note]	DMA transfer request status flag
0	DMA transfer not requested
1	DMA transfer requested

Note Do not set the DFn bit to “1” by software.
If the interrupt specified as the DMA transfer start factor occurs and it is necessary to clear the DMA transfer request while DMA transfer is disabled (including when it is forcibly terminated by software), stop the operation of the source causing the interrupt, and then write 0 to the DFn bit (for example, disable reception in the case of serial reception). If it is clear that the interrupt will not occur until DMA transfer is resumed next, it is not necessary to stop the operation of the source causing the interrupt.

Cautions 1. For the IFCn5 to IFCn0 bits, see Table 16-1 DMA Start Factors.
2. Be sure to clear bit 6 to “0”. If it is set to 1, the operation is not guaranteed.

Table 16-1. DMA Start Factors (1/2)

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	Interrupt Source
0	0	0	0	0	0	DMA request from on-chip peripheral I/O disabled
0	0	0	0	0	1	INTP6
0	0	0	0	1	0	INTP7
0	0	0	0	1	1	INTCMP0
0	0	0	1	0	0	INTCMP1
0	0	0	1	0	1	INTTQ0OV
0	0	0	1	1	0	INTTQ0CC0
0	0	0	1	1	1	INTTQ0CC1
0	0	1	0	0	0	INTTQ0CC2
0	0	1	0	0	1	INTTQ0CC3
0	0	1	0	1	0	INTTQ1OV
0	0	1	0	1	1	INTTQ1CC0
0	0	1	1	0	0	INTTQ1CC1
0	0	1	1	0	1	INTTQ1CC2
0	0	1	1	1	0	INTTQ1CC3
0	0	1	1	1	1	INTCC00
0	1	0	0	0	0	INTCC01
0	1	0	0	0	1	INTCM00
0	1	0	0	1	0	INTCM01
0	1	0	0	1	1	INTCC10[Note]
0	1	0	1	0	0	INTCC11[Note]
0	1	0	1	0	1	INTCM10[Note]
0	1	0	1	1	0	INTCM11[Note]
0	1	0	1	1	1	INTTP0OV
0	1	1	0	0	0	INTTP0CC0
0	1	1	0	0	1	INTTP0CC1
0	1	1	0	1	0	INTTP1OV
0	1	1	0	1	1	INTTP1CC0
0	1	1	1	0	0	INTTP1CC1
0	1	1	1	0	1	INTTP2OV
0	1	1	1	1	0	INTTP2CC0
0	1	1	1	1	1	INTTP2CC1
1	0	0	0	0	0	INTTP3OV
1	0	0	0	0	1	INTTP3CC0
1	0	0	0	1	0	INTTP3CC1
1	0	0	0	1	1	INTDMA0
1	0	0	1	0	0	INTDMA1
1	0	0	1	0	1	INTDMA2
1	0	0	1	1	0	INTDMA3

Note V850E/IA4 only

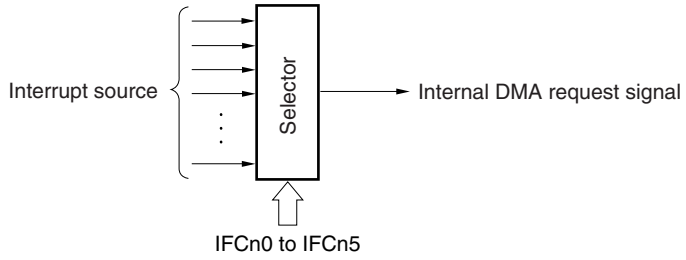
Remark n = 0 to 3

Table 16-1. DMA Start Factors (2/2)

IFCn5	IFCn4	IFCn3	IFCn2	IFCn1	IFCn0	Interrupt Source
1	0	0	1	1	1	INTUA0RE
1	0	1	0	0	0	INTUA0R
1	0	1	0	0	1	INTUA0T
1	0	1	0	1	0	INTCB0RE
1	0	1	0	1	1	INTCB0R
1	0	1	1	0	0	INTCB0T
1	0	1	1	0	1	INTUA1RE
1	0	1	1	1	0	INTUA1R
1	0	1	1	1	1	INTUA1T
1	1	0	0	0	0	INTCB1RE
1	1	0	0	0	1	INTCB1R
1	1	0	0	1	0	INTCB1T
1	1	0	0	1	1	INTAD0
1	1	0	1	0	0	INTAD1
1	1	0	1	0	1	INTAD2
1	1	0	1	1	0	INTTM0EQ0
Other than above						Setting prohibited

Remark n = 0 to 3

The relationship between the interrupt source and the DMA transfer trigger is as follows (n = 0 to 3).



Cautions 1. DMA transfer starts by interrupt factor setting in IFCn5 to IFCn0 bits. To prevent an interrupt from being executed, mask the interrupt by setting the interrupt control register. DMA transfer starts even if an interrupt is masked.

2. If the frequency of the CPU clock falls below the clock of each on-chip peripheral I/O because of the setting of prescaler 2 of the clock generator, the DMA transfer start factor may not be acknowledged.

16.4 Transfer Modes

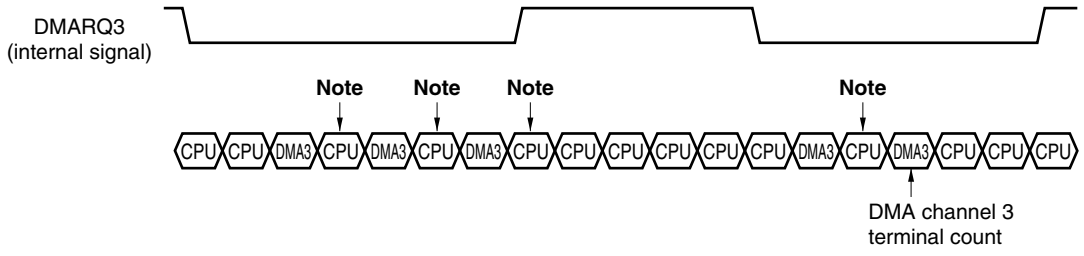
16.4.1 Single transfer mode

In single transfer mode, the DMAC releases the bus at each byte/halfword transfer. If there is a subsequent DMA transfer request, transfer is performed again once. This operation continues until a terminal count occurs.

When the DMAC has released the bus, if another higher priority DMA transfer request is issued, the higher priority DMA request always takes precedence. If another DMA transfer request with a lower priority occurs one clock after single transfer has been completed, however, this request does not take precedence even if the previous DMA transfer request signal with a higher priority remains active. DMA transfer with the newly requested lower priority request is executed after the CPU bus has been released.

Figures 16-1 to 16-4 show examples of single transfer.

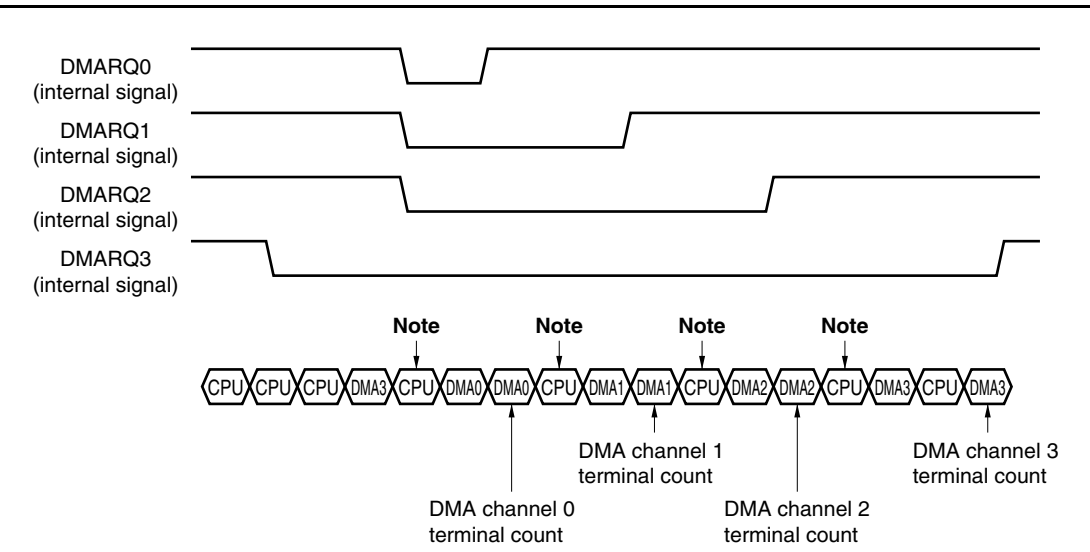
Figure 16-1. Single Transfer Example 1



Note The bus is always released.

Figure 16-2 shows an example of a single transfer in which a higher priority DMA request is issued. DMA channels 0 to 2 are in the block transfer mode and channel 3 is in the single transfer mode.

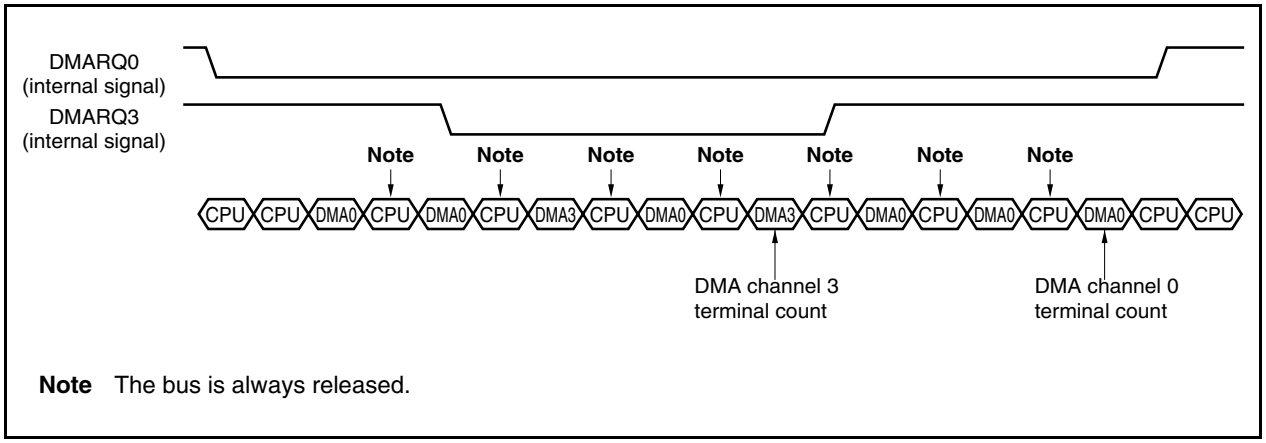
Figure 16-2. Single Transfer Example 2



Note The bus is always released.

Figure 16-3 is an example of single transfer where a DMA transfer request with a lower priority is issued one clock after single transfer has been completed. DMA channels 0 and 3 are used for single transfer. If two DMA transfer request signals become active at the same time, two DMA transfer operations are alternately executed.

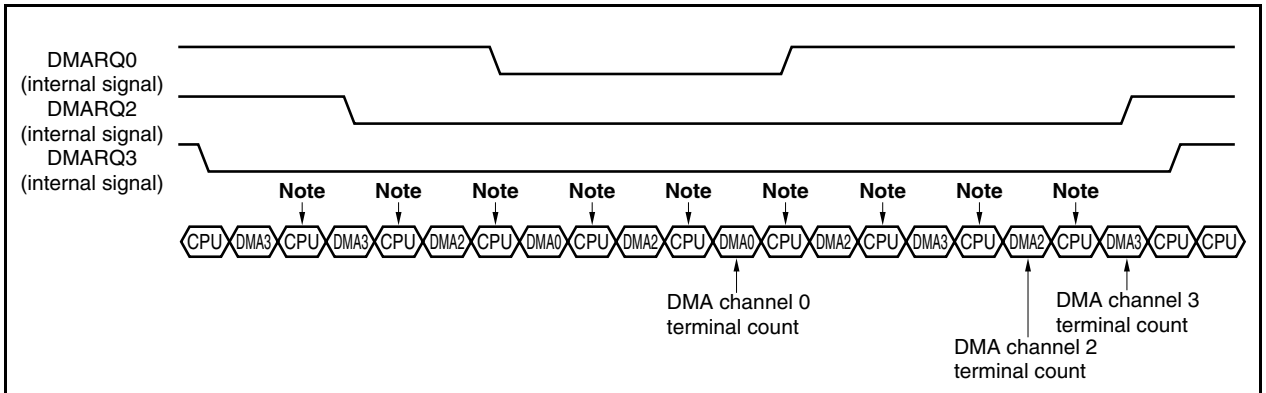
Figure 16-3. Single Transfer Example 3



Note The bus is always released.

Figure 16-4 is an example of single transfer where two or more DMA transfer requests with a lower priority are issued one clock after single transfer has been completed. DMA channels 0, 2, and 3 are used for single transfer. If three or more DMA transfer request signals become active at the same time, two DMA transfer operations are alternately executed, starting from the one with the highest priority.

Figure 16-4. Single Transfer Example 4



Note The bus is always released.

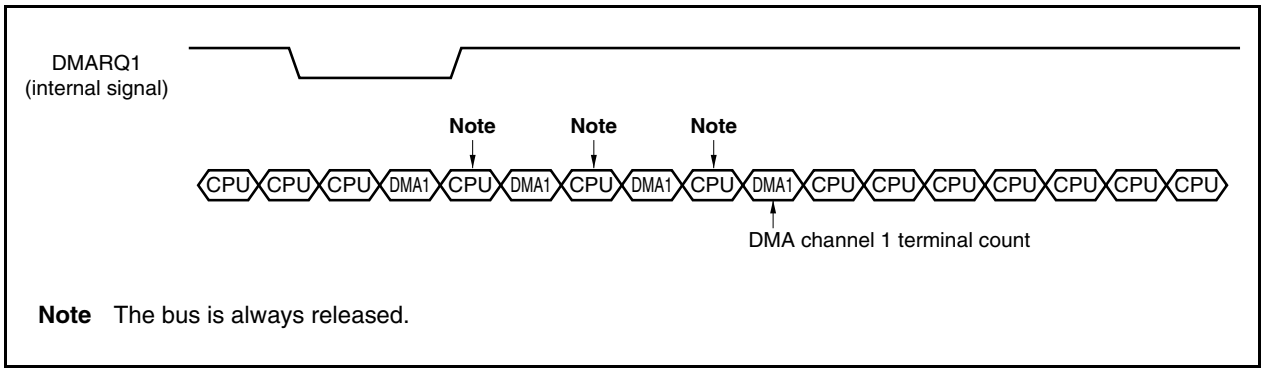
16.4.2 Single-step transfer mode

In single-step transfer mode, the DMAC releases the bus at each byte/halfword transfer. If there is a subsequent DMA transfer request signal, transfer is performed again. This operation continues until a terminal count occurs.

When the DMAC has released the bus, if another higher priority DMA transfer request is issued, the higher priority DMA request always takes precedence.

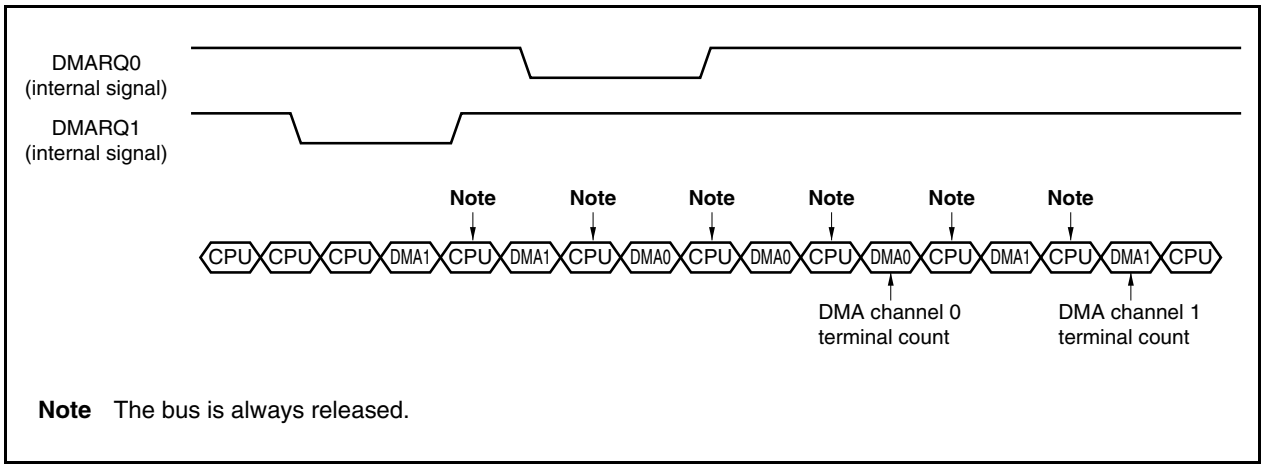
The following shows an example of a single-step transfer. Figure 16-6 shows an example of single-step transfer made in which a higher priority DMA request is issued. DMA channels 0 and 1 are in the single-step transfer mode.

Figure 16-5. Single-Step Transfer Example 1



Note The bus is always released.

Figure 16-6. Single-Step Transfer Example 2



Note The bus is always released.

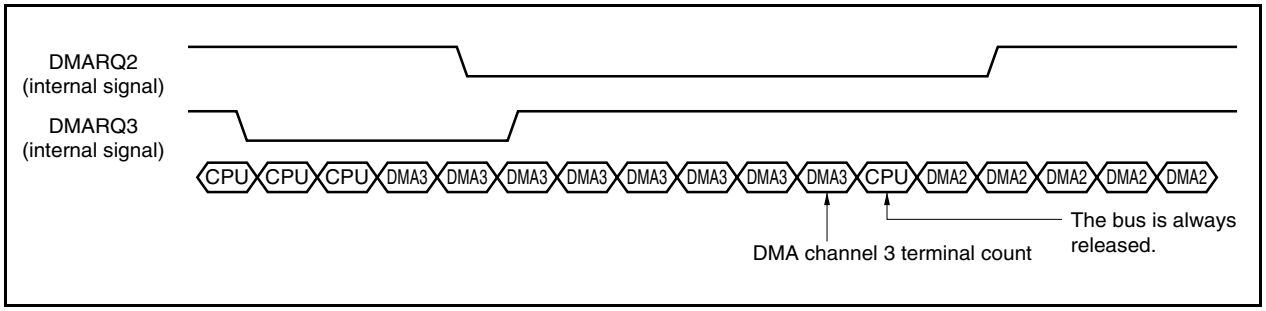
16.4.3 Block transfer mode

In the block transfer mode, once transfer starts, the DMAC continues the transfer operation without releasing the bus until a terminal count occurs. No other DMA requests are acknowledged during block transfer.

After the block transfer ends and the DMAC releases the bus, another DMA transfer can be acknowledged.

The following shows an example of block transfer in which a higher priority DMA request is issued. DMA channels 2 and 3 are in the block transfer mode.

Figure 16-7. Block Transfer Example



16.5 Transfer Types

16.5.1 2-cycle transfer

In 2-cycle transfer, data transfer is performed in two cycles, a read cycle (source to DMAC) and a write cycle (DMAC to destination).

In the first cycle, the source address is output and reading is performed from the source to the DMAC. In the second cycle, the destination address is output and writing is performed from the DMAC to the destination.

Caution An idle cycle of 1 to 2 clocks is always inserted between a read cycle and a write cycle.

16.6 Transfer Target

16.6.1 Transfer type and transfer target

Table 16-2 lists the relationship between the transfer type and transfer target. The mark "√" means "transfer possible", and the mark "×" means "transfer impossible".

Table 16-2. Relationship Between Transfer Type and Transfer Target

		Destination		
		Internal ROM	On-Chip Peripheral I/O[Note]	Internal RAM
Source	On-chip peripheral I/O[Note]	×	√	√
	Internal RAM	×	√	×
	Internal ROM	×	×	×

Note If the transfer target is the on-chip peripheral I/O, only the single transfer mode can be used.

Cautions 1. The operation is not guaranteed for combinations of transfer destination and source marked with "×" in Table 16-2.

2. Addresses between 3FFF000H and 3FFFFFFH cannot be specified for the source and destination address of DMA transfer. Be sure to specify an address between FFFF000H and FFFFFFFH.

Remark If DMA transfer is executed to transfer data of an on-chip peripheral I/O register (as a transfer source or destination), be sure to specify the same transfer size as the register size. For example, to execute DMA transfer of an 8-bit register, be sure to specify byte (8-bit) transfer.

16.7 DMA Channel Priorities

The DMA channel priorities are fixed as follows.

DMA channel 0 > DMA channel 1 > DMA channel 2 > DMA channel 3

In the block transfer mode, the channel used for transfer is never switched.

In the single-step transfer mode, if a higher priority DMA transfer request is issued while the bus is released, the higher priority DMA transfer request is acknowledged.

16.8 Next Address Setting Function

The DSAnH, DSAnL, DDAnH, DDAnL, and DBCn registers are two-stage FIFO buffer registers consisting of a master register and a slave register (n = 0 to 3).

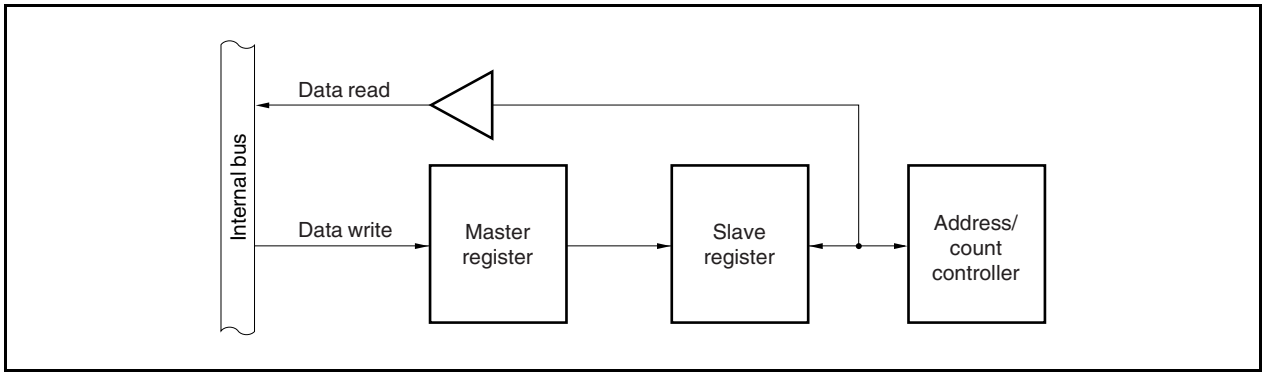
When the terminal count is issued, these registers are automatically rewritten with the value that was set immediately before.

If new DMA transfer setting is made to these registers during DMA transfer, therefore, the values of the registers are automatically updated to the new value after completion of transfer[Note].

Note To make new DMA transfer setting, confirm that DMA transfer has been started. If a new setting is made before the start of DMA transfer, the set value is overwritten to both the master and slave registers.

Figure 16-8 shows the configuration of the buffer register.

Figure 16-8. Buffer Register Configuration



The actual DMA transfer is executed in accordance with the contents of the slave register.

The set value to be reflected upon the master register and slave register differs as follows, depending on the timing (period) of setting.

(1) Period from system reset to the generation of the first DMA transfer request

The set values are reflected on both the master and slave registers.

(2) During DMA transfer (period from the generation of DMA transfer request to completion of DMA transfer)

The set value is reflected only on the master register and not on the slave register (the slave register holds the set value for the next DMA transfer).

After completion of DMA transfer, however, the contents of the master register are automatically overwritten to the slave register.

If the value of a register is read during this period, the value of the slave register is read.

To check that DMA transfer has been started, confirm that the first transfer has been executed by reading the DBCn register (n = 0 to 3).

(3) Period from completion of DMA transfer to start of next DMA transfer

The set value is reflected on both the master and slave registers.

Remark "Completion of DMA transfer" means either of the following cases.

- Completion of DMA transfer (terminal count)
- Forced termination of DMA transfer (setting DCHCn.INITn bit to 1).

16.9 DMA Transfer Start Factors

There are two types of DMA transfer start factors, as shown below.

Cautions 1. Do not use both start factors ((1) and (2)) in combination for the same channel (if both start factors are generated at the same time, only one of them is valid, but the valid start factor cannot be identified).

The operation is not guaranteed if both start factors are used in combination.

2. If DMA transfer is started via software and if the software does not correctly detect whether the expected DMA transfer operation has been completed through manipulation (setting to 1) of the DCHCn.STGn bit, it cannot be guaranteed whether the next (second) manipulation of the STGn bit corresponds to the start of "the next DMA transfer expected by software" (n = 0 to 3).

For example, suppose single transfer is started by manipulating the STGn bit. Even if the STGn bit is manipulated next (the second time) without checking by software whether the single transfer has actually been executed, the next (second) DMA transfer is not always executed. This is because the STGn bit may be manipulated the second time before the first DMA transfer is started or completed because, for example, DMA transfer with a higher priority had already been started when the STGn bit was manipulated for the first time.

It is therefore necessary to manipulate the STGn bit the next time (the second time) after checking whether DMA transfer started by the first manipulation of the STGn bit has been completed.

Completion of DMA transfer can be checked by checking the contents of the DBCn register.

(1) Request from software

If the DCHCn.STGn, DCHCn.Enn, and DCHCn.TCn bits are set as follows, DMA transfer starts (n = 0 to 3).

- STGn bit = 1
- Enn bit = 1
- TCn bit = 0

(2) Request from on-chip peripheral I/O

If, when the DCHCn.Enn and DCHCn.TCn bits are set as shown below, an interrupt request is issued from the on-chip peripheral I/O that is set in the DTFRn register, DMA transfer starts (n = 0 to 3).

- Enn bit = 1
- TCn bit = 0

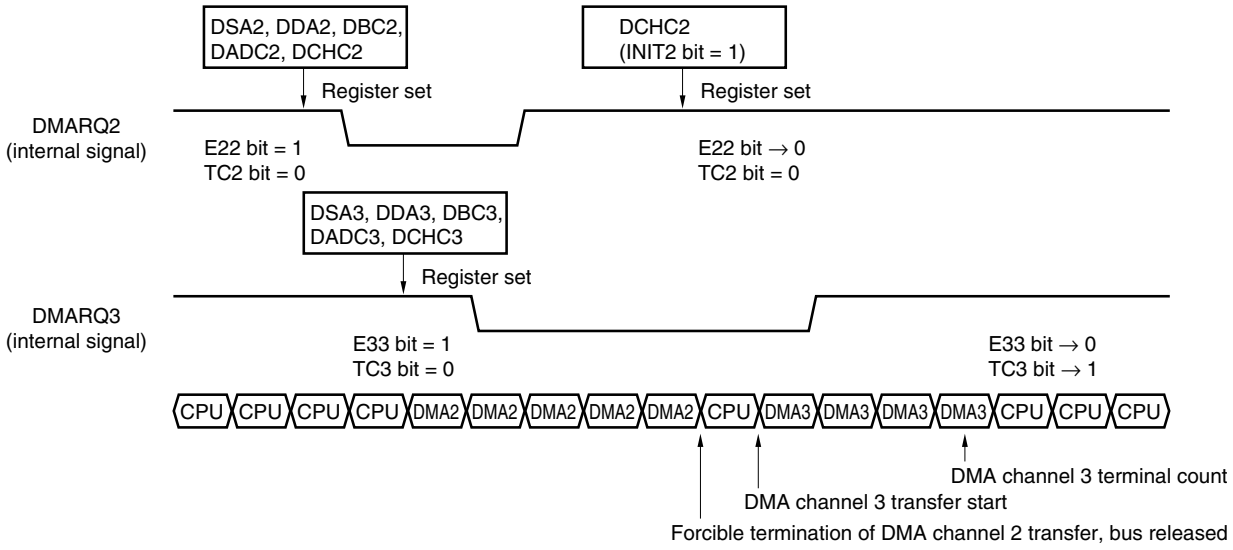
16.10 Forcible Termination

DMA transfer can be forcibly terminated by the DCHCn.INITn bit (n = 0 to 3).

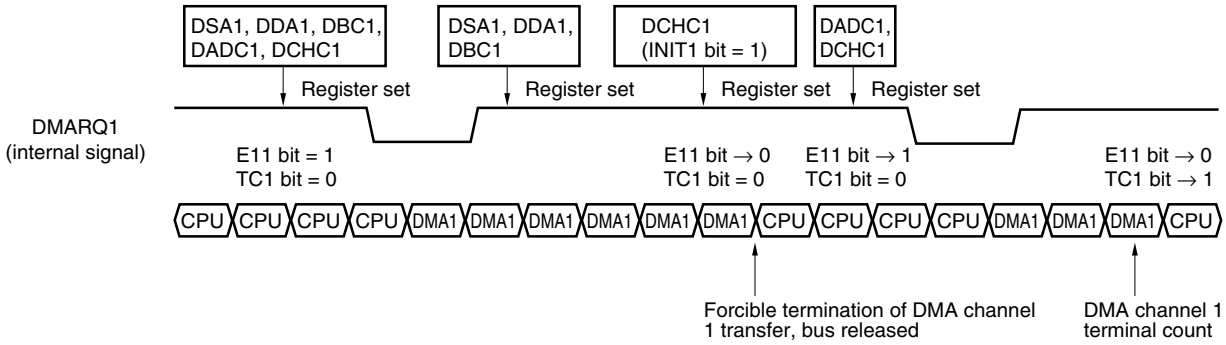
An example of forcible termination by the DCHCn.INITn bit is illustrated below (n = 0 to 3).

Figure 16-9. Example of Forcible Termination of DMA Transfer

(a) Block transfer via DMA channel 3 is started during block transfer via DMA channel 2



(b) When transfer is suspended during DMA channel 1 block transfer, and transfer under another condition is executed



Remark The values of the DSAn, DDAn, and DBCn registers (n = 0 to 3) are retained even when DMA transfer is forcibly terminated, because these registers are FIFO-format buffer registers. The next transfer condition can be set to these registers even while DMA transfer is in progress. On the other hand, the setting of the DADCn and DCHCn registers is invalid during DMA transfer because these registers are not buffer registers (see **16.8 Next Address Setting Function**, **16.3.4 DMA addressing control registers 0 to 3 (DADC0 to DADC3)**, and **16.3.5 DMA channel control registers 0 to 3 (DCHC0 to DCHC3)**).

16.11 Times Related to DMA Transfer

The overhead before and after DMA transfer and minimum execution clock for DMA transfer are shown below.

Table 16-3. Number of Minimum Execution Clocks in DMA Cycle

DMA Cycle		Minimum Number of Execution Clocks
<1> Response time to DMA request		4 clocks[Note 1]
<2> Memory access	Internal RAM access	2 clocks[Note 2]
	On-chip peripheral I/O register access	4 clocks + Number of wait cycles specified by VSWC register

Notes 1. If an external interrupt (INTPn) is specified as the DMA transfer start factor, noise elimination time is added (n = 6, 7).

2. Two clocks for the DMA cycle

The minimum number of execution clocks during the DMA cycle in each mode is as follows.

Single transfer: DMA response time (<1>) + Transfer source memory access (<2>) + 1[Note] + Transfer destination memory access (<2>)

Block transfer: DMA response time (<1>) + Transfer source memory access (<2>) + 1[Note] + Transfer destination memory access (<2>) × Number of transfers

Note One clock is always inserted between the read cycle and write cycle of DMA transfer.

16.12 Cautions

(1) Memory boundary

The transfer operation is not guaranteed if the source or the destination address exceeds the area of DMA targets (internal RAM or on-chip peripheral I/O) during DMA transfer.

(2) Transfer of misaligned data

DMA transfer of 32-/16-bit bus width misaligned data is not supported. If the source or the destination address is set to an odd address, the LSB of the address is forcibly handled as “0”.

<R> **(3) Bus arbitration for CPU**

Because the DMA controller has a higher priority bus mastership than the CPU, a CPU access that takes place during DMA transfer is held pending until the DMA transfer cycle is completed and the bus is released to the CPU.

However, the CPU can access the internal ROM and RAM to/from which DMA transfer is not being executed.

- The CPU can access the internal ROM when DMA transfer is being executed between the on-chip peripheral I/O and internal RAM.

<R> **(4) DMA start factors**

Note with caution when setting two or more DMA channels with the same factor.

If two or more DMA channels are started with the same factor, the DMA channel with a lower priority may be acknowledged before the DMA channel with a higher priority.

(5) Program execution and DMA transfer with internal RAM

Do not execute DMA transfer to/from the internal RAM and an instruction in the internal RAM simultaneously.

(6) Restrictions related to automatic clearing of DCHCn.TCn bit

The DCHCn.TCn bit is automatically cleared to 0 when it is read. When DMA transfer is executed to transfer data to or from the internal RAM when two or more DMA transfer channels are simultaneously used, the TCn bit may not be cleared even if it is read after completion of DMA transfer (n = 0 to 3).

Caution This restriction does not apply if one of the following conditions is satisfied.

- **Only one channel of DMA transfer is used.**
- **DMA is not executed to transfer data to or from the internal RAM.**

[Preventive measures]

To read the DCHCn.TCn bit of the DMA channel that is used to transfer data to or from the internal RAM, be sure to read the TCn bit three times in a row. This can accurately clear the TCn bit to 0.

(7) Read values of DSAn and DDAn registers

If the values of the DSAn and DDAn registers are read during DMA transfer, values in the middle of being updated may be read (n = 0 to 3).

For example, if the DSAnH register and the DSAnL register are read in that order when the value of the DMA transfer source address (DSAn register) is “0000FFFFH” and the counting direction is incremental (when the SADn1 and SADn0 bits of the DADCn register = 00), the value of the DSAnL register differs as follows depending on whether DMA transfer is executed immediately after the DSAnH register has been read.

(a) If DMA transfer does not occur while the DSAn register is being read

<1> Reading DSAnH register: DSAnH = 0000H
<2> Reading DSAnL register: DSAnL = FFFFH

(b) If DMA transfer occurs while the DSAn register is being read

<1> Reading DSAnH register: DSAnH = 0000H
<2> Occurrence of DMA transfer
<3> Incrementing DSAn register : DSAn = 00010000H
<4> Reading DSAnL register: DSAnL = 0000H

16.13 DMA Transfer End

When DMA transfer ends and the DCHCn.TCn bit is set to 1, a DMA transfer end interrupt (INTDMAn) is issued to the interrupt controller (INTC) (n = 0 to 3).

CHAPTER 17 INTERRUPT/EXCEPTION PROCESSING FUNCTION

The V850E/IA3 and V850E/IA4 are provided with a dedicated interrupt controller (INTC) for interrupt servicing and can process a total of 56 to 61 interrupt requests.

An interrupt is an event that occurs independently of program execution, and an exception is an event whose occurrence is dependent on program execution.

The V850E/IA3 and V850E/IA4 can process interrupt requests from the on-chip peripheral hardware and external sources. Moreover, exception processing can be started by the TRAP instruction (software exception) or by generation of an exception event (i.e. fetching of an illegal opcode) (exception trap).

17.1 Features

○ Interrupts

- Non-maskable interrupts: 1 source (external: none, internal: 1 source)
- Maskable interrupts (the number of maskable interrupt sources differs depending on the product)
  V850E/IA3: 55 sources (external: 7 sources, internal: 48 sources)
  V850E/IA4: 60 sources (external: 8 sources, internal: 52 sources)
- 8 levels of programmable priorities (maskable interrupts)
- Multiple interrupt control according to priority
- Masks can be specified for each maskable interrupt request.
- Noise elimination, edge detection, and valid edge specification for external interrupt request signals.

○ Exceptions

- Software exceptions: 32 sources
- Exception traps: 2 sources (illegal opcode exception and debug trap)

Interrupt sources are listed in Table 17-1.

Table 17-1. Interrupt Source List (1/3)

Type	Classification	Interrupt/Exception Source				Default Priority	Exception Code	Handler Address	Restored PC
		Name	Control Register	Generating Source	Generating Unit				
Reset	Interrupt	RESET	–	$\overline{\text{RESET}}$ pin input	Pin	–	0000H	00000000H	Undefined
			–	WDT overflow (WDTRES)	WDT				
Non-maskable	Interrupt	INTWDT	–	WDT overflow	WDT		0010H	00000010H	nextPC
Software exception	Exception	TRAP0n[Note 1]	–	TRAP instruction	–	–	004nH	00000040H	nextPC
	Exception	TRAP1n[Note 1]	–	TRAP instruction	–	–	005nH	00000050H	nextPC
Exception trap	Exception	ILGOP/ DBG0	–	Illegal instruction code/ DBTRAP instruction	–	–	0060H	00000060H	nextPC
Maskable	Interrupt	INTP0	PIC0	INTP0 pin valid edge input	Pin	0	0080H	00000080H	nextPC
	Interrupt	INTP1[Note 2]	PIC1[Note 2]	INTP1 pin valid edge input	Pin	1	0090H	00000090H	nextPC
	Interrupt	INTP2	PIC2	INTP2 pin valid edge input	Pin	2	00A0H	000000A0H	nextPC
	Interrupt	INTP3	PIC3	INTP3 pin valid edge input	Pin	3	00B0H	000000B0H	nextPC
	Interrupt	INTP4	PIC4	INTP4 pin valid edge input	Pin	4	00C0H	000000C0H	nextPC
	Interrupt	INTP5	PIC5	INTP5 pin valid edge input	Pin	5	00D0H	000000D0H	nextPC
	Interrupt	INTP6	PIC6	INTP6 pin valid edge input	Pin	6	00E0H	000000E0H	nextPC
	Interrupt	INTP7	PIC7	INTP7 pin valid edge input	Pin	7	00F0H	000000F0H	nextPC
	Interrupt	INTCMP0	CMPIC0	ADC0 overvoltage detection (comparator output)	ADC0 (comparator)	8	0100H	00000100H	nextPC
	Interrupt	INTCMP1	CMPIC1	ADC1 overvoltage detection (comparator output)	ADC1 (comparator)	9	0110H	00000110H	nextPC
	Interrupt	INTTQ0OV	TQ0OVIC	TMQ0 overflow[Note 3]	TMQ0	10	0120H	00000120H	nextPC
	Interrupt	INTTQ0CC0	TQ0CCIC0	TQ0CCR0 capture input/ compare match[Note 4]	TMQ0	11	0130H	00000130H	nextPC
	Interrupt	INTTQ0CC1	TQ0CCIC1	TQ0CCR1 capture input/ compare match	TMQ0	12	0140H	00000140H	nextPC
	Interrupt	INTTQ0CC2	TQ0CCIC2	TQ0CCR2 capture input/ compare match	TMQ0	13	0150H	00000150H	nextPC
	Interrupt	INTTQ0CC3	TQ0CCIC3	TQ0CCR3 capture input/ compare match	TMQ0	14	0160H	00000160H	nextPC
	Interrupt	INTTQ1OV	TQ1OVIC	TMQ1 overflow[Note 3]	TMQ1	15	0170H	00000170H	nextPC
	Interrupt	INTTQ1CC0	TQ1CCIC0	TQ1CCR0 compare match[Note 4]	TMQ1	16	0180H	00000180H	nextPC
	Interrupt	INTTQ1CC1	TQ1CCIC1	TQ1CCR1 compare match	TMQ1	17	0190H	00000190H	nextPC

Notes 1. n = 0 to FH

2. V850E/IA4 only

3. When TMQm is used in the 6-phase PWM output mode, it functions as INTTQmOV (valley interrupt) from the TMQm option (TMQOPm) (V850E/IA3: m = 0, V850E/IA4: m = 0, 1).

4. When TMQm is used in 6-phase PWM output mode, it functions as INTTQmCC0 (crest interrupt) from the TMQm option (TMQOPm) (V850E/IA3: m = 0, V850E/IA4: m = 0, 1).

Table 17-1. Interrupt Source List (2/3)

Type	Classification	Interrupt/Exception Source				Default Priority	Exception Code	Handler Address	Restored PC
		Name	Control Register	Generating Source	Generating Unit				
Maskable	Interrupt	INTTQ1CC2	TQ1CCIC2	TQ1CCR2 compare match	TMQ1	18	01A0H	000001A0H	nextPC
	Interrupt	INTTQ1CC3	TQ1CCIC3	TQ1CCR3 compare match	TMQ1	19	01B0H	000001B0H	nextPC
	Interrupt	INTCC00	CC0IC0	CC100 capture input/ compare match	TMENC10	20	01C0H	000001C0H	nextPC
	Interrupt	INTCC01	CC0IC1	CC101 capture input/ compare match	TMENC10	21	01D0H	000001D0H	nextPC
	Interrupt	INTCM00	CM0IC0	CM100 compare match	TMENC10	22	01E0H	000001E0H	nextPC
	Interrupt	INTCM01	CM0IC1	CM101 compare match	TMENC10	23	01F0H	000001F0H	nextPC
	Interrupt	INTCC10[Note]	CC1IC0[Note]	CC110 capture input/ compare match	TMENC11	24	0200H	00000200H	nextPC
	Interrupt	INTCC11[Note]	CC1IC1[Note]	CC111 capture input/ compare match	TMENC11	25	0210H	00000210H	nextPC
	Interrupt	INTCM10[Note]	CM1IC0[Note]	CM110 compare match	TMENC11	26	0220H	00000220H	nextPC
	Interrupt	INTCM11[Note]	CM1IC1[Note]	CM111 compare match	TMENC11	27	0230H	00000230H	nextPC
	Interrupt	INTTP0OV	TP0OVIC	TMP0 overflow	TMP0	28	0240H	00000240H	nextPC
	Interrupt	INTTP0CC0	TP0CCIC0	TP0CCR0 capture input/ compare match	TMP0	29	0250H	00000250H	nextPC
	Interrupt	INTTP0CC1	TP0CCIC1	TP0CCR1 capture input/ compare match	TMP0	30	0260H	00000260H	nextPC
	Interrupt	INTTP1OV	TP1OVIC	TMP1 overflow	TMP1	31	0270H	00000270H	nextPC
	Interrupt	INTTP1CC0	TP1CCIC0	TP1CCR0 compare match	TMP1	32	0280H	00000280H	nextPC
	Interrupt	INTTP1CC1	TP1CCIC1	TP1CCR1 compare match	TMP1	33	0290H	00000290H	nextPC
	Interrupt	INTTP2OV	TP2OVIC	TMP2 overflow	TMP2	34	02A0H	000002A0H	nextPC
	Interrupt	INTTP2CC0	TP2CCIC0	TP2CCR0 capture input/ compare match	TMP2	35	02B0H	000002B0H	nextPC
	Interrupt	INTTP2CC1	TP2CCIC1	TP2CCR1 capture input/ compare match	TMP2	36	02C0H	000002C0H	nextPC
	Interrupt	INTTP3OV	TP3OVIC	TMP3 overflow	TMP3	37	02D0H	000002D0H	nextPC
	Interrupt	INTTP3CC0	TP3CCIC0	TP3CCR0 compare match	TMP3	38	02E0H	000002E0H	nextPC
	Interrupt	INTTP3CC1	TP3CCIC1	TP3CCR1 compare match	TMP3	39	02F0H	000002F0H	nextPC
	Interrupt	INTDMA0	DMAIC0	DMA channel 0 transfer completion	DMA0	40	0300H	00000300H	nextPC
	Interrupt	INTDMA1	DMAIC1	DMA channel 1 transfer completion	DMA1	41	0310H	00000310H	nextPC
	Interrupt	INTDMA2	DMAIC2	DMA channel 2 transfer completion	DMA2	42	0320H	00000320H	nextPC
	Interrupt	INTDMA3	DMAIC3	DMA channel 3 transfer completion	DMA3	43	0330H	00000330H	nextPC
	Interrupt	INTUA0RE	UA0REIC	UARTA0 receive error	UARTA0	44	0340H	00000340H	nextPC

Note V850E/IA4 only

Table 17-1. Interrupt Source List (3/3)

Type	Classification	Interrupt/Exception Source				Default Priority	Exception Code	Handler Address	Restored PC
		Name	Control Register	Generating Source	Generating Unit				
Maskable	Interrupt	INTUA0R	UA0RIC	UARTA0 reception completion	UARTA0	45	0350H	00000350H	nextPC
	Interrupt	INTUA0T	UA0TIC	UARTA0 transmission enable	UARTA0	46	0360H	00000360H	nextPC
	Interrupt	INTCB0RE	CB0REIC	CSIB0 receive error	CSIB0	47	0370H	00000370H	nextPC
	Interrupt	INTCB0R	CB0RIC	CSIB0 reception completion	CSIB0	48	0380H	00000380H	nextPC
	Interrupt	INTCB0T	CB0TIC	CSIB0 transmission enable	CSIB0	49	0390H	00000390H	nextPC
	Interrupt	INTUA1RE	UA1REIC	UARTA1 receive error	UARTA1	50	03A0H	000003A0H	nextPC
	Interrupt	INTUA1R	UA1RIC	UARTA1 reception completion	UARTA1	51	03B0H	000003B0H	nextPC
	Interrupt	INTUA1T	UA1TIC	UARTA1 transmission enable	UARTA1	52	03C0H	000003C0H	nextPC
	Interrupt	INTCB1RE	CB1REIC	CSIB1 receive error	CSIB1	53	03D0H	000003D0H	nextPC
	Interrupt	INTCB1R	CB1RIC	CSIB1 reception completion	CSIB1	54	03E0H	000003E0H	nextPC
	Interrupt	INTCB1T	CB1TIC	CSIB1 transmission enable	CSIB1	55	03F0H	000003F0H	nextPC
	Interrupt	INTAD0	AD0IC	ADC0 conversion completion	ADC0	56	0400H	00000400H	nextPC
	Interrupt	INTAD1	AD1IC	ADC1 conversion completion	ADC1	57	0410H	00000410H	nextPC
	Interrupt	INTAD2	AD2IC	ADC2 conversion completion	ADC2	58	0420H	00000420H	nextPC
	Interrupt	INTTM0EQ0	TM0EQIC0	TM0CMP0 compare match	TMM0	59	0430H	00000430H	nextPC

Remarks 1. Default Priority: The priority order when two or more maskable interrupt requests occur at the same time. The highest priority is 0.

Restored PC: The value of the program counter (PC) saved to EIPC, FEPC, or DBPC of CPU when interrupt servicing is started. Note, however, that the restored PC when a non-maskable or maskable interrupt is acknowledged while one of the following instructions is being executed does not become the nextPC. (If an interrupt is acknowledged during interrupt execution, execution stops, and then resumes after the interrupt servicing has finished. In this case, the address of the aborted instruction is the restore PC.)

- Load instructions (SLD.B, SLD.BU, SLD.H, SLD.HU, SLD.W)
- Division instructions (DIV, DIVH, DIVU, DIVHU)
- PREPARE, DISPOSE instructions (only if an interrupt is generated before the stack pointer is updated)

nextPC: The PC value that starts the processing following interrupt/exception processing.

2. The execution address of the illegal instruction when an illegal opcode exception occurs is calculated by (Restored PC – 4).

17.2 Non-Maskable Interrupts

A non-maskable interrupt request signal is acknowledged unconditionally, even when interrupts are in the interrupt disabled (DI) status. An NMI is not subject to priority control and takes precedence over all the other interrupt request signals.

The non-maskable interrupt signals of the V850E/IA3 and V850E/IA4 are the non-maskable interrupt request signals generated by the overflow of the watchdog timer (INTWDT).

INTWDT functions when the WDTM.WDM1 and WDTM.WDM0 bits are set to “01”.

17.2.1 Operation

If a non-maskable interrupt request signal (INTWDT) is generated, the CPU performs the following processing, and transfers control to the handler routine.

(1) Saves the restored PC to FEPC.
(2) Saves the current PSW to FEPSW.
(3) Writes the exception code (0010H) to the higher halfword (FECC) of ECR.
(4) Sets the PSW.NP and PSW.ID bits (1) and clears the PSW.EP bit (0).
(5) Loads the handler address (00000010H) of the non-maskable interrupt routine to the PC, and transfers control.

The following shows the non-maskable interrupt servicing.

Figure 17-1. Non-Maskable Interrupt Servicing

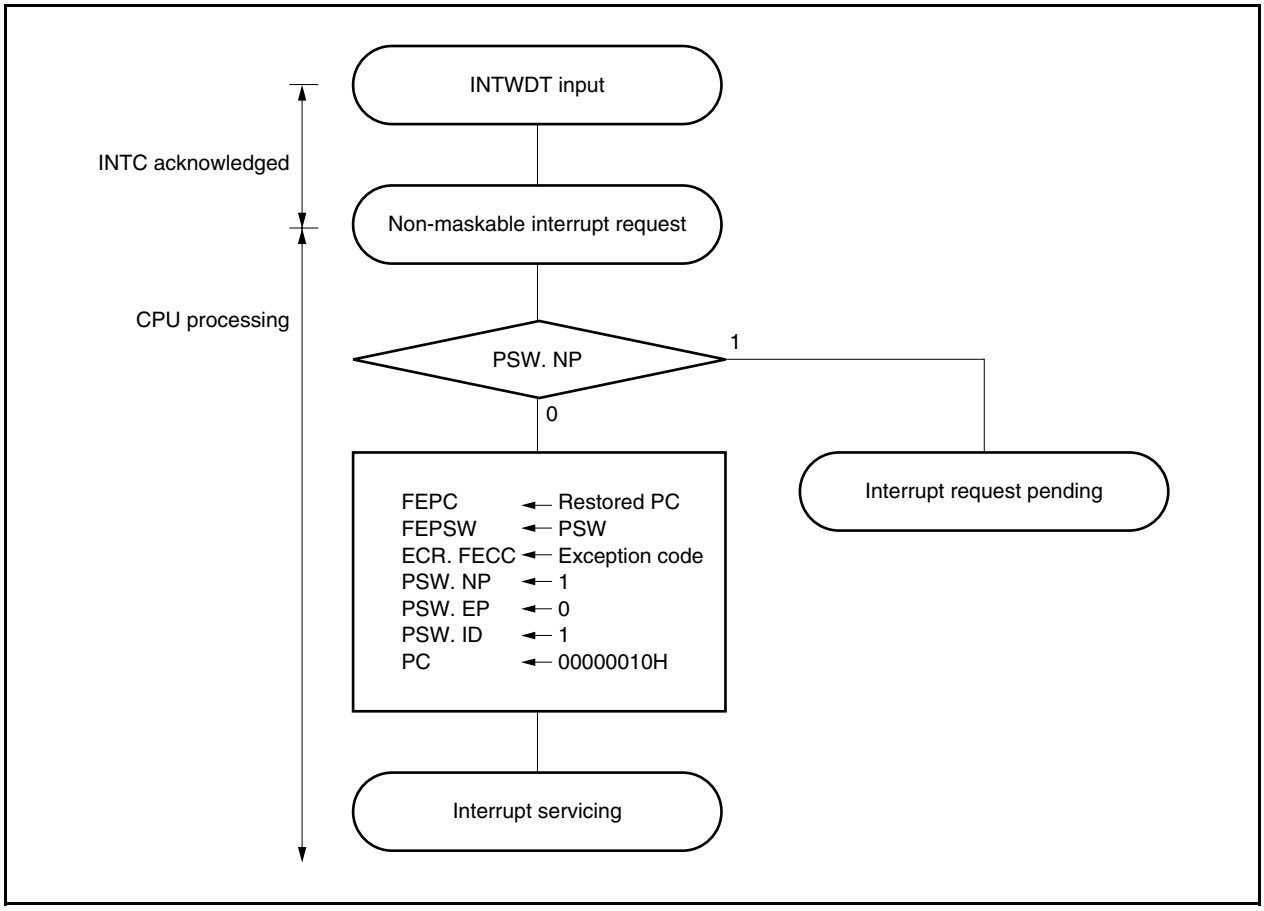
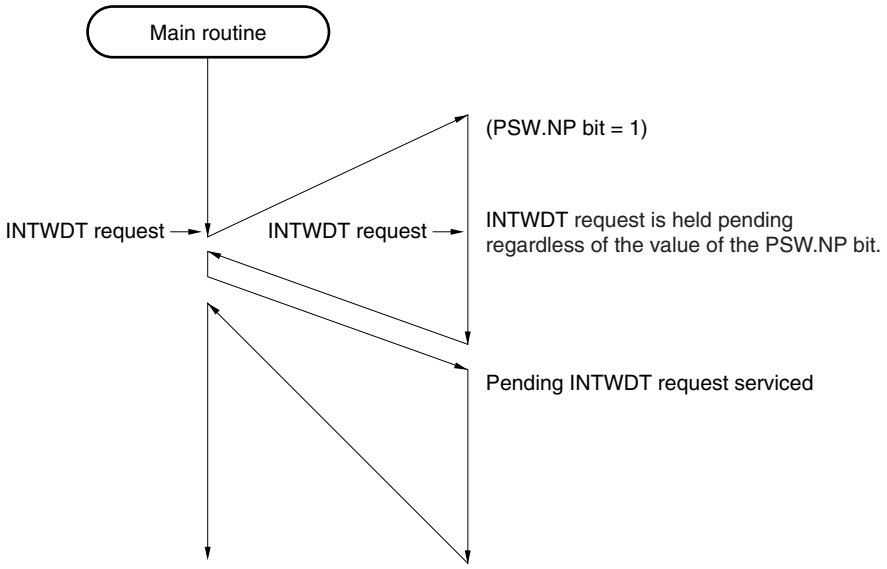
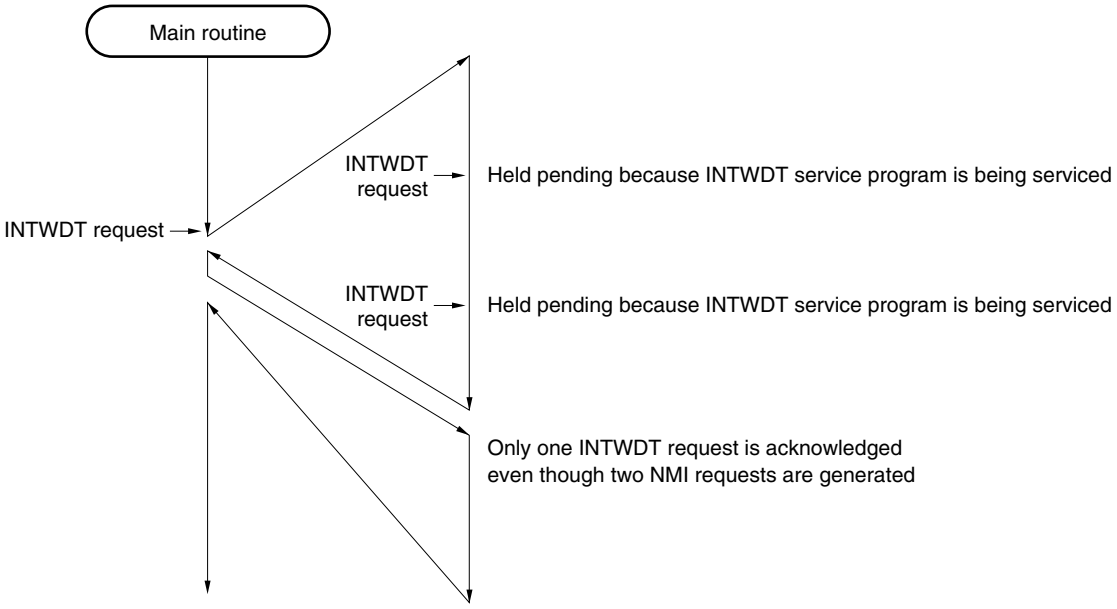


Figure 17-2. Acknowledging Non-Maskable Interrupt Request

(a) If a new INTWDT request is generated while an INTWDT service program is being executed



(b) If a new INTWDT request is generated twice while an INTWDT service program is being executed



17.2.2 Restore

Execution is restored from non-maskable interrupt servicing by the RETI instruction.

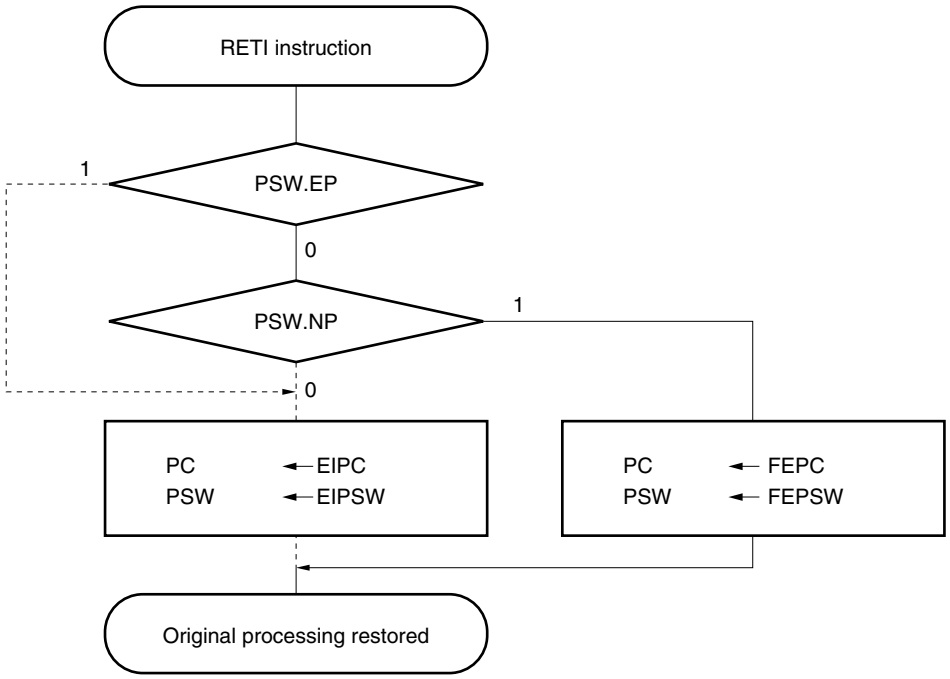
When the RETI instruction is executed, the CPU performs the following processing, and transfers control to the address of the restored PC.

<1> Restores the values of the PC and the PSW from FEPC and FEPSW, respectively, because the PSW.EP bit is 0 and the PSW.NP bit is 1.

<2> Transfers control back to the address of the restored PC and PSW.

The following illustrates how the RETI instruction is processed.

Figure 17-3. RETI Instruction Processing



Caution When the EP and NP bits are changed by the LDSR instruction during non-maskable interrupt servicing, in order to restore the PC and PSW correctly during recovery by the RETI instruction, it is necessary to set EP back to 0 and NP back to 1 using the LDSR instruction immediately before the RETI instruction.

Remark The solid line shows the CPU processing flow.

17.2.3 Non-maskable interrupt status flag (NP)

The NP flag is a status flag that indicates that non-maskable interrupt (INTWDT) servicing is in progress. The NP flag is allocated to the PSW.

This flag is set when an INTWDT interrupt signal has been acknowledged, and masks all interrupt requests and exceptions to prohibit multiple interrupts from being acknowledged.

The flag is cleared to 00000020H after reset.

After reset: 00000020H

	31	8	7	6	5	4	3	2	1	0
PSW	0		**NP**	EP	ID	SAT	CY	OV	S	Z

NP	Non-maskable interrupt (INTWDT) servicing status
0	No non-maskable interrupt servicing
1	Non-maskable interrupt servicing in progress

17.3 Maskable Interrupts

Maskable interrupt request signals can be masked by interrupt control registers. The V850E/IA3 and V850E/IA4 have 60 maskable interrupt sources.

If two or more maskable interrupt request signals are generated at the same time, they are acknowledged according to the default priority. In addition to the default priority, eight levels of priorities can be specified by using the interrupt control registers (programmable priority control).

When an interrupt request signal has been acknowledged, the acknowledgment of other maskable interrupt request signals is disabled and the interrupt disabled (DI) status is set.

When the EI instruction is executed in an interrupt service routine, the interrupt enabled (EI) status is set, which enables servicing of interrupts having a higher priority than the interrupt request signal in progress (specified by the interrupt control register). Note that only interrupts with a higher priority will have this capability; interrupts with the same priority level cannot be serviced as multiple interrupts.

To enable multiple interrupt servicing, however, save EIPC and EIPSW to memory or registers before executing the EI instruction, and execute the DI instruction before the RETI instruction to restore the original values of EIPC and EIPSW.

17.3.1 Operation

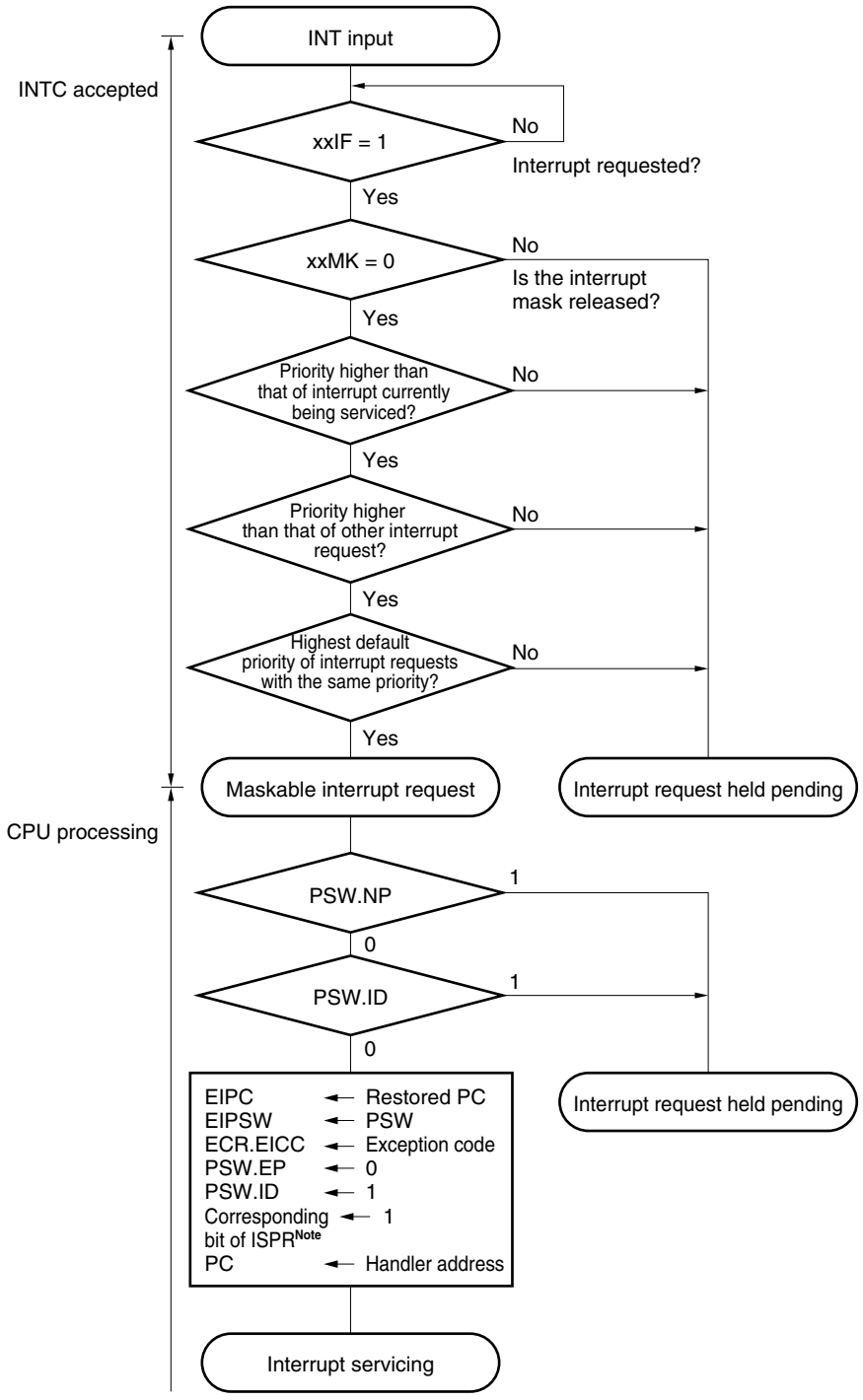
If a maskable interrupt occurs, the CPU performs the following processing, and transfers control to the handler routine.

<1> Saves the restored PC to EIPC.
<2> Saves the current PSW to EIPSW.
<3> Writes an exception code to the lower halfword of ECR (EICC).
<4> Sets the PSW.ID bit to 1 and clears the PSW.EP bit to 0.
<5> Sets the handler address corresponding to each interrupt to the PC, and transfers control.

The maskable interrupt request signal masked by interrupt controller (INTC) and the maskable interrupt request signal generated while another interrupt is being serviced (while PSW.NP bit = 1 or ID bit = 1) are held pending inside the INTC. In this case, servicing a new maskable interrupt is started in accordance with the priority of the pending maskable interrupt request signal if either the maskable interrupt is unmasked or NP and ID bits are cleared to 0 by using the RETI or LDSR instruction.

How maskable interrupts are serviced is illustrated below.

Figure 17-4. Maskable Interrupt Servicing



Note For details of the ISPR register, see **17.3.6 In-service priority register (ISPR)**.

17.3.2 Restore

Recovery from maskable interrupt servicing is carried out by the RETI instruction.

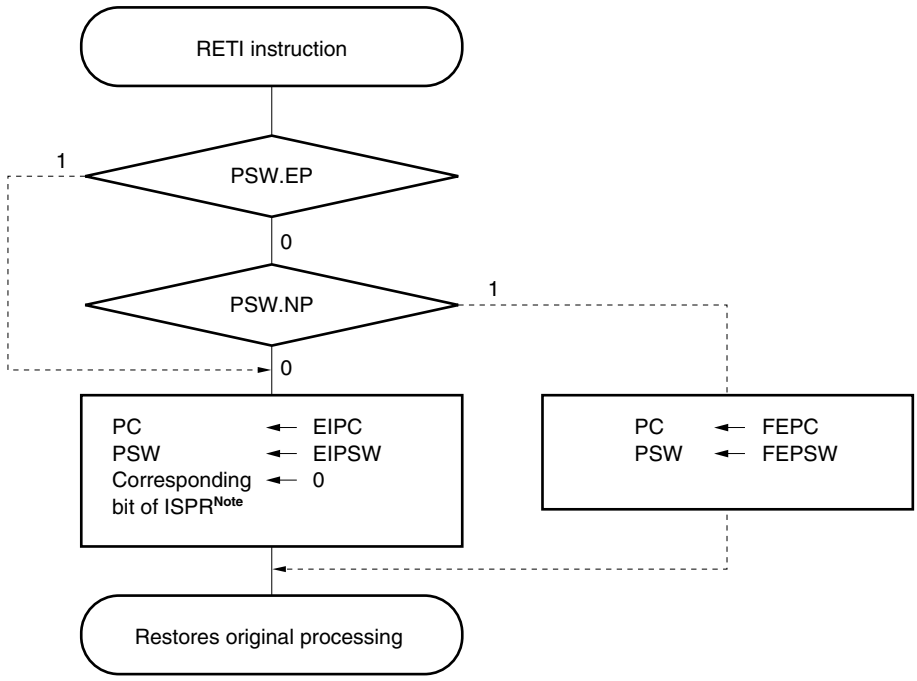
When the RETI instruction is executed, the CPU performs the following steps, and transfers control to the address of the restored PC.

<1> Restores the values of the PC and the PSW from EIPC and EIPSW because the PSW.EP bit is 0 and the PSW.NP bit is 0.

<2> Transfers control to the address of the restored PC and PSW.

The processing of the RETI instruction is shown below.

Figure 17-5. RETI Instruction Processing



Note For the ISPR register, see **17.3.6 In-service priority register (ISPR)**.

Caution When the EP and NP bits are changed by the LDSR instruction during non-maskable interrupt servicing, in order to restore the PC and PSW correctly during recovery by the RETI instruction, it is necessary to set EP back to 0 and NP back to 1 using the LDSR instruction immediately before the RETI instruction.

Remark The solid line shows the CPU processing flow.

17.3.3 Priorities of maskable interrupts

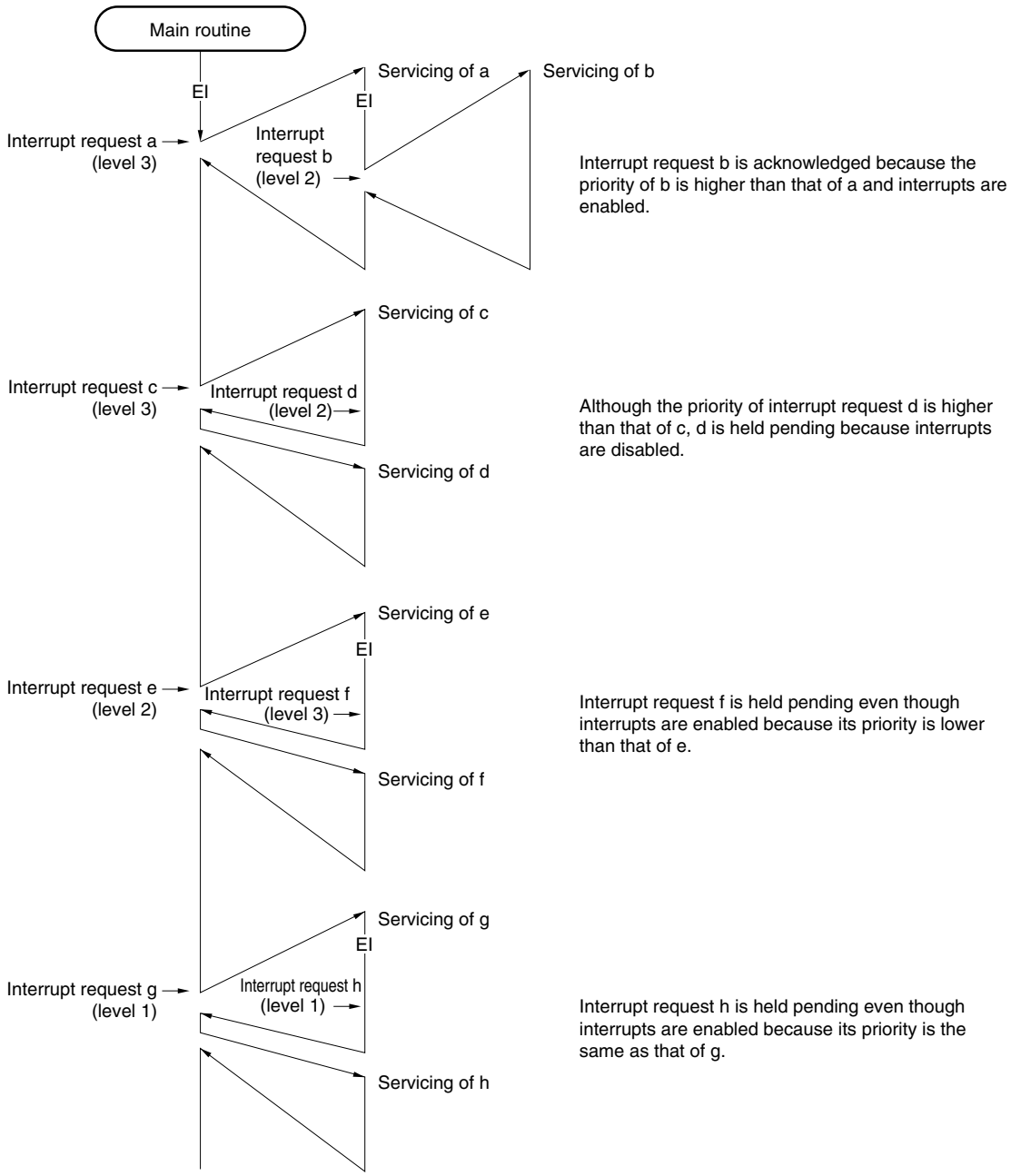
The INTC provides multiple interrupt servicing in which an interrupt is acknowledged while another interrupt is being serviced. Multiple interrupts can be controlled by priority levels.

There are two types of priority level control: control based on the default priority levels, and control based on the programmable priority levels that are specified by the interrupt priority level specification bit (xxPRn) of the interrupt control register (xxICn). When two or more interrupts having the same priority level specified by the xxPRn bit are generated at the same time, interrupts are serviced in order depending on the priority level allocated to each interrupt request signal type (default priority level) beforehand. For more information, see **Table 17-1 Interrupt Source List**. Programmable priority control customizes interrupt request signals into eight levels by the setting of the priority level specification flag.

Note that when an interrupt request signal is acknowledged, the PSW.ID flag is automatically set to 1. Therefore, when multiple interrupts are to be used, clear the ID flag to 0 beforehand (for example, by placing the EI instruction in the interrupt servicing program) to set the interrupt enabled mode.

Remark xx: Identification name of each peripheral unit (see **Table 17-2**)
n: Peripheral unit number (see **Table 17-2**)

Figure 17-6. Example of Processing in Which Another Interrupt Request Signal Is Issued While an Interrupt Is Being Serviced (1/2)

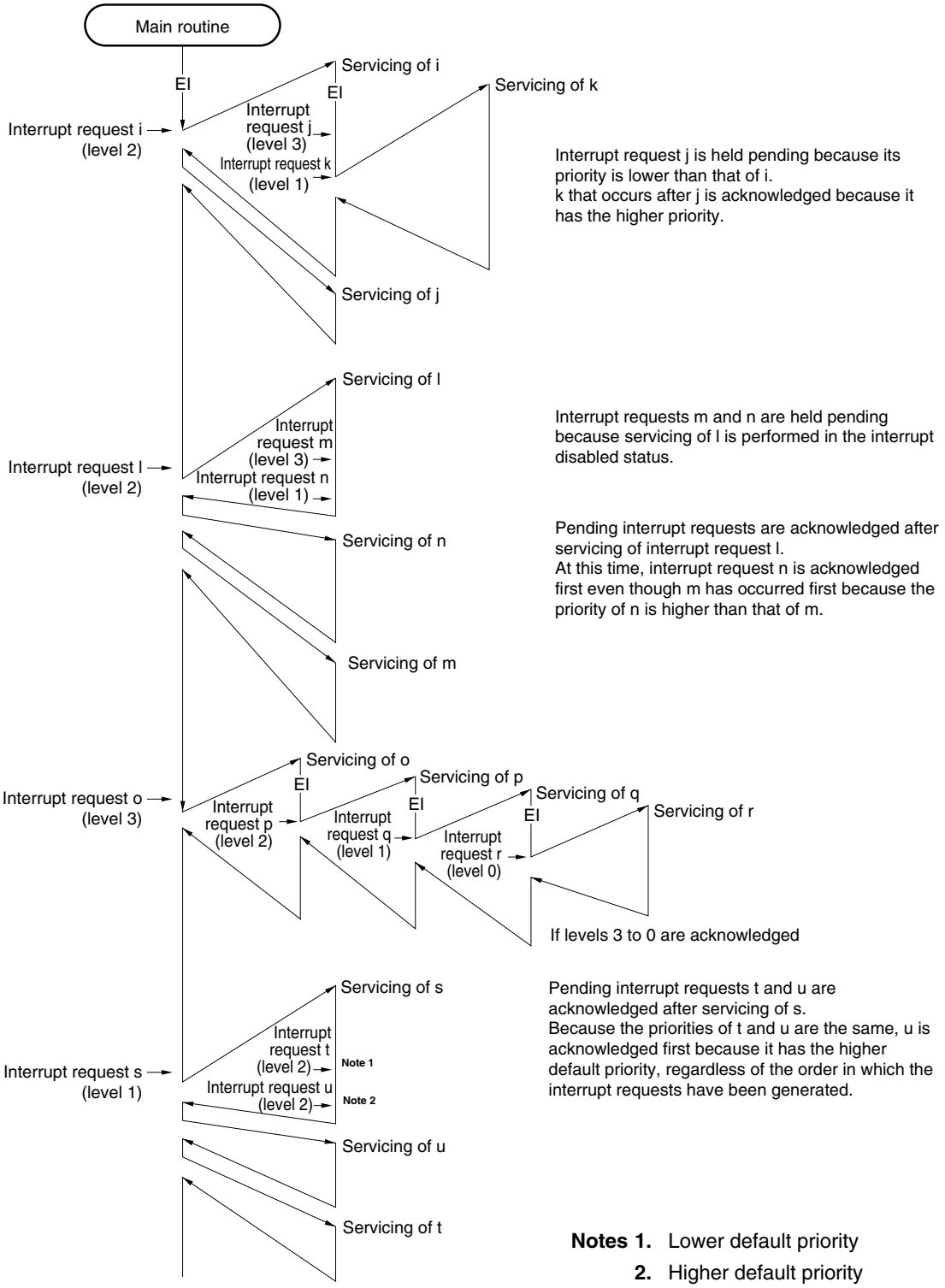


Caution To perform multiple interrupt servicing, the values of the EIPC and EIPSW registers must be saved before executing the EI instruction. When returning from multiple interrupt servicing, restore the values of EIPC and EIPSW after executing the DI instruction.

Remarks 1. a to u in the figure are the temporary names of interrupt request signals shown for the sake of explanation.

2. The default priority in the figure indicates the relative priority between two interrupt request signals.

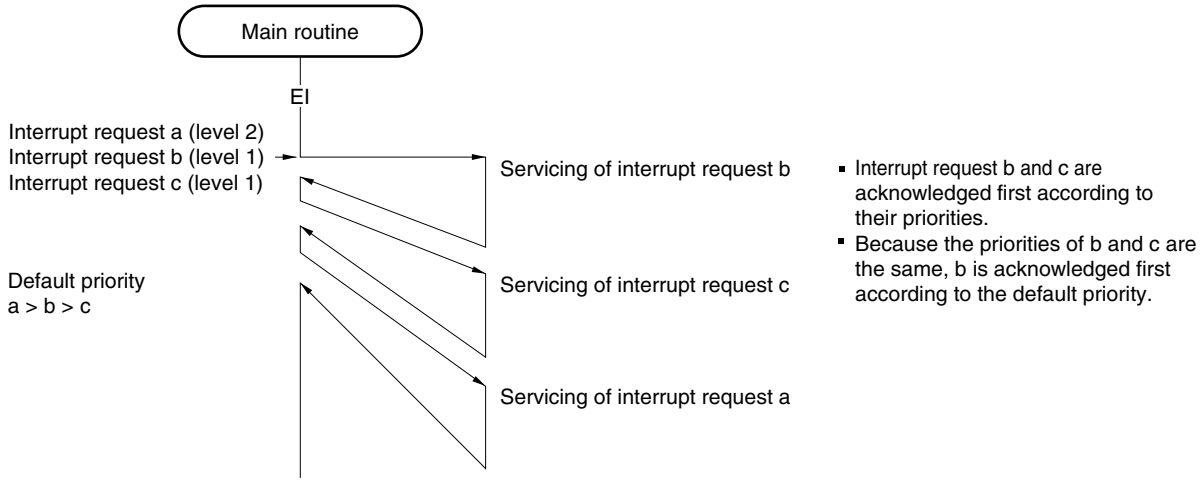
Figure 17-6. Example of Processing in Which Another Interrupt Request Signal Is Issued While an Interrupt Is Being Serviced (2/2)



Notes 1. Lower default priority
2. Higher default priority

Caution To perform multiple interrupt servicing, the values of the EIPC and EIPSW registers must be saved before executing the EI instruction. When returning from multiple interrupt servicing, restore the values of EIPC and EIPSW after executing the DI instruction.

Figure 17-7. Example of Servicing Interrupt Request Signals Generated Simultaneously



Caution To perform multiple interrupt servicing, the values of the EIPC and EIPSW registers must be saved before executing the EI instruction. When returning from multiple interrupt servicing, restore the values of EIPC and EIPSW after executing the DI instruction.

Remarks 1. a to c in the figure are assumed names given to interrupt request signals for the sake of explanation.

2. The default priority in the figure indicates the relative priority between two interrupt request signals.

17.3.4 Interrupt control registers (xxICn)

An xxICn register is assigned to each interrupt request signal (maskable interrupt) and sets the control conditions for each maskable interrupt request.

These registers can be read or written in 8-bit or 1-bit units.

Reset sets these registers to 47H.

Cautions 1. Disable interrupts (DI) to read the xxICn.xxIFn bit. If the xxIFn bit is read while interrupts are enabled (EI), the correct value may not be read when acknowledging an interrupt and reading the bit conflict.

<R> **2. When manipulating the xxICn.xxMKn bit while interrupt requests may occur (including the state in which interrupts are disabled (DI)), be sure to use a bit manipulation instruction or use the IMRm.xxMKn bit (m = 0 to 3).**

After reset: 47H R/W Address: FFFFF110H to FFFFF186H

	<7>	<6>	5	4	3	2	1	0
xxICn	xxIFn	xxMKn	0	0	0	xxPRn2	xxPRn1	xxPRn0

xxIFn	Interrupt request flag[Note]
0	Interrupt request not issued
1	Interrupt request issued

xxMKn	Interrupt mask flag
0	Interrupt servicing enabled
1	Interrupt servicing disabled (pending)

xxPRn2	xxPRn1	xxPRn0	Interrupt priority specification bit
0	0	0	Specifies level 0 (highest).
0	0	1	Specifies level 1.
0	1	0	Specifies level 2.
0	1	1	Specifies level 3.
1	0	0	Specifies level 4.
1	0	1	Specifies level 5.
1	1	0	Specifies level 6.
1	1	1	Specifies level 7 (lowest).

Note The flag xxIFn is reset automatically by the hardware if an interrupt request signal is acknowledged.

Remark xx: Identification name of each peripheral unit (see **Table 17-2**)
n: Peripheral unit number (see **Table 17-2**)

The addresses and bits of the interrupt control registers are as follows.

Table 17-2. Addresses and Bits of Interrupt Control Registers (1/2)

Address	Register	Bit							
		<7>	<6>	5	4	3	2	1	0
FFFFF110H	PIC0	PIF0	PMK0	0	0	0	PPR02	PPR01	PPR00
FFFFF112H	PIC1[Note]	PIF1	PMK1	0	0	0	PPR12	PPR11	PPR10
FFFFF114H	PIC2	PIF2	PMK2	0	0	0	PPR22	PPR21	PPR20
FFFFF116H	PIC3	PIF3	PMK3	0	0	0	PPR32	PPR31	PPR30
FFFFF118H	PIC4	PIF4	PMK4	0	0	0	PPR42	PPR41	PPR40
FFFFF11AH	PIC5	PIF5	PMK5	0	0	0	PPR52	PPR51	PPR50
FFFFF11CH	PIC6	PIF6	PMK6	0	0	0	PPR62	PPR61	PPR60
FFFFF11EH	PIC7	PIF7	PMK7	0	0	0	PPR72	PPR71	PPR70
FFFFF120H	CMPIC0	CMPIF0	CMPMK0	0	0	0	CMPPR02	CMPPR01	CMPPR00
FFFFF122H	CMPIC1	CMPIF1	CMPMK1	0	0	0	CMPPR12	CMPPR11	CMPPR10
FFFFF124H	TQ0OVIC	TQ0OVIF	TQ0OVMK	0	0	0	TQ0OVPR2	TQ0OVPR1	TQ0OVPR0
FFFFF126H	TQ0CCIC0	TQ0CCIF0	TQ0CCMK0	0	0	0	TQ0CCPR02	TQ0CCPR01	TQ0CCPR00
FFFFF128H	TQ0CCIC1	TQ0CCIF1	TQ0CCMK1	0	0	0	TQ0CCPR12	TQ0CCPR11	TQ0CCPR10
FFFFF12AH	TQ0CCIC2	TQ0CCIF2	TQ0CCMK2	0	0	0	TQ0CCPR22	TQ0CCPR21	TQ0CCPR20
FFFFF12CH	TQ0CCIC3	TQ0CCIF3	TQ0CCMK3	0	0	0	TQ0CCPR32	TQ0CCPR31	TQ0CCPR30
FFFFF12EH	TQ1OVIC	TQ1OVIF	TQ1OVMK	0	0	0	TQ1OVPR2	TQ1OVPR1	TQ1OVPR0
FFFFF130H	TQ1CCIC0	TQ1CCIF0	TQ1CCMK0	0	0	0	TQ1CCPR02	TQ1CCPR01	TQ1CCPR00
FFFFF132H	TQ1CCIC1	TQ1CCIF1	TQ1CCMK1	0	0	0	TQ1CCPR12	TQ1CCPR11	TQ1CCPR10
FFFFF134H	TQ1CCIC2	TQ1CCIF2	TQ1CCMK2	0	0	0	TQ1CCPR22	TQ1CCPR21	TQ1CCPR20
FFFFF136H	TQ1CCIC3	TQ1CCIF3	TQ1CCMK3	0	0	0	TQ1CCPR32	TQ1CCPR31	TQ1CCPR30
FFFFF138H	CC0IC0	CC0IF0	CC0MK0	0	0	0	CC0PR02	CC0PR01	CC0PR00
FFFFF13AH	CC0IC1	CC0IF1	CC0MK1	0	0	0	CC0PR12	CC0PR11	CC0PR10
FFFFF13CH	CM0IC0	CM0IF0	CM0MK0	0	0	0	CM0PR02	CM0PR01	CM0PR00
FFFFF13EH	CM0IC1	CM0IF1	CM0MK1	0	0	0	CM0PR12	CM0PR11	CM0PR10
FFFFF140H	CC1IC0[Note]	CC1IF0	CC1MK0	0	0	0	CC1PR02	CC1PR01	CC1PR00
FFFFF142H	CC1IC1[Note]	CC1IF1	CC1MK1	0	0	0	CC1PR12	CC1PR11	CC1PR10
FFFFF144H	CM1IC0[Note]	CM1IF0	CM1MK0	0	0	0	CM1PR02	CM1PR01	CM1PR00
FFFFF146H	CM1IC1[Note]	CM1IF1	CM1MK1	0	0	0	CM1PR12	CM1PR11	CM1PR10
FFFFF148H	TP0OVIC	TP0OVIF	TP0OVMK	0	0	0	TP0OVPR2	TP0OVPR1	TP0OVPR0
FFFFF14AH	TP0CCIC0	TP0CCIF0	TP0CCMK0	0	0	0	TP0CCPR02	TP0CCPR01	TP0CCPR00
FFFFF14CH	TP0CCIC1	TP0CCIF1	TP0CCMK1	0	0	0	TP0CCPR12	TP0CCPR11	TP0CCPR10
FFFFF14EH	TP1OVIC	TP1OVIF	TP1OVMK	0	0	0	TP1OVPR2	TP1OVPR1	TP1OVPR0
FFFFF150H	TP1CCIC0	TP1CCIF0	TP1CCMK0	0	0	0	TP1CCPR02	TP1CCPR01	TP1CCPR00
FFFFF152H	TP1CCIC1	TP1CCIF1	TP1CCMK1	0	0	0	TP1CCPR12	TP1CCPR11	TP1CCPR10
FFFFF154H	TP2OVIC	TP2OVIF	TP2OVMK	0	0	0	TP2OVPR2	TP2OVPR1	TP2OVPR0
FFFFF156H	TP2CCIC0	TP2CCIF0	TP2CCMK0	0	0	0	TP2CCPR02	TP2CCPR01	TP2CCPR00
FFFFF158H	TP2CCIC1	TP2CCIF1	TP2CCMK1	0	0	0	TP2CCPR12	TP2CCPR11	TP2CCPR10
FFFFF15AH	TP3OVIC	TP3OVIF	TP3OVMK	0	0	0	TP3OVPR2	TP3OVPR1	TP3OVPR0
FFFFF15CH	TP3CCIC0	TP3CCIF0	TP3CCMK0	0	0	0	TP3CCPR02	TP3CCPR01	TP3CCPR00
FFFFF15EH	TP3CCIC1	TP3CCIF1	TP3CCMK1	0	0	0	TP3CCPR12	TP3CCPR11	TP3CCPR10
FFFFF160H	DMAIC0	DMAIF0	DMAMK0	0	0	0	DMAPR02	DMAPR01	DMAPR00

Note V850E/IA4 only

Table 17-2. Addresses and Bits of Interrupt Control Registers (2/2)

Address	Register	Bit							
		<7>	<6>	5	4	3	2	1	0
FFFFF162H	DMAIC1	DMAIF1	DMAMK1	0	0	0	DMAPR12	DMAPR11	DMAPR10
FFFFF164H	DMAIC2	DMAIF2	DMAMK2	0	0	0	DMAPR22	DMAPR21	DMAPR20
FFFFF166H	DMAIC3	DMAIF3	DMAMK3	0	0	0	DMAPR32	DMAPR31	DMAPR30
FFFFF168H	UA0REIC	UA0REIF	UA0REMK	0	0	0	UA0REPR2	UA0REPR1	UA0REPR0
FFFFF16AH	UA0RIC	UA0RIF	UA0RMK	0	0	0	UA0RPR2	UA0RPR1	UA0RPR0
FFFFF16CH	UA0TIC	UA0TIF	UA0TMK	0	0	0	UA0TPR2	UA0TPR1	UA0TPR0
FFFFF16EH	CB0REIC	CB0REIF	CB0REMK	0	0	0	CB0REPR2	CB0REPR1	CB0REPR0
FFFFF170H	CB0RIC	CB0RIF	CB0RMK	0	0	0	CB0RPR2	CB0RPR1	CB0RPR0
FFFFF172H	CB0TIC	CB0TIF	CB0TMK	0	0	0	CB0TPR2	CB0TPR1	CB0TPR0
FFFFF174H	UA1REIC	UA1REIF	UA1REMK	0	0	0	UA1REPR2	UA1REPR1	UA1REPR0
FFFFF176H	UA1RIC	UA1RIF	UA1RMK	0	0	0	UA1RPR2	UA1RPR1	UA1RPR0
FFFFF178H	UA1TIC	UA1TIF	UA1TMK	0	0	0	UA1TPR2	UA1TPR1	UA1TPR0
FFFFF17AH	CB1REIC	CB1REIF	CB1REMK	0	0	0	CB1REPR2	CB1REPR1	CB1REPR0
FFFFF17CH	CB1RIC	CB1RIF	CB1RMK	0	0	0	CB1RPR2	CB1RPR1	CB1RPR0
FFFFF17EH	CB1TIC	CB1TIF	CB1TMK	0	0	0	CB1TPR2	CB1TPR1	CB1TPR0
FFFFF180H	AD0IC	AD0IF	AD0MK	0	0	0	AD0PR2	AD0PR1	AD0PR0
FFFFF182H	AD1IC	AD1IF	AD1MK	0	0	0	AD1PR2	AD1PR1	AD1PR0
FFFFF184H	AD2IC	AD2IF	AD2MK	0	0	0	AD2PR2	AD2PR1	AD2PR0
FFFFF186H	TM0EQIC0	TM0EQIF0	TM0EQMK0	0	0	0	TM0EQPR02	TM0EQPR01	TM0EQPR00

17.3.5 Interrupt mask registers 0 to 3 (IMR0 to IMR3)

The IMR0 to IMR3 registers set the interrupt mask state for the maskable interrupts. The IMR0.xxMKn to IMR3.xxMKn bits are equivalent to the xxICn.xxMKn bit.

The IMRm register (m = 0 to 3) can be read or written in 16-bit units.

If the higher 8 bits of the IMRm register are used as the IMRmH register and the lower 8 bits as the IMRmL register, these registers can be read or written in 8-bit or 1-bit units.

Reset sets these registers to FFFFH.

Caution The device file defines the xxICn.xxMKn bit as a reserved word. If a bit is manipulated using the name of xxMKn, the contents of the xxICn register, instead of the IMRm register, are rewritten (as a result, the contents of the IMRm register are also rewritten).

After reset: FFFFH R/W Address: IMR3 FFFFF106H
IMR3L FFFFF106H, IMR3H FFFFF107H

	15	14	13	12	11	10	9	8
IMR3 (IMR3H[Note 1])	1	1	1	1	TM0EQMK0	AD2MK	AD1MK	AD0MK
	7	6	5	4	3	2	1	0
(IMR3L)	CB1TMK	CB1RMK	CB1REMK	UA1TMK	UA1RMK	UA1REMK	CB0TMK	CB0RMK

After reset: FFFFH R/W Address: IMR2 FFFFF104H
IMR2L FFFFF104H, IMR2H FFFFF105H

	15	14	13	12	11	10	9	8
IMR2 (IMR2H[Note 1])	CB0REMK	UA0TMK	UA0RMK	UA0REMK	DMAMK3	DMAMK2	DMAMK1	DMAMK0
	7	6	5	4	3	2	1	0
(IMR2L)	TP3CCMK1	TP3CCMK0	TP3OVMK	TP2CCMK1	TP2CCMK0	TP2OVMK	TP1CCMK1	TP1CCMK0

After reset: FFFFH R/W Address: IMR1 FFFFF102H
IMR1L FFFFF102H, IMR1H FFFFF103H

	15	14	13	12	11	10	9	8
IMR1 (IMR1H[Note 1])	TP1OVMK	TP0CCMK1	TP0CCMK0	TP0OVMK	CM1MK1[Note 2]	CM1MK0[Note 2]	CC1MK1[Note 2]	CC1MK0[Note 2]
	7	6	5	4	3	2	1	0
(IMR1L)	CM0MK1	CM0MK0	CC0MK1	CC0MK0	TQ1CCMK3	TQ1CCMK2	TQ1CCMK1	TQ1CCMK0

After reset: FFFFH R/W Address: IMR0 FFFFF100H
IMR0L FFFFF100H, IMR0H FFFFF101H

	15	14	13	12	11	10	9	8
IMR0 (IMR0H[Note 1])	TQ1OVMK	TQ0CCMK3	TQ0CCMK2	TQ0CCMK1	TQ0CCMK0	TQ0OVMK	CMPMK1	CMPMK0
	7	6	5	4	3	2	1	0
(IMR0L)	PMK7	PMK6	PMK5	PMK4	PMK3	PMK2	PMK1[Note 2]	PMK0

xxMKn	Interrupt mask flag setting
0	Interrupt servicing enabled
1	Interrupt servicing disabled

Notes 1. When reading/writing bits 15 to 8 of the IMR0 to IMR3 registers in 8-bit or 1-bit units, specify these bits as bits 7 to 0 of the IMR0H to IMR3H registers.

2. These bits are valid only for the V850E/IA4.
Be sure to set these bits to 1 in the V850E/IA3.

Caution Set bits 15 to 12 of the IMR3 register (bits 7 to 4 of the IMR3H register) to 1. The operation when these settings are changed is not guaranteed.

Remark xx: Identification name of each peripheral unit (see **Table 17-2**)
n: Peripheral unit number (see **Table 17-2**)

17.3.6 In-service priority register (ISPR)

The ISPR register holds the priority level of the maskable interrupt currently acknowledged. When an interrupt request signal is acknowledged, the bit of this register corresponding to the priority level of that interrupt signal request is set to 1 and remains set while the interrupt is serviced.

When the RETI instruction is executed, the bit corresponding to the interrupt request signal having the highest priority is automatically cleared to 0 by hardware. However, it is not cleared to 0 when execution is returned from non-maskable interrupt servicing or exception processing.

This register is read-only, in 8-bit or 1-bit units.

Reset sets this register to 00H.

Caution In the interrupt enabled (EI) state, if an interrupt is acknowledged during the reading of the ISPR register, the value of the ISPR register may be read after the bit is set (1) by this interrupt acknowledgment. To read the value of the ISPR register properly before interrupt acknowledgment, read it in the interrupt disabled (DI) state.

After reset: 00H R Address: FFFFF1FAH

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
ISPR	ISPR7	ISPR6	ISPR5	ISPR4	ISPR3	ISPR2	ISPR1	ISPR0

ISPRn	Priority of interrupt currently being acknowledged
0	Interrupt request signal with priority n is not acknowledged
1	Interrupt request signal with priority n is being acknowledged

Remark n: 0 to 7 (priority level)

17.3.7 Maskable interrupt status flag (ID)

The ID flag controls the maskable interrupt's operating state, and stores control information regarding enabling or disabling of interrupt requests. The ID flag is allocated to the PSW.

This flag is set to 00000020H after reset.

After rest: 00000020H

	31	8	7	6	5	4	3	2	1	0
PSW	0		NP	EP	ID	SAT	CY	OV	S	Z

ID	Maskable interrupt servicing specification[Note]
0	Maskable interrupt request signal acknowledgment enabled
1	Maskable interrupt request signal acknowledgment disabled (pending)

Note Interrupt disable flag (ID) function

ID is set (1) by the DI instruction and cleared (0) by the EI instruction. Its value is also modified by the RETI instruction or LDSR instruction when referencing the PSW.

Non-maskable interrupt signals and exceptions are acknowledged regardless of this flag. When a maskable interrupt signal is acknowledged, the ID flag is automatically set (1) by hardware.

An interrupt request signal generated during the acknowledgment disabled period (ID flag = 1) can be acknowledged when the xxICn.xxIFn bit is set (1), and the ID flag is cleared (0).

17.4 External Interrupt Request Input Pins (INTP0 to INTP7)

17.4.1 Noise elimination

(1) Noise elimination of INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, and INTP7 pins

The INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, and INTP7 pins incorporate a noise eliminator that uses analog delay. Unless, therefore, the input level of each pin is held for a certain time, an edge cannot be detected. An edge is detected after a certain time has elapsed.

(2) Noise elimination of INTP6 pin

The INTP6 pin incorporates a digital noise eliminator.

The sampling clock that performs digital sampling can be selected from fxx/4, fxx/8, fxx/16, or fxx/32.

The system clock stops in the IDLE and STOP modes, so the INTP6 pin cannot be used to cancel the IDLE and STOP modes.

(a) External interrupt noise elimination control register (INTPNRC)

The INTPNRC register is used to select the sampling clock that is used to eliminate digital noise on the INTP6 pin. If the same level is not detected five times in a row, the signal is eliminated as noise.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

Cautions 1. If the input pulse lasts for the duration of 4 to 5 clocks, it is undefined whether the pulse is detected as a valid edge or eliminated as noise. So that the pulse is actually detected as a valid edge, the same pulse level must be input for the duration of 5 clocks or more.

2. If noise is generated in synchronization with the sampling clock, eliminate the noise by attaching a filter to the input pin.

3. Noise is not eliminated if the pin is used as a normal input port pin.

After reset: 00H R/W Address: FFFFF310H

	7	6	5	4	3	2	1	0
INTPNRC	0	0	0	0	0	0	INTPNRC1	INTPNRC0

INTPNRC1	INTPNRC0	Selection of sampling clock
0	0	fxx/32
0	1	fxx/16
1	0	fxx/8
1	1	fxx/4

17.4.2 Edge detection

The valid edges of the INTPn pin can be selected by program (V850E/IA3: n = 0, 2 to 7, V850E/IA4: n = 0 to 7). The edge that can be selected as the valid edge is one of the following.

- Rising edge
- Falling edge
- Both the rising and falling edges

The edge-detected INTPn signal becomes an interrupt source.
The valid edge is specified by the INTR0 and INTF0 registers.

(1) External interrupt rising edge specification register 0 (INTR0), external interrupt falling edge specification register 0 (INTF0)

The INTR0 and INTF0 registers are 8-bit registers that specify the trigger mode of the INTP0, INTP1 (V850E/IA4 only), and INTP2 to INTP7 pins and can specify the valid edge independently for each pin (rising edge, falling edge, or both rising and falling edges).

These registers can be read or written in 8-bit or 1-bit units.

Reset sets these registers to 00H.

Caution When the function is changed from the external interrupt function (alternate function) to the port mode, an edge may be detected. Therefore, be sure to clear the INTF0n and INTR0n bits to 00, and then set the port mode (V850E/IA3: n = 0, 2 to 7, V850E/IA4: n = 0 to 7).

After reset: 00H R/W Address: FFFFFC20H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTR0	INTR07	INTR06	INTR05	INTR04	INTR03	INTR02	INTR01Note	INTR00

After reset: 00H R/W Address: FFFFFC00H

	<7>	<6>	<5>	<4>	<3>	<2>	<1>	<0>
INTF0	INTF07	INTF06	INTF05	INTF04	INTF03	INTF02	INTF01Note	INTF00

Note Valid only for the V850E/IA4.
In the V850E/IA3, be sure to clear these bits to 0.

Remark For the valid edge specification, see **Table 17-3**.

Table 17-3. Valid Edge Specification of INTP0 to INTP7 Pins

INTF0n	INTR0n	Valid Edge Specification
0	0	No edge detected
0	1	Rising edge
1	0	Falling edge
1	1	Both rising and falling edges

Caution When not using these pins as the INTPn pins, be sure to clear the INTF0n and INTR0n bits to 00.

Remark V850E/IA3: n = 0, 2 to 7
V850E/IA4: n = 0 to 7

17.5 Software Exception

A software exception is generated when the CPU executes the TRAP instruction, and can always be acknowledged.

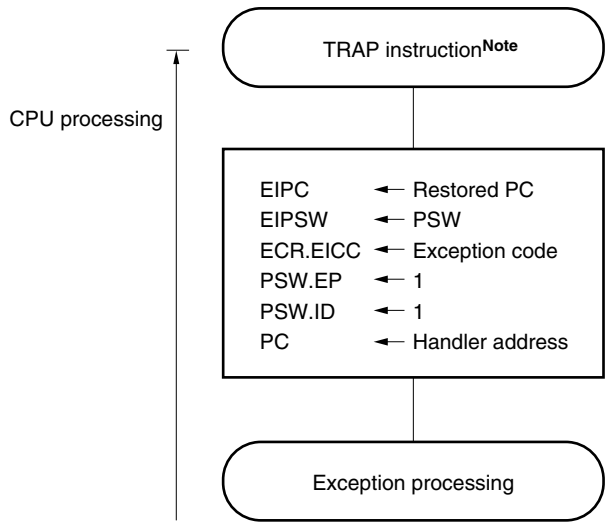
17.5.1 Operation

If a software exception occurs, the CPU performs the following processing, and transfers control to the handler routine.

<1> Saves the restored PC to EIPC.
<2> Saves the current PSW to EIPSW.
<3> Writes an exception code to the lower 16 bits (EICC) of ECR (interrupt source).
<4> Sets the PSW.EP and PSW.ID bits (1).
<5> Sets the handler address (00000040H or 00000050H) corresponding to the software exception to the PC, and transfers control.

The processing of a software exception is shown below.

Figure 17-8. Software Exception Processing



Note TRAP instruction format: TRAP vector (the vector is a value from 00H to 1FH.)

The handler address is determined by the TRAP instruction's operand (vector). If the vector is 00H to 0FH, it becomes 00000040H, and if the vector is 10H to 1FH, it becomes 00000050H.

17.5.2 Restore

Recovery from software exception processing is carried out by the RETI instruction.

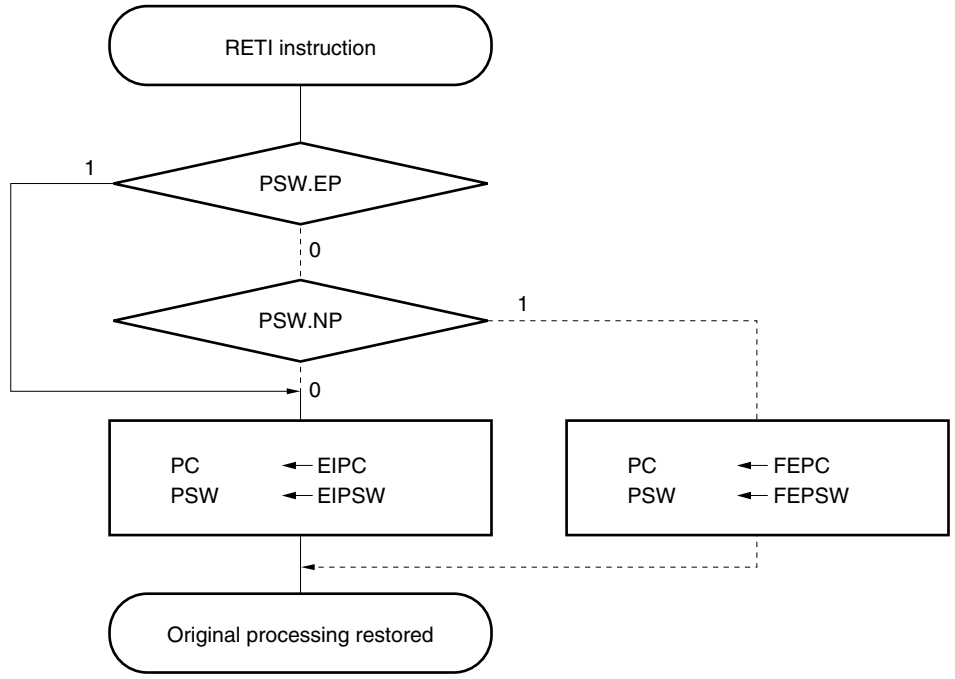
By executing the RETI instruction, the CPU carries out the following processing and shifts control to the restored PC's address.

<1> Loads the restored PC and PSW from EIPC and EIPSW because the PSW.EP bit is 1.

<2> Transfers control to the address of the restored PC and PSW.

The processing of the RETI instruction is shown below.

Figure 17-9. RETI Instruction Processing



Caution When the EP and NP bits are changed by the LDSR instruction during software exception processing, in order to restore the PC and PSW correctly during recovery by the RETI instruction, it is necessary to set the EP bit back to 1 and clear the NP bit to 0 using the LDSR instruction immediately before the RETI instruction.

Remark The solid line shows the CPU processing flow.

17.5.3 Exception status flag (EP)

The EP flag is a status flag used to indicate that exception processing is in progress. This flag is set when an exception occurs. The EP flag is allocated to the PSW.

This flag is set to 00000020H after reset.

After reset: 00000020H

	31	8	7	6	5	4	3	2	1	0
PSW	0		NP	EP	ID	SAT	CY	OV	S	Z

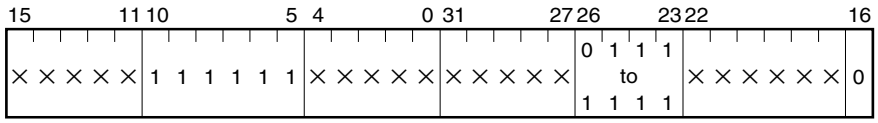
EP	Exception processing status
0	Exception processing not in progress
1	Exception processing in progress

17.6 Exception Trap

An exception trap is an interrupt that is requested when the illegal execution of an instruction takes place. In the V850E/IA3 and V850E/IA4, an illegal opcode trap (ILGOP: Illegal Opcode Trap) is considered as an exception trap.

17.6.1 Illegal opcode definition

The illegal instruction has an opcode (bits 10 to 5) of 111111B, a sub-opcode (bits 26 to 23) of 0111B to 1111B, and a sub-opcode (bit 16) of 0B. An exception trap is generated when an instruction applicable to this illegal instruction is executed.



×: Arbitrary

Caution Since it is possible that this instruction may be assigned to an illegal opcode in the future, it is recommended that it not be used.

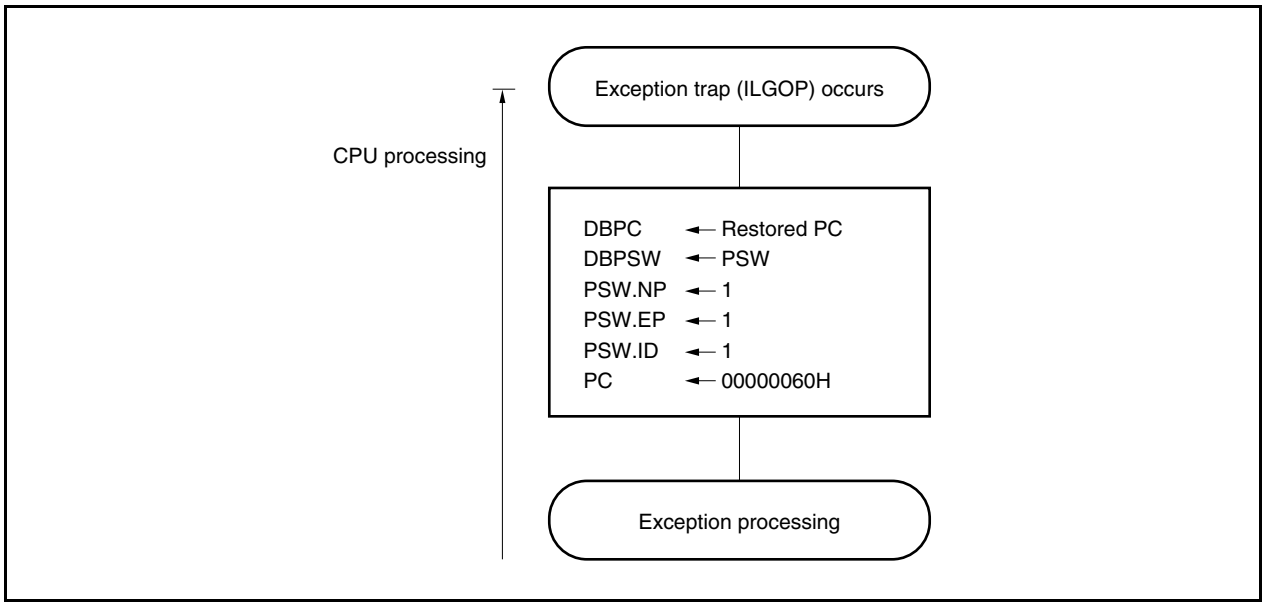
(1) Operation

If an exception trap occurs, the CPU performs the following processing, and transfers control to the handler routine.

<1> Saves the restored PC to DBPC.
<2> Saves the current PSW to DBPSW.
<3> Sets the PSW.NP, PSW.EP, and PSW.ID bits (1).
<4> Sets the handler address (00000060H) corresponding to the exception trap to the PC, and transfers control.

The processing of the exception trap is shown below.

Figure 17-10. Exception Trap Processing



(2) Restore

Recovery from an exception trap is carried out by the DBRET instruction. By executing the DBRET instruction, the CPU carries out the following processing and controls the address of the restored PC.

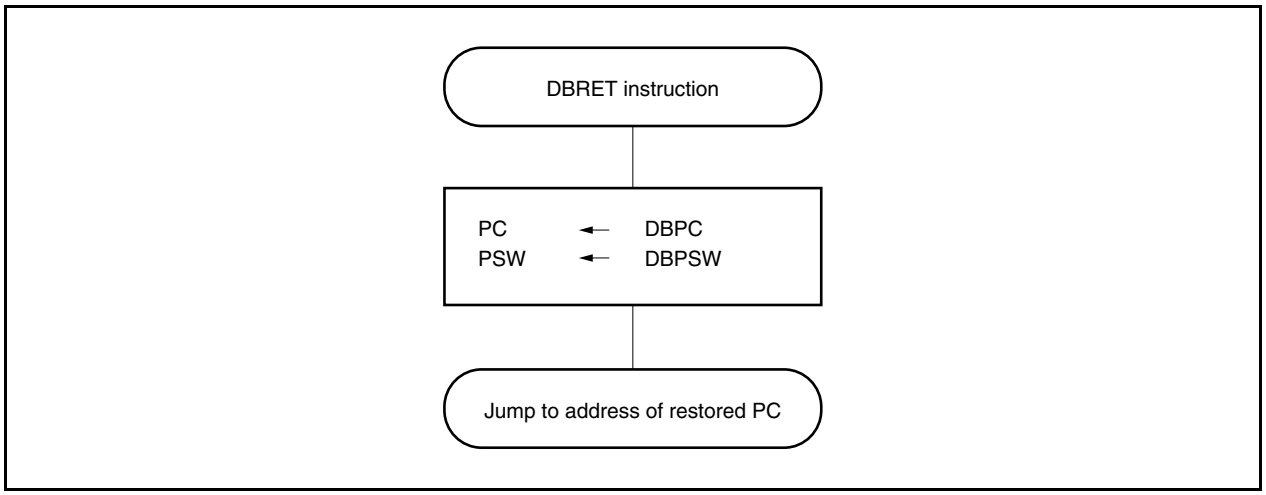
<1> Loads the restored PC and PSW from DBPC and DBPSW.

<2> Transfers control to the address indicated by the restored PC and PSW.

<R> **Caution DBPC and DBPSW can be accessed only during the period between when the illegal opcode is executed and when the DBRET instruction is executed.**

The restore processing from an exception trap is shown below.

Figure 17-11. Restore Processing from Exception Trap



17.6.2 Debug trap

The debug trap is an exception that can be acknowledged anytime and is generated by execution of the DBTRAP instruction.

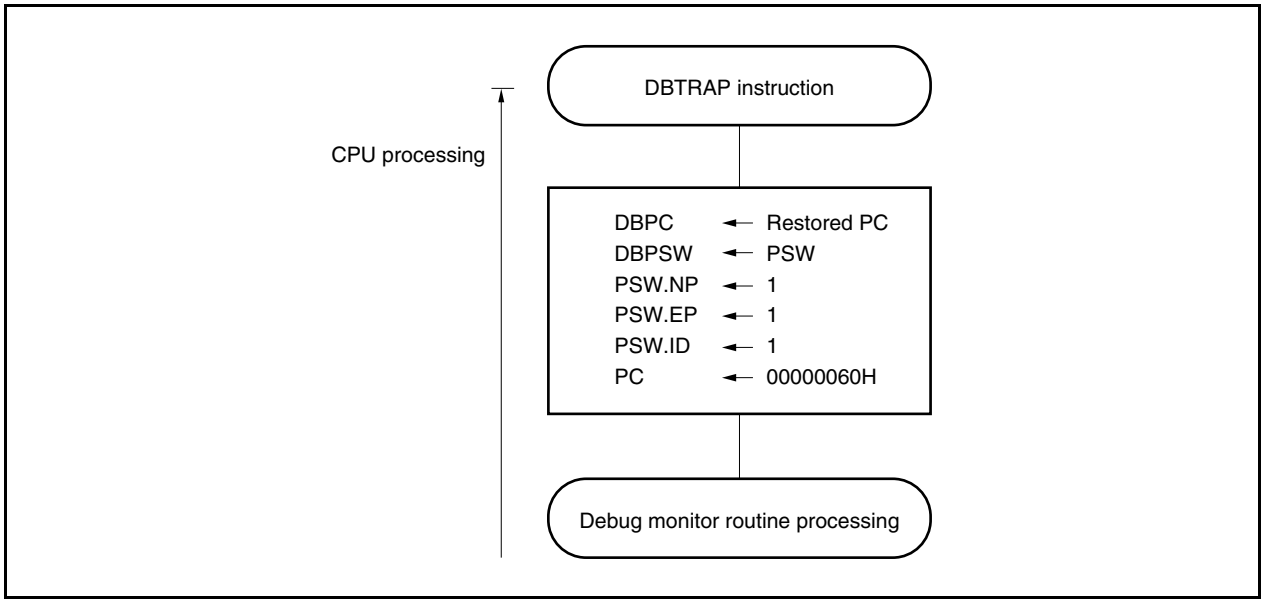
When the debug trap is generated, the CPU performs the following processing.

(1) Operation

<1> Saves the restored PC to DBPC.
<2> Saves the current PSW to DBPSW.
<3> Sets the PSW.NP, PSW.EP and PSW.ID bits (1).
<4> Sets the handler address (00000060H) corresponding to the debug trap to the PC and transfers control.

The processing of the debug trap is shown below.

Figure 17-12. Debug Trap Processing



(2) Restore

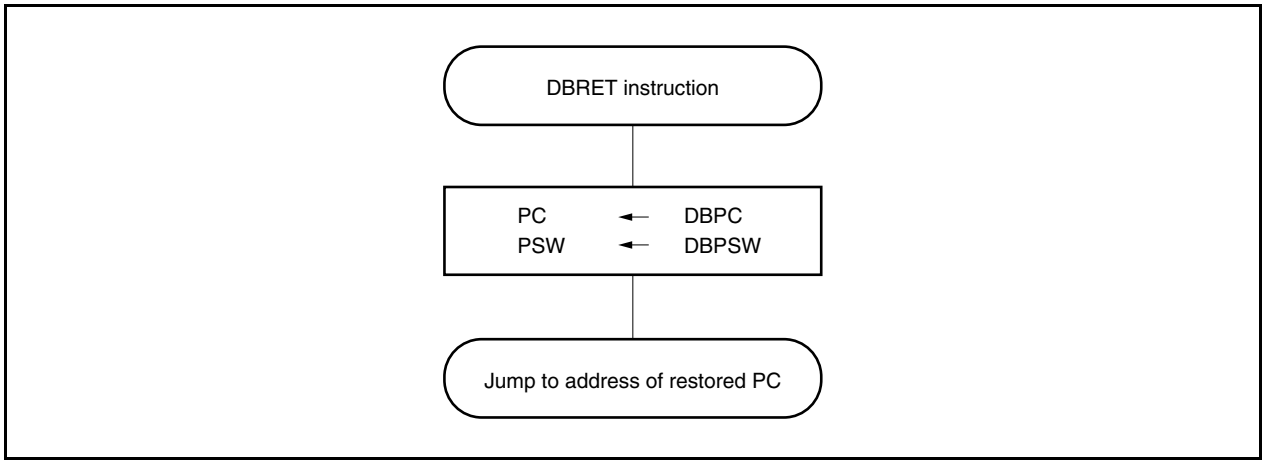
Recovery from a debug trap is carried out by the DBRET instruction. By executing the DBRET instruction, the CPU carries out the following processing and controls the address of the restored PC.

<1> Loads the restored PC and PSW from DBPC and DBPSW.
<2> Transfers control to the address indicated by the restored PC and PSW.

Caution DBPC and DBPSW can be accessed only during the period between when the DBTRAP is executed and when the DBRET instruction is executed.

The restore processing from a debug trap is shown below.

Figure 17-13. Restore Processing from Debug Trap



17.7 Multiple Interrupt Servicing Control

Multiple interrupt servicing control is a process by which an interrupt request that is currently being serviced can be interrupted during servicing if there is an interrupt request signal with a higher priority level, and the higher priority interrupt request signal is acknowledged and serviced first.

If there is an interrupt request signal with a lower priority level than the interrupt request currently being serviced, that interrupt request signal is held pending.

Multiple interrupt servicing control of maskable interrupts is executed when interrupts are enabled (PSW.ID bit = 0). Thus, to execute multiple interrupts, it is necessary to set the interrupt enabled state (PSW.ID bit = 0) even in an interrupt servicing routine.

If maskable interrupts are enabled or a software exception is generated in a maskable interrupt or software exception servicing program, it is necessary to save EIPC and EIPSW.

This is accomplished by the following procedure.

(1) Acknowledgment of maskable interrupt signals in servicing program

Service program of maskable interrupt or exception

```
        ...
        ...
• EIPC saved to memory or register
• EIPSW saved to memory or register
• EI instruction (interrupt acknowledgment enabled)
        ...
        ...                          ← Maskable interrupt acknowledgment
        ...
        ...
• DI instruction (interrupt acknowledgment disabled)
• Saved value restored to EIPSW
• Saved value restored to EIPC
• RETI instruction
```

(2) Generation of exception in servicing program

Servicing program of maskable interrupt or exception

... ... • EIPC saved to memory or register • EIPSW saved to memory or register ... • TRAP instruction ... • Saved value restored to EIPSW • Saved value restored to EIPC • RETI instruction	← Exception such as TRAP instruction acknowledged.

The priority order for multiple interrupt servicing control has 8 levels, from 0 to 7 for each maskable interrupt request signal (0 is the highest priority), but it can be set as desired via software. The priority order is set using the xxPRn0 to xxPRn2 bits of the interrupt control request register (xxICn), provided for each maskable interrupt request signal. After system reset, an interrupt request signal is masked by the xxMKn bit and the priority order is set to level 7 by the xxPRn0 to xxPRn2 bits.

The priority order of maskable interrupts is as follows.

(High) Level 0 > Level 1 > Level 2 > Level 3 > Level 4 > Level 5 > Level 6 > Level 7 (Low)

Interrupt servicing that has been suspended as a result of multiple servicing control is resumed after the servicing of the higher priority interrupt has been completed and the RETI instruction has been executed.
A pending interrupt request signal is acknowledged after the current interrupt servicing has been completed and the RETI instruction has been executed.

Caution In a non-maskable interrupt servicing routine (time until the RETI instruction is executed), maskable interrupts are suspended and not acknowledged.

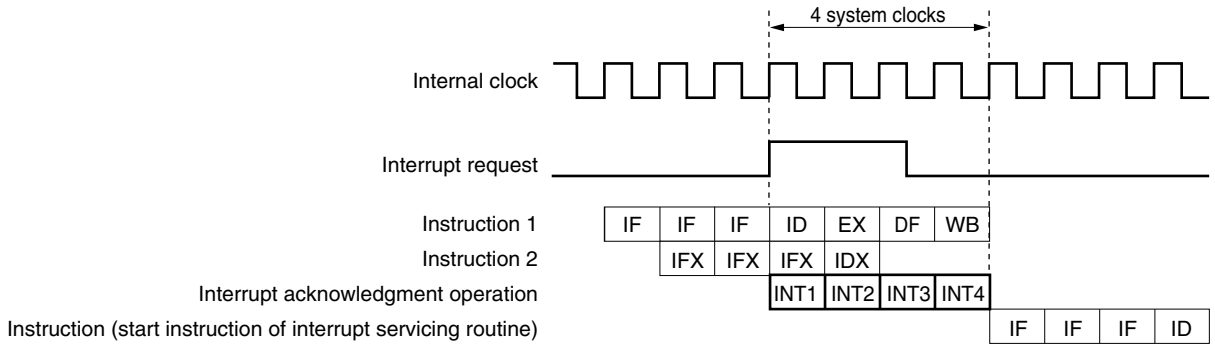
Remark xx: Identification name of each peripheral unit (see **Table 17-2**)
n: Peripheral unit number (see **Table 17-2**)

17.8 Interrupt Response Time of CPU

Except the following cases, the interrupt response time of the CPU is 4 clocks minimum. To input interrupt request signals successively, input the next interrupt request signal at least 4 clocks after the preceding interrupt.

- In IDLE/STOP mode
- When interrupt request non-sampling instructions are successively executed (see **17.9 Periods in Which CPU Does Not Acknowledge Interrupts**.)
- When an on-chip peripheral I/O register is accessed

Figure 17-14. Pipeline Operation at Interrupt Request Acknowledgment (Outline)



Remark INT1 to INT4: Interrupt acknowledgment processing
IFX: Invalid instruction fetch
IDX: Invalid instruction decode

	Interrupt latency time (internal system clock)			Conditions
	Internal interrupt	External interrupt		
		INTP0, INTP1[Note 1], INTP2 to INTP5, INTP7	INTP6	
Minimum	4	4 + Analog delay time	4 + **Note 2** + Digital noise filter	The following cases are exceptions. • In IDLE/STOP mode • Two or more interrupt request non-sample instructions are executed in succession • Access to on-chip peripheral I/O register
Maximum	8	8 + Analog delay time	8 + **Note 2** + Digital noise filter	

Notes 1. V850E/IA4 only

2. For details, see **4.6 (1) External interrupt noise elimination control register (INTPNRC)**.

17.9 Periods in Which CPU Does Not Acknowledge Interrupts

The CPU acknowledges an interrupt while an instruction is being executed. However, no interrupt will be acknowledged between an interrupt request non-sample instruction and the next instruction (interrupt is held pending).

The interrupt request non-sample instructions are as follows.

- EI instruction
- DI instruction
- LDSR reg2, 0x5 instruction (for PSW)
- Store instruction for the command register (PRCMD).
- Store instructions or bit manipulation instructions excluding tst1 instruction for the following registers.
  - Interrupt-related registers:
    Interrupt control register (xxICn) and interrupt mask registers 0 to 3 (IMR0 to IMR3)
  - Power save control register (PSC)
  - Internal memory size switching register (IMS)

Remark xx: Identification name of each peripheral unit (see **Table 17-2**)
n: Peripheral unit number (see **Table 17-2**)

17.10 Caution

Note that if a port is set to external interrupt input (INTPn), the timer/counter-related interrupt and A/D converter-related interrupt, which are alternate functions, do not occur (V850E/IA3: n = 0, 2 to 7, V850E/IA4: n = 0 to 7).

CHAPTER 18 STANDBY FUNCTION

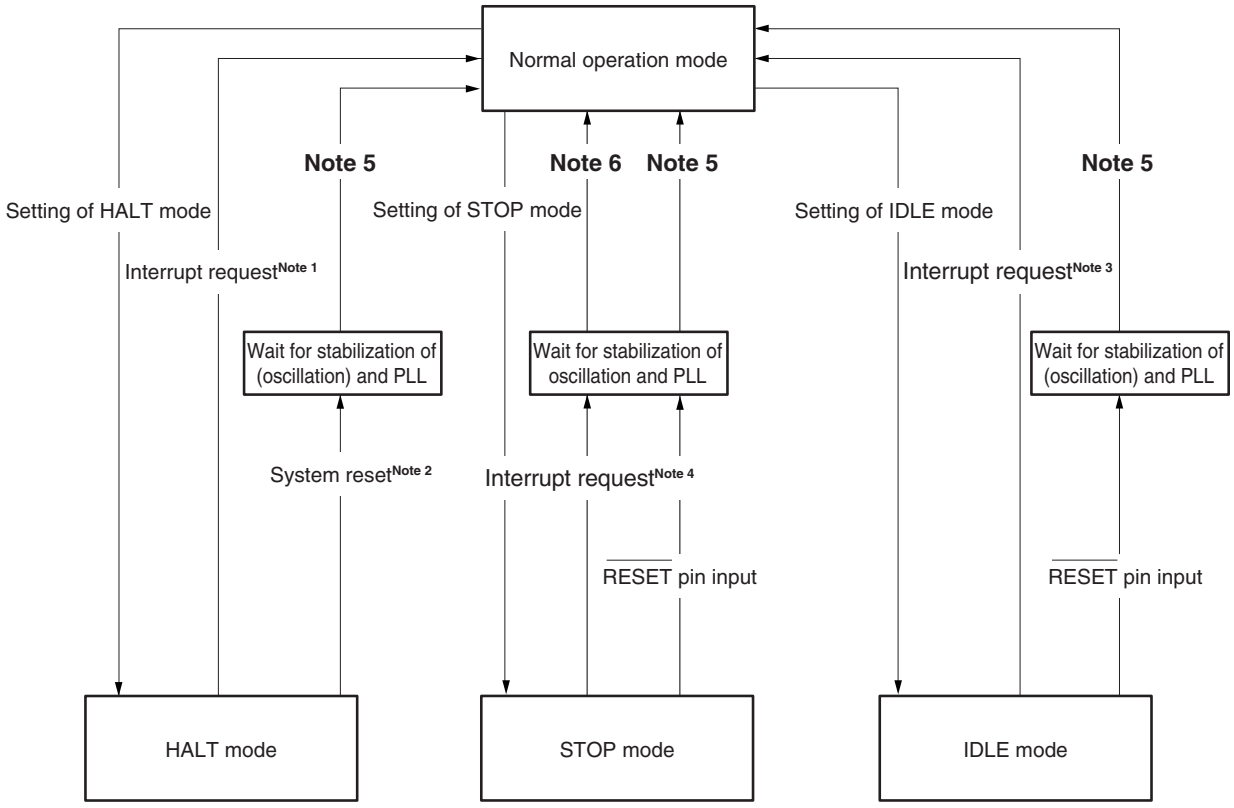
18.1 Overview

The power consumption of the system can be effectively reduced by using the standby modes in combination and selecting the appropriate mode for the application. The available standby modes are listed in Table 18-1.

Table 18-1. Standby Modes

Mode	Functional Outline
HALT mode	Mode to stop only the operating clock of the CPU
IDLE mode	Mode to stop all the operations of the internal circuit except the oscillator, PLL, CSIB in the slave mode
STOP mode	Mode to stop all the operations of the internal circuit except the CSIB in the slave mode

Figure 18-1. Status Transition



Notes 1. Non-maskable interrupt request signal (INTWDT) or unmasked maskable interrupt request signal

2. $\overline{\text{RESET}}$ pin input or reset signal (WDTRES) generation by watchdog timer overflow

3. Unmasked external interrupt request signal (INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, INTP7) or unmasked internal interrupt request signal (CSIB-related interrupt request signal in the slave mode) from peripheral functions operable in IDLE mode

4. Unmasked external interrupt request signal (INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, INTP7) or unmasked internal interrupt request signal from (CSIB-related interrupt request signal in the slave mode) peripheral functions operable in STOP mode

5. Oscillation stabilization time count by oscillation stabilization time wait control (OST)
The oscillation stabilization time is necessary after release of reset because the PLL is initialized by a reset. The stabilization time is the time determined by default.

6. Oscillation stabilization time count by oscillation stabilization time wait control (OST)
The stabilization time is determined by the setting of the OSTS register.

18.2 Control Registers

(1) Power save control register (PSC)

The PSC register is an 8-bit register that controls the standby function. The STB bit of this register is used to specify the standby mode. This register is a special register (see **3.4.8 Special registers**). This register can be written only by a combination of specific sequences.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF1FEH

	7	6	5	<4>	3	2	<1>	0
PSC	0	0	0	INTM	0	0	STB	0

INTM	Standby mode control by maskable interrupt request (INTxx[Note])
0	Standby mode release by INTxx request enabled
1	Standby mode release by INTxx request disabled

STB	Sets operation mode
0	Normal mode
1	Standby mode

Note For details, see **Table 17-1 Interrupt Source List**.

Cautions 1. Be sure to clear bits 0, 2, 3, and 5 to 7 to "0".

2. Before setting a standby mode by setting the STB bit to 1, be sure to set the PCC register to 03H and then set the STB bit to 1. Otherwise, the standby mode may not be set or released. After releasing the standby mode, change the value of the PCC register to the desired value.

3. To set the IDLE mode or STOP mode, set the PCC register to 03H, and the PSMR.PSM0 bit in that order and then set the STB bit to 1.

(2) Power save mode register (PSMR)

The PSMR register is an 8-bit register that controls the operation in the software standby mode.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF820H

	7	6	5	4	3	2	1	<0>
PSMR	0	0	0	0	0	0	0	PSM0

PSM0	Operation in software standby mode specification
0	IDLE mode
1	STOP mode

Cautions 1. Be sure to clear bits 1 to 7 to "0".

2. The PSM0 bit is valid only when the PSC.STB bit is 1.

18.3 HALT Mode

18.3.1 Setting and operation status

The HALT mode is set when a dedicated instruction (HALT) is executed in the normal operation mode.

When HALT mode is set, clock supply is stopped to the CPU only. The clock generator and PLL continue operating. Clock supply to the other on-chip peripheral functions continues.

As a result, program execution is stopped, and the internal RAM retains the contents before the HALT mode was set. The on-chip peripheral functions that are independent of instruction processing by the CPU continue operating.

Table 18-3 shows the operation status in the HALT mode.

The average power consumption of the system can be reduced by using the HALT mode in combination with the normal operation mode for intermittent operation.

Cautions 1. Insert five or more NOP instructions after the HALT instruction.

2. If the HALT instruction is executed while an interrupt request is being held pending, the HALT mode is set but is released immediately by the pending interrupt request.

18.3.2 Releasing HALT mode

The HALT mode is released by a non-maskable interrupt request signal (INTWDT), an unmasked maskable interrupt request signal, $\overline{\text{RESET}}$ pin input, and WDTRES signal generation.

After the HALT mode has been released, the normal operation mode is restored.

(1) Releasing HALT mode by non-maskable interrupt request signal or unmasked maskable interrupt request signal

The HALT mode is released by a non-maskable interrupt request signal (INTWDT) or an unmasked maskable interrupt request signal, regardless of the priority of the interrupt request. If the HALT mode is set in an interrupt servicing routine, however, an interrupt request that is issued later is serviced as follows.

(a) If an interrupt request signal with a priority lower than or same as the interrupt currently being serviced is generated, the HALT mode is released, but the newly generated interrupt request signal is not acknowledged. The interrupt request signal itself is retained. Therefore, execution starts at the next instruction after the HALT instruction.

(b) If an interrupt request signal with a priority higher than that of the interrupt currently being serviced is issued (including a non-maskable interrupt request signal), the HALT mode is released and that interrupt request signal is acknowledged. Therefore, execution branches to the handler address.

Table 18-2. Operation After Releasing HALT Mode by Interrupt Request Signal

Release Source	Interrupt Enabled (EI) Status	Interrupt Disabled (DI) Status
Non-maskable interrupt request signal	Execution branches to the handler address	
Unmasked maskable interrupt request signal	Execution branches to the handler address or the next instruction is executed	The next instruction is executed

(2) Releasing HALT mode by $\overline{\text{RESET}}$ pin input or WDTRES signal generation

The same operation as the normal reset operation is performed.

Table 18-3. Operation Status in HALT Mode

Setting of HALT Mode / Item		Operation Status
Clock generator, PLL		Operates
System clock (f_{XX})		Supply
CPU		Stops operation
DMA		Operable
Interrupt controller		Operable
ROM correction		Stops operation
Timer	TMM0	Operable
	TMP0 to TMP3	Operable
	TMQ0, TMQ1	Operable
	TMENC10, TMENC11[Note]	Operable
Watchdog timer		Operable
Serial interface	CSIB0, CSIB1	Operable
	UARTA0, UARTA1	Operable
A/D converters 0 to 2		Operable
Port function		Retains status before HALT mode was set.
Internal data		The CPU registers, statuses, data, and all other internal data such as the contents of the internal RAM are retained as they were before the HALT mode was set.

Note V850E/IA4 only

18.4 IDLE Mode

18.4.1 Setting and operation status

The IDLE mode is set by clearing (0) the PSMR.PSM0 bit and setting (1) the PSC.STB bit in the normal operation mode.

In the IDLE mode, the clock generator and PLL continue operation but clock supply to the CPU and other on-chip peripheral functions stops.

As a result, program execution stops and the contents of the internal RAM before the IDLE mode was set are retained. The CPU and other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate with an external clock continue operating.

Table 18-5 shows the operation status in the IDLE mode.

The IDLE mode can reduce the power consumption more than the HALT mode because it stops the operation of the on-chip peripheral functions. The clock generator and PLL do not stop, so the normal operation mode can be restored without waiting for the oscillation stabilization time after the IDLE mode has been released, in the same manner as when the HALT mode is released.

Caution Insert five or more NOP instructions after the instruction that stores data in the PSC register to set the IDLE mode.

18.4.2 Releasing IDLE mode

The IDLE mode is released by an unmasked external interrupt request signal (INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, INTP7 pin input), unmasked internal interrupt request signal (CSIB-related interrupt request signal in the slave mode) from the peripheral functions operable in the IDLE mode, or $\overline{\text{RESET}}$ pin input.

After the IDLE mode has been released, the normal operation mode is restored.

(1) Releasing IDLE mode by unmasked maskable interrupt request signal

The IDLE mode is released by an unmasked maskable interrupt request signal, regardless of the priority of the interrupt request. If the IDLE mode is set in an interrupt servicing routine, however, an interrupt request that is issued later is processed as follows.

Caution When PSC.INTM bit = 1, the IDLE mode cannot be released by the unmasked maskable interrupt request signal.

(a) If an interrupt request with a priority lower than or same as the interrupt request signal currently being serviced is generated, the IDLE mode is released, but the newly generated interrupt is not acknowledged. The interrupt request signal itself is retained. Therefore, execution starts at the next instruction after the IDLE instruction.

(b) If an interrupt request signal with a priority higher than that of the interrupt request signal currently being serviced is issued (including a non-maskable interrupt request signal), the IDLE mode is released and that interrupt request signal is acknowledged. Therefore, execution branches to the handler address.

Table 18-4. Operation After Releasing IDLE Mode by Interrupt Request Signal

Release Source	Interrupt Enabled (EI) Status	Interrupt Disabled (DI) Status
Unmasked maskable interrupt request	Execution branches to the handler address or the next instruction is executed	The next instruction is executed

(2) Releasing IDLE mode by $\overline{\text{RESET}}$ pin input

The same operation as the normal reset operation is performed.

Table 18-5. Operation Status in IDLE Mode

Setting of IDLE Mode / Item		Operation Status
Clock generator, PLL		Operates
System clock (f_{XX})		Stops supply
CPU		Stops operation
DMA		Stops operation
Interrupt controller		Stops operation
ROM correction		Stops operation
Timer	TMM0	Stops operation
	TMP0 to TMP3	Stops operation
	TMQ0, TMQ1	Stops operation
	TMENC10, TMENC11[Note]	Stops operation
Watchdog timer		Stops operation
Serial interface	CSIB0, CSIB1	Operable when $\overline{\text{SCKBn}}$ input clock is selected as count clock (in slave mode) (n = 0, 1)
	UARTA0, UARTA1	Stops operation
A/D converters 0 to 2		Stops operation
Port function		Retains status before IDLE mode was set.
Internal data		The CPU registers, statuses, data, and all other internal data such as the contents of the internal RAM are retained as they were before the IDLE mode was set.

Note V850E/IA4 only

18.5 STOP Mode

18.5.1 Setting and operation status

The STOP mode is set by setting (1) the PSMR.PSM0 bit and setting (1) the PSC.STB bit in the normal operation mode.

In the STOP mode, the clock generator stops operation. Clock supply to the CPU and the on-chip peripheral functions is stopped.

As a result, program execution is stopped, and the contents of the internal RAM before the STOP mode was set <R> are retained. The CPU and other on-chip peripheral functions stop operating. However, the on-chip peripheral functions that can operate with an external clock continue operating.

Table 18-7 shows the operation status in the STOP mode.

Because the STOP stops operation of the clock generator, it reduces the power consumption to a level lower than the IDLE mode. When the external clock is not used, the power consumption can be minimized with only leakage current flowing.

Caution Insert five or more NOP instructions after the instruction that stores data in the PSC register to set the STOP mode.

18.5.2 Releasing STOP mode

The STOP mode is released by an unmasked external interrupt request signal (INTP0, INTP1 (V850E/IA4 only), INTP2 to INTP5, INTP7 pin input), unmasked internal interrupt request signal (CSIB-related interrupt signal in the slave mode) from the peripheral functions operable in the STOP mode, or $\overline{\text{RESET}}$ pin input.

After the STOP mode has been released, the normal operation mode is restored after the oscillation stabilization time has been secured.

(1) Releasing STOP mode by unmasked maskable interrupt request signal

The STOP mode is released by an unmasked maskable interrupt request signal, regardless of the priority of the interrupt request. If the STOP mode is set in an interrupt servicing routine, however, an interrupt request that is issued later is serviced as follows.

Caution When PSC.INTM bit = 1, the IDLE mode cannot be released by the unmasked maskable interrupt request signal.

(a) If an interrupt request with a priority lower than or same as the interrupt request currently being serviced is generated, the STOP mode is released, but the newly generated interrupt is not acknowledged. The interrupt request itself is retained. Therefore, execution starts at the next instruction after the STOP instruction.

(b) If an interrupt request with a priority higher than that of the interrupt request currently being serviced is issued, the STOP mode is released and that interrupt request is acknowledged. Therefore, execution branches to the handler address.

Table 18-6. Operation After Releasing STOP Mode by Interrupt Request Signal

Release Source	Interrupt Enabled (EI) Status	Interrupt Disabled (DI) Status
Unmasked maskable interrupt request	Execution branches to the handler address or the next instruction is executed after securing oscillation stabilization time	The next instruction is executed after securing oscillation stabilization time

(2) Releasing STOP mode by $\overline{\text{RESET}}$ pin input

The same operation as the normal reset operation is performed.

Table 18-7. Operation Status in STOP Mode

Setting of STOP Mode / Item		Operation Status
Clock generator, PLL		Stops operation
System clock (f_{XX})		Stops supply
CPU		Stops operation
DMA		Stops operation
Interrupt controller		Stops operation
ROM correction		Stops operation
Timer	TMM0	Stops operation
	TMP0 to TMP3	Stops operation
	TMQ0, TMQ1	Stops operation
	TMENC10, TMENC11[Note]	Stops operation
Watchdog timer		Stops operation
Serial interface	CSIB0, CSIB1	Operable when $\overline{\text{SCKBn}}$ input clock is selected as count clock (in slave mode) (n = 0, 1)
	UARTA0, UARTA1	Stops operation
A/D converters 0 to 2		Stops operation
Port function		Retains status before STOP mode was set.
Internal data		The CPU registers, statuses, data, and all other internal data such as the contents of the internal RAM are retained as they were before the STOP mode was set.

Note V850E/IA4 only

18.6 Securing Oscillation Stabilization Time

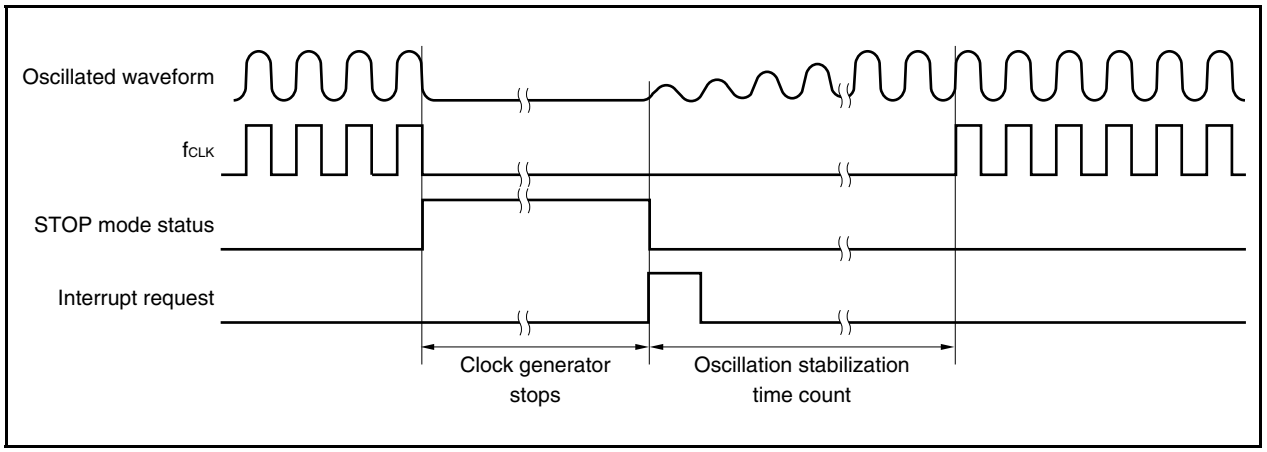
When the STOP mode is released, the oscillation stabilization time set by the OSTS register elapses. The oscillation stabilization time is the reset value of the OSTS register, $2^{14}/f_X$ (2.048 ms at f_X = 8 MHz), if the STOP mode is released by $\overline{\text{RESET}}$ pin input.

However, the actual oscillation stabilization time is half this value (after reset: $2^{13}/f_X$ (1.024 ms at f_X = 8 MHz), and the other half is the stabilization time of the PLL. Set an oscillation stabilization time double that of the oscillation stabilization time of the oscillator used when the STOP mode is released. If the oscillation stabilization time of the oscillator used is longer than $2^{13}/f_X$ when the STOP mode is released by $\overline{\text{RESET}}$ pin input, secure the oscillation stabilization time with the low-level width of the $\overline{\text{RESET}}$ signal.

The timer for counting the oscillation stabilization time secures oscillation stabilization time equal to the overflow time of the watchdog timer.

The operation performed when the STOP mode is released by an interrupt request signal is shown below.

Figure 18-2. Oscillation Stabilization Time



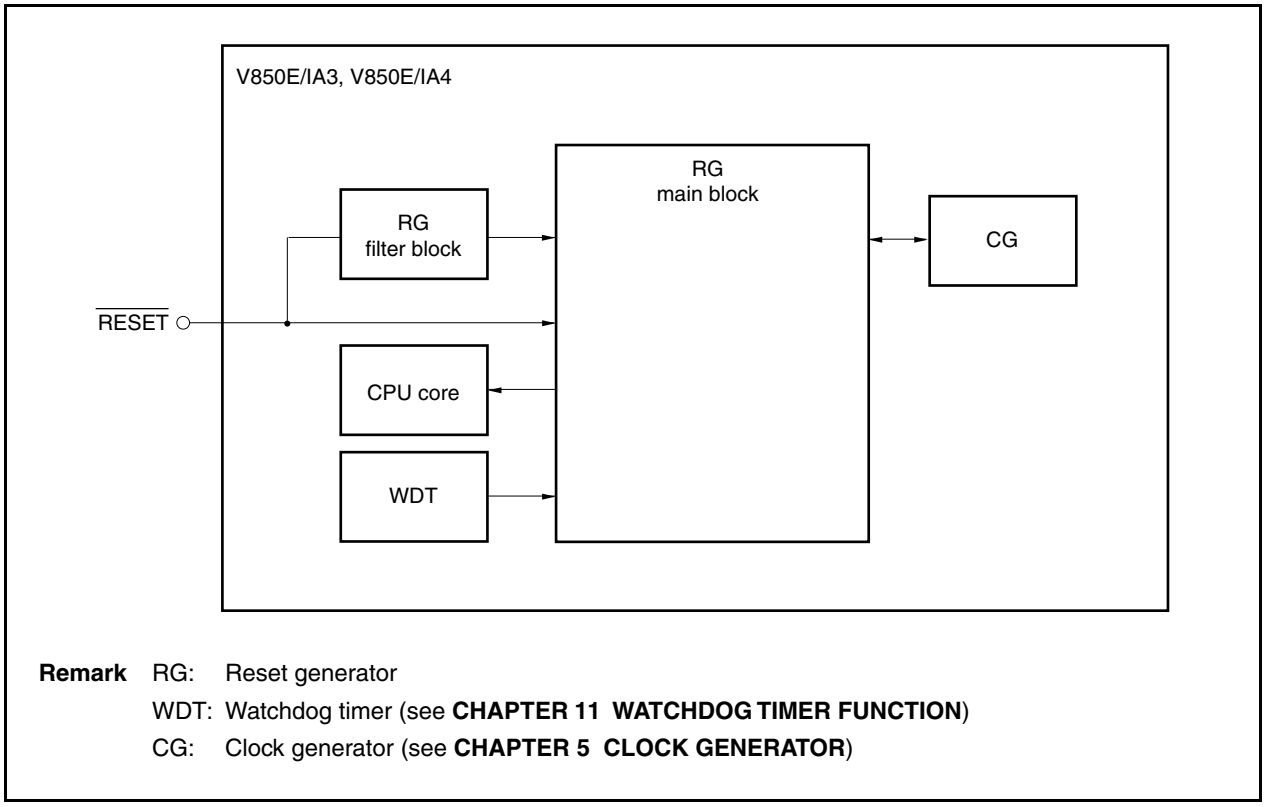
Caution For details of the OSTS register, see 5.3 (5) Oscillation stabilization time select register (OSTS).

CHAPTER 19 RESET FUNCTIONS

19.1 Overview

- System reset by $\overline{\text{RESET}}$ pin input
- System reset signal (WDTRES) generation by watchdog timer (WDT) overflow
- Forced reset by on-chip debug function (DCU) and reset mask function (see **CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)**.)

19.2 Configuration



Remark RG: Reset generator
WDT: Watchdog timer (see **CHAPTER 11 WATCHDOG TIMER FUNCTION**)
CG: Clock generator (see **CHAPTER 5 CLOCK GENERATOR**)

19.3 Control Register

(1) Reset source flag register (RESF)

The RESF register is an 8-bit register that indicates occurrence of a reset request from the watchdog timer (WDT).

The RESF. RESFH4 bit of this register is set to 1 when the internal reset source signal from WDT is asserted. The RESFH4 bit is cleared by reset via the $\overline{\text{RESET}}$ pin or by a bit manipulation instruction or store instruction (writing 0 to the RESFH4 bit).

The RESF register is a special register and can be written only in a combination of specific sequences (see **3.4.8 Special registers**).

This register can be read or written in 8-bit or 1-bit units. However, bit 4 can only be cleared (0) when writing.

Reset via the $\overline{\text{RESET}}$ pin input sets this register to 00H, and reset by the watchdog timer sets this register to 10H. For details on reset conflict, see **Caution** below.

After reset: **Note** R/W Address: FFFFF888H

	7	6	5	4	3	2	1	0
RESF	0	0	0	RESFH4	0	0	0	0

RESFH4	Occurrence of reset request from watchdog timer (WDT)
0	Read: No reset request, Write: Clear
1	Reset request

Note After reset by $\overline{\text{RESET}}$ pin input: 00H
After reset by watchdog timer: 10H

Caution If setting (occurrence of reset of set source) and clearing (occurrence of system reset or writing 0 to the RESFH4 bit) of the RESF register conflict, the priorities are as follows.

1. **Occurrence of reset via $\overline{\text{RESET}}$ pin input (clearing RESF register)**
2. **Occurrence of reset by watchdog timer (setting RESF register)**
3. **Writing 0 to the RESFH4 bit by a bit manipulation or store instruction (clearing RESF register)**

If the occurrence of reset via the $\overline{\text{RESET}}$ pin input and the occurrence of reset by the watchdog timer conflict, the RESF register is not set but cleared (00H).

19.4 Operation

(1) Reset operation by $\overline{\text{RESET}}$ pin input

When a low level is input to the $\overline{\text{RESET}}$ pin, the V850E/IA3 and V850E/IA4 are reset, and each hardware unit is initialized to a specific status.

The oscillator continues oscillation even while a low level is input to the $\overline{\text{RESET}}$ pin but the oscillation mode is initialized to the clock-through mode (PLLCTL register = 01H) and the CPU clock (f_{CPU}) division to $f_{XX}/8$ (PCC register = 03H).

The reset status is released when the $\overline{\text{RESET}}$ pin input goes from low to high. After the reset status is released, the oscillation stabilization time of the oscillator and lockup time of PLL (default value of OSTS register for the total time: $2^{14}/f_X$ (2.05 ms (f_X = 8 MHz)) elapse, and then the CPU starts program execution. After release of reset, therefore, the operation is started in the clock-through mode and at $f_{XX}/8$.

The status of each hardware unit during the reset period and after the reset status is released is shown below.

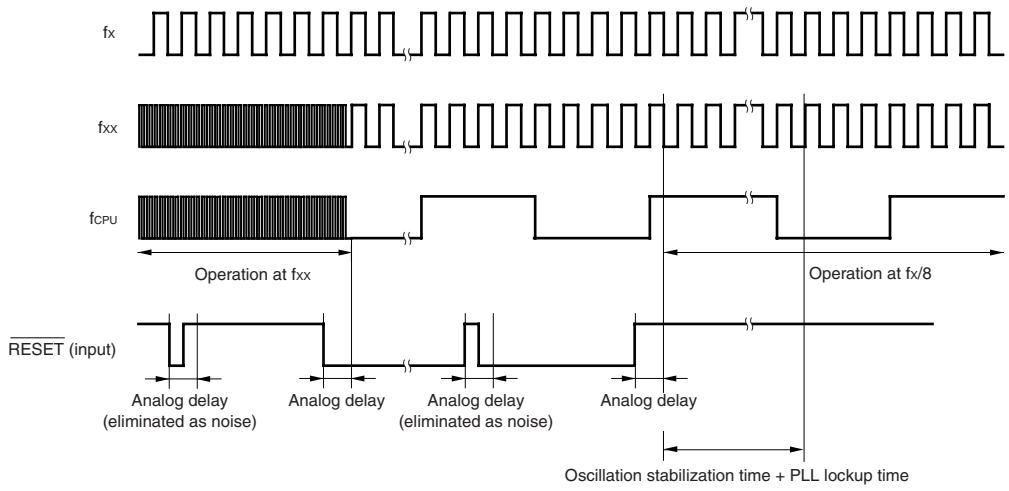
Hardware	During Reset Period	After Reset Is Released
Clock generator: Oscillator (f_X) Internal system clock (f_{CLK}) CPU clock (f_{CPU})	Oscillation/supply continues However, the CPU clock (f_{CPU}) is initialized to $f_{XX}/8$.	
Clock generator: Peripheral clock (f_{XX} to $f_{XX}/1,024$)	Oscillation/supply stops	Oscillation/supply starts after securing of oscillation stabilization time
Clock generator: Watchdog timer clock ($f_{XX}/1,024$)	Oscillation/supply stops	Oscillation/supply starts
CPU	Initialized	Program execution starts[Note 1] after securing of oscillation stabilization time
Internal RAM	Undefined if power-on reset or writing data to RAM (by CPU) conflicts with reset input (data destroyed). Otherwise, retains value immediately before reset input[Note 2].	
Ports (including alternate-function pins)	High impedance	
On-chip peripheral I/O registers (other than ports)	Initialized to specific status	
On-chip peripheral functions other than above	Stops operation	Can start operation

Notes 1. With the μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4), program execution is delayed by insertion of internal processing for boot switching.

2. The firmware of the μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4) uses part of the internal RAM (used RAM area: 3FFD800H to 3FFD895H, 3FFEFBAH to 3FFEFFFH) after the internal system reset operation has been released because it supports a boot switching function. Therefore, the contents of some RAM areas are not retained on power-on reset.

The reset operation by $\overline{\text{RESET}}$ pin input is illustrated below.

Figure 19-1. Reset Operation by $\overline{\text{RESET}}$ Pin Input



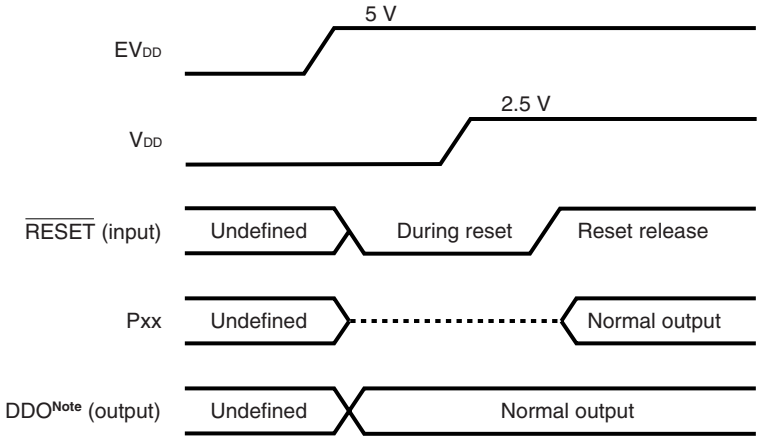
Caution After release of reset, make sure that the oscillation stabilization time (1.024 ms (at f_X = 8 MHz)) and PLL lockup time (1.024 ms (at f_X = 8 MHz)) elapse. If an oscillation stabilization time of 1.024 ms is not sufficient to secure stable oscillation, keep the $\overline{\text{RESET}}$ pin low for the deficient time.

Remark The relationship between f_{XX} and f_X in the above timing chart is $f_{XX} = 8 \times f_X$.

The operation after release of reset is the same in both the PLL mode and clock-through mode and is started in the clock-through mode. Set the PLL mode by software control (setting PLLCTL.SELPLL bit to 1). To improve noise immunity, it is recommended to set the PLL mode and then speed up the CPU clock (example: PCC register = 00H (f_{XX} operation)).

The timing and pin status at power-on are shown below.

Figure 19-2. Timing and Pin Status at Power-On



Note μPD70F3186 (V850E/IA4) only

Remark The broken lines indicate the high-impedance state.

(2) Reset operation (WDTRES) by overflow of watchdog timer (WDT)

If the reset mode is set to reset upon overflow of the watchdog timer (WDT) (WDTM.WDM1 and WDTM.WDM0 bits = 10 or 11), the system is reset and each hardware is initialized to a specific state when WDT overflows (INTWDT).

If the INTWDT interrupt request signal is generated, the RESF.RESFH4 bit is set to 1, indicating that internal reset has occurred.

The operations during the reset period and after release of reset, other than the operation of the RESF register, are the same as the reset operation by $\overline{\text{RESET}}$ pin input (see **(1) Reset operation by $\overline{\text{RESET}}$ pin input**).

CHAPTER 20 ROM CORRECTION FUNCTION

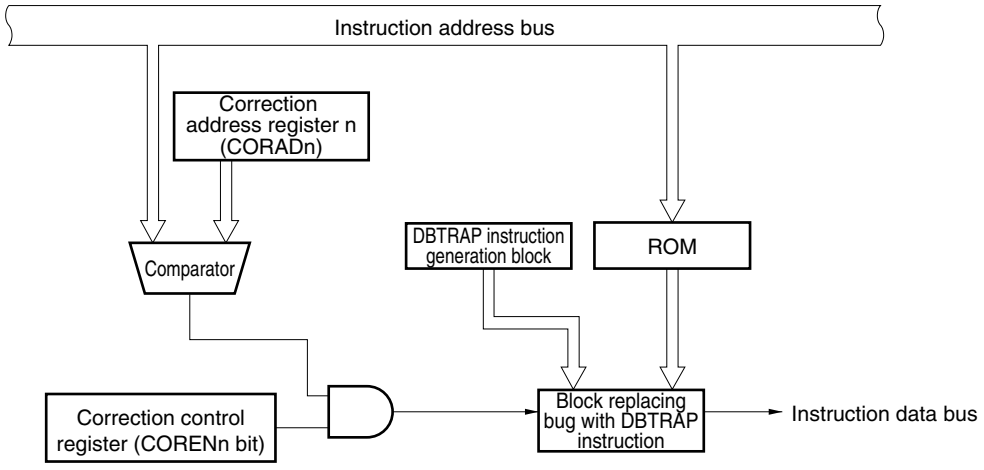
20.1 Overview

The ROM correction function is used to replace part of the program in the mask ROM or flash memory with the program of the internal RAM.

By using this function, program bugs found in the mask ROM or flash memory can be corrected.

The correction address can be specified at up to four places by the ROM correction function.

Figure 20-1. Block Diagram of ROM Correction



Remark n = 0 to 3

20.2 Control Registers

(1) Correction address registers 0 to 3 (CORAD0 to CORAD3)

The CORAD0 to CORAD3 registers set the first address of the correction program.

The program can be corrected at up to four places because four CORADn registers are provided (n = 0 to 3).

The CORADn register can be read or written in 32-bit units.

If the higher 16 bits of the CORADn register are used as the CORADnH register, and the lower 16 bits as the CORADnL register, these registers can be read or written in 16-bit units.

Reset sets these registers to 00000000H.

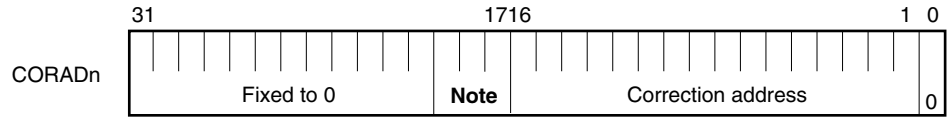
Because the ROM capacity differs depending on the product, set correction addresses in the following ranges.

μPD703183, 703185 (128 KB): 0000000H to 001FFFEH
μPD70F3184, 703186, 70F3186 (256 KB): 0000000H to 003FFFEH

Fix bits 0 and 20 to 31 to “0”.

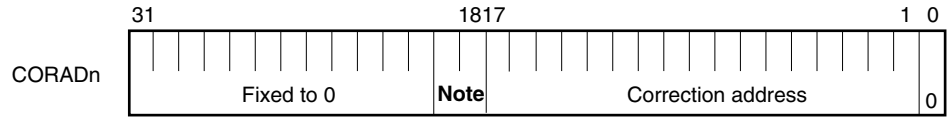
After reset: 00000000H R/W Address: CORAD0 FFFFF840H
CORAD0L FFFFF840H, CORAD0H FFFFF842H
CORAD1 FFFFF844H
CORAD1L FFFFF844H, CORAD1H FFFFF846H
CORAD2 FFFFF848H
CORAD2L FFFFF848H, CORAD2H FFFFF84AH
CORAD3 FFFFF84CH
CORAD3L FFFFF84CH, CORAD3H FFFFF84EH

(a) When 128 KB



(n = 0 to 3)

(b) When 256 KB



(n = 0 to 3)

Note Be sure to clear these bits to 0.

(2) Correction control register (CORCN)

This register disables or enables the correction operation at the address set by each CORADn register (n = 0 to 3).

Each channel can be enabled or disabled by this register.

This register can be read or written in 8-bit or 1-bit units.

Reset sets this register to 00H.

After reset: 00H R/W Address: FFFFF880H

	7	6	5	4	<3>	<2>	<1>	<0>
CORCN	0	0	0	0	COREN3	COREN2	COREN1	COREN0

CORENn	Enables/disables correction operation
0	Disabled
1	Enabled

Remark n = 0 to 3

Table 20-1. Correspondence Between CORCN Register Bits and CORADn Registers

CORCN Register Bit	Corresponding CORADn Register
COREN3	CORAD3
COREN2	CORAD2
COREN1	CORAD1
COREN0	CORAD0

20.3 ROM Correction Operation and Program Flow

<1> If the address to be corrected and the fetch address of the internal ROM match, the fetch code is replaced by the DBTRAP instruction.

<2> When the DBTRAP instruction is executed, execution branches to address 00000060H.

<3> Software processing after branching causes the result of ROM correction to be judged (the fetch address and ROM correction operation are confirmed) and execution to branch to the correction software.

<4> After the correction software has been executed, the return address is set, and return processing is started by the DBRET instruction.

Cautions 1. The software that performs <3> and <4> must be executed in the internal ROM/RAM.

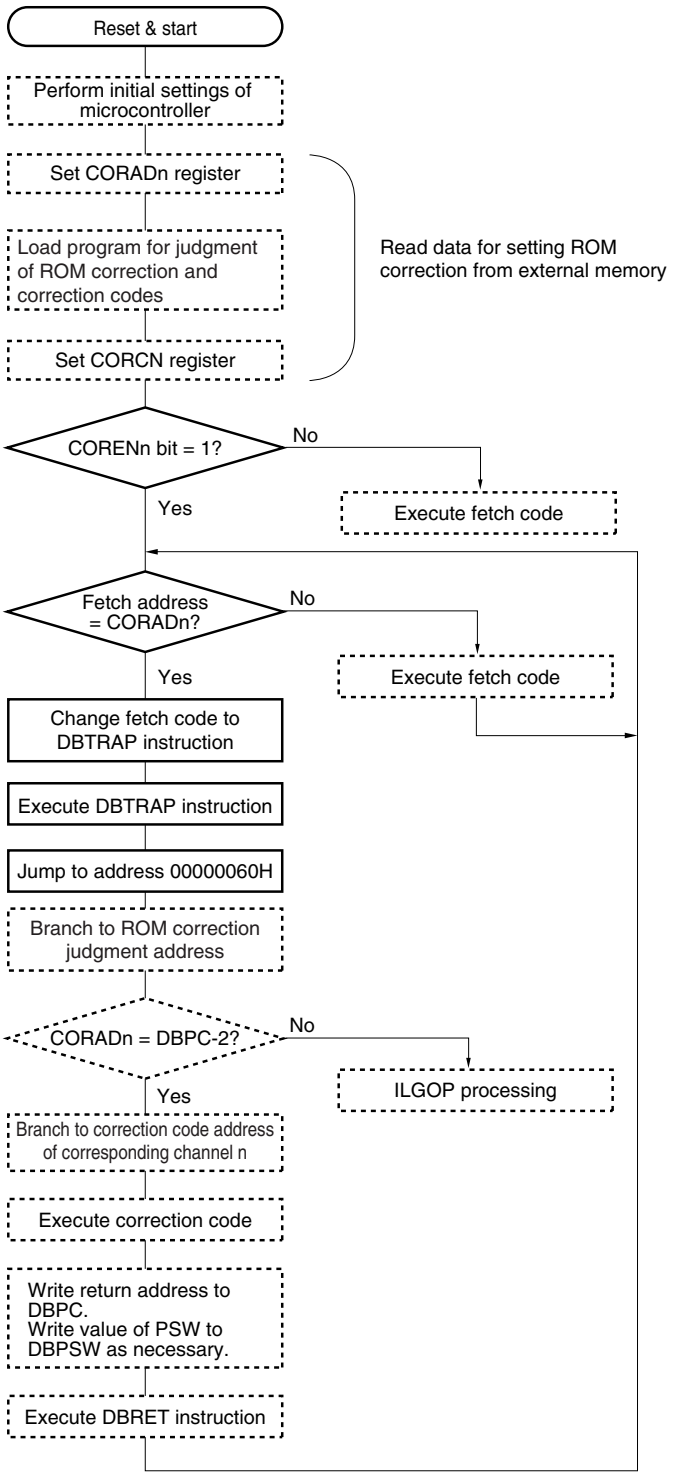
2. When setting an address to be corrected to the CORADn register, clear the higher bits to 0 in accordance with the capacity of the internal ROM.

3. The ROM correction function cannot be used to correct the data of the internal ROM. It can only be used to correct instruction codes. If ROM correction is used to correct data, that data is replaced with the DBTRAP instruction code.

4. Use of ROM correction is prohibited if self-programming is performed in the μPD70F3184 (V850E/IA3) or 70F3186 (V850E/IA4).

5. ROM correction cannot be used when DMA transfer is executed in the internal RAM (do not execute DMA transfer in the internal RAM and instructions in the internal RAM at the same time).

Figure 20-2. ROM Correction Operation and Program Flow



Remarks 1. : Processing by user program (software)

: Processing by ROM correction (hardware)

2. n = 0 to 3

CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)

An on-chip debug unit is provided in the μPD70F3186 (V850E/IA4) and realizes standalone on-chip debugging of the μPD70F3186 by connecting an on-chip debug emulator (an on-chip debug function is not provided in the μPD703185 (V850E/IA4), μPD703186 (V850E/IA4), and the V850E/IA3).

<R> **Caution The debug function explained in this chapter is the function that can be realized by using the μPD70F3186 (V850E/IA4), the NEC Electronics' QB-V850MINI (on-chip debug emulator), and the debugger ID850QB. When using a partner manufacturer's on-chip debug emulator, refer to the manual for the debugger used.**

21.1 Functional Overview

21.1.1 On-chip debug unit type

The on-chip debug unit incorporated in the μPD70F3186 (V850E/IA4) is RCU1 (run control unit 1). The on-chip unit incorporated differs depending on the microcontroller, and also features different functions.

21.1.2 Debug function

For details of the debug function, refer to the **ID850QB Operation User's Manual**.

(1) Debug interface

This interface establishes communication with the host machine by using the $\overline{\text{DRST}}$, DCK, DMS, DDI, and DDO signals, via an on-chip debug emulator. The communication specifications of on-chip debug are used for this interface. It does not support a boundary scan function.

(2) On-chip debug

On-chip debugging can be performed by providing wiring and connectors for debugging on the target system. Connect an on-chip debug emulator to the emulator connector.

(3) Forced reset function

The μPD70F3186 (V850E/IA4) can be forcibly reset.

(4) Break reset function

The CPU can be started in the debug mode immediately after resetting the CPU has been cleared.

(5) Forced break function

Execution of the user program can be forcibly stopped (however, the handler of the illegal instruction code exception (first address: 00000060H) cannot be used).

(6) Hardware break function

Two common instruction fetch/access breakpoints can be used. By using the instruction breakpoint, program execution can be suspended at an arbitrary address. By using the access breakpoint, program execution can be suspended by data-accessing an arbitrary address. In addition to these two breakpoints, a software break function is available. Up to four software breakpoints can be set in the internal ROM area. The number of software breakpoints that can be set in the internal RAM area differs depending on the debugger used.

(7) Debug monitor function

During debugging, a memory space for debugging that differs from the user memory space is used (background monitor format). The user program can be executed starting from any address.

While execution of the user program is stopped, the user resources (such as memory and I/O) can be read or written, and the user program can be downloaded.

(8) Mask function

Signals can be masked.

<R> The mask function of the debugger (ID850QB) for the on-chip debug emulator (QB-V850MINI) and the corresponding functions are shown below.

Mask Function of Debugger (ID850QB)	Corresponding Function of μPD70F3186 (V850E/IA4)
NMI0 mask function	INTWDT interrupt
RESET mask function	$\overline{\text{RESET}}$ pin input, reset signal (WDRES) generation by overflow of watchdog timer

(9) Timer function

The execution time of the user program can be measured.

(10) Function to select operation/stop of peripheral functions during a break

Depending on the debugger used, whether to operate or stop peripheral functions during a break can be selected.

- Peripheral functions that always stop during a break
  Watchdog timer, clock monitor function
- Peripheral functions for which operation/stop can be selected (however, cannot be selected individually)
  A/D converters 0 and 1, timer M (TMM0), timer P (TMP0 to TMP3), timer Q (TMQ0, TMQ1), timer Q option unit (TMQOP0, TMQOP1)
- Peripheral functions that continue operating during a break (cannot be stopped)
  All peripheral functions other than above

(11) Open break function

If a break occurs during on-chip debugging, the specified timer output can be set to a high-impedance state. By using this function, devices and systems that are driven by timer output can be protected during a break.

21.1.3 ROM security function

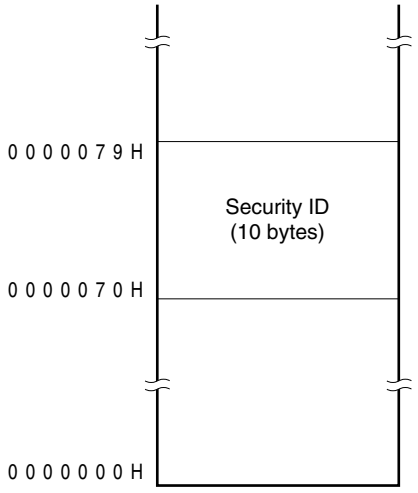
(1) Security ID

The flash memory versions of the V850E/IA3 and V850E/IA4 perform authentication using a 10-byte ID code to prevent the contents of the flash memory from being read by an unauthorized person during on-chip debugging by the on-chip debug emulator.

Set the ID code in the 10-byte on-chip flash memory area from 0000070H to 0000079H to allow the debugger perform ID authentication.

If the IDs match, the security is released and reading flash memory and using the on-chip debug emulator are enabled.

- Set the 10-byte ID code to 0000070H to 0000079H.
- Bit 7 of 0000079H is the on-chip debug emulator enable flag.
  (0: Disable, 1: Enable)
- When the on-chip debug emulator is started, the debugger requests ID input. When the ID code input on the debugger and the ID code set in 0000070H to 0000079H match, the debugger starts.
- Debugging cannot be performed if the on-chip debug emulator enable flag is 0, even if the ID codes match.



Caution When the data in the flash memory has been deleted, all the bits are set to 0xFF. Therefore, ID code is FFFFFFFFFFFFFFFFFFFFH.

(2) Setting

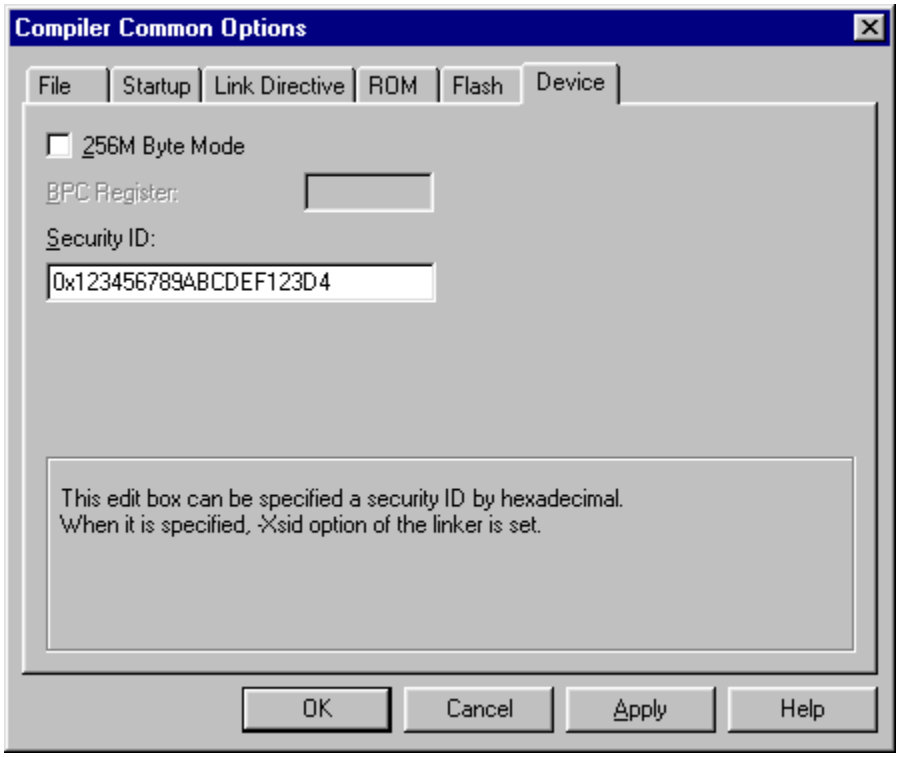
How to set the ID code as shown in Table 21-1 is shown below.

When the ID code is set as shown in Table 21-1, the ID code input in the configuration dialog box of the ID850QB is “123456789ABCDEF123D4” (no distinction is made between uppercase and lowercase for the ID code).

Table 21-1. ID Code

Address	Value
0x70	0x12
0x71	0x34
0x72	0x56
0x73	0x78
0x74	0x9A
0x75	0xBC
0x76	0xDE
0x77	0xF1
0x78	0x23
0x79	0xD4

<R> If the device file supports CA850 Ver. 3.10 or later and the security ID, the ID code can be specified using the compiler common option settings of PM+.



21.2 Selecting On-Chip Debug Function and Port Function (Including Alternate Functions)

In the μPD70F3186 (V850E/IA4), pins P50 to P52 also function as on-chip debug pins.

The on-chip debug function or port function (including the alternate functions) can be selected by using the level of the $\overline{\text{DRST}}$ pin, as shown in the table below.

Port 5 Functions	
$\overline{\text{DRST}}$ Pin Low-Level Input	$\overline{\text{DRST}}$ Pin High-Level Input
P50/TIUD11/TO11	DDI
P51/TCUD11	DCK
P52/TCLR11	DMS

Caution Because the DDI, DCK, and DMS pins function alternately as the I/O pins of timer ENC11 (TIUD11, TO11, TCUD11, TCLR11), timer ENC11 cannot be used while the on-chip debug function is being used.

21.3 Connection with On-Chip Debug Emulator

To connect an on-chip debug emulator, it is necessary to mount an emulator connector and circuit for connection on the target system.

Select either the KEL connector, MICTOR connector (Part number: 2-767004-2, distributor: Tyco Electronics AMP K.K.), or 2.54 mm pitch 20-pin general-purpose connector as the emulator connector. Connectors other than the KEL connector may not be supported, depending on the emulator, so when using a connector, refer to the manual of the emulator used.

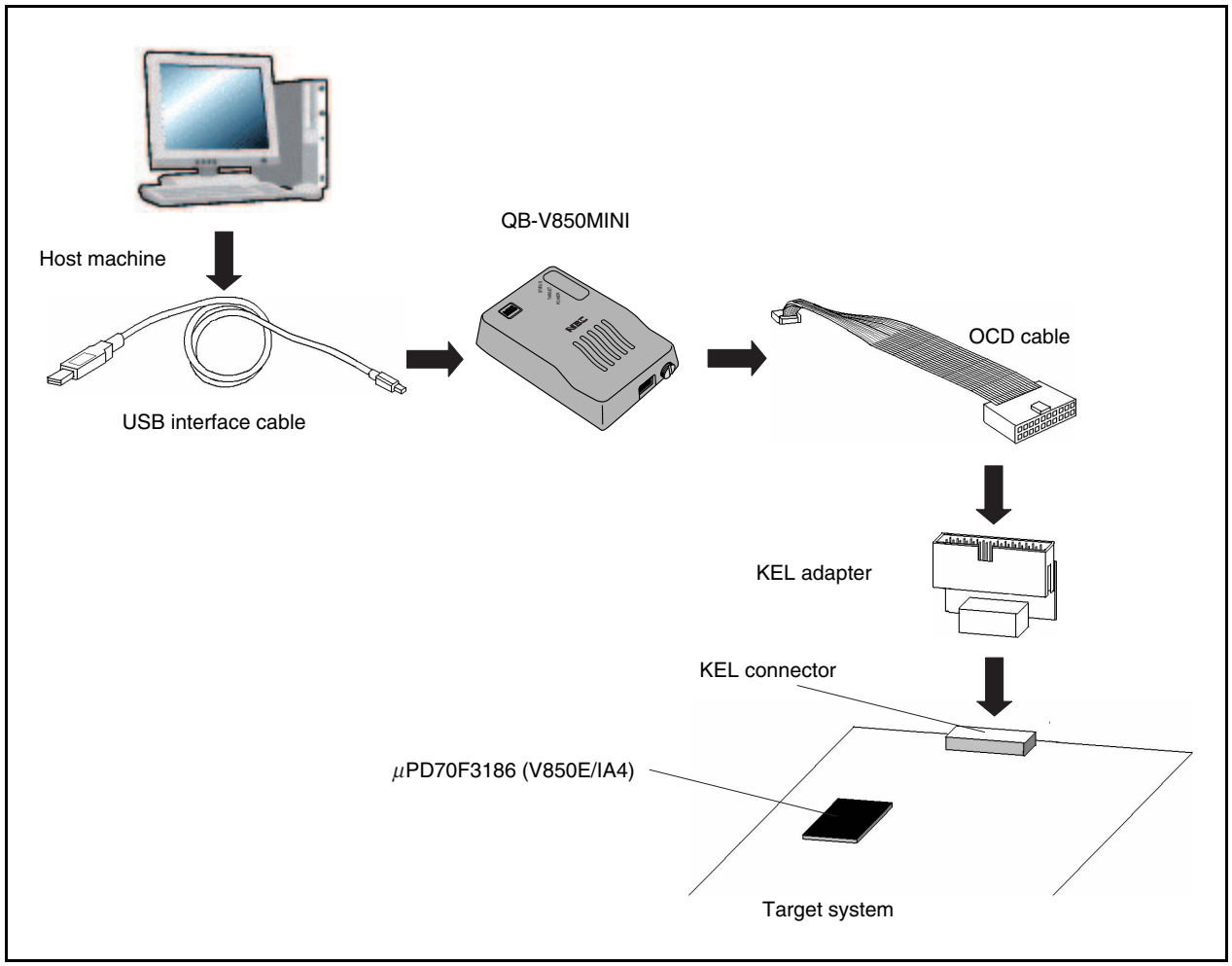
21.3.1 KEL connector

<R> When the QB-V850MINI is used, use of the following connector is recommended.

○ Part number
- 8830E-026-170S: Straight type
- 8830E-026-170L: Right-angle type

<R> It is necessary to mount an emulator and circuit for connection on the target system.

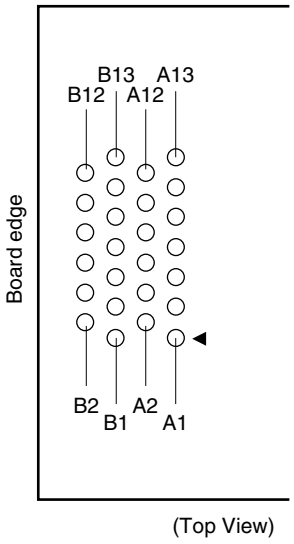
<R> **Figure 21-1. Connecting On-Chip Debug Emulator (QB-V850MINI)**



(1) Pin configuration

Figure 21-2 shows the pin configuration of the emulator connector (target system side), and Table 21-2 shows the pin functions.

Figure 21-2. Pin Configuration of Emulator Connector (on Target System Side)



Caution Design the board based on the dimensions of the connector when actually mounting the connector on the board.

(2) Pin functions

The following table shows the pin functions of the emulator connector (on the target system side).

<R> **Table 21-2. Pin Functions of Connector for QB-V850MINI (on Target System Side)**

Pin No.	Pin Name	I/O	Pin Function
A1	(Reserved 1)	–	(Connect to GND)
A2	(Reserved 2)	–	(Connect to GND)
A3	(Reserved 3)	–	(Connect to GND)
A4	(Reserved 4)	–	(Connect to GND)
A5	(Reserved 5)	–	(Connect to GND)
A6	(Reserved 6)	–	(Connect to GND)
A7	DDI	Output	Data output for N-Wire interface
A8	DCK	Output	Clock output for N-Wire interface
A9	DMS	Output	Transfer mode select output for N-Wire interface
A10	DDO	Input	Data input for N-Wire interface
A11	$\overline{\text{DRST}}$	Output	On-chip debug unit reset output
A12	(Reserved 7)	–	(Leave open)
A13	FLMD0	Output	Control signal for flash memory downloading
B1	GND	–	–
B2	GND	–	–
B3	GND	–	–
B4	GND	–	–
B5	GND	–	–
B6	GND	–	–
B7	GND	–	–
B8	GND	–	–
B9	GND	–	–
B10	GND	–	–
B11	PORT0_IN	–	(Connect to GND)
B12	PORT1_IN	–	(Connect to GND)
B13	V_{DD}	–	5 V input (for monitoring power application to target)

Cautions 1. The connection of the pins not supported in the μPD70F3186 (V850E/IA4) depends on the emulator used.

2. The pattern on the target board must satisfy the following conditions.

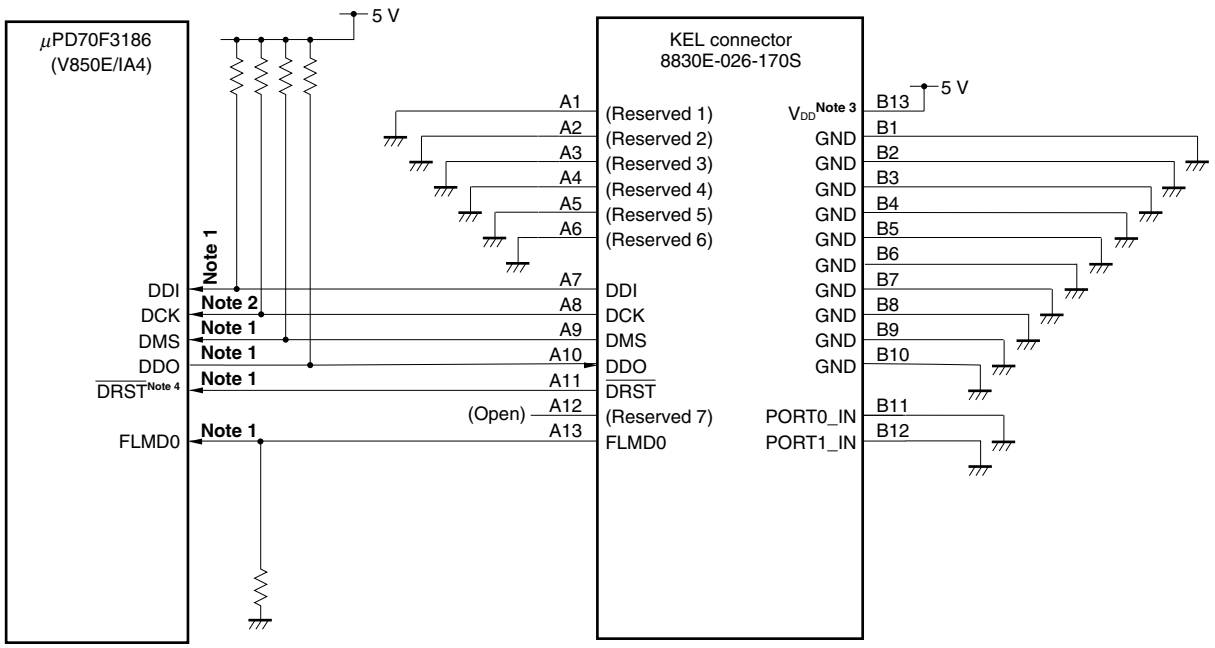
- **Keep the pattern length to within 100 mm.**
- **Shield the clock signal with GND.**

Remark Input/output is as viewed from the emulator side.

(3) Recommended circuit example

The following figure shows an example of the recommended circuit of the emulator connector (on the target system side).

Figure 21-3. Example of Recommended Connection of μPD70F3186 (V850E/IA4) and KEL Connector



Notes 1. Keep the pattern length to within 100 mm.

2. Shield the DCK signal with GND.

<R> **3.** For detecting power supply to the target board.

4. When $\overline{\text{DRST}}$ pin is high level: On-chip debug mode
When $\overline{\text{DRST}}$ pin is low level: Normal operation mode
The $\overline{\text{DRST}}$ pin is internally pulled down in the μPD70F3186.

Cautions 1. The DDO signal is 5 V output, and the input level of the DDI, DCK, DMS, and $\overline{\text{DRST}}$ signals is TTL level.

2. A 5 V interface may not be supported, so a level shifter may be required by some on-
<R> **chip debug emulators. Refer to the manual of the emulator used. Note that the QB-V850MINI supports a 5 V interface.**

21.4 Cautions

(1) The flash memory of the device used in debugging is rewritten during debugging, so the number of flash memory rewrites cannot be guaranteed. Therefore, do not use the device used in debugging for a mass production product.

(2) If a reset ($\overline{\text{RESET}}$ signal input from the target system or reset input by an internal reset source) occurs during RUN (program execution), the break function may malfunction.

(3) Even if reset is masked by using the mask function, the I/O buffers (port pins, etc.) are set to the reset state when the $\overline{\text{RESET}}$ signal is input.

(4) $\overline{\text{RESET}}$ signal input during a break is masked.

(5) The ROM correction function cannot be emulated.

(6) Because the DDI, DCK, and DMS pins function alternately as the I/O pins of timer ENC11 (TIUD11, TO11, TCUD11, TCLR11), timer ENC11 cannot be used while the on-chip debug function is being used.

(7) When the on-chip debug function is used, the clock generator and PLL continue operating even if the STOP mode is set.

CHAPTER 22 FLASH MEMORY

The μPD70F3184 (V850E/IA3) and μPD70F3186 (V850E/IA4) are the flash memory versions and incorporate 256 KB of flash memory.

Caution There are differences in the amount of noise tolerance and noise radiation between flash memory versions and mask ROM versions. When considering changing from a flash memory version to a mask ROM version during the process from experimental manufacturing to mass production, make sure to sufficiently evaluate commercial samples (CS) (not engineering samples (ES)) of the mask ROM versions.

Flash memory versions are commonly used in the following development environments and mass production applications.

- For altering software after the V850E/IA3 and V850E/IA4 is soldered onto the target system.
- For data adjustment when starting mass production.
- For differentiating software according to the specification in small scale production of various models.
- For facilitating inventory management.
- For updating software after shipment.

22.1 Features

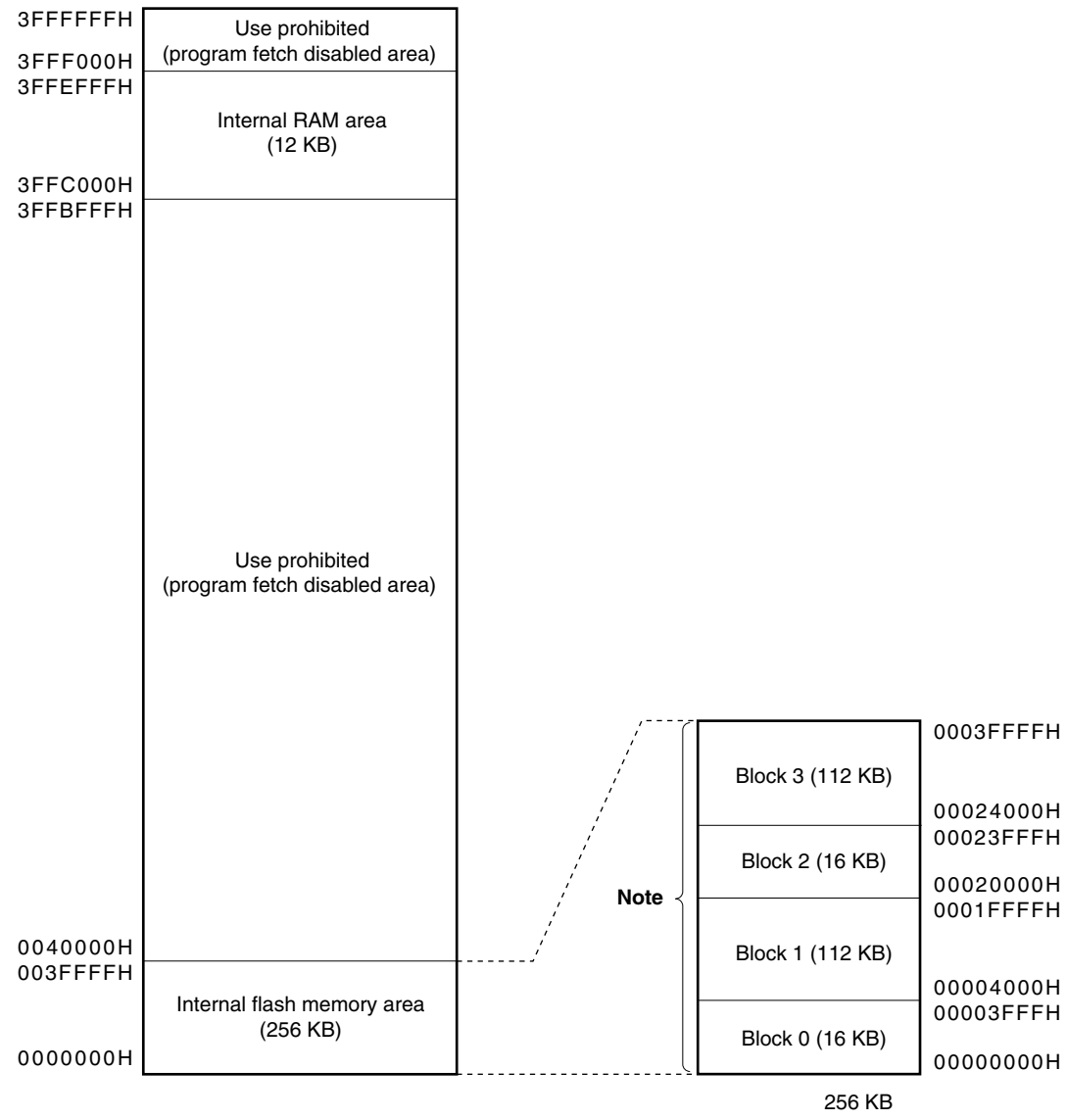
- 4-byte/1-clock access (when instruction is fetched)
- Capacity: 256 KB
- Write voltage: Erase/write with a single power supply
- Rewriting method
  - Rewriting by communication with dedicated flash memory programmer via serial interface (on-board/off-board programming)
  - Rewriting flash memory by user program (self programming)
- Flash memory write prohibit function supported (security function)
- Safe rewriting of entire flash memory area by self programming using boot swap function
- Interrupts can be acknowledged during self programming.

22.2 Memory Configuration

The 256 KB internal flash memory area is divided into 4 blocks and can be programmed/erased in block units. All the blocks can also be erased at once.

When the boot swap function is used, the physical memory located at the addresses of blocks 0 and 1 is replaced by the physical memory located at the addresses of blocks 2 and 3. For details of the boot swap function, see **22.5 Rewriting by Self Programming**.

Figure 22-1. Flash Memory Mapping



Note Blocks 0 and 1: Boot area
Blocks 2 and 3: Area used to replace boot area via boot swap function

22.3 Functional Overview

The internal flash memory of the V850E/IA3 and V850E/IA4 can be rewritten by using the rewrite function of the dedicated flash memory programmer, regardless of whether the V850E/IA3 and V850E/IA4 have already been mounted on the target system or not (on-board/off-board programming).

In addition, a security function that prohibits rewriting the user program written to the internal flash memory is also supported, so that the program cannot be changed by an unauthorized person.

The rewrite function using the user program (self programming) is ideal for an application where it is assumed that the program is changed after production/shipment of the target system. A boot swap function that rewrites the entire flash memory area safely is also supported. In addition, interrupt servicing is supported during self programming, so that the flash memory can be rewritten under various conditions, such as while communicating with an external device.

Table 22-1. Rewrite Method

Rewrite Method	Functional Outline	Operation Mode
On-board programming	Flash memory can be rewritten after the device is mounted on the target system, by using a dedicated flash memory programmer.	Flash memory programming mode
Off-board programming	Flash memory can be rewritten before the device is mounted on the target system, by using a dedicated flash memory programmer and a dedicated program adapter board (FA series).	
Self programming	Flash memory can be rewritten by executing a user program that has been written to the flash memory in advance by means of on-board/off-board programming. (During self-programming, instructions cannot be fetched from or data access cannot be made to the internal flash memory area. Therefore, the rewrite program must be transferred to the internal RAM or external memory in advance).	Normal operation mode

Remark The FA series is a product of Naito Densei Machida Mfg. Co., Ltd.

Table 22-2. Basic Functions

Function	Functional Outline	Support (√: Supported, ×: Not supported)	
		On-Board/Off-Board Programming	Self Programming
Block erasure	The contents of specified memory blocks are erased.	√	√
Chip erasure	The contents of the entire memory area are erased all at once.	√	×
Write	Writing to specified addresses, and a verify check to see if write level is secured are performed.	√	√
Verify/checksum	Data read from the flash memory is compared with data transferred from the flash memory programmer.	√	× (Can be read by user program)
Blank check	The erasure status of the entire memory is checked.	√	√
<R> Security setting	Use of the block erase command, chip erase command, program command, and read command can be prohibited, and rewriting boot area can be prohibited.	√	× (Supported only when setting is changed from enable to prohibit)

The following table lists the security functions. The block erase command prohibit, chip erase command prohibit, and program command prohibit functions are enabled by default after shipment, and security can be set by rewriting via on-board/off-board programming. Each security function can be used in combination with the others at the same time.

<R> **Table 22-3. Security Functions**

Function	Function Outline
Block erase command prohibit	Execution of a block erase command on all blocks is prohibited. Setting of prohibition can be initialized by execution of a chip erase command.
Chip erase command prohibit	Execution of block erase and chip erase commands on all the blocks is prohibited. Once prohibition is set, setting of prohibition cannot be initialized because the chip erase command cannot be executed.
Program command prohibit	Execution of program and block erase commands on all the blocks are prohibited. Setting of prohibition can be initialized by execution of the chip erase command.
Read command prohibit	Execution of read command on all the blocks is prohibited. Setting of prohibition can be initialized by execution of the chip erase command.
Boot area rewrite prohibit	Boot areas from block 0 to the specified last block can be protected. The protected boot area cannot be rewritten (erased and written). Setting of prohibition cannot be initialized by execution of the chip erase command.

<R> **Table 22-4. Security Setting**

Function	Erase, Write, Read Operations When Each Security Is Set (√: Executable, ×: Not Executable, –: Not Supported)		Notes on Security Setting	
	On-Board/ Off-Board Programming	Self Programming	On-Board/ Off-Board Programming	Self Programming
Block erase command prohibit	Block erase command: × Chip erase command: √ Program command: √ Read command: √	Block erasure (FlashBlockErase): √ Chip erasure: – Write (FlashWordWrite): √ Read (FlashWordRead): √	Setting of prohibition can be initialized by chip erase command.	Supported only when setting is changed from enable to prohibit
Chip erase command prohibit	Block erase command: × Chip erase command: × Program command: √Note 1 Read command: √	Block erasure (FlashBlockErase): √ Chip erasure: – Write (FlashWordWrite): √ Read (FlashWordRead): √	Setting of prohibition cannot be initialized.	
Program command prohibit	Block erase command: × Chip erase command: √ Program command: × Read command: √	Block erasure (FlashBlockErase): √ Chip erasure: – Write (FlashWordWrite): √ Read (FlashWordRead): √	Setting of prohibition can be initialized by chip erase command.	
Read command prohibit	Block erase command: √ Chip erase command: √ Program command: √ Read command: ×	Block erasure (FlashBlockErase): √ Chip erasure: – Write (FlashWordWrite): √ Read (FlashWordRead): √	Setting of prohibition can be initialized by chip erase command.	
Boot area rewrite prohibit	Block erase command: √Note 2 Chip erase command: × Program command: √Note 2 Read command: √	Block erasure (FlashBlockErase): √ Chip erasure: – Write (FlashWordWrite): √ Read (FlashWordRead): √	Setting of prohibition cannot be initialized.	

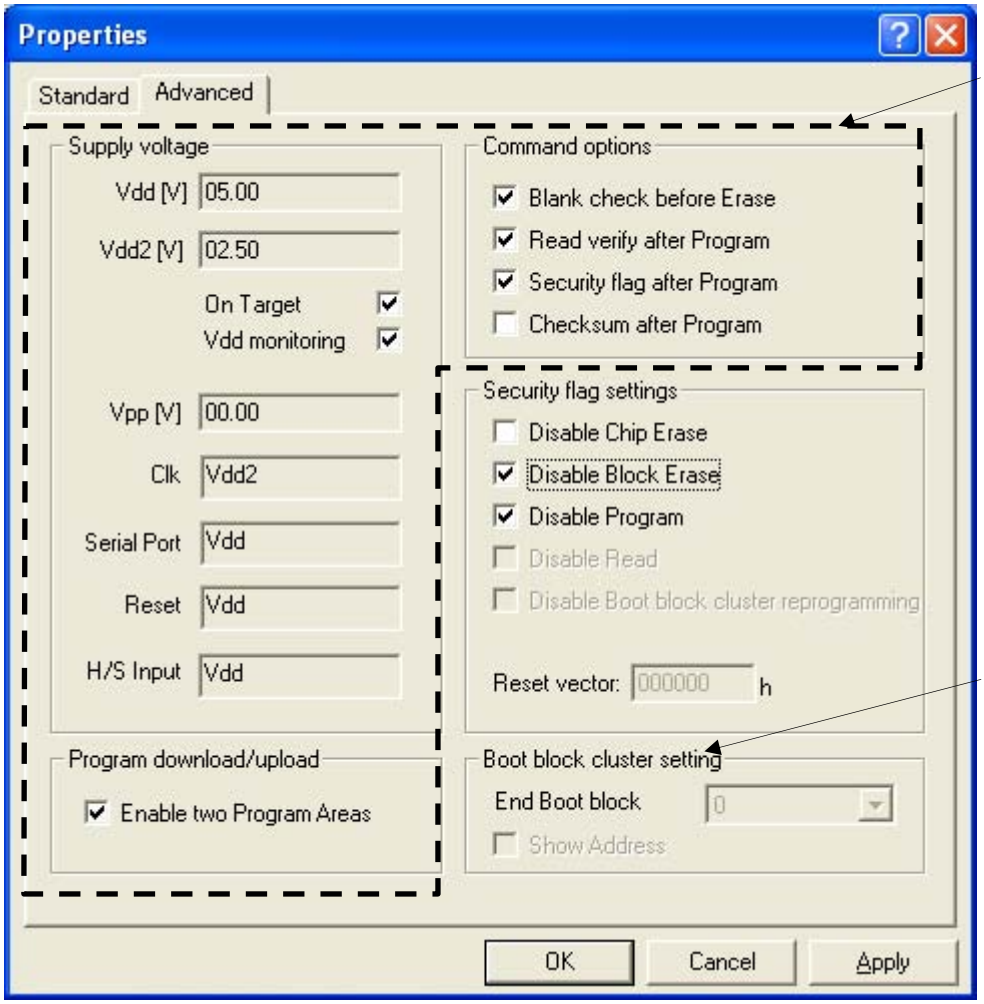
Notes 1. In this case, since the erase command is invalid, data different from the data already written in the flash memory cannot be written.

2. The boot area for which rewriting is prohibited is invalid.

<R> **(1) Security setting by PG-FP4 and PG-FP5 (Security flag settings)**

When disabling the read command (Disable Read), to raise the security level, it is recommended to also disable the block erase command (Disable Block Erase) and program command (Disable Program).

Furthermore, when rewriting program is not necessary similarly to the mask ROM versions, additionally disable the chip erase command (Disable Chip Erase).



Notes 1. Set “Supply voltage”, “Program download/upload”, and “Command options” in broken lines in accordance with the use conditions.

2. To disable rewriting the boot area (Boot block cluster setting), select “Disable Boot block cluster reprogramming” in “Security flag settings” and select the last block of the boot area for which rewriting is to be disabled.

22.4 Rewriting by Dedicated Flash Memory Programmer

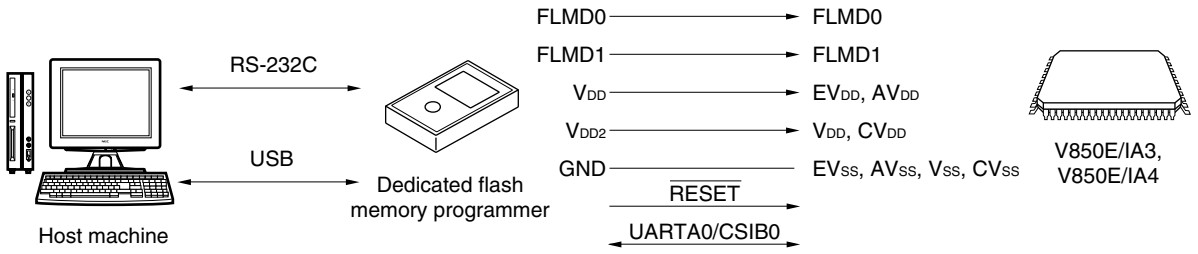
The flash memory can be rewritten by using a dedicated flash memory programmer after the V850E/IA3 and V850E/IA4 are mounted on the target system (on-board programming). The flash memory can also be rewritten before the device is mounted on the target system (off-board programming) by using a dedicated program adapter (FA series).

22.4.1 Programming environment

The following shows the environment required for writing programs to the flash memory of the V850E/IA3 and V850E/IA4.

Figure 22-2. Environment Required for Writing Programs to Flash Memory

<R>



Caution Connect the PLLSIN pin as follows in accordance with the range of the PLL output clock frequency (f_{XX}).

- **f_{XX} = 32 to 55 MHz: Connect to EV_{SS}.**
- **f_{XX} = 55 to 64 MHz: Connect to EV_{DD}.**

A host machine is required for controlling the dedicated flash memory programmer.

UARTA0 or CSIB0 is used for the interface between the dedicated flash memory programmer and the V850E/IA3, V850E/IA4 to perform writing, erasing, etc. A dedicated program adapter (FA series) required for off-board writing.

Remark The FA series is a product of Naito Densei Machida Mfg. Co., Ltd.

22.4.2 Communication mode

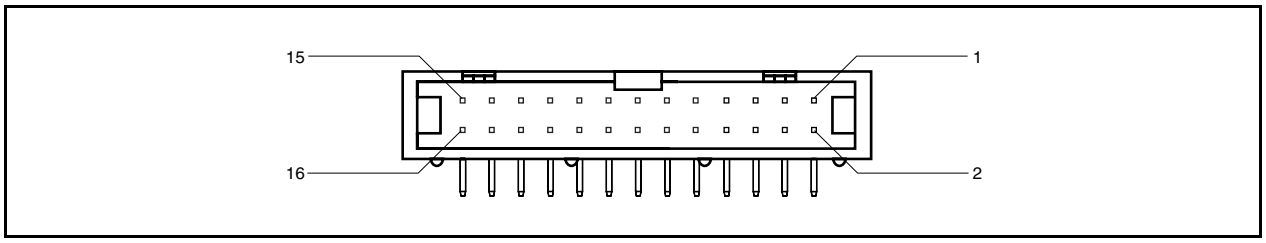
Communication between the dedicated flash memory programmer and the V850E/IA3 and V850E/IA4 is performed via serial communication using UARTA0 or CSIB0.

Remark The recommended target connector is as follows.

- 7616-5002SC (Sumitomo 3M Ltd.)

The following figure outlines the connector (when viewed from the connector insertion side).

Figure 22-3. Target Connector Outline (Viewed from Connector Insertion Side)



(1) UARTA0

Transfer rate: 9,600 to 153,600 bps

Table 22-5. Wiring Correspondence Between Dedicated Flash Memory Programmer and V850E/IA3, V850E/IA4

<R>

Pin No.	Dedicated Flash Memory Programmer (PG-FP4 and PG-FP5)	I/O (PG-FP4 and PG-FP5 Side)	V850E/IA3, V850E/IA4			
			Pin Name	Pin No.		
				V850E/IA3	V850E/IA4	
				GC	GC	GF
1	GND	–	EV_{SS}, AV_{SS}, V_{SS}, CV_{SS}	8, 10, 26, 31, 35, 39, 52, 67, 73	10, 12, 31, 39, 46, 51, 65, 72, 85, 91	13, 19, 38, 40, 59, 67, 74, 79, 93, 100
2	$\overline{RESET}$	Output	$\overline{RESET}$	36	47	75
3	SI/RxD	Input	TXDA0	42	53	81
4	V_{DD}	–	EV_{DD}, AV_{DD}	7, 11, 30, 74	9, 13, 38, 73, 92	1, 20, 37, 41, 66
5	SO/TxD	Output	RXDA0	41	52	80
6	V_{PP}	×	NC	–	–	–
7	SCK	×	NC	–	–	–
8	H/S	×	NC	–	–	–
9	CLK[Note 1]	Output	X1[Note 1]	34	45	73
10	VDE	×	NC	–	–	–
11	V_{DD2}	–	V_{DD}, CV_{DD}	25, 32, 40, 51, 68	30, 43, 50, 64, 86	14, 58, 71, 78, 92
12	FLMD1	Output	**Note 2**	56	69	97
13	RFU-1	×	NC	–	–	–
14	FLMD0	Output	FLMD0	69	87	15
15	Not used	×	NC	–	–	–
16	Not used	×	NC	–	–	–

Notes 1. In the V850E/IA3 and V850E/IA4, external clock input is prohibited. Mount the resonator on board.

2. Connect to FLMD1 or GND via a resistor.

Caution Connect the PLLSIN pin as follows in accordance with the range of the PLL output clock frequency (f_{XX}).

- **f_{XX} = 32 to 55 MHz: Connect to EV_{SS}.**
- **f_{XX} = 55 to 64 MHz: Connect to EV_{DD}.**

Remark NC: No Connection

GC (V850E/IA3): 80-pin plastic QFP (14 × 14)

GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)

GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(2) CSIB0

Transfer rate: 2.4 kHz to 2,500 kHz (MSB first)

Table 22-6. Wiring Correspondence Between Dedicated Flash Memory Programmer and V850E/IA3, V850E/IA4

<R>

Pin No.	Dedicated Flash Memory Programmer (PG-FP4 and PG-FP5)	I/O (PG-FP4 and PG-FP5 Side)	V850E/IA3, V850E/IA4			
			Pin Name	Pin No.		
				V850E/IA3	V850E/IA4	
				GC	GC	GF
1	GND	–	EV_{SS}, AV_{SS}, V_{SS}, CV_{SS}	8, 10, 26, 31, 35, 39, 52, 67, 73	10, 12, 31, 39, 46, 51, 65, 72, 85, 91	13, 19, 38, 40, 59, 67, 74, 79, 93, 100
2	$\overline{RESET}$	Output	$\overline{RESET}$	36	47	75
3	SI/RxD	Input	SOB0	28	33	61
4	V_{DD}	–	EV_{DD}, AV_{DD}	7, 11, 30, 74	9, 13, 38, 73, 92	1, 20, 37, 41, 66
5	SO/TxD	Output	SIB0	27	32	60
6	V_{PP}	×	NC	–	–	–
7	SCK	Output	$\overline{SCKB0}$	29	34	62
8	H/S	×	NC	–	–	–
9	CLK[Note 1]	Output	X1[Note 1]	34	45	73
10	VDE	×	NC	–	–	–
11	V_{DD2}	–	V_{DD}, CV_{DD}	25, 32, 40, 51, 68	30, 43, 50, 64, 86	14, 58, 71, 78, 92
12	FLMD1	Output	**Note 2**	56	69	97
13	RFU-1	×	NC	–	–	–
14	FLMD0	Output	FLMD0	69	87	15
15	Not used	×	NC	–	–	–
16	Not used	×	NC	–	–	–

Notes 1. In the V850E/IA3 and V850E/IA4, external clock input is prohibited. Mount the resonator on board.

2. Connect to FLMD1 or GND via a resistor.

Caution Connect the PLLSIN pin as follows in accordance with the range of the PLL output clock frequency (f_{XX}).

- **f_{XX} = 32 to 55 MHz: Connect to EV_{SS}.**
- **f_{XX} = 55 to 64 MHz: Connect to EV_{DD}.**

Remark NC: No Connection

GC (V850E/IA3): 80-pin plastic QFP (14 × 14)

GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)

GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

(3) CSIB0 + HS

Transfer rate: 2.4 kHz to 2,500 kHz (MSB first)

Table 22-7. Wiring Correspondence Between Dedicated Flash Memory Programmer and V850E/IA3, V850E/IA4

Pin No.	Dedicated Flash Memory Programmer (PG-FP4 and PG-FP5)	I/O (PG-FP4 and PG-FP5 Side)	V850E/IA3, V850E/IA4			
			Pin Name	Pin No.		
				V850E/IA3	V850E/IA4	
				GC	GC	GF
1	GND	–	EV_{SS}, AV_{SS}, V_{SS}, CV_{SS}	8, 10, 26, 31, 35, 39, 52, 67, 73	10, 12, 31, 39, 46, 51, 65, 72, 85, 91	13, 19, 38, 40, 59, 67, 74, 79, 93, 100
2	$\overline{\text{RESET}}$	Output	$\overline{\text{RESET}}$	36	47	75
3	SI/RxD	Input	SOB0	28	33	61
4	V_{DD}	–	EV_{DD}, AV_{DD}	7, 11, 30, 74	9, 13, 38, 73, 92	1, 20, 37, 41, 66
5	SO/TxD	Output	SIB0	27	32	60
6	V_{PP}	×	NC	–	–	–
7	SCK	Output	$\overline{\text{SCKB0}}$	29	34	62
8	H/S	Input	P43	37	48	76
9	CLK[Note 1]	Output	X1[Note 1]	34	45	73
10	VDE	×	NC	–	–	–
11	V_{DD2}	–	V_{DD}, CV_{DD}	25, 32, 40, 51, 68	30, 43, 50, 64, 86	14, 58, 71, 78, 92
12	FLMD1	Output	**Note 2**	56	69	97
13	RFU-1	×	NC	–	–	–
14	FLMD0	Output	FLMD0	69	87	15
15	Not used	×	NC	–	–	–
16	Not used	×	NC	–	–	–

Notes 1. In the V850E/IA3 and V850E/IA4, external clock input is prohibited. Mount the resonator on board.

2. Connect to FLMD1 or GND via a resistor.

Caution Connect the PLLSIN pin as follows in accordance with the range of the PLL output clock frequency (f_{XX}).

- **f_{XX} = 32 to 55 MHz: Connect to EV_{SS}.**
- **f_{XX} = 55 to 64 MHz: Connect to EV_{DD}.**

Remark NC: No Connection
GC (V850E/IA3): 80-pin plastic QFP (14 × 14)
GC (V850E/IA4): 100-pin plastic LQFP (fine pitch) (14 × 14)
GF (V850E/IA4): 100-pin plastic QFP (14 × 20)

The dedicated flash memory programmer outputs the transfer clock, and the V850E/IA3 and V850E/IA4 operate as slaves.

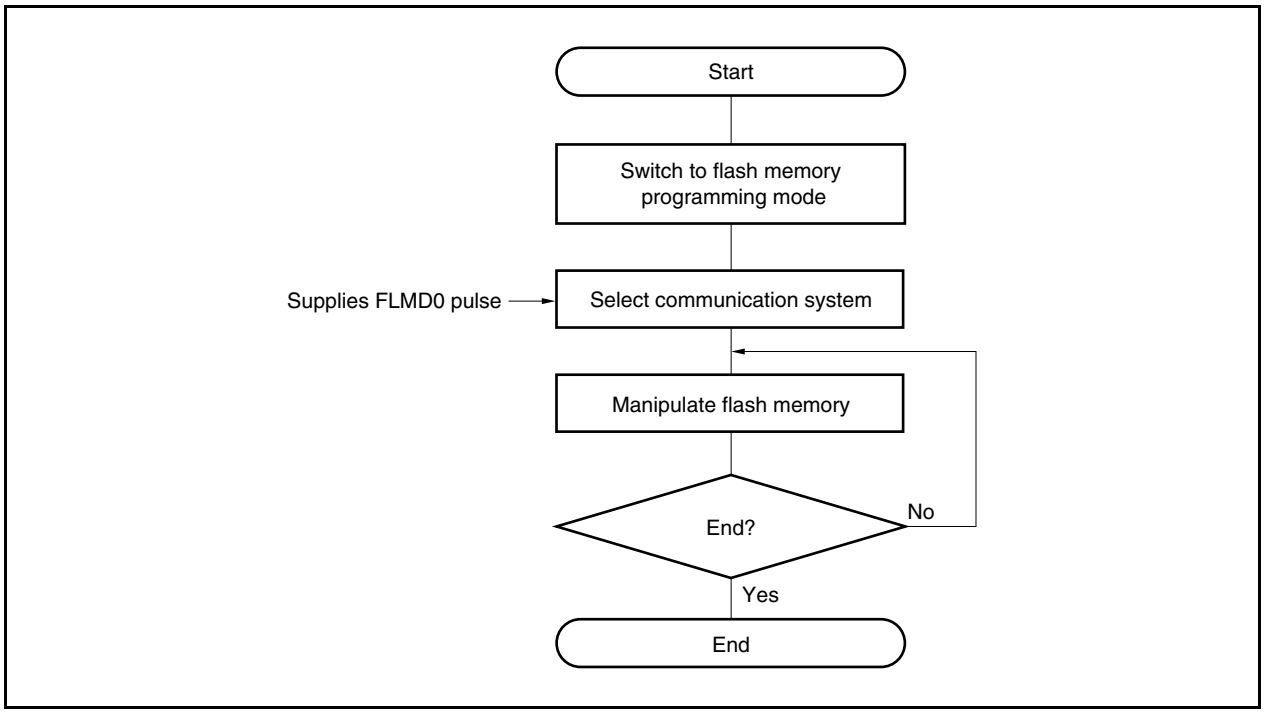
When the PG-FP4 or PG-FP5 is used as the dedicated flash memory programmer, it sends the following signals to the V850E/IA3 and V850E/IA4. For details, refer to the **PG-FP4 User's Manual (U15260E)** and **PG-FP5 User's Manual (U18865E)**.

22.4.3 Flash memory control

The following shows the procedure for manipulating the flash memory.

<R>

Figure 22-4. Procedure for Manipulating Flash Memory

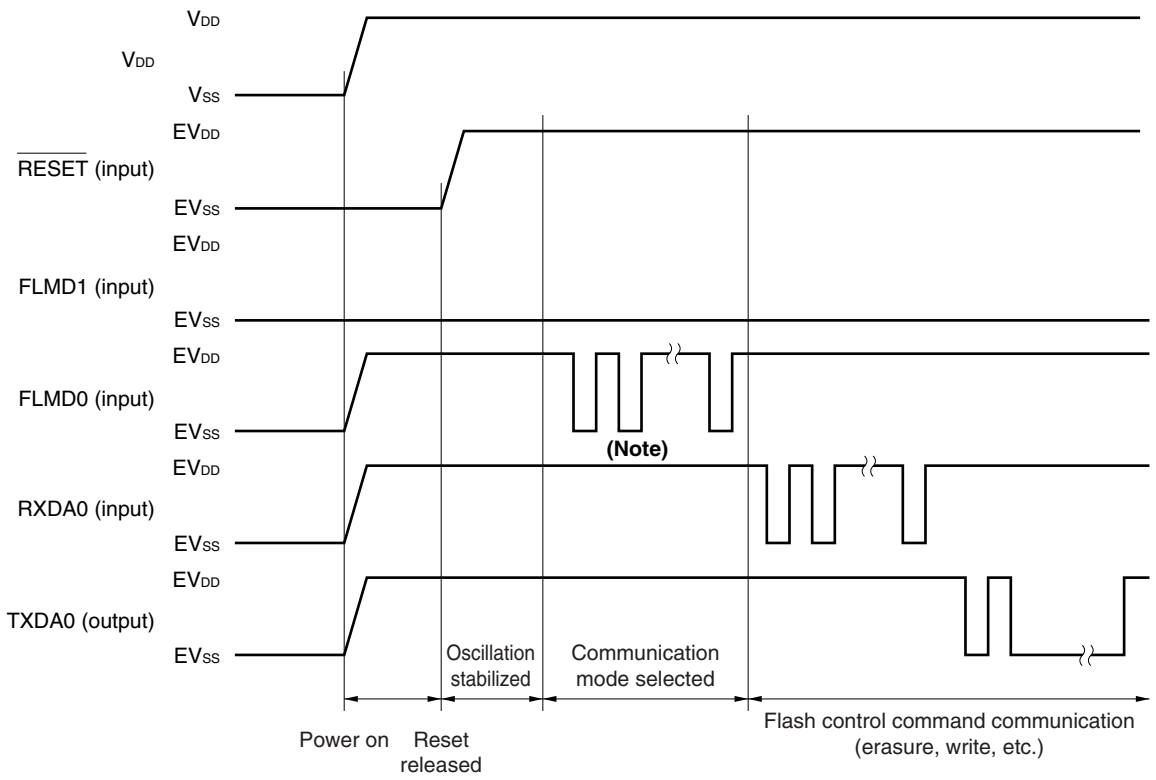


22.4.4 Selection of communication mode

In the V850E/IA3 and V850E/IA4, the communication mode is selected by inputting pulses (11 pulses max.) to the FLMD0 pin after switching to the flash memory programming mode. The FLMD0 pulse is generated by the dedicated flash memory programmer.

The following shows the relationship between the number of pulses and the communication mode.

Figure 22-5. Selection of Communication Mode



Note The number of clocks is as follows depending on the communication mode.

FLMD0 Pulse	Communication Mode	Remarks
0	UARTA0	Communication rate: 9,600 bps (after reset), LSB first
8	CSIB0	V850E/IA3 and V850E/IA4 perform slave operation, MSB first
11	CSIB0 + HS	V850E/IA3 and V850E/IA4 perform slave operation, MSB first
Other	RFU	Setting prohibited

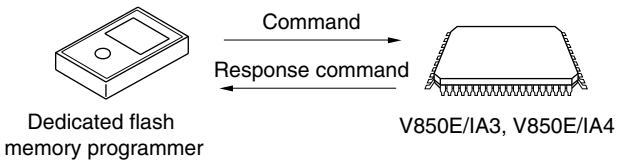
Caution When UARTA0 is selected, the receive clock is calculated based on the reset command sent from the dedicated flash memory programmer after receiving the FLMD0 pulse.

22.4.5 Communication commands

The V850E/IA3 and V850E/IA4 communicates with the dedicated flash memory programmer by means of commands. The signals sent from the dedicated flash memory programmer to the V850E/IA3 and V850E/IA4 are called “commands”. The response signals sent from the V850E/IA3 and V850E/IA4 to the dedicated flash memory programmer are called “response commands”.

Figure 22-6. Communication Commands

<R>



The following shows the commands for flash memory control in the V850E/IA3 and V850E/IA4. All of these commands are issued from the dedicated flash memory programmer, and the V850E/IA3 and V850E/IA4 perform the processing corresponding to the commands.

Table 22-8. Flash Memory Control Commands

Classification	Command Name	Support			Function
		CSIB0	CSIB0 + HS	UARTA0	
Blank check	Block blank check command	√	√	√	Checks if the contents of the memory in the specified block have been correctly erased.
Erase	Chip erase command	√	√	√	Erases the contents of the entire memory.
	Block erase command	√	√	√	Erases the contents of the memory of the specified block.
<R> Write	Program command	√	√	√	Writes the specified address range, and executes a contents verify check.
Verify	Verify command	√	√	√	Compares the contents of memory in the specified address range with data transferred from the flash memory programmer.
	Checksum command	√	√	√	Reads the checksum in the specified address range.
System setting, control	Silicon signature command	√	√	√	Reads silicon signature information.
<R>	Security setting command	√	√	√	Disables the chip erase command, block erase command, program command, read command, and boot area rewrite.

22.4.6 Pin connection

When performing on-board writing, mount a connector recommended target connector: 7616-5002SC (SUMITOMO 3M Limited) on the target system to connect to the dedicated flash memory programmer. Also, incorporate a function on-board to switch from the normal operation mode (single-chip mode) to the flash memory programming mode.

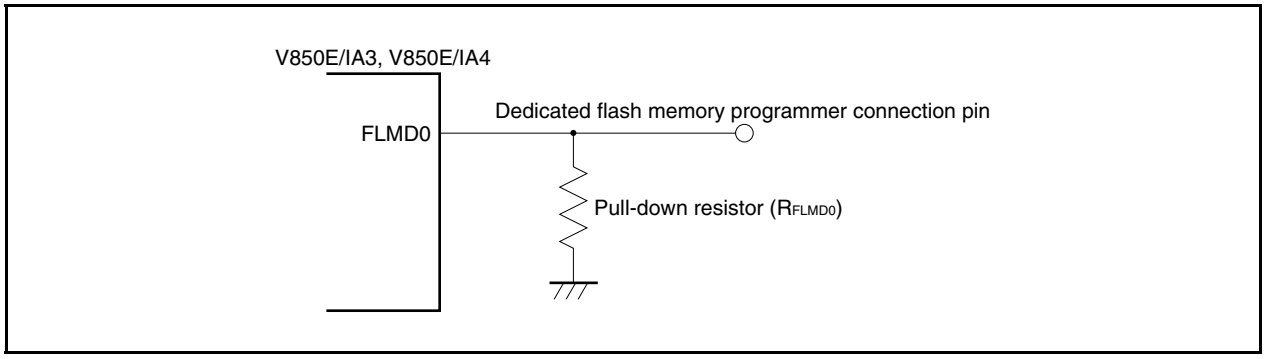
In the flash memory programming mode, all the pins not used for flash memory programming become the same status as that immediately after reset. Therefore, pin handling is required when the external device does not acknowledge the status immediately after a reset.

(1) FLMD0 pin

In the normal operation mode, input a voltage of EV_{SS} level to the FLMD0 pin. In the flash memory programming mode, supply a write voltage of EV_{DD} level to the FLMD0 pin.

Because the FLMD0 pin serves as a write protection pin in the self programming mode, a voltage of EV_{DD} level must be supplied to the FLMD0 pin via port control, etc., before writing to the flash memory. For details, see **22.5.5 (1) FLMD0 pin**.

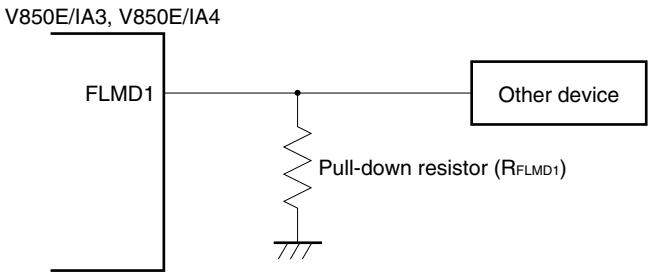
Figure 22-7. FLMD0 Pin Connection Example



(2) FLMD1 pin

When 0 V is input to the FLMD0 pin, the FLMD1 pin does not function. When EV_{DD} is supplied to the FLMD0 pin, the flash memory programming mode is entered, so 0 V must be input to the FLMD1 pin. The following shows an example of the connection of the FLMD1 pin.

Figure 22-8. FLMD1 Pin Connection Example



Caution If the EV_{DD} signal is input to the FLMD1 pin from another device during on-board writing and immediately after reset, isolate this signal.

Table 22-9. Relationship Between FLMD0 and FLMD1 Pins and Operation Mode When Reset Is Released

FLMD0	FLMD1	Operation Mode
0	Don't care	Normal operation mode
EV_{DD}	0	Flash memory programming mode
EV_{DD}	EV_{DD}	Setting prohibited

(3) Serial interface pin

The following shows the pins used by each serial interface.

Table 22-10. Pins Used by Serial Interfaces

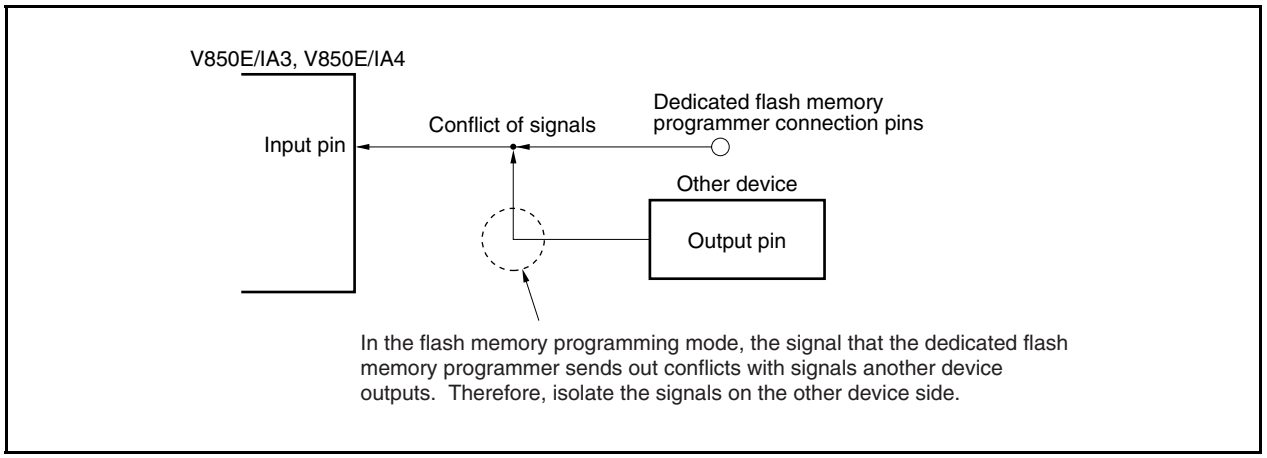
Serial Interface	Pins Used
UARTA0	TXDA0, RXDA0
CSIB0	SOB0, SIB0, $\overline{\text{SCKB0}}$
CSIB0 + HS	SOB0, SIB0, $\overline{\text{SCKB0}}$, P43

When connecting a dedicated flash memory programmer to a serial interface pin that is connected to another device on-board, care should be taken to avoid conflict of signals and malfunction of the other device.

(a) Conflict of signals

When the dedicated flash memory programmer (output) is connected to a serial interface pin (input) that is connected to another device (output), a conflict of signals occurs. To avoid the conflict of signals, isolate the connection to the other device or set the other device to the output high-impedance status.

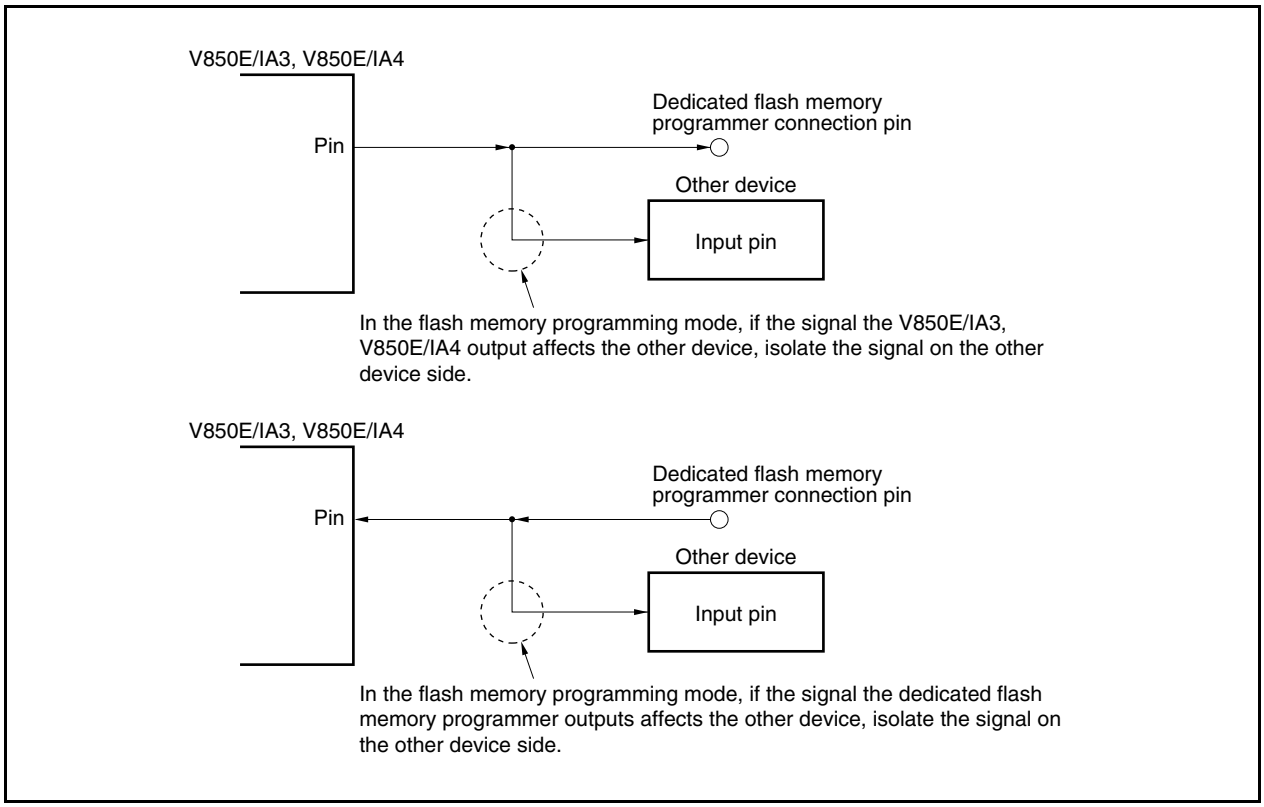
Figure 22-9. Conflict of Signals (Serial Interface Input Pin)



(b) Malfunction of other device

When the dedicated flash memory programmer (output or input) is connected to a serial interface pin (input or output) that is connected to another device (input), the signal is output to the other device, causing the device to malfunction. To avoid this, isolate the connection to the other device.

Figure 22-10. Malfunction of Other Device

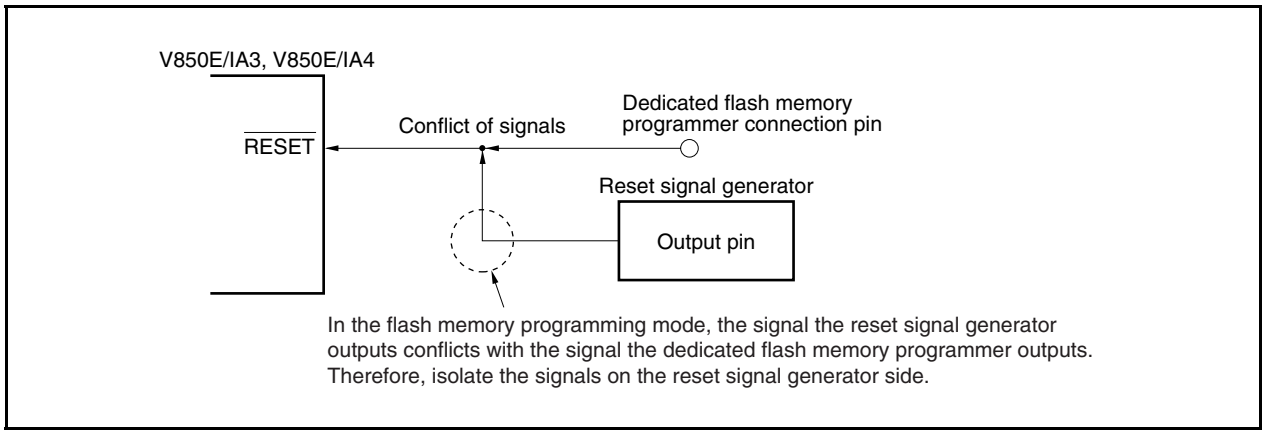


(4) $\overline{\text{RESET}}$ pin

When the reset signals of the dedicated flash memory programmer are connected to the $\overline{\text{RESET}}$ pin that is connected to the reset signal generator on-board, a conflict of signals occurs. To avoid the conflict of signals, isolate the connection to the reset signal generator.

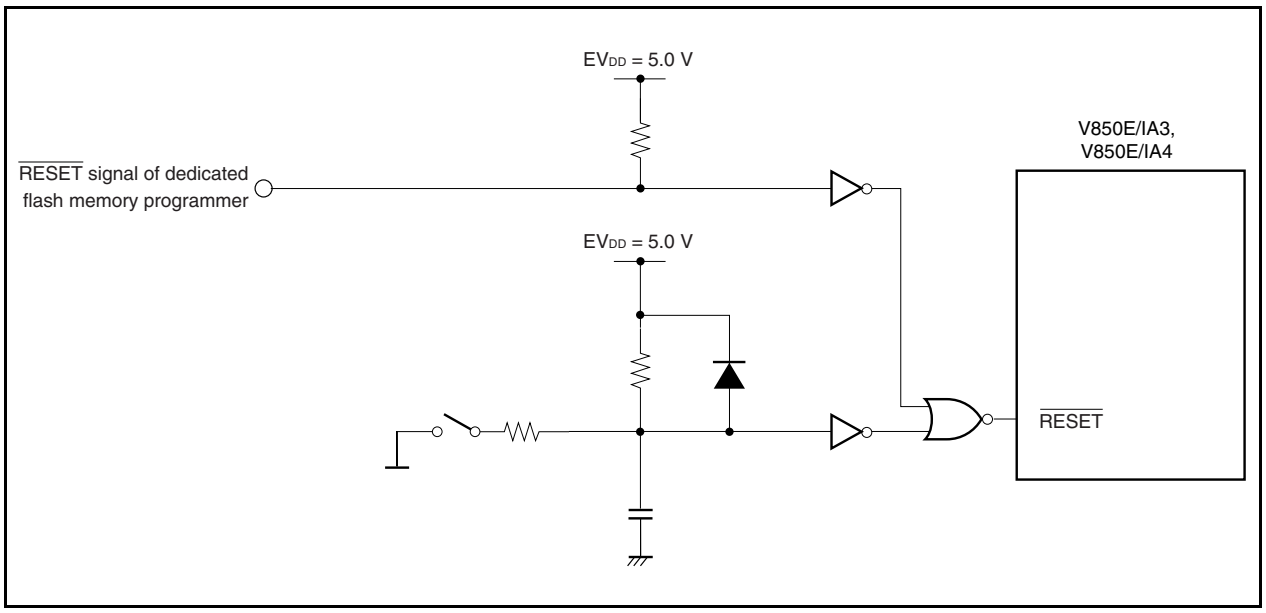
When a reset signal is input from the user system in the flash memory programming mode, the programming operation will not be performed correctly. Therefore, do not input signals other than the reset signals from the dedicated flash memory programmer.

Figure 22-11. Conflict of Signals ($\overline{\text{RESET}}$ Pin)



Connect the reset pin of the dedicated flash memory programmer to the $\overline{\text{RESET}}$ pin of the V850E/IA3 and V850E/IA4 at the location where the two reset signals are the same.

<R>



(5) Port pins

When the system shifts to the flash memory programming mode, all the pins that are not used for flash memory programming are in the same status as that immediately after reset. If the external device connected to each port does not recognize the status of the port immediately after reset, pins require appropriate processing, such as connecting to EV_{DD} via a resistor or connecting to EV_{SS} via a resistor.

(6) PLLSIN pin

Connect the PLLSIN pin as follows in accordance with the range of the PLL output clock frequency (f_{XX}).

- f_{XX} = 32 to 55 MHz: Connect to EV_{SS}.
- f_{XX} = 55 to 64 MHz: Connect to EV_{DD}.

(7) Other signal pins

Connect X1 and X2 in the same status as in the normal operation mode.

(8) Power supply

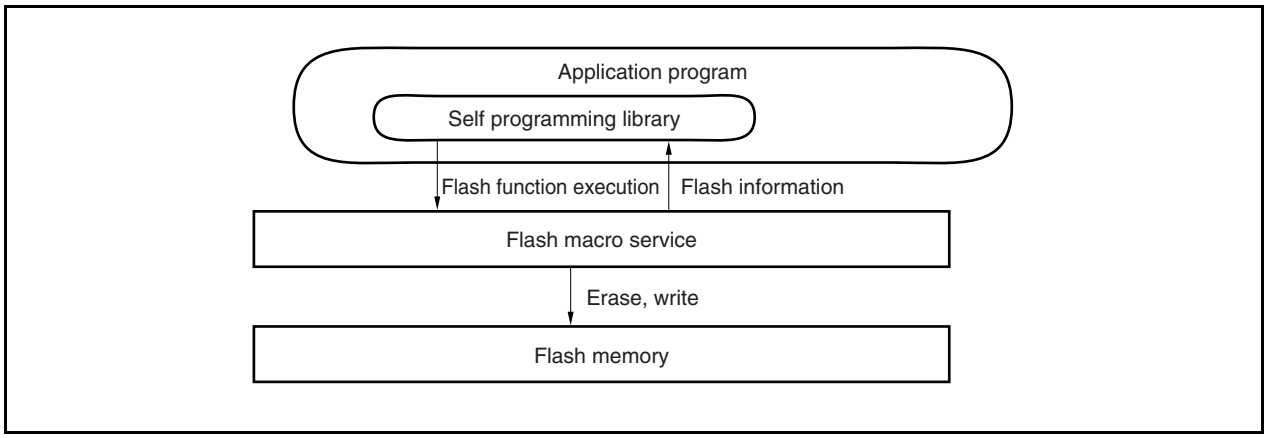
Supply the same power (V_{DD}, V_{SS}, AV_{DD}, AV_{SS}, CV_{DD}, CV_{SS}, EV_{DD}, and EV_{SS}) as in normal operation mode. Connect V_{DD} of the dedicated flash memory programmer to EV_{DD} and AV_{DD}, GND of the dedicated flash memory programmer to EV_{SS}, AV_{SS}, V_{SS}, and CV_{SS}, and V_{DD2} of the dedicated flash memory programmer to V_{DD} and CV_{DD}.

22.5 Rewriting by Self Programming

22.5.1 Overview

The V850E/IA3 and V850E/IA4 support a flash macro service that allows the user program to rewrite the internal flash memory by itself. By using this interface and a self programming library that is used to rewrite the flash memory with a user application program, the flash memory can be rewritten by a user application transferred in advance to the internal RAM or external memory. Consequently, the user program can be upgraded and constant data can be rewritten in the field.

Figure 22-12. Concept of Self Programming

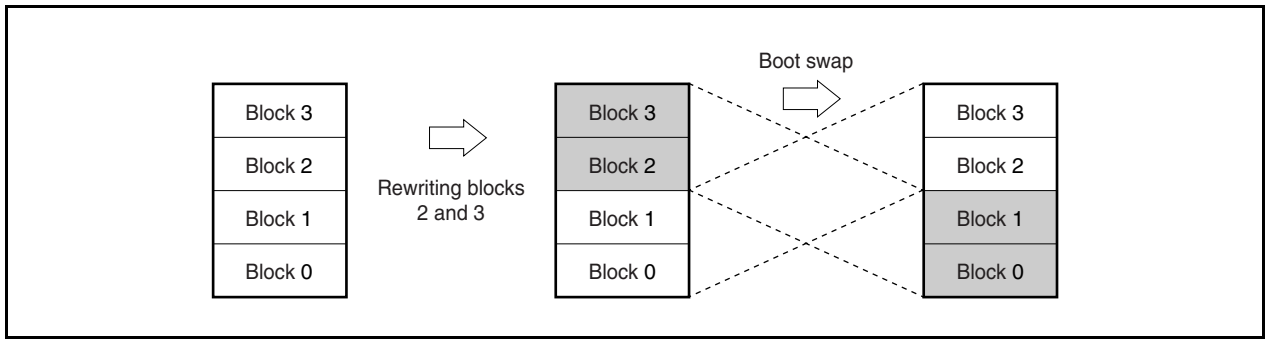


22.5.2 Features

(1) Secure self programming (boot swap function)

The V850E/IA3 and V850E/IA4 support a boot swap function that can exchange the physical memory of blocks 0 and 1 with the physical memory of blocks 2 and 3. By writing the start program to be rewritten to blocks 2 and 3 in advance and then swapping the physical memory, the entire area can be safely rewritten even if a power failure occurs during rewriting because the correct user program always exists in blocks 0 and 1.

Figure 22-13. Rewriting Entire Memory Area (Boot Swap)



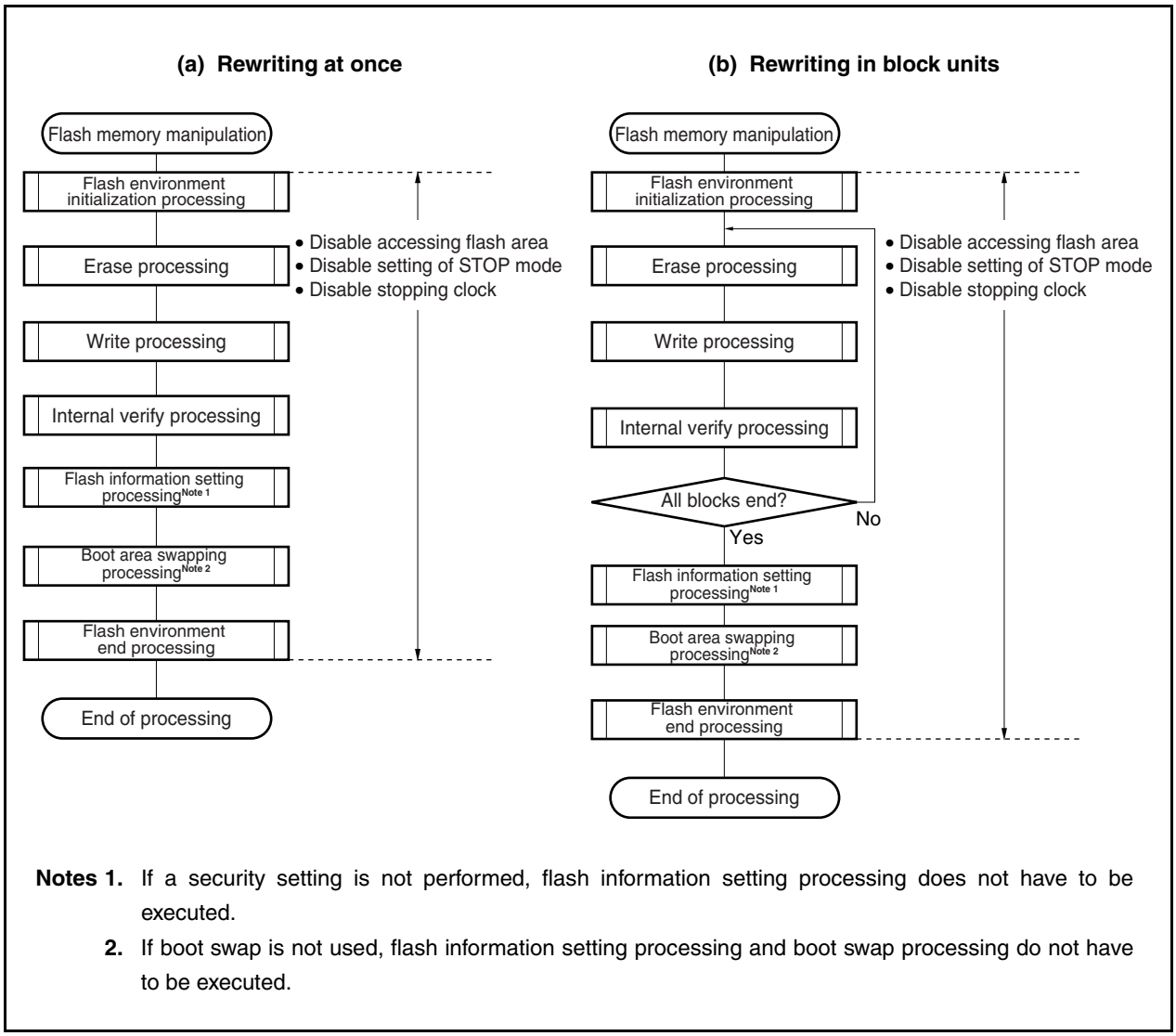
(2) Interrupt support

Instructions cannot be fetched from the flash memory during self programming. Conventionally, therefore, a user handler written to the flash memory could not be used even if an interrupt occurred. With the V850E/IA3 and V850E/IA4, a user handler can be registered to an entry RAM area by using a library function, so that interrupt servicing can be performed by internal RAM or external memory execution.

22.5.3 Standard self programming flow

The entire processing to rewrite the flash memory by flash self programming is illustrated below.

<R> **Figure 22-14. Standard Self Programming Flow**



Notes 1. If a security setting is not performed, flash information setting processing does not have to be executed.

2. If boot swap is not used, flash information setting processing and boot swap processing do not have to be executed.

22.5.4 Flash functions

<R>

Table 22-11. Flash Function List

Function Name	Outline	Support
FlashEnv	Initialization of flash control macro	√
FlashBlockErase	Erasure of only specified one block	√
FlashWordWrite	Writing from specified address	√
FlashBlockIVerify	Internal verification of specified block	√
FlashBlockBlankCheck	Blank check of specified block	√
FlashFLMDCheck	Check of FLMD pin	√
FlashGetInfo	Reading of flash information	√
FlashSetInfo	Setting of flash information	√
FlashBootSwap	Swapping of boot area	√
FlashWordRead	Reading data from specified address	√

Remark For details, refer to the **V850 Series Flash Memory Self Programming (Single Power Supply Flash Memory) User's Manual**.
Contact an NEC Electronics sales representative for the above manual.

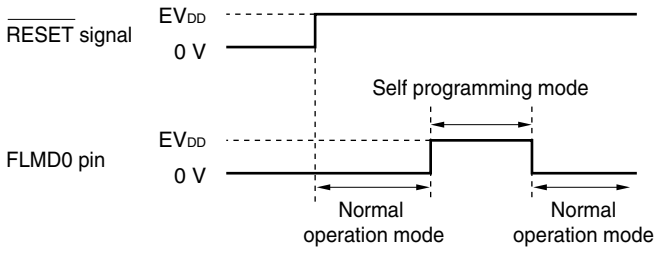
22.5.5 Pin processing

(1) FLMD0 pin

The FLMD0 pin is used to set the operation mode when reset is released and to protect the flash memory from being written during self rewriting. It is therefore necessary to keep the voltage applied to the FLMD0 pin at 0 V when reset is released and a normal operation is executed. It is also necessary to apply a voltage of EV_{DD} level to the FLMD0 pin during the self programming mode period via port control before the memory is rewritten.

When self programming has been completed, the voltage on the FLMD0 pin must be returned to 0 V.

Figure 22-15. Mode Change Timing



Caution Make sure that the FLMD0 pin is at 0 V when reset is released.

22.5.6 Internal resources used

The following table lists the internal resources used for self programming. These internal resources can also be used freely for purposes other than self programming.

Table 22-12. Internal Resources Used

Resource Name	Description
Entry RAM area (internal RAM/external RAM size[Note])	Routines and parameters used for the flash macro service are located in this area. The entry program and default parameters are copied by calling a library initialization function.
Stack area (stack size[Note])	An extension of the stack used by the user is used by the library (can be used in both the internal RAM and external RAM).
Library code (code size[Note])	Program entity of library (can be used anywhere other than the flash memory block to be manipulated).
Application program	Executed as user application. Calls flash functions.
Maskable interrupt	Can be used in user application execution status or self programming status. To use this interrupt in the self programming status, the interrupt servicing start address must be registered in advance by a registration function.

Note For details, refer to the **V850 Series Flash Memory Self Programming (Single Power Supply Flash Memory) User's Manual**.
Contact an NEC Electronics sales representative for the above manual.

Absolute Maximum Ratings (T_A = 25°C)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V_{DD}	$V_{DD} = CV_{DD}$		−0.5 to +3.6	V
	V_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	EV_{DD}	$EV_{DD} = AV_{DD}$		−0.5 to +6.5	V
	EV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	CV_{DD}	$V_{DD} = CV_{DD}$		−0.5 to +3.6	V
	CV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	AV_{DD}	$EV_{DD} = AV_{DD}$		−0.5 to +6.5	V
	AV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
Input voltage	V_{I1}	**Note 1**		−0.5 to EV_{DD} + 0.5[Note 2]	V
	V_{I2}	X1, X2		−0.5 to CV_{DD} + 0.5[Note 2]	V
Output current, low	I_{OL}	P10 to P15	Per pin	18	mA
		Pins other than P10 to P15	Per pin	4	mA
		P10 to P17	Total of all pins	50	mA
		P00, P02 to P07, P30 to P37, P40 to P44, PDL0 to PDL15	Total of all pins	40	mA
Output current, high	I_{OH}	All pins	Per pin	−4.0	mA
		P10 to P17	Total of all pins	−10	mA
		P00, P02 to P07, P30 to P37, P40 to P44, PDL0 to PDL15	Total of all pins	−40	mA
Analog input voltage	V_{IAN}	P70/ANI20 to P75/ANI25, ANI00, ANI01, ANI10 to ANI13		−0.5 to AV_{DD} + 0.5[Note 2]	V
Analog input reference voltage	V_{IREF}	CMPREF		−0.5 to AV_{DD} + 0.5[Note 2]	V
Operating ambient temperature	T_A	In normal operating mode		−40 to +85	°C
		In flash memory programming mode		−40 to +85	°C
Storage temperature	T_{stg}			−40 to +125	°C

Notes 1. P00, P02 to P07, P10 to P17, P30 to P37, P40 to P44, PDL0 to PDL15, $\overline{\text{RESET}}$, IC1 (μPD703183 only)/FLMD0 (μPD70F3184 only), PLLSIN

2. Be sure not to exceed the absolute maximum ratings (MAX. value) of each supply voltage.

Cautions 1. Do not directly connect the output pins (or I/O pins in the output state) of IC products to other output pins (including I/O pins in the output state), power supply pins such as V_{DD} and EV_{DD}, or GND pin. Direct connection of the output pins between an IC product and an external circuit is possible, if the output pins can be set to the high-impedance state and the output timing of the external circuit is designed to avoid output conflict.

2. 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.

The ratings and conditions indicated for DC characteristics and AC characteristics represent the quality assurance range during normal operation.

Capacitance (T_A = 25°C, V_{DD} = V_{SS} = EV_{DD} = EV_{SS} = CV_{DD} = CV_{SS} = AV_{DD} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input capacitance	C_I	fc = 1 MHz	**Note 1**			15	pF
I/O capacitance	C_{IO}	Unmeasured pins returned to 0 V	**Note 2**			15	pF

Notes 1. ANI00, ANI01, ANI10 to ANI13, P70 to P75, PLLSIN, $\overline{\text{RESET}}$, CMPREF

2. P00, P02 to P07, P10 to P17, P30 to P37, P40 to P44, PDL0 to PDL15

Cautions 1. Excludes the FLMD0, X1, and X2 pins.

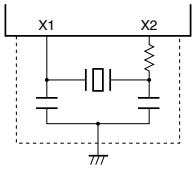
2. In addition to input capacitance, sampling capacitance is added to the ANI00, ANI01, ANI10 to ANI13, and ANI20 to ANI25 pins for sampling.

Operating Conditions (T_A = –40 to +85°C, V_{SS} = EV_{SS} = CV_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
System clock frequency	f_{XX}	PLL mode	PLLSIN = Low level	32		55	MHz
			PLLSIN = High level	55		64	MHz
		Clock through mode		4		8	MHz
CPU clock frequency	f_{CPU}	PLL mode	PLLSIN = Low level	4		55	MHz
			PLLSIN = High level	6.875		64	MHz
		Clock through mode		0.5		8	MHz
V_{DD}, CV_{DD} voltage	V_{DD}, CV_{DD}	V_{DD} = CV_{DD}		2.3		2.7	V
EV_{DD} voltage	EV_{DD}	EV_{DD} = AV_{DD}		4.0		5.5	V
AV_{DD} voltage	AV_{DD}	When A/D converters 0 to 2 are operating		4.5		5.5	V
		When A/D converters 0 to 2 are not operating		4.0		5.5	V

Clock Oscillator Characteristics

(T_A = –40 to +85°C, CV_{DD} = 2.3 to 2.7 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic /crystal resonator		Oscillation frequency (fx)		4		8	MHz
		Oscillation stabilization time	After reset release		2^{14}/fx		ms
			After STOP mode release		**Note**		ms

Note The value varies depending on the setting of the oscillation stabilization time select register (OSTS).

Cautions 1. Connect the oscillator as close to the X1 and X2 pins as possible.

2. Do not cross the wiring with the other signal lines in the area enclosed by the broken lines in the above figure.

3. The duty factor of the oscillation waveform must be within 45% to 55%.

4. Inputting an external clock to the V850E/IA3 is prohibited.

(i) Murata Mfg. Co., Ltd.: Ceramic resonator (T_A = –40 to +85°C)

Type	Part Number	Oscillation Frequency fx (MHz)	Recommended Circuit Constant			Oscillation Voltage Range	
			C1	C2	Rd (kΩ)	MIN. (V)	MAX. (V)
Lead	CSTLS4M00G56-B0	4	On chip (47 pF)	On chip (47 pF)	0	2.3	2.7
	CSTLS5M00G56-B0	5	On chip (47 pF)	On chip (47 pF)	0	2.3	2.7
	CSTLS8M00G53-B0	8	On chip (15 pF)	On chip (15 pF)	0	2.3	2.7
Surface mounting	CSTCR4M00G55-R0	4	On chip (39 pF)	On chip (39 pF)	0	2.3	2.7
	CSTCR5M00G55-R0	5	On chip (39 pF)	On chip (39 pF)	0	2.3	2.7
	CSTCE8M00G52-R0	8	On chip (10 pF)	On chip (10 pF)	0	2.3	2.7

Caution These oscillator constants are reference values based on evaluation under a specific environment by the resonator manufacturer. When optimization of the oscillator characteristics on the actual application is necessary, request evaluation on the mounting circuit from the resonator manufacturer.

The oscillation voltage and oscillation frequency indicate only oscillator characteristics, therefore use the V850E/IA3 within the DC characteristics and AC characteristics for internal operation conditions.

DC Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.[Note 1]	MAX.	Unit
Input voltage, high	V_{IH1}	**Note 2**		0.7EV_{DD}		EV_{DD}	V
	V_{IH2}	**Note 3**		0.8EV_{DD}		EV_{DD}	V
	V_{IH4}	**Note 4**		0.7AV_{DD}		AV_{DD}	V
Input voltage, low	V_{IL1}	**Note 2**		EV_{SS}		0.3EV_{DD}	V
	V_{IL2}	**Note 3**		EV_{SS}		0.2EV_{DD}	V
	V_{IL4}	**Note 4**		AV_{SS}		0.3AV_{DD}	V
Input leakage current, high	I_{LIH1}	V_I = AV_{DD} = EV_{DD}	Other than X1			5	μA
	I_{LIH2}		X1			20	μA
Input leakage current, low	I_{LIL1}	V_I = 0 V	Other than X1			–5	μA
	I_{LIL2}		X1			–20	μA
Output leakage current, high	I_{LOH}	V_O = AV_{DD} = EV_{DD}				5	μA
Output leakage current, low	I_{LOL}	V_O = 0 V				–5	μA
Output voltage, high	V_{OH}	**Note 5**	I_{OH} = –1.0 mA	EV_{DD} – 1.0			V
Output voltage, low	V_{OL1}	P10 to P15	I_{OL} = 15 mA			2.0	V
			I_{OL} = 1.0 mA			0.4	V
	V_{OL2}	**Note 6**	I_{OL} = 1.0 mA			0.4	V
Pull-up resistor	R_1			10	30	100	kΩ
Supply current[Note 7] (μPD70F3184)	I_{DD1}	f_{XX} = 64 MHz	Normal operation		90	120	mA
	I_{DD2}		HALT mode		50	75	mA
	I_{DD3}		IDLE mode		10	20	mA
	I_{DD4}	STOP mode			40	800[Note 8]	μA
Supply current[Note 7] (μPD703183)	I_{DD1}	f_{XX} = 64 MHz	Normal operation		70	95	mA
	I_{DD2}		HALT mode		45	70	mA
	I_{DD3}		IDLE mode		10	20	mA
	I_{DD4}	STOP mode			40	800[Note 8]	μA

Notes 1. The TYP. value is a reference value at T_A = 25°C, V_{DD} = 2.5 V.

2. P31, P33, P41, and PDL0 to PDL15 pins

3. P00, P02 to P07, P10 to P17, P30, P32, P34 to P37, P40, P42 to P44, $\overline{\text{RESET}}$, IC1 (μPD703183 only)/FLMD0 (μPD70F3184 only), and PLLSIN pins

4. P70 to P75 pins

5. P00, P02 to P07, P10 to P17, P30 to P37, P40 to P44, PDL0 to PDL15 pins

6. P00, P02 to P07, P16, P17, P30 to P37, P40 to P44, PDL0 to PDL15 pins

7. Current flowing through the output buffer and pull-up resistor is not included.

8. T_J (junction temperature) = 85°C

Remarks 1. The characteristics of alternate-function pins are the same as those of port pins.

2. When the I_{OH} and I_{OL} conditions are not satisfied for a pin but the total value of all pins is satisfied, only that pin does not satisfy the DC characteristics.

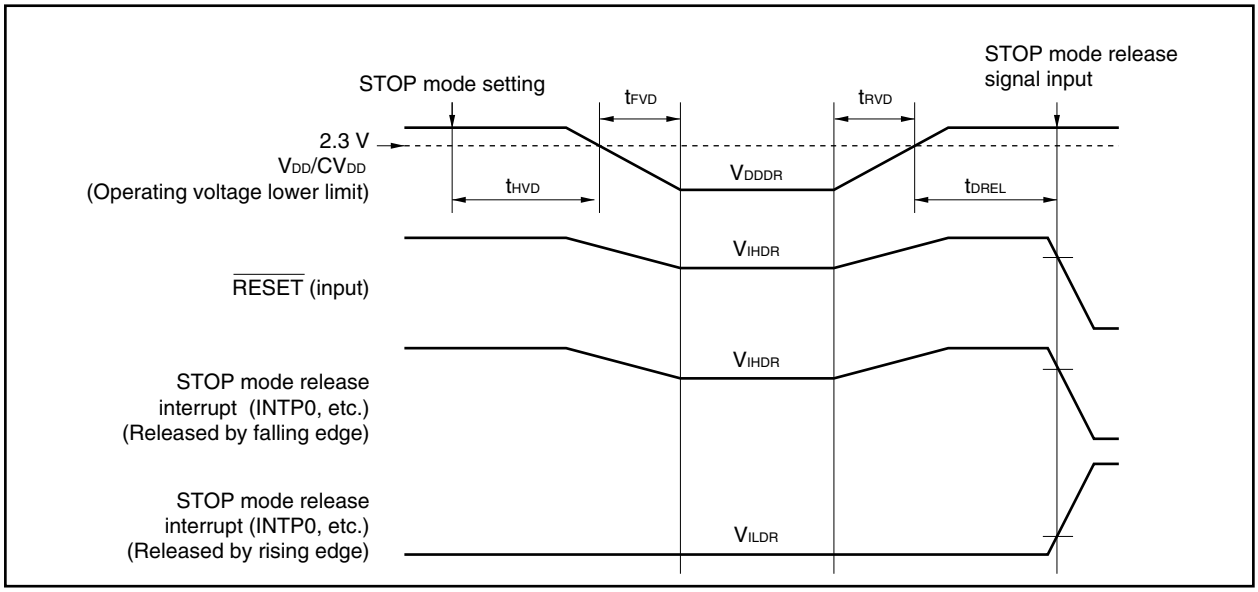
Data Retention Characteristics

STOP mode (T_A = –40 to +85°C, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Data retention voltage	V_{DDDR}	STOP mode	V_{DD}, CV_{DD}	1.8		2.7	V
			AV_{DD}, EV_{DD}	4.0		5.5	V
Data retention current	I_{DDDR}	V_{DD} = V_{DDDR}	μPD70F3184		40	800 Note	μA
			μPD703183		40	800 Note	μA
Supply voltage rise time	t_{RVD}	V_{DD}, CV_{DD}		200			μs
Supply voltage fall time	t_{FVD}	V_{DD}, CV_{DD}		200			μs
Supply voltage retention time (from STOP mode setting)	t_{HVD}			0			ms
STOP mode release signal input time	t_{DREL}			0			ms
Data retention input voltage, high	V_{IHDR}	All input pins		0.9V_{DDDR}		V_{DDDR}	V
Data retention input voltage, low	V_{ILDR}	All input pins		EV_{SS}		0.1V_{DDDR}	V

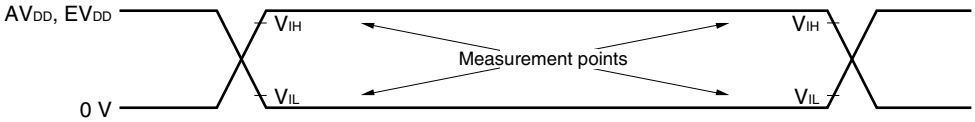
Note T_J (junction temperature) = 85°C

Caution Shifting to STOP mode and restoring from STOP mode must be performed at V_{DD} = CV_{DD} ≥ 2.3 V (AV_{DD} = EV_{DD} = 4.0 to 5.5 V).

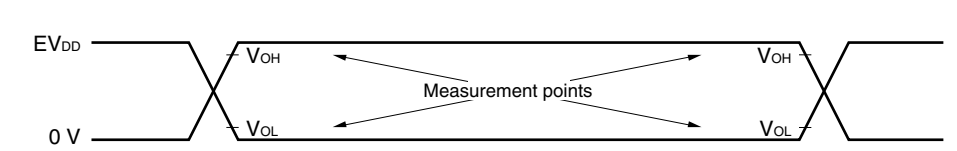


AC Characteristics

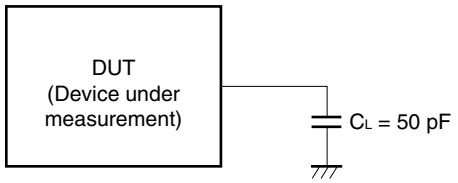
AC Test Input Measurement Points



AC Test Output Measurement Points



Load Conditions

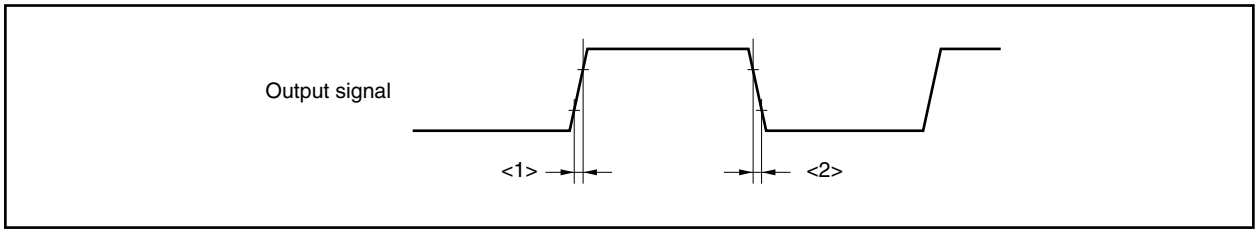


Caution If the load capacitance exceeds 50 pF due to the circuit configuration, bring the load capacitance of the device to 50 pF or less by inserting a buffer or by some other means.

Output Signal Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Output rise time	t_{OR}	<1>			15	ns
Output fall time	t_{OF}	<2>			15	ns



Reset, External Interrupt Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

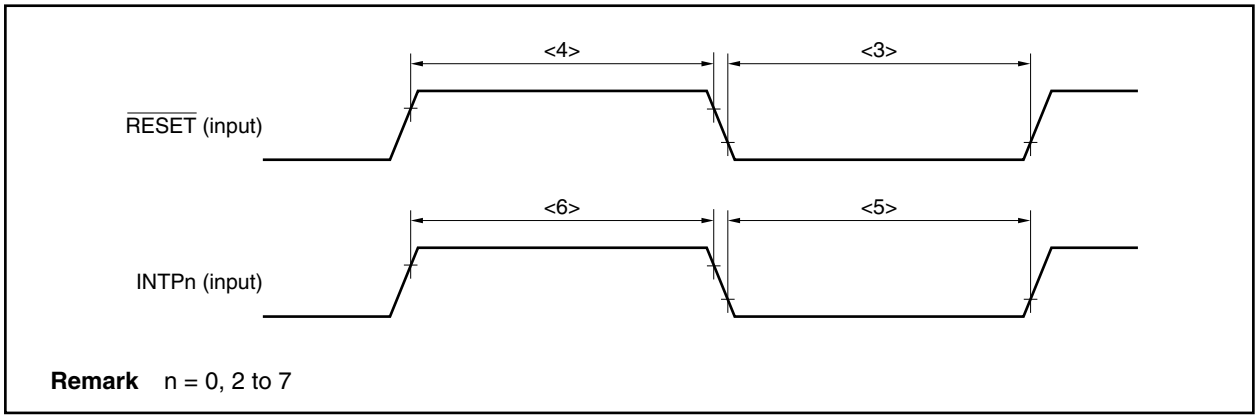
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{RESET}}$ low-level width	t_{WRSL}	<3>	Power is on, STOP mode is released	500 + T_{os}		ns
			Other than above	500		ns
$\overline{\text{RESET}}$ high-level width	t_{WRSH}	<4>		500		ns
INTPn low-level width	t_{WITL}	<5>	n = 0, 2 to 5, 7 (analog noise elimination)	500		ns
			n = 6 (digital noise elimination)	5T_{smp} + 10		ns
INTPn high-level width	t_{WITH}	<6>	n = 0, 2 to 5, 7 (analog noise elimination)	500		ns
			n = 6 (digital noise elimination)	5T_{smp} + 10		ns

Remarks 1. T_{os}: Oscillation stabilization time

T_{smp}: Noise elimination sampling clock cycle (set by INTPNRC register)

2. After reset release, an oscillation stabilization time is internally secured for 1 ms. The oscillation stabilization time is therefore (T_{os} + 1) ms. After STOP mode release, an oscillation stabilization time half the value set to the OSTS register is internally secured. Therefore, T_{os} = 0 ns is acceptable if sufficient stabilization time can be secured by the OSTS register setting.

Reset/Interrupt



Remark n = 0, 2 to 7

Timer Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
TIn high-level width[Note 1]	t_{WTIHn}	<7>		10T + 10		ns
TIn low-level width[Note 1]	t_{WTILn}	<8>		10T + 10		ns
EVTQ0 high-level width[Note 1]	t_{WEVHn}	<9>		10T + 10		ns
EVTQ0 low-level width[Note 1]	t_{WEVLn}	<10>		10T + 10		ns
TRGQ0 high-level width[Note 1]	t_{WTRHn}	<11>		10T + 10		ns
TRGQ0 low-level width[Note 1]	t_{WTRLn}	<12>		10T + 10		ns
TIUD10/TCUD10 high-level width[Note 2]	t_{WUDHm}	<13>		$5T_{smp}$ + 10		ns
TIUD10/TCUD10 low-level width[Note 2]	t_{WUDLm}	<14>		$5T_{smp}$ + 10		ns
TCLR10 high-level width[Note 2]	t_{WTCHm}	<15>		$5T_{smp}$ + 10		ns
TCLR10 low-level width[Note 2]	t_{WTCLm}	<16>		$5T_{smp}$ + 10		ns
TIUD10/TCUD10 input time differential[Note 2]	t_{PHUD}	<17>		$5T_{smp}$ + 10		ns

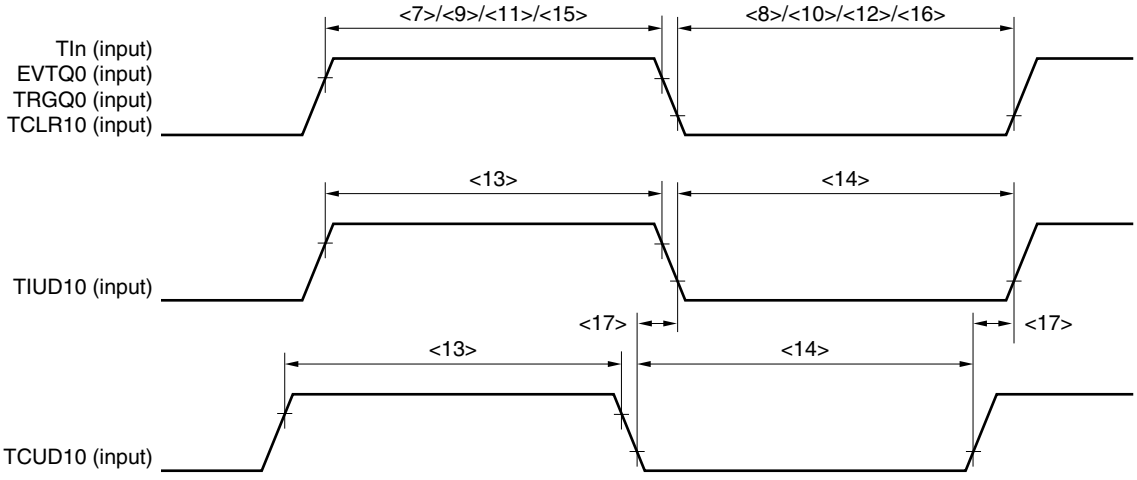
Notes 1. T = 1/fxx

2. T_{smp}: Noise elimination sampling clock cycle (set by NRC10 register)

Remarks 1. n = Q00, Q01, Q02, Q03, P00, P01, P20, P21

2. The above specification shows a pulse width that is accurately detected as a valid edge. Even if a pulse narrower than the above specification is input, therefore, it may be detected as a valid edge.

Timer Input Timing



Remark n = Q00, Q01, Q02, Q03, P00, P01, P20, P21

CSIB Timing

(1) Master mode

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

	Parameter	Symbol		Conditions	MIN.	MAX.	Unit
	$\overline{SCKBn}$ cycle	t_{KCYM}	<18>		125		ns
	$\overline{SCKBn}$ high-/low-level width	t_{KWHM}, t_{KWLM}	<19>		t_{KCYM}/2 – 10		ns
	SIBn setup time (to $\overline{SCKBn}$↑)	t_{SSIM}	<20>		30		ns
	SIBn hold time (from $\overline{SCKBn}$↑)	t_{HSIM}	<21>		30		ns
	SOBn output delay time (from $\overline{SCKBn}$↓)	t_{DSOM}	<22>			30	ns
	SOBn output delay time (from $\overline{SCKBn}$↑)					30	ns
<R>	SOBn output hold time (from $\overline{SCKBn}$↑)	t_{HSOM}	<23>		t_{KCYM}/2 – 10		ns
<R>	SOBn output hold time (from $\overline{SCKBn}$↓)				t_{KCYM}/2 – 10		ns

Remark n = 0, 1

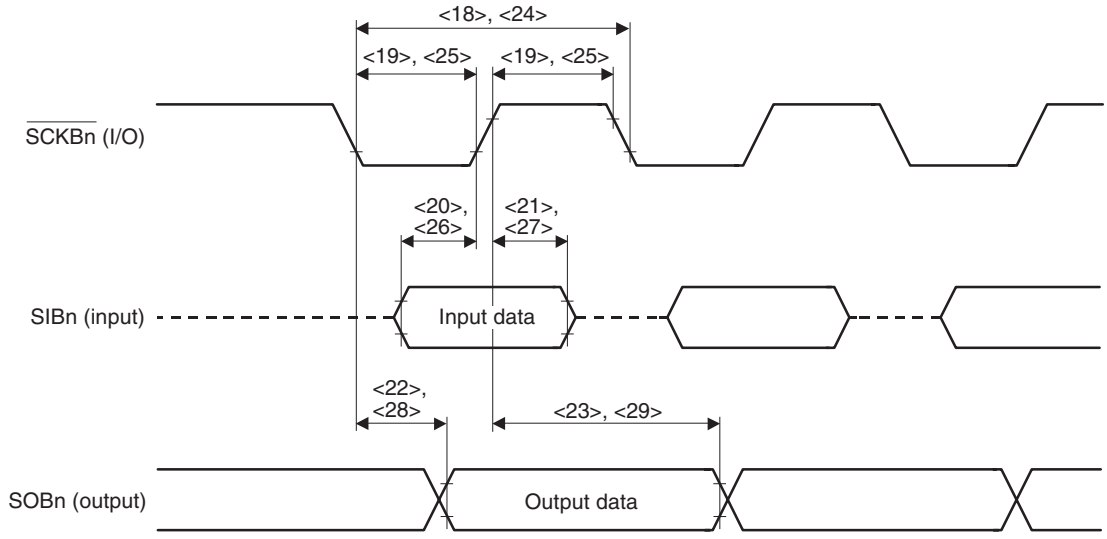
(2) Slave mode

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

	Parameter	Symbol		Conditions	MIN.	MAX.	Unit
	$\overline{SCKBn}$ cycle	t_{KCYS}	<24>		125		ns
	$\overline{SCKBn}$ high-/low-level width	t_{KWHS}, t_{KWLS}	<25>		t_{KCYS}/2 – 10		ns
	SIBn setup time (to $\overline{SCKBn}$↑)	t_{SSIS}	<26>		30		ns
	SIBn hold time (from $\overline{SCKBn}$↑)	t_{HSIS}	<27>		30		ns
	SOBn output delay time (from $\overline{SCKBn}$↓)	t_{DSOS}	<28>			30	ns
	SOBn output delay time (from $\overline{SCKBn}$↑)					30	ns
<R>	SOBn output hold time (from $\overline{SCKBn}$↑)	t_{HSOS}	<29>		t_{KCYS}/2 – 10		ns
<R>	SOBn output hold time (from $\overline{SCKBn}$↓)				t_{KCYS}/2 – 10		ns

Remark n = 0, 1

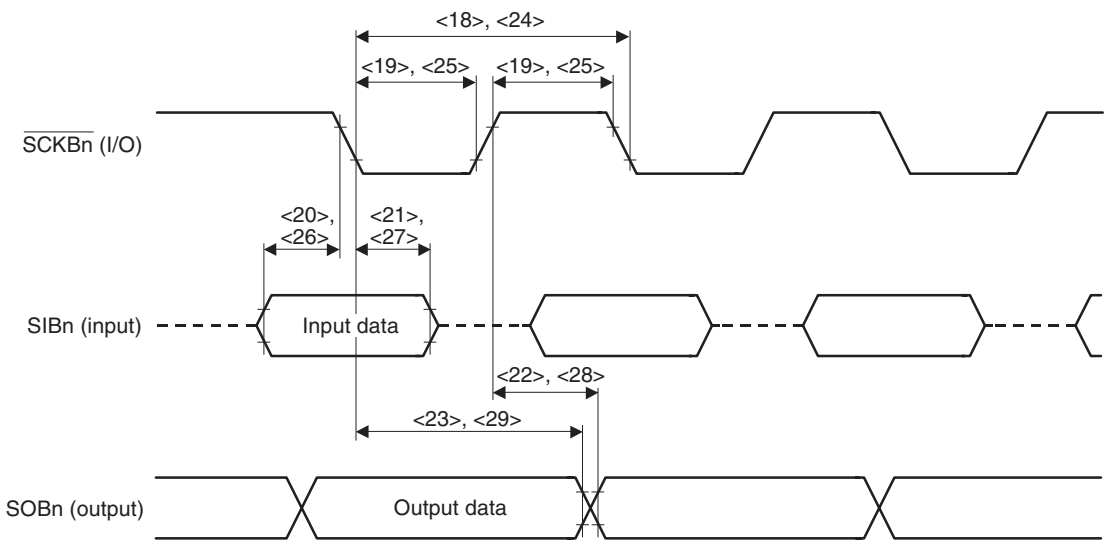
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 00



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

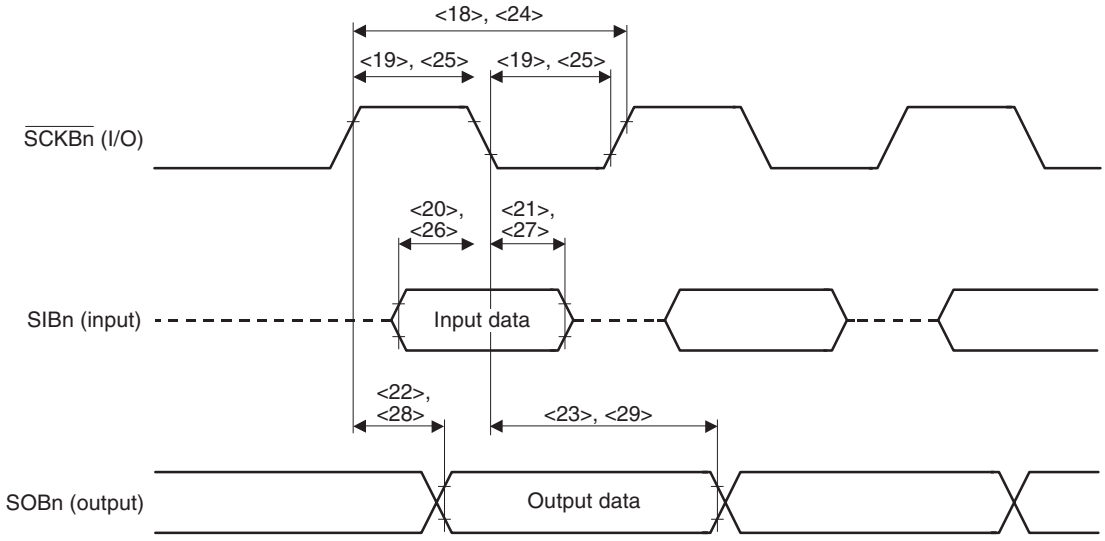
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 01



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

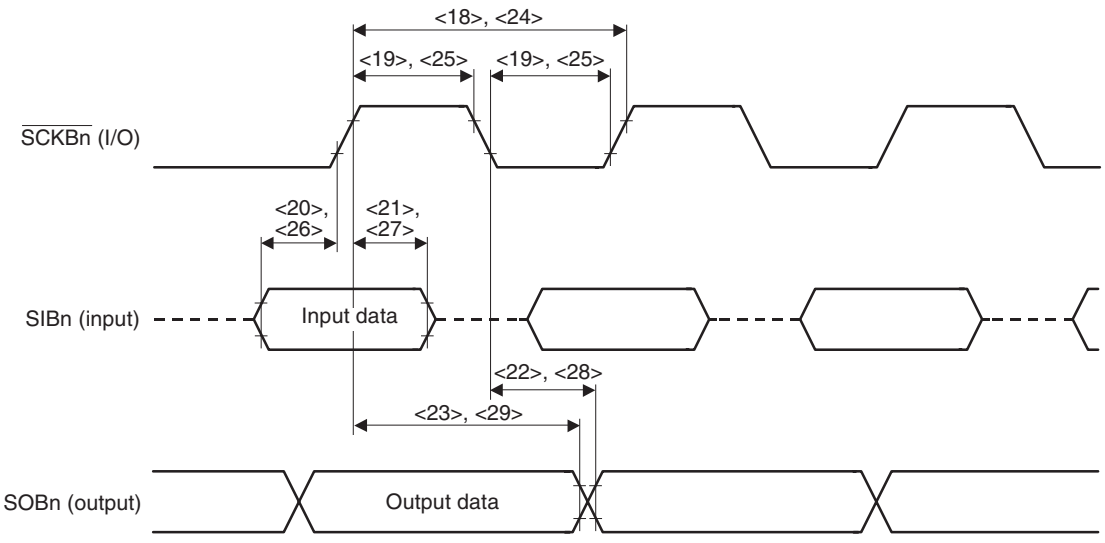
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 10



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 11



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

High-Impedance Control Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Oscillation stop → timer output high impedance	t_{CLM}	When clock monitor is operating		65	μs
Input to TOQ0OFF → timer output high impedance	t_{HTQ0}			300	ns
Input to TOP2OFF → timer output high impedance	t_{HTP2}			300	ns
Input to ANI00, ANI01 → timer output high impedance	t_{ANI0}			10	μs
Input to ANI10 to ANI12 → timer output high impedance	t_{ANI1}			10	μs

Characteristics of A/D Converters 0, 1

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10	10	10	bit
Overall error[Note 1]					±4.0	LSB
Conversion time	t_{CONV}	f_{XX} = 64 MHz, ADAnM1 register = 01H	1.94			μs
		f_{XX} = 32 MHz, ADAnM1 register = 03H			7.75	μs
Zero scale error[Note 1]					±4.0	LSB
Full-scale error[Note 1]					±4.0	LSB
Integral linearity error[Note 1]					±4.0	LSB
Differential linearity error[Note 1]					±2.0	LSB
Analog reference voltage	AV_{DD}		4.5		5.5	V
Analog input voltage	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} supply current[Note 2]	AI_{DD}	During operation		5	10	mA
	AI_{DDS}	In STOP mode[Note 3]		1	10	μA

Notes 1. Excludes quantization error (±0.5 LSB).

2. The value for either of A/D converter 0 or A/D converter 1.

3. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. LSB: Least Significant Bit

2. n = 0, 1

Characteristics of A/D Converter 2

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8		10	Bits
Overall error		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Conversion time	t_{CONV}	10-bit resolution, serial mode		512		μs
		10-bit resolution, parallel mode		128 Note 1		μs
		8-bit resolution, serial mode		128		μs
		8-bit resolution, parallel mode		32 Note 1		μs
Zero scale error Note 2		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Full-scale error Note 2		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Integral linearity error Note 2		10-bit resolution			±2.0	LSB
		8-bit resolution			±1.5	LSB
Differential linearity error Note 2		10-bit resolution			±2.0	LSB
		8-bit resolution			±1.5	LSB
Analog reference voltage	AV_{DD}		4.5		5.5	V
Analog input voltage	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} supply current	AI_{DD}	During operation		1	3	mA
	AI_{DDS}	In STOP mode Note 3		1	10	μA

Notes 1. The time taken for the first conversion to end in parallel mode is the same as that in serial mode. The above specification shows the value for the second and subsequent conversions.

2. Excludes quantization error (±0.5 LSB).

3. Stop the operation of A/D converter 2 (ADA2CTL0.ADA2CE bit = 0) before setting STOP mode.

Remark LSB: Least Significant Bit

Operational Amplifier Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input offset voltage	V_{IO}			±5.0		mV
Input voltage range	V_I	Gain = 2.5	0.04AV_{DD}		0.32AV_{DD}	V
		Gain = 5	0.03AV_{DD}		0.16AV_{DD}	V
Slew rate	S_R			10		V/μs
Gain error		Gain = 2.5		±1.0	±5.0	%
		Gain = 5		±1.0	±5.0	%
Operating current[Note 1]	I_{OPDD}	During operation		1.0	3.0	mA
	A_{IDDS}	In STOP mode[Note 2]		1.0	10	μA

Notes 1. Five operational amplifiers are provided in total. The value shows the operating current per operational amplifier.

2. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. The operating current of the operational amplifiers is included in AV_{DD}.

2. n = 0, 1

Comparator Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input offset voltage	V_{IO}			±3.0		mV
Input voltage range	V_I	CMPREF, ANIm	0.1AV_{DD}		0.5AV_{DD}	V
Response time	t_{CR}	Input amplitude = 100 mV, at rising edge[Note 1]		4.0		μs
	t_{CF}	Input amplitude = 100 mV, at falling edge[Note 2]		2.0		μs
Operating current[Note 3]	I_{CPDD}	During operation		50	150	μA
	A_{IDDS}	In STOP mode[Note 4]		1.0	10	μA

Notes 1. Characteristics of pulse response when ANIm input changes from CMPREF − 100 mV to CMPREF + 100 mV

2. Characteristics of pulse response when ANIm input changes from CMPREF + 100 mV to CMPREF − 100 mV

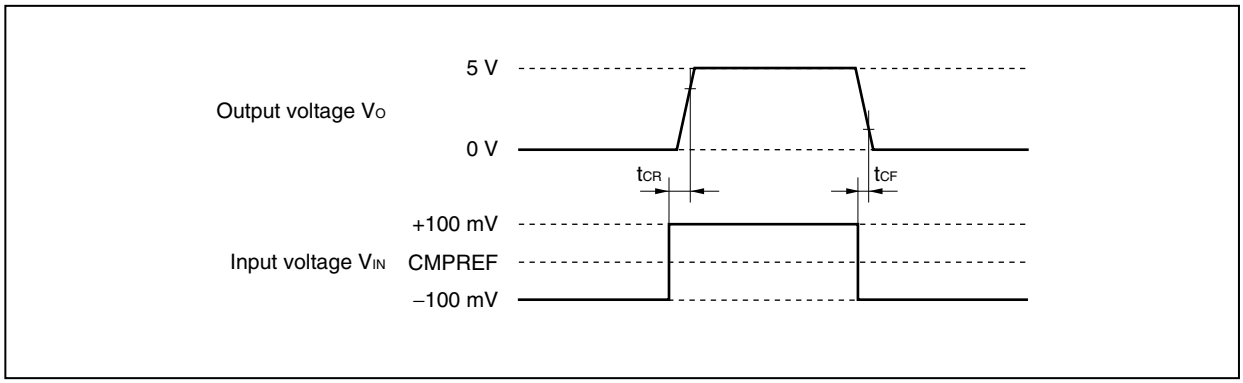
3. Five comparators are provided in total. The value shows the operating current per comparator.

4. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. The operating current of the comparators is included in AV_{DD}.

2. m = 00, 01, 10 to 12
n = 0, 1

Comparator Characteristics



Supply Voltage Application/Cutoff Timing

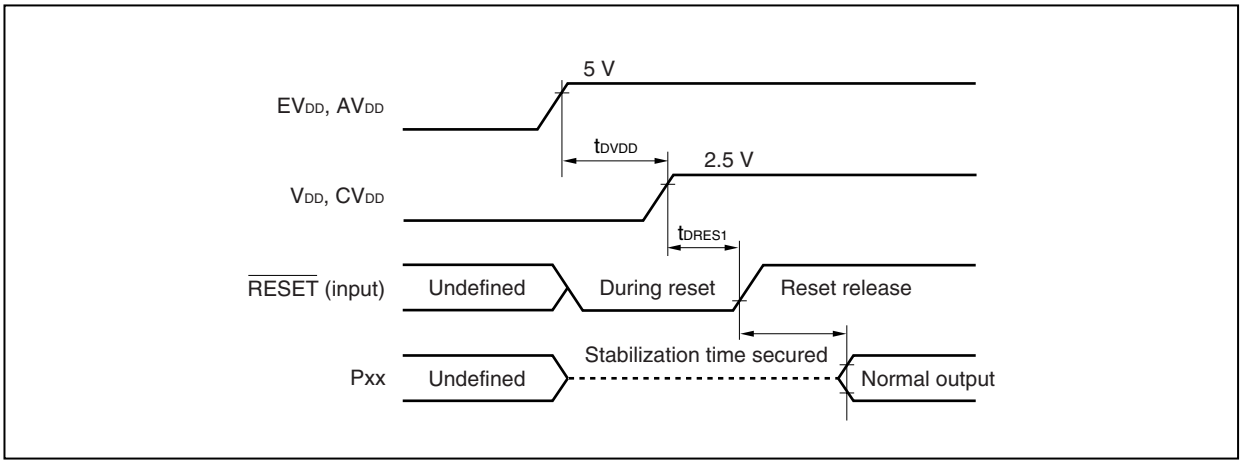
(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Delay time from EV_{DD}, AV_{DD} rise to V_{DD}, CV_{DD} rise	t_{DVDD}		–50	50	ms
Delay time from V_{DD}, CV_{DD} rise to $\overline{RESET}$↑	t_{DRES1}		T_{osc} – 1		ms
Delay time from EV_{DD}, AV_{DD} rise to $\overline{RESET}$↑	t_{DRES2}		T_{osc} – 1		ms
Delay time from EV_{DD}, AV_{DD} fall to V_{DD}, CV_{DD} fall	t_{DEVDD}		0		ns

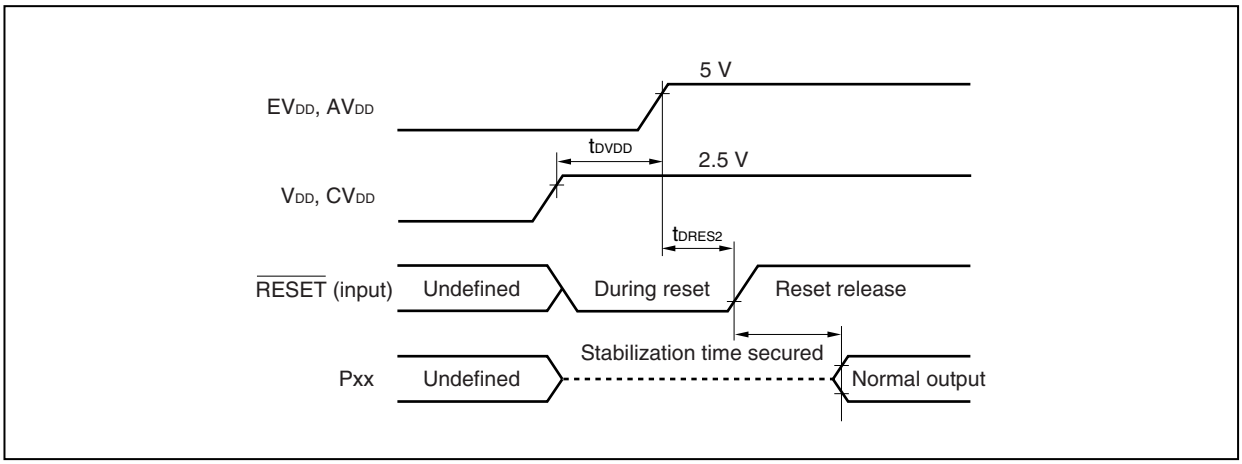
Remark T_{osc}: Oscillation stabilization time of oscillator (varies depending on the resonator or oscillator used)

Supply Voltage Application Timing

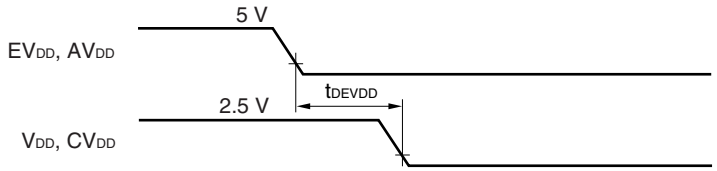
(a) Power supply sequence recommended condition 1



(b) Power supply sequence recommended condition 2



Supply Voltage Cutoff Timing



Cautions 1. There are no regulations for the voltage level and time of 2.5 V V_{DD} and CV_{DD}, and 5 V EV_{DD} and AV_{DD} in the process of natural discharge after power supply cutoff.

2. The application of only one of power supplies V_{DD}, CV_{DD}, EV_{DD}, or AV_{DD}, and only either the 2.5 V V_{DD} and CV_{DD} power supplies or the 5 V EV_{DD} and AV_{DD} power supplies is prohibited.

Flash Memory Programming Characteristics (μPD70F3184 only)

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Write count	C_{ERWR}	**Note**		100		Times
Write current				90	120	mA
Erase current				90	120	mA

Note When writing initially to shipped products, it is counted as one rewrite for both “erase to write” and “write only”.

Example (P: Write, E: Erase)

Shipped product ⟶ P→ E→ P→ E→ P: 3 rewrites

Shipped product → E→ P→ E→ P→ E→ P: 3 rewrites

Absolute Maximum Ratings (T_A = 25°C)

Parameter	Symbol	Conditions		Ratings	Unit
Supply voltage	V_{DD}	$V_{DD} = CV_{DD}$		−0.5 to +3.6	V
	V_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	EV_{DD}	$EV_{DD} = AV_{DD}$		−0.5 to +6.5	V
	EV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	CV_{DD}	$V_{DD} = CV_{DD}$		−0.5 to +3.6	V
	CV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
	AV_{DD}	$EV_{DD} = AV_{DD}$		−0.5 to +6.5	V
	AV_{SS}	$V_{SS} = CV_{SS} = EV_{SS} = AV_{SS}$		−0.5 to +0.5	V
Input voltage	V_{I1}	**Note 1**		−0.5 to EV_{DD} + 0.5 Note 2	V
	V_{I2}	X1, X2		−0.5 to CV_{DD} + 0.5 Note 2	V
Output current, low	I_{OL}	P10 to P15, P20 to P25	Per pin	18	mA
		Pins other than P10 to P15, P20 to P25	Per pin	4	mA
		P00 to P07, P20 to P25, P40 to P42	Total of all pins	50	mA
		P10 to P17, DDO Note 3	Total of all pins	50	mA
		P26, P27, P30 to P37, P43, P44, P50 to P52, PDL0 to PDL15	Total of all pins	50	mA
Output current, high	I_{OH}	All pins	Per pin	−4.0	mA
		P00 to P07, P20 to P25, P40 to P42	Total of all pins	−20	mA
		P10 to P17, DDO Note 3	Total of all pins	−10	mA
		P26, P27, P30 to P37, P43, P44, P50 to P52, PDL0 to PDL15	Total of all pins	−40	mA
Analog input voltage	V_{IAN}	P70/ANI20 to P77/ANI27, ANI00 to ANI03, ANI10 to ANI13		−0.5 to AV_{DD} + 0.5 Note 2	V
Analog input reference voltage	V_{IREF}	CMPREF		−0.5 to AV_{DD} + 0.5 Note 2	V
Operating ambient temperature	T_A	In normal operating mode		−40 to +85	°C
		In flash memory programming mode		−40 to +85	°C
Storage temperature	T_{stg}			−40 to +125	°C

Notes 1. P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15, $\overline{\text{RESET}}$, IC1 (μPD703185, 703186 only)/FLMD0 (μPD70F3186 only), PLLSIN, $\overline{\text{DRST}}$ (μPD70F3186 only)

2. Be sure not to exceed the absolute maximum ratings (MAX. value) of each supply voltage.

3. μPD70F3186 only

Cautions 1. Do not directly connect the output pins (or I/O pins in the output state) of IC products to other output pins (including I/O pins in the output state), power supply pins such as V_{DD} and EV_{DD}, or GND pin. Direct connection of the output pins between an IC product and an external circuit is possible, if the output pins can be set to the high-impedance state and the output timing of the external circuit is designed to avoid output conflict.

2. 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.

The ratings and conditions indicated for DC characteristics and AC characteristics represent the quality assurance range during normal operation.

Capacitance (T_A = 25°C, V_{DD} = V_{SS} = EV_{DD} = EV_{SS} = CV_{DD} = CV_{SS} = AV_{DD} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input capacitance	C_I	fc = 1 MHz	**Note 1**			15	pF
I/O capacitance	C_{IO}	Unmeasured pins returned to 0 V	**Note 2**			15	pF
Output capacitance	C_O		**Note 3**			15	pF

Notes 1. ANI00 to ANI03, ANI10 to ANI13, P70 to P77, PLLSIN, $\overline{\text{RESET}}$, CMPREF

2. P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15

3. DDO (μPD70F3186 only)

Cautions 1. Excludes the FLMD0 (μPD70F3186 only), $\overline{\text{DRST}}$ (μPD70F3186 only), X1, and X2 pins.

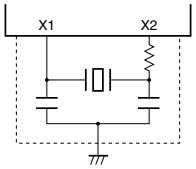
2. In addition to input capacitance, sampling capacitance is added to the ANI00 to ANI03, ANI10 to ANI13, and ANI20 to ANI27 pins for sampling.

Operating Conditions (T_A = –40 to +85°C, V_{SS} = EV_{SS} = CV_{SS} = AV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
System clock frequency	f_{XX}	PLL mode	PLLSIN = Low level	32		55	MHz
			PLLSIN = High level	55		64	MHz
		Clock through mode		4		8	MHz
CPU clock frequency	f_{CPU}	PLL mode	PLLSIN = Low level	4		55	MHz
			PLLSIN = High level	6.875		64	MHz
		Clock through mode		0.5		8	MHz
V_{DD}, CV_{DD} voltage	V_{DD}, CV_{DD}	V_{DD} = CV_{DD}		2.3		2.7	V
EV_{DD} voltage	EV_{DD}	EV_{DD} = AV_{DD}		4.0		5.5	V
AV_{DD} voltage	AV_{DD}	When A/D converters 0 to 2 are operating		4.5		5.5	V
		When A/D converters 0 to 2 are not operating		4.0		5.5	V

Clock Oscillator Characteristics

(T_A = –40 to +85°C, CV_{DD} = 2.3 to 2.7 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Resonator	Recommended Circuit	Parameter	Conditions	MIN.	TYP.	MAX.	Unit
Ceramic /crystal resonator		Oscillation frequency (fx)		4		8	MHz
		Oscillation stabilization time	After reset release		2^{14}/fx		ms
			After STOP mode release		**Note**		ms

Note The value varies depending on the setting of the oscillation stabilization time select register (OSTS).

Cautions 1. Connect the oscillator as close to the X1 and X2 pins as possible.

2. Do not cross the wiring with the other signal lines in the area enclosed by the broken lines in the above figure.

3. The duty factor of the oscillation waveform must be within 45% to 55%.

4. Inputting an external clock to the V850E/IA4 is prohibited.

(i) Murata Mfg. Co., Ltd.: Ceramic resonator (T_A = –40 to +85°C)

Type	Part Number	Oscillation Frequency fx (MHz)	Recommended Circuit Constant			Oscillation Voltage Range	
			C1	C2	Rd (kΩ)	MIN. (V)	MAX. (V)
Lead	CSTLS4M00G56-B0	4	On chip (47 pF)	On chip (47 pF)	0	2.3	2.7
	CSTLS5M00G56-B0	5	On chip (47 pF)	On chip (47 pF)	0	2.3	2.7
	CSTLS8M00G53-B0	8	On chip (15 pF)	On chip (15 pF)	0	2.3	2.7
Surface mounting	CSTCR4M00G55-R0	4	On chip (39 pF)	On chip (39 pF)	0	2.3	2.7
	CSTCR5M00G55-R0	5	On chip (39 pF)	On chip (39 pF)	0	2.3	2.7
	CSTCE8M00G52-R0	8	On chip (10 pF)	On chip (10 pF)	0	2.3	2.7

Caution These oscillator constants are reference values based on evaluation under a specific environment by the resonator manufacturer. When optimization of the oscillator characteristics on the actual application is necessary, request evaluation on the mounting circuit from the resonator manufacturer.

The oscillation voltage and oscillation frequency indicate only oscillator characteristics, therefore use the V850E/IA4 within the DC characteristics and AC characteristics for internal operation conditions.

DC Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V) (1/2)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Input voltage, high	V_{IH1}	**Note 1**		0.7EV_{DD}		EV_{DD}	V
	V_{IH2}	**Note 2**		0.8EV_{DD}		EV_{DD}	V
	V_{IH3}	**Note 3**		2.2		EV_{DD}	V
	V_{IH4}	**Note 4**		0.7AV_{DD}		AV_{DD}	V
Input voltage, low	V_{IL1}	**Note 1**		EV_{SS}		0.3EV_{DD}	V
	V_{IL2}	**Note 2**		EV_{SS}		0.2EV_{DD}	V
	V_{IL3}	**Note 3**		EV_{SS}		0.8	V
	V_{IL4}	**Note 4**		AV_{SS}		0.3AV_{DD}	V
Input leakage current, high	I_{LIH1}	V_I = AV_{DD} = EV_{DD},	Other than X1			5	μA
	I_{LIH2}	**Note 5**	X1			20	μA
Input leakage current, low	I_{LIL1}	V_I = 0 V	Other than X1			–5	μA
	I_{LIL2}		X1			–20	μA
Output leakage current, high	I_{LOH}	V_O = AV_{DD} = EV_{DD}				5	μA
Output leakage current, low	I_{LOL}	V_O = 0 V				–5	μA
Output voltage, high	V_{OH}	**Note 6**	I_{OH} = –1.0 mA	EV_{DD} – 1.0			V
Output voltage, low	V_{OL1}	**Note 7**	I_{OL} = 15 mA			2.0	V
			I_{OL} = 1.0 mA			0.4	V
	V_{OL2}	**Note 8**	I_{OL} = 1.0 mA			0.4	V
Pull-up resistor	R_1			10	30	100	kΩ
Pull-down resistor[Note 9]	R_2			10	30	100	kΩ

Notes 1. P20 to P27, P31, P33, P41, and PDL0 to PDL15 pins

2. P00 to P07, P10 to P17, P30, P32, P34 to P37, P40, P42 to P44, P50 to P52, $\overline{\text{RESET}}$, IC1 (μPD703185, 703186 only)/FLMD0 (μPD70F3186 only), and PLLSIN pins

3. $\overline{\text{DRST}}$, DDI, DCK, DMS pins (μPD70F3186 only)

4. P70 to P77 pins

5. $\overline{\text{DRST}}$ pin (μPD70F3186 only)

6. P00 to P07, P10 to P17, P20 to P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15 pins, and DDO pin (μPD70F3186 only)

7. P10 to P15, P20 to P25 pin

8. P00 to P07, P16, P17, P26, P27, P30 to P37, P40 to P44, P50 to P52, PDL0 to PDL15 pins, and DDO pin (μPD70F3186 only)

9. $\overline{\text{DRST}}$ pin (μPD70F3186 only) only

Remarks 1. The characteristics of alternate-function pins are the same as those of port pins.

2. When the I_{OH} and I_{OL} conditions are not satisfied for a pin but the total value of all pins is satisfied, only that pin does not satisfy the DC characteristics.

DC Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V) (2/2)

Parameter	Symbol	Conditions		MIN.	TYP.[Note 1]	MAX.	Unit
Supply current[Note 2] (μPD70F3186)	I_{DD1}	f_{XX} = 64 MHz	Normal operation		90	120	mA
	I_{DD2}		HALT mode		50	75	mA
	I_{DD3}		IDLE mode		10	20	mA
	I_{DD4}	STOP mode			40	800[Note 3]	μA
Supply current[Note 2] (μPD703185, 703186)	I_{DD1}	f_{XX} = 64 MHz	Normal operation		70	95	mA
	I_{DD2}		HALT mode		45	70	mA
	I_{DD3}		IDLE mode		10	20	mA
	I_{DD4}	STOP mode			40	800[Note 3]	μA

Notes 1. The TYP. value is a reference value when V_{DD} = 2.5 V and T_A = 25°C.

2. Current flowing through the output buffer and pull-up resistor is not included.

3. T_J (junction temperature) = 85°C

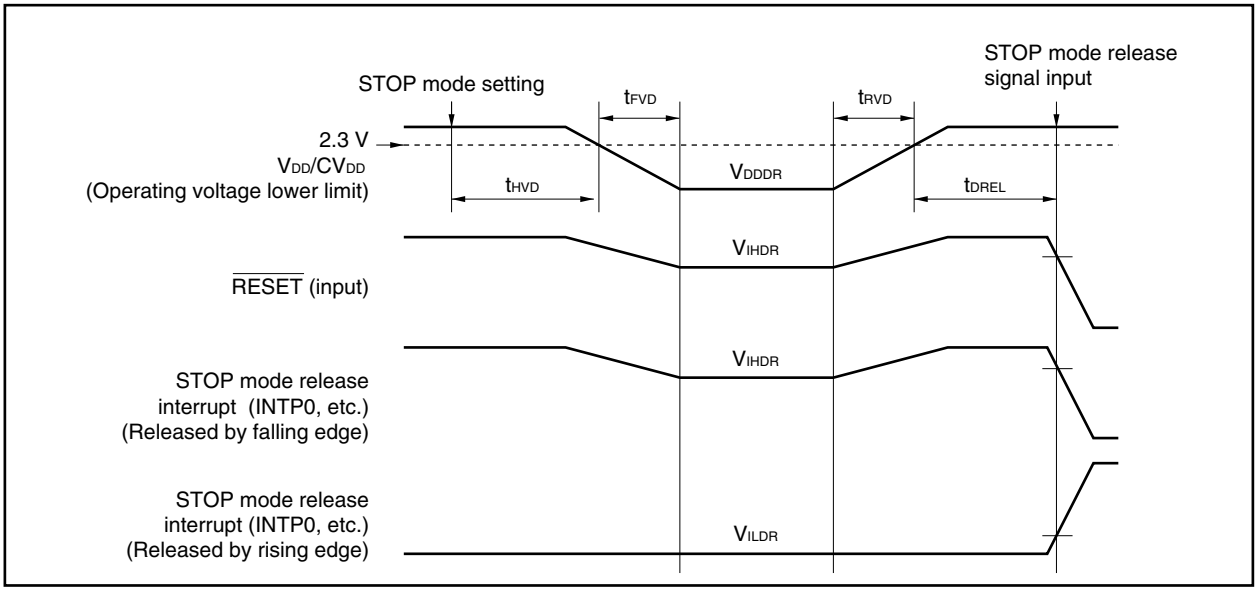
Data Retention Characteristics

STOP mode (T_A = –40 to +85°C, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions		MIN.	TYP.	MAX.	Unit
Data retention voltage	V_{DDDR}	STOP mode	V_{DD}, CV_{DD}	1.8		2.7	V
			AV_{DD}, EV_{DD}	4.0		5.5	V
Data retention current	I_{DDDR}	V_{DD} = V_{DDDR}	μPD70F3186		40	800[Note]	μA
			μPD703185		40	800[Note]	μA
			μPD703186		40	800[Note]	μA
Supply voltage rise time	t_{RVD}	V_{DD}, CV_{DD}		200			μs
Supply voltage fall time	t_{FVD}	V_{DD}, CV_{DD}		200			μs
Supply voltage retention time (from STOP mode setting)	t_{HVD}			0			ms
STOP mode release signal input time	t_{DREL}			0			ms
Data retention input voltage, high	V_{IHDR}	All input pins		0.9V_{DDDR}		V_{DDDR}	V
Data retention input voltage, low	V_{ILDR}	All input pins		EV_{SS}		0.1V_{DDDR}	V

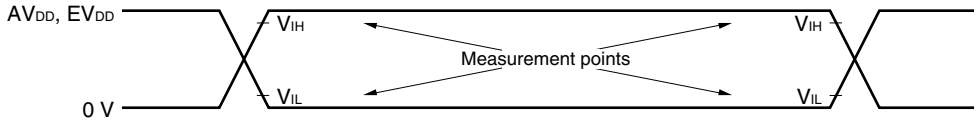
Note T_J (junction temperature) = 85°C

Caution Shifting to STOP mode and restoring from STOP mode must be performed at V_{DD} = CV_{DD} ≥ 2.3 V (AV_{DD} = EV_{DD} = 4.0 to 5.5 V).

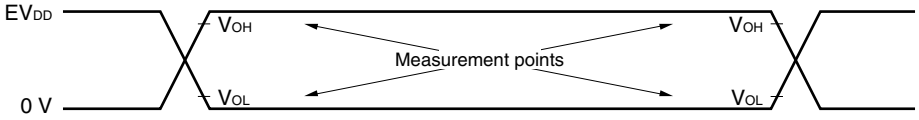


AC Characteristics

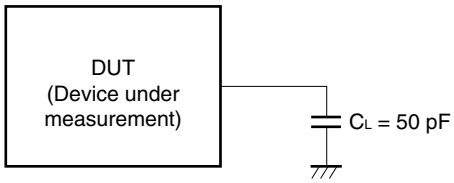
AC Test Input Measurement Points



AC Test Output Measurement Points



Load Conditions

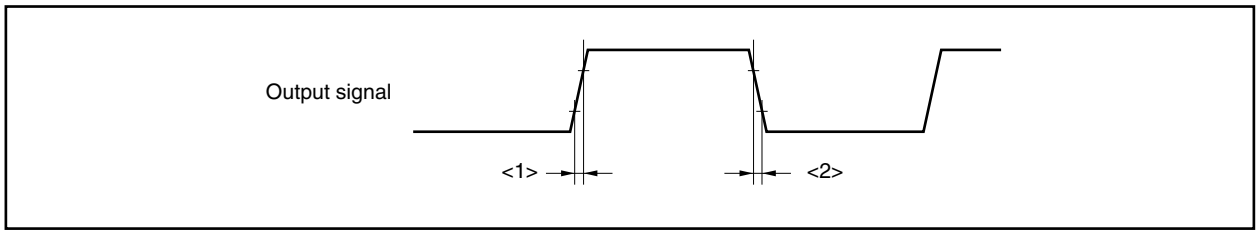


Caution If the load capacitance exceeds 50 pF due to the circuit configuration, bring the load capacitance of the device to 50 pF or less by inserting a buffer or by some other means.

Output Signal Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
Output rise time	t_{OR}	<1>			15	ns
Output fall time	t_{OF}	<2>			15	ns



Reset, External Interrupt Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

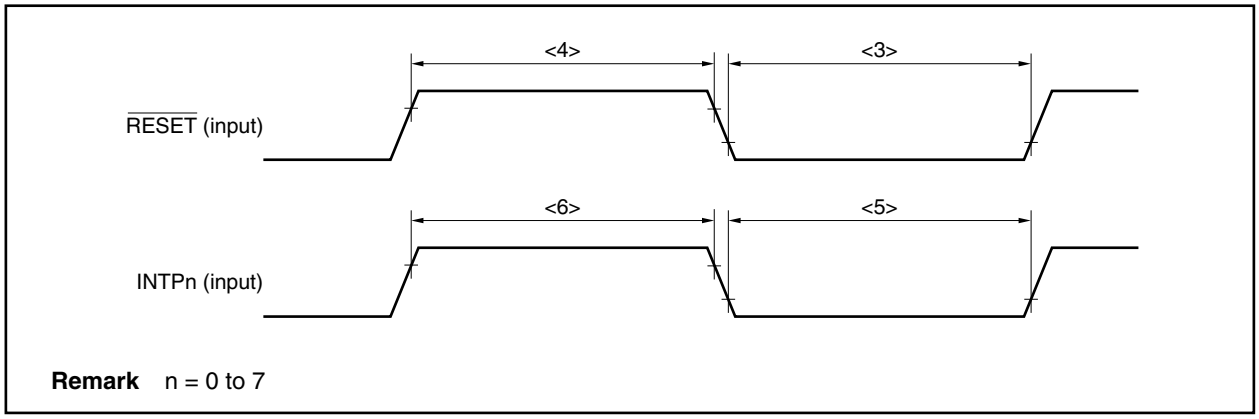
Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{RESET}}$ low-level width	t_{WRSL}	<3>	Power is on, STOP mode is released	500 + T_{os}		ns
			Other than above	500		ns
$\overline{\text{RESET}}$ high-level width	t_{WRSH}	<4>		500		ns
INTPn low-level width	t_{WITL}	<5>	n = 0 to 5, 7 (analog noise elimination)	500		ns
			n = 6 (digital noise elimination)	$5T_{smp}$ + 10		ns
INTPn high-level width	t_{WITH}	<6>	n = 0 to 5, 7 (analog noise elimination)	500		ns
			n = 6 (digital noise elimination)	$5T_{smp}$ + 10		ns

Remarks 1. T_{os}: Oscillation stabilization time

T_{smp}: Noise elimination sampling clock cycle (set by INTPNRC register)

2. After reset release, an oscillation stabilization time is internally secured for 1 ms. The oscillation stabilization time is therefore (T_{os} + 1) ms. After STOP mode release, an oscillation stabilization time half the value set to the OSTS register is internally secured. Therefore, T_{os} = 0 ns is acceptable if sufficient stabilization time can be secured by the OSTS register setting.

Reset/Interrupt



Remark n = 0 to 7

Timer Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
TIn high-level width[Note 1]	t_{WTIHn}	<7>		10T + 10		ns
TIn low-level width[Note 1]	t_{WTILn}	<8>		10T + 10		ns
EVTQ0 high-level width[Note 1]	t_{WEVHn}	<9>		10T + 10		ns
EVTQ0 low-level width[Note 1]	t_{WEVLn}	<10>		10T + 10		ns
TRGQ0 high-level width[Note 1]	t_{WTRHn}	<11>		10T + 10		ns
TRGQ0 low-level width[Note 1]	t_{WTRLn}	<12>		10T + 10		ns
TIUD1m/TCUD1m high-level width[Note 2]	t_{WUDHm}	<13>		$5T_{smp}$ + 10		ns
TIUD1m/TCUD1m low-level width[Note 2]	t_{WUDLm}	<14>		$5T_{smp}$ + 10		ns
TCLR1m high-level width[Note 2]	t_{WTCHm}	<15>		$5T_{smp}$ + 10		ns
TCLR1m low-level width[Note 2]	t_{WTCLm}	<16>		$5T_{smp}$ + 10		ns
TIUD1m/TCUD1m input time differential[Note 2]	t_{PHUD}	<17>		$5T_{smp}$ + 10		ns

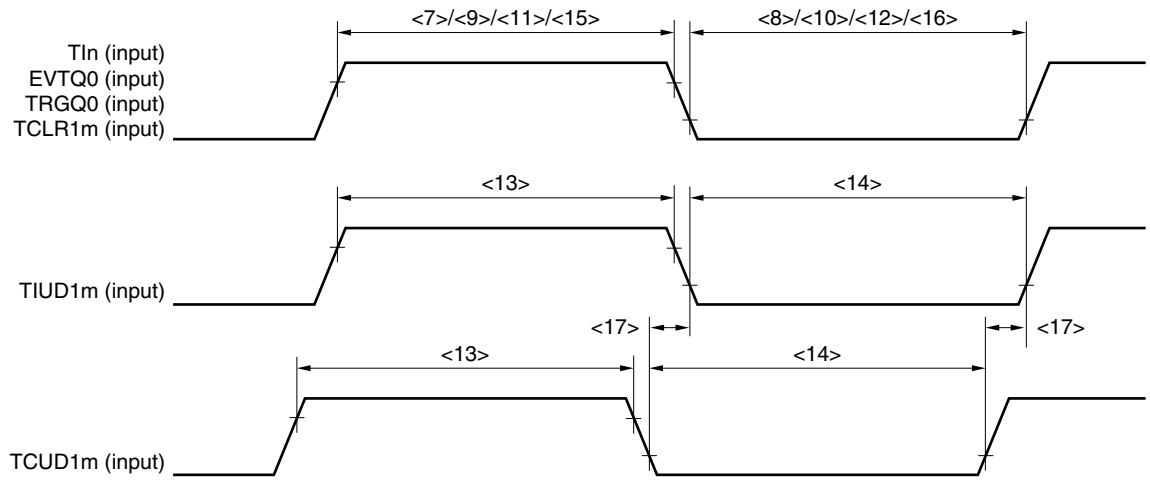
Notes 1. T = 1/fxx

2. T_{smp}: Noise elimination sampling clock cycle (set by NRC1m register)

Remarks 1. n = Q00, Q01, Q02, Q03, P00, P01, P20, P21
m = 0, 1

2. The above specification shows a pulse width that is accurately detected as a valid edge. Even if a pulse narrower than the above specification is input, therefore, it may be detected as a valid edge.

Timer Input Timing



Remark n = Q00, Q01, Q02, Q03, P00, P01, P20, P21
m = 0, 1

CSIB Timing

(1) Master mode

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{SCKBn}}$ cycle	t_{KCYM}	<18>		125		ns
$\overline{\text{SCKBn}}$ high-/low-level width	t_{KWHM}, t_{KWLM}	<19>		t_{KCYM}/2 – 10		ns
SIBn setup time (to $\overline{\text{SCKBn}}$↑)	t_{SSIM}	<20>		30		ns
SIBn hold time (from $\overline{\text{SCKBn}}$↑)	t_{HSIM}	<21>		30		ns
SOBn output delay time (from $\overline{\text{SCKBn}}$↓)	t_{DSOM}	<22>			30	ns
SOBn output delay time (from $\overline{\text{SCKBn}}$↑)					30	ns
<R> SOBn output hold time (from $\overline{\text{SCKBn}}$↑)	t_{HSOM}	<23>		t_{KCYM}/2 – 10		ns
<R> SOBn output hold time (from $\overline{\text{SCKBn}}$↓)				t_{KCYM}/2 – 10		ns

Remark n = 0, 1

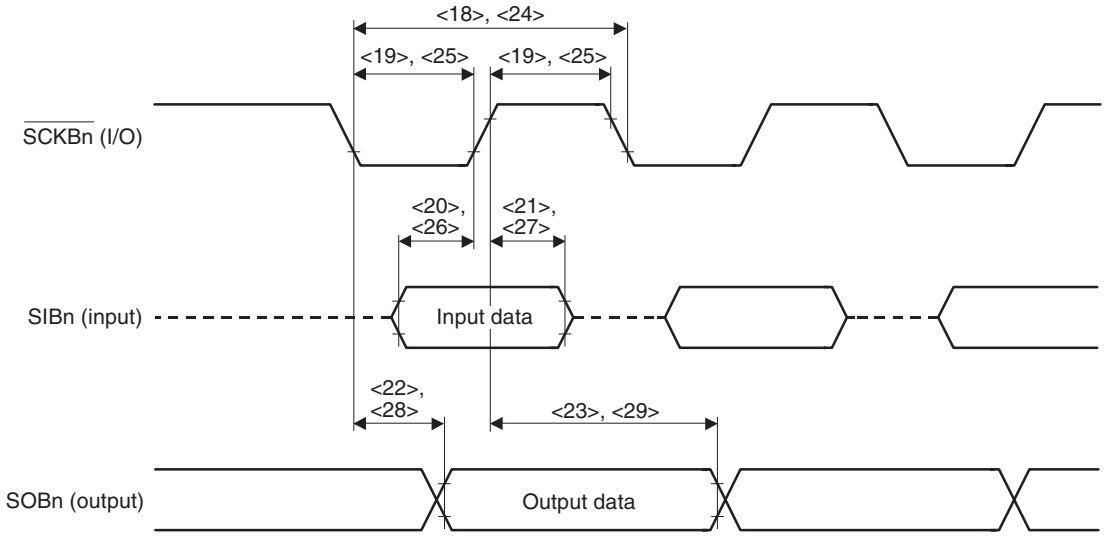
(2) Slave mode

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol		Conditions	MIN.	MAX.	Unit
$\overline{\text{SCKBn}}$ cycle	t_{KCYS}	<24>		125		ns
$\overline{\text{SCKBn}}$ high-/low-level width	t_{KWHS}, t_{KWLS}	<25>		t_{KCYS}/2 – 10		ns
SIBn setup time (to $\overline{\text{SCKBn}}$↑)	t_{SSIS}	<26>		30		ns
SIBn hold time (from $\overline{\text{SCKBn}}$↑)	t_{HSIS}	<27>		30		ns
SOBn output delay time (from $\overline{\text{SCKBn}}$↓)	t_{DSOS}	<28>			30	ns
SOBn output delay time (from $\overline{\text{SCKBn}}$↑)					30	ns
<R> SOBn output hold time (from $\overline{\text{SCKBn}}$↑)	t_{HSOS}	<29>		t_{KCYS}/2 – 10		ns
<R> SOBn output hold time (from $\overline{\text{SCKBn}}$↓)				t_{KCYS}/2 – 10		ns

Remark n = 0, 1

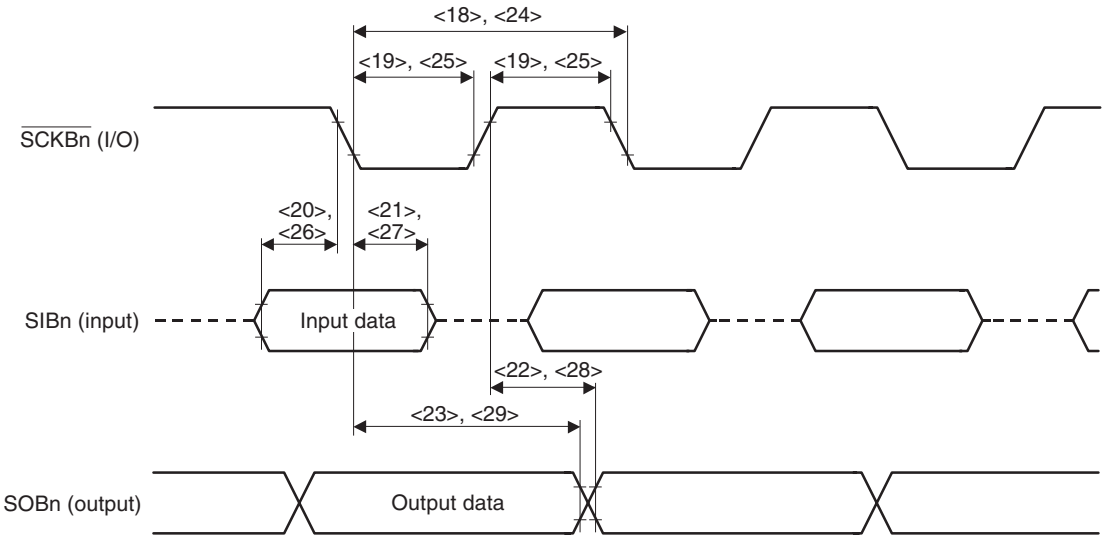
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 00



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

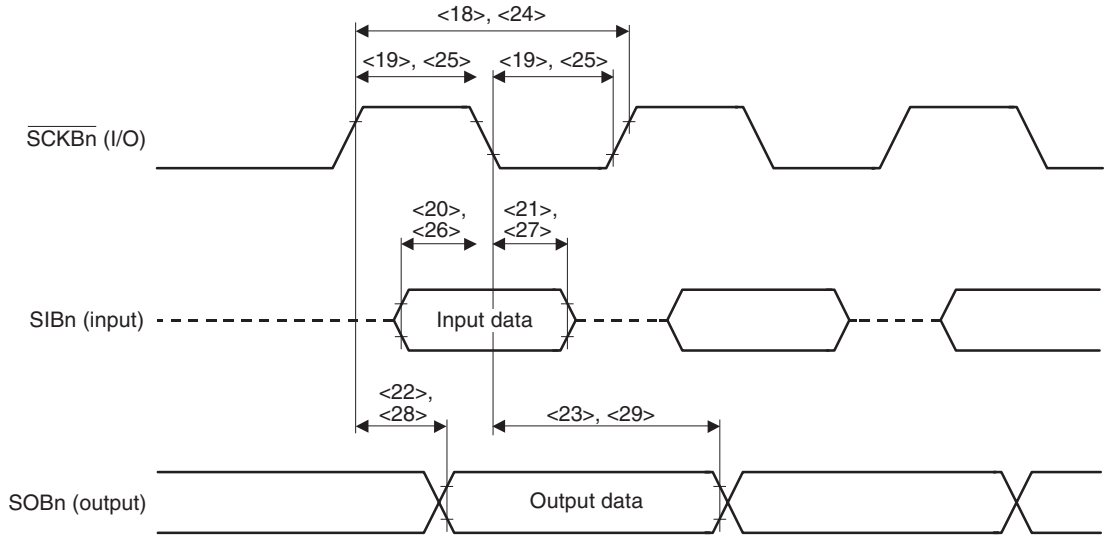
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 01



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

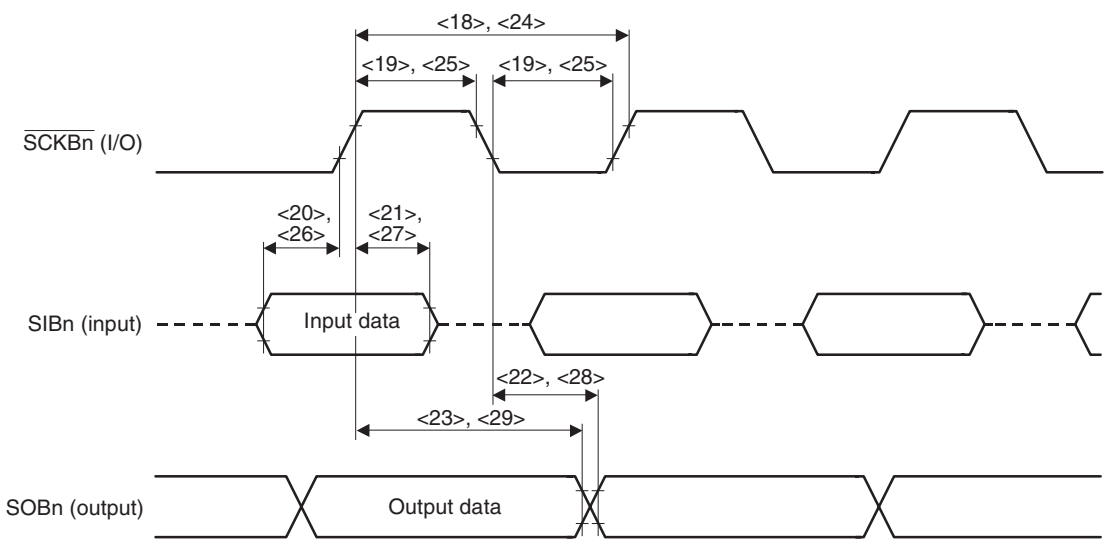
CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 10



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

CSIB timing when CBnCKP and CBnDAP bits of CBnCTL1 register = 11



Remarks 1. Broken lines indicate high impedance.

2. n = 0, 1

High-Impedance Control Timing

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Oscillation stop → timer output high impedance	t_{CLM}	When clock monitor is operating		65	μs
Input to TOQnOFF → timer output high impedance	t_{HTQn}			300	ns
Input to TOPmOFF → timer output high impedance	t_{HTPm}			300	ns
Input to ANI00 to ANI03 → timer output high impedance	t_{ANI0}			10	μs
Input to ANI10 to ANI12 → timer output high impedance	t_{ANI1}			10	μs

Remark n = 0, 1
m = 2, 3

Characteristics of A/D Converters 0, 1

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			10	10	10	bit
Overall error[Note 1]					±4.0	LSB
Conversion time	t_{CONV}	f_{XX} = 64 MHz, ADAnM1 register = 01H	1.94			μs
		f_{XX} = 32 MHz, ADAnM1 register = 03H			7.75	μs
Zero scale error[Note 1]					±4.0	LSB
Full-scale error[Note 1]					±4.0	LSB
Integral linearity error[Note 1]					±4.0	LSB
Differential linearity error[Note 1]					±2.0	LSB
Analog reference voltage	AV_{DD}		4.5		5.5	V
Analog input voltage	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} supply current[Note 2]	AI_{DD}	During operation		5	10	mA
	AI_{DDS}	In STOP mode[Note 3]		1	10	μA

Notes 1. Excludes quantization error (±0.5 LSB).

2. The value for either of A/D converter 0 or A/D converter 1.

3. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. LSB: Least Significant Bit

2. n = 0, 1

Characteristics of A/D Converter 2

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Resolution			8		10	Bits
Overall error		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Conversion time	t_{CONV}	10-bit resolution, serial mode		512		μs
		10-bit resolution, parallel mode		128[Note 1]		μs
		8-bit resolution, serial mode		128		μs
		8-bit resolution, parallel mode		32[Note 1]		μs
Zero scale error[Note 2]		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Full-scale error[Note 2]		10-bit resolution			±9.5	LSB
		8-bit resolution			±3.5	LSB
Integral linearity error[Note 2]		10-bit resolution			±2.0	LSB
		8-bit resolution			±1.5	LSB
Differential linearity error[Note 2]		10-bit resolution			±2.0	LSB
		8-bit resolution			±1.5	LSB
Analog reference voltage	AV_{DD}		4.5		5.5	V
Analog input voltage	V_{IAN}		AV_{SS}		AV_{DD}	V
AV_{DD} supply current	AI_{DD}	During operation		1	3	mA
	AI_{DDS}	In STOP mode[Note 3]		1	10	μA

Notes 1. The time taken for the first conversion to end in parallel mode is the same as that in serial mode. The above specification shows the value for the second and subsequent conversions.

2. Excludes quantization error (±0.5 LSB).

3. Stop the operation of A/D converter 2 (ADA2CTL0.ADA2CE bit = 0) before setting STOP mode.

Remark LSB: Least Significant Bit

Operational Amplifier Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input offset voltage	V_{IO}			±5.0		mV
Input voltage range	V_I	Gain = 2.5	0.04AV_{DD}		0.32AV_{DD}	V
		Gain = 5	0.03AV_{DD}		0.16AV_{DD}	V
Slew rate	S_R			10		V/μs
Gain error		Gain = 2.5		±1.0	±5.0	%
		Gain = 5		±1.0	±5.0	%
Operating current[Note 1]	I_{OPDD}	During operation		1.0	3.0	mA
	A_{IDDS}	In STOP mode[Note 2]		1.0	10	μA

Notes 1. Six operational amplifiers are provided in total. The value shows the operating current per operational amplifier.

2. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. The operating current of the operational amplifiers is included in AV_{DD}.

2. n = 0, 1

Comparator Characteristics

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.5 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Input offset voltage	V_{IO}			±3.0		mV
Input voltage range	V_I	CMPREF, ANIm	0.1AV_{DD}		0.5AV_{DD}	V
Response time	t_{CR}	Input amplitude = 100 mV, at rising edge[Note 1]		4.0		μs
	t_{CF}	Input amplitude = 100 mV, at falling edge[Note 2]		2.0		μs
Operating current[Note 3]	I_{CPDD}	During operation		50	150	μA
	A_{IDDS}	In STOP mode[Note 4]		1.0	10	μA

Notes 1. Characteristics of pulse response when ANIm input changes from CMPREF – 100 mV to CMPREF + 100 mV

2. Characteristics of pulse response when ANIm input changes from CMPREF + 100 mV to CMPREF – 100 mV

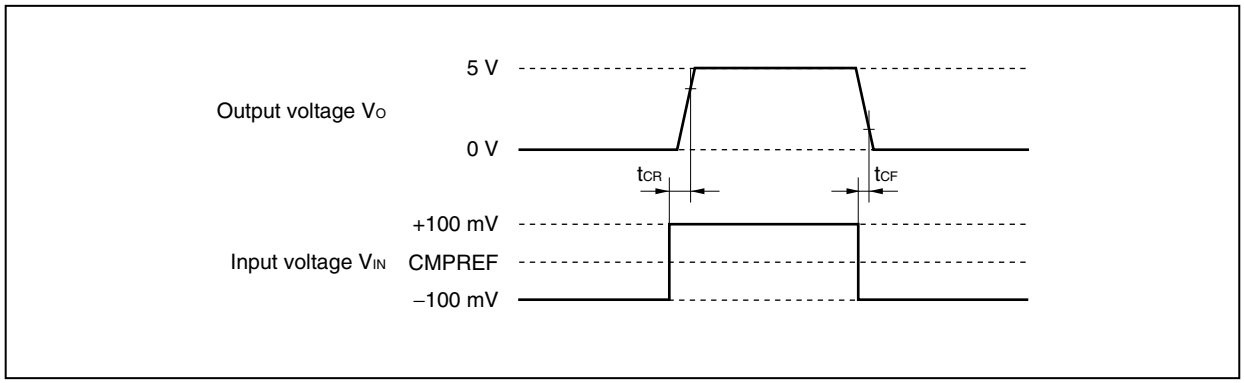
3. Six comparators are provided in total. The value shows the operating current per comparator.

4. Stop the operation of A/D converters 0 and 1 (ADAnM0.ADAnCE bit = 0) before setting STOP mode.

Remarks 1. The operating current of the comparators is included in AV_{DD}.

2. m = 00 to 02, 10 to 12
n = 0, 1

Comparator Characteristics



Supply Voltage Application/Cutoff Timing

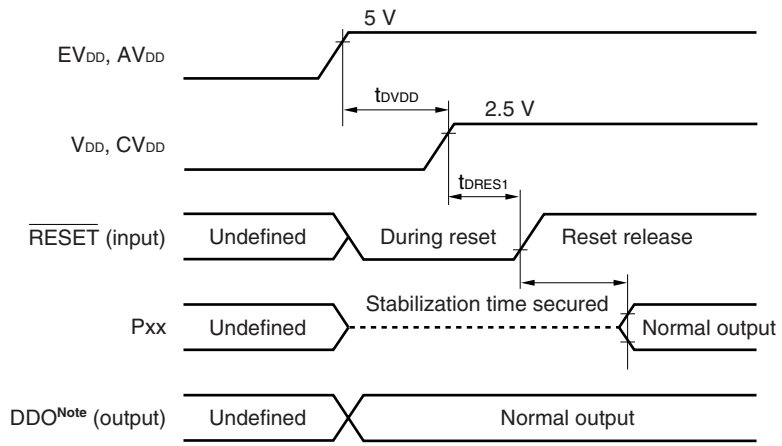
(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	MAX.	Unit
Delay time from EV_{DD}, AV_{DD} rise to V_{DD}, CV_{DD} rise	t_{DVDD}		–50	50	ms
Delay time from V_{DD}, CV_{DD} rise to $\overline{RESET}$↑	t_{DRES1}		T_{osc} – 1		ms
Delay time from EV_{DD}, AV_{DD} rise to $\overline{RESET}$↑	t_{DRES2}		T_{osc} – 1		ms
Delay time from EV_{DD}, AV_{DD} fall to V_{DD}, CV_{DD} fall	t_{DEVDD}		0		ns

Remark T_{osc}: Oscillation stabilization time of oscillator (varies depending on the resonator or oscillator used)

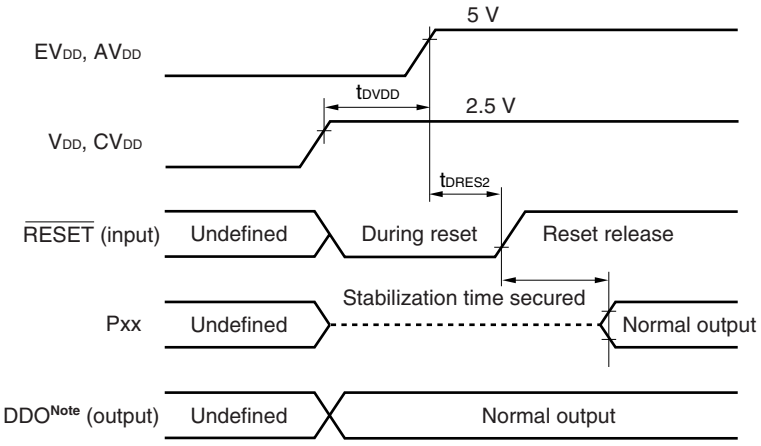
Supply Voltage Application Timing

(a) Power supply sequence recommended condition 1



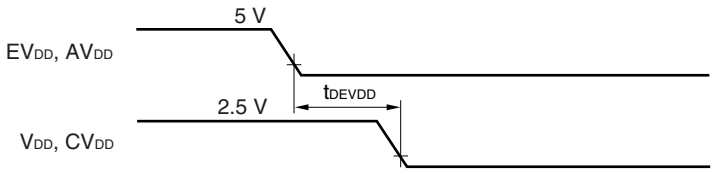
Note μPD70F3186 only

(b) Power supply sequence recommended condition 2



Note μPD70F3186 only

Supply Voltage Cutoff Timing



Cautions 1. There are no regulations for the voltage level and time of 2.5 V V_{DD} and CV_{DD}, and 5 V EV_{DD} and AV_{DD} in the process of natural discharge after power supply cutoff.

2. The application of only one of power supplies V_{DD}, CV_{DD}, EV_{DD}, or AV_{DD}, and only either the 2.5 V V_{DD} and CV_{DD} power supplies or the 5 V EV_{DD} and AV_{DD} power supplies is prohibited.

Flash Memory Programming Characteristics (μPD70F3186 only)

(T_A = –40 to +85°C, V_{DD} = CV_{DD} = 2.3 to 2.7 V, AV_{DD} = EV_{DD} = 4.0 to 5.5 V, V_{SS} = AV_{SS} = CV_{SS} = EV_{SS} = 0 V, C_L = 50 pF)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Write count	C_{ERWR}	**Note**		100		Times
Write current				90	120	mA
Erase current				90	120	mA

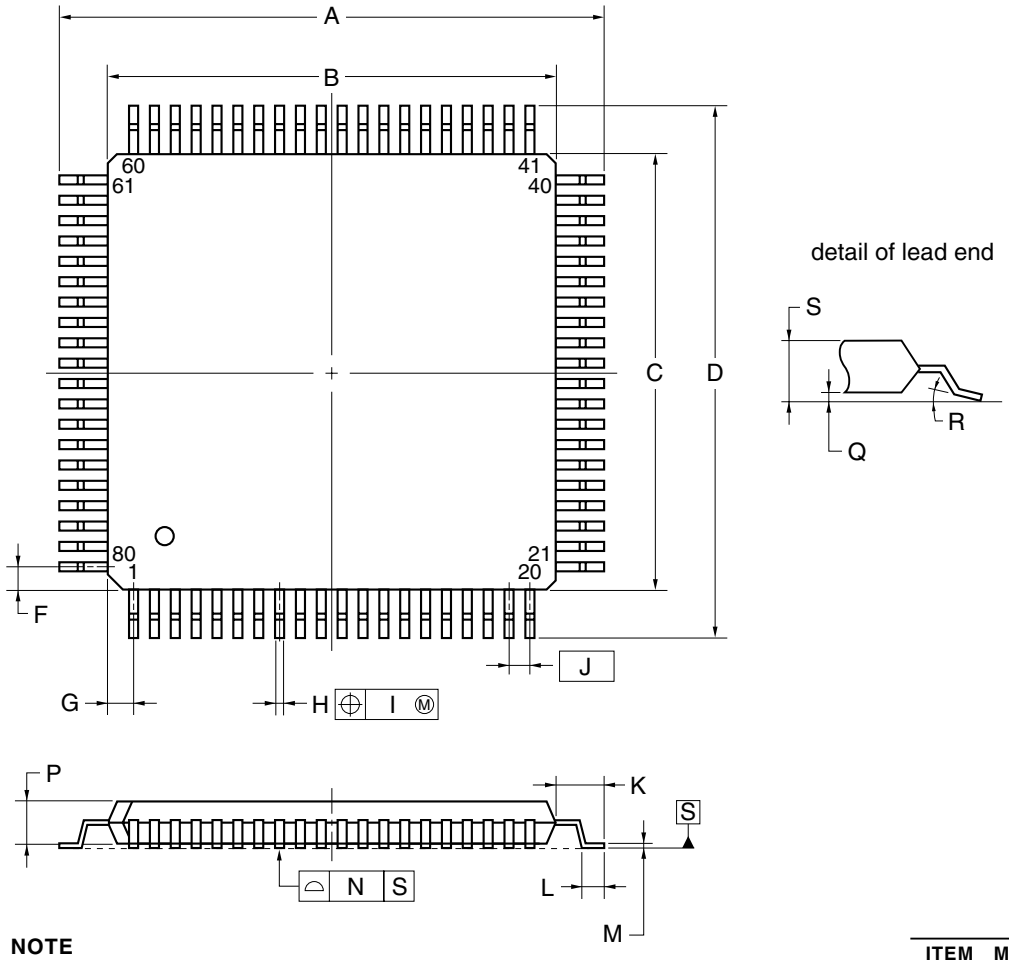
Note When writing initially to shipped products, it is counted as one rewrite for both “erase to write” and “write only”.

Example (P: Write, E: Erase)

Shipped product ⟶ P→ E→ P→ E→ P: 3 rewrites

Shipped product → E→ P→ E→ P→ E→ P: 3 rewrites

80-PIN PLASTIC QFP (14x14)



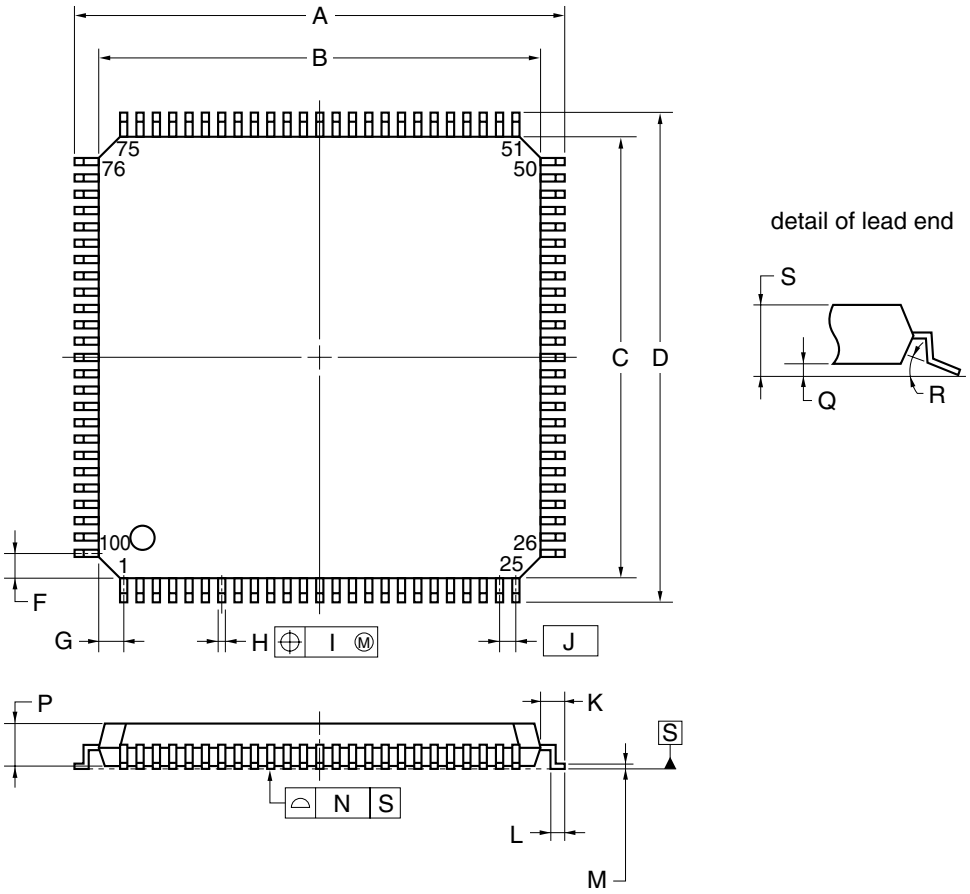
NOTE

Each lead centerline is located within 0.13 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	17.20±0.20
B	14.00±0.20
C	14.00±0.20
D	17.20±0.20
F	0.825
G	0.825
H	0.32±0.06
I	0.13
J	0.65 (T.P.)
K	1.60±0.20
L	0.80±0.20
M	$0.17^{+0.03}_{-0.07}$
N	0.10
P	1.40±0.10
Q	0.125±0.075
R	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$
S	1.70 MAX.

P80GC-65-8BT-1

100-PIN PLASTIC LQFP (FINE PITCH) (14x14)



detail of lead end

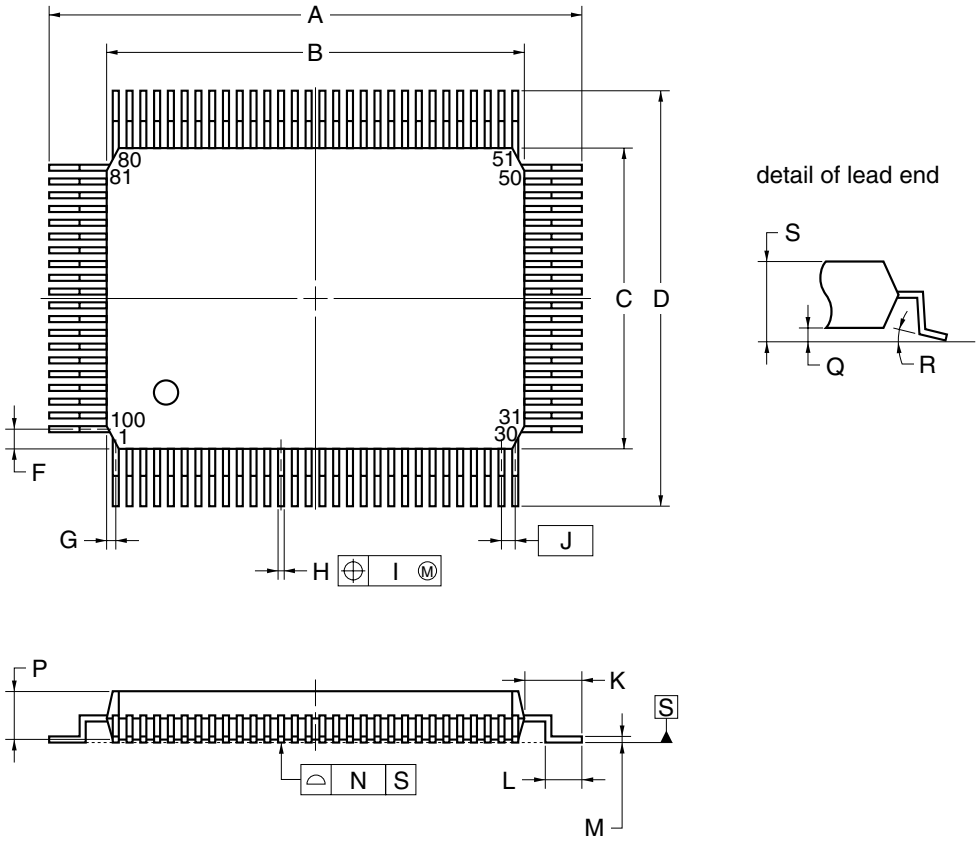
NOTE

Each lead centerline is located within 0.08 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	16.00±0.20
B	14.00±0.20
C	14.00±0.20
D	16.00±0.20
F	1.00
G	1.00
H	$0.22^{+0.05}_{-0.04}$
I	0.08
J	0.50 (T.P.)
K	1.00±0.20
L	0.50±0.20
M	$0.17^{+0.03}_{-0.07}$
N	0.08
P	1.40±0.05
Q	0.10±0.05
R	$3^{\circ}{}^{+7^{\circ}}_{-3^{\circ}}$
S	1.60 MAX.

S100GC-50-8EU, 8EA-2

100-PIN PLASTIC QFP (14x20)



NOTE

Each lead centerline is located within 0.15 mm of its true position (T.P.) at maximum material condition.

ITEM	MILLIMETERS
A	23.6±0.4
B	20.0±0.2
C	14.0±0.2
D	17.6±0.4
F	0.8
G	0.6
H	0.30±0.10
I	0.15
J	0.65 (T.P.)
K	1.8±0.2
L	0.8±0.2
M	$0.15^{+0.10}_{-0.05}$
N	0.10
P	2.7±0.1
Q	0.1±0.1
R	5°±5°
S	3.0 MAX.

P100GF-65-3BA1-4

CHAPTER 26 RECOMMENDED SOLDERING CONDITIONS

These products should be soldered and mounted under the following recommended conditions.
For technical information, see the following website.

Semiconductor Device Mount Manual (http://www.necel.com/pkg/en/mount/index.html)

Table 26-1. Surface Mounting Type Soldering Conditions

μPD703183GC-×××-8BT-A: 80-pin plastic QFP (14 × 14)
μPD70F3184GC-8BT-A: 80-pin plastic QFP (14 × 14)
μPD703185GC-×××-8EU-A: 100-pin plastic LQFP (fine pitch) (14 × 14)
μPD703185GF-×××-3BA-A: 100-pin plastic QFP (14 × 20)
μPD703186GC-×××-8EU-A: 100-pin plastic LQFP (fine pitch) (14 × 14)
μPD703186GF-×××-3BA-A: 100-pin plastic QFP (14 × 20)
μPD70F3186GC-8EU-A: 100-pin plastic LQFP (fine pitch) (14 × 14)
μPD70F3186GF-3BA-A: 100-pin plastic QFP (14 × 20)

Soldering Method	Soldering Conditions	Recommended Condition Symbol
Infrared reflow	Package peak temperature: 260°C, Time: 60 seconds max. (at 220°C or higher), Count: 3 times or less, Exposure limit: 3 days[Note] (after that, prebake at 125°C for 20 to 72 hours)	IR60-203-3
Wave soldering	For details, contact an NEC Electronics sales representative.	–
Partial heating	Pin temperature: 350°C max., Time: 3 seconds max. (per pin row)	–

Note After opening the dry pack, store it at 25°C or less and 65% RH or less for the allowable storage period.

Remarks 1. Products with -A at the end of the part number are lead-free products.

2. For soldering methods and conditions other than those recommended, please contact an NEC Electronics sales representative.

APPENDIX A CAUTIONS

A.1 Restriction on Conflict Between sld Instruction and Interrupt Request

A.1.1 Description

If a conflict occurs between the decode operation of an instruction in <2> immediately before the sld instruction following an instruction in <1> and an interrupt request before the instruction in <1> is complete, the execution result of the instruction in <1> may not be stored in a register.

Instruction <1>

- ld instruction: ld.b, ld.h, ld.w, ld.bu, ld.hu
- sld instruction: sld.b, sld.h, sld.w, sld.bu, sld.hu
- Multiplication instruction: mul, mulh, mulhi, mulu

Instruction <2>

mov reg1, reg2	not reg1, reg2	satsubr reg1, reg2	satsub reg1, reg2
satadd reg1, reg2	satadd imm5, reg2	or reg1, reg2	xor reg1, reg2
and reg1, reg2	tst reg1, reg2	subr reg1, reg2	sub reg1, reg2
add reg1, reg2	add imm5, reg2	cmp reg1, reg2	cmp imm5, reg2
mulh reg1, reg2	shr imm5, reg2	sar imm5, reg2	shl imm5, reg2

<Example>

<i> ld.w [r11], r10
•
•
•
<ii> mov r10, r28
<iii> sld.w 0x28, r10

If the decode operation of the mov instruction <ii> immediately before the sld instruction <iii> and an interrupt request conflict before execution of the ld instruction <i> is complete, the execution result of instruction <i> may not be stored in a register.

A.1.2 Countermeasure

(1) When compiler (CA850) is used

Use CA850 Ver. 2.61 or later because generation of the corresponding instruction sequence can be automatically suppressed.

(2) For assembler

When executing the sld instruction immediately after instruction <ii>, avoid the above operation using either of the following methods.

- Insert a nop instruction immediately before the sld instruction.
- Do not use the same register as the sld instruction destination register in the above instruction <ii> executed immediately before the sld instruction.

APPENDIX B REGISTER INDEX

(1/9)

Symbol	Name	Unit	Page
AD0IC	Interrupt control register	INTC	684
AD1IC	Interrupt control register	INTC	684
AD2IC	Interrupt control register	INTC	684
ADA0CR0	A/D0 conversion result register 0	ADC0	506
ADA0CR0H	A/D0 conversion result register 0H	ADC0	506
ADA0CR1	A/D0 conversion result register 1	ADC0	506
ADA0CR1H	A/D0 conversion result register 1H	ADC0	506
ADA0CR2	A/D0 conversion result register 2	ADC0	506
ADA0CR2H	A/D0 conversion result register 2H	ADC0	506
ADA0CR3	A/D0 conversion result register 3	ADC0	506
ADA0CR3H	A/D0 conversion result register 3H	ADC0	506
ADA0CR4	A/D0 conversion result register 4	ADC0	507
ADA0CR4H	A/D0 conversion result register 4H	ADC0	507
ADA0CR5	A/D0 conversion result register 5	ADC0	507
ADA0CR5H	A/D0 conversion result register 5H	ADC0	507
ADA0CR6	A/D0 conversion result register 6	ADC0	507
ADA0CR6H	A/D0 conversion result register 6H	ADC0	507
ADA0CR7	A/D0 conversion result register 7	ADC0	507
ADA0CR7H	A/D0 conversion result register 7H	ADC0	507
ADA0M0	A/D converter 0 mode register 0	ADC0	498
ADA0M1	A/D converter 0 mode register 1	ADC0	500
ADA0M2	A/D converter 0 mode register 2	ADC0	502
ADA0S	A/D converter 0 channel specification register	ADC0	501
ADA1CR0	A/D1 conversion result register 0	ADC1	506
ADA1CR0H	A/D1 conversion result register 0H	ADC1	506
ADA1CR1	A/D1 conversion result register 1	ADC1	506
ADA1CR1H	A/D1 conversion result register 1H	ADC1	506
ADA1CR2	A/D1 conversion result register 2	ADC1	506
ADA1CR2H	A/D1 conversion result register 2H	ADC1	506
ADA1CR3	A/D1 conversion result register 3	ADC1	506
ADA1CR3H	A/D1 conversion result register 3H	ADC1	506
ADA1CR4	A/D1 conversion result register 4	ADC1	507
ADA1CR4H	A/D1 conversion result register 4H	ADC1	507
ADA1CR5	A/D1 conversion result register 5	ADC1	507
ADA1CR5H	A/D1 conversion result register 5H	ADC1	507
ADA1CR6	A/D1 conversion result register 6	ADC1	507
ADA1CR6H	A/D1 conversion result register 6H	ADC1	507
ADA1CR7	A/D1 conversion result register 7	ADC1	507
ADA1CR7H	A/D1 conversion result register 7H	ADC1	507
ADA1M0	A/D converter 1 mode register 0	ADC1	498
ADA1M1	A/D converter 1 mode register 1	ADC1	500

(2/9)

Symbol	Name	Unit	Page
ADA1M2	A/D converter 1 mode register 2	ADC1	502
ADA1S	A/D converter 1 channel specification register	ADC1	501
ADA2CR0	A/D2 conversion result register 0	ADC2	558
ADA2CR0H	A/D2 conversion result register 0H	ADC2	558
ADA2CR1	A/D2 conversion result register 1	ADC2	558
ADA2CR1H	A/D2 conversion result register 1H	ADC2	558
ADA2CR2	A/D2 conversion result register 2	ADC2	558
ADA2CR2H	A/D2 conversion result register 2H	ADC2	558
ADA2CR3	A/D2 conversion result register 3	ADC2	558
ADA2CR3H	A/D2 conversion result register 3H	ADC2	558
ADA2CR4	A/D2 conversion result register 4	ADC2	558
ADA2CR4H	A/D2 conversion result register 4H	ADC2	558
ADA2CR5	A/D2 conversion result register 5	ADC2	558
ADA2CR5H	A/D2 conversion result register 5H	ADC2	558
ADA2CR6	A/D2 conversion result register 6	ADC2	558
ADA2CR6H	A/D2 conversion result register 6H	ADC2	558
ADA2CR7	A/D2 conversion result register 7	ADC2	558
ADA2CR7H	A/D2 conversion result register 7H	ADC2	558
ADA2CTL0	A/D converter 2 control register 0	ADC2	552
ADA2CTL1	A/D converter 2 control register 1	ADC2	553
ADA2CTL2	A/D converter 2 control register 2	ADC2	554
ADA2CTL3	A/D converter 2 control register 3	ADC2	555
ADA2STR	A/D converter 2 status register	ADC2	557
CB0CTL0	CSIB0 control register 0	CSIB0	602
CB0CTL1	CSIB0 control register 1	CSIB0	605
CB0CTL2	CSIB0 control register 2	CSIB0	606
CB0REIC	Interrupt control register	INTC	684
CB0RIC	Interrupt control register	INTC	684
CB0RX	CSIB0 receive data register	CSIB0	601
CB0RXL	CSIB0 receive data register L	CSIB0	601
CB0STR	CSIB0 status register	CSIB0	608
CB0TIC	Interrupt control register	INTC	684
CB0TX	CSIB0 transmit data register	CSIB0	601
CB0TXL	CSIB0 transmit data register L	CSIB0	601
CB1CTL0	CSIB1 control register 0	CSIB1	602
CB1CTL1	CSIB1 control register 1	CSIB1	605
CB1CTL2	CSIB1 control register 2	CSIB1	606
CB1REIC	Interrupt control register	INTC	684
CB1RIC	Interrupt control register	INTC	684
CB1RX	CSIB1 receive data register	CSIB1	601
CB1RXL	CSIB1 receive data register L	CSIB1	601
CB1STR	CSIB1 status register	CSIB1	608
CB1TIC	Interrupt control register	INTC	684
CB1TX	CSIB1 transmit data register	CSIB1	601
CB1TXL	CSIB1 transmit data register L	CSIB1	601

(3/9)

Symbol	Name	Unit	Page
CC0IC0	Interrupt control register	INTC	684
CC0IC1	Interrupt control register	INTC	684
CC100	Capture/compare register 100	Timer	401
CC101	Capture/compare register 101	Timer	402
CC110	Capture/compare register 110	Timer	401
CC111	Capture/compare register 111	Timer	402
CC1IC0	Interrupt control register	INTC	684
CC1IC1	Interrupt control register	INTC	684
CCR10	Capture/compare control register 10	Timer	394
CCR11	Capture/compare control register 11	Timer	394
CLM	Clock monitor mode register	CG	167
CM0IC0	Interrupt control register	INTC	684
CM0IC1	Interrupt control register	INTC	684
CM100	Compare register 100	Timer	399
CM101	Compare register 101	Timer	400
CM110	Compare register 110	Timer	399
CM111	Compare register 111	Timer	400
CM1IC0	Interrupt control register	INTC	684
CM1IC1	Interrupt control register	INTC	684
CMPIC0	Interrupt control register	INTC	684
CMPIC1	Interrupt control register	INTC	684
CORAD0	Correction address register 0	CPU	723
CORAD0H	Correction address register 0H	CPU	723
CORAD0L	Correction address register 0L	CPU	723
CORAD1	Correction address register 1	CPU	723
CORAD1H	Correction address register 1H	CPU	723
CORAD1L	Correction address register 1L	CPU	723
CORAD2	Correction address register 2	CPU	723
CORAD2H	Correction address register 2H	CPU	723
CORAD2L	Correction address register 2L	CPU	723
CORAD3	Correction address register 3	CPU	723
CORAD3H	Correction address register 3H	CPU	723
CORAD3L	Correction address register 3L	CPU	723
CORCN	Correction control register	CPU	724
CSL10	CC101 capture input select register	Timer	398
CSL11	CC111 capture input select register	Timer	398
DADC0	DMA addressing control register 0	DMAC	650
DADC1	DMA addressing control register 1	DMAC	650
DADC2	DMA addressing control register 2	DMAC	650
DADC3	DMA addressing control register 3	DMAC	650
DBC0	DMA transfer count register 0	DMAC	649
DBC1	DMA transfer count register 1	DMAC	649
DBC2	DMA transfer count register 2	DMAC	649
DBC3	DMA transfer count register 3	DMAC	649
DCHC0	DMA channel control register 0	DMAC	652

(4/9)

Symbol	Name	Unit	Page
DCHC1	DMA channel control register 1	DMAC	652
DCHC2	DMA channel control register 2	DMAC	652
DCHC3	DMA channel control register 3	DMAC	652
DDA0H	DMA destination address register 0H	DMAC	647
DDA0L	DMA destination address register 0L	DMAC	648
DDA1H	DMA destination address register 1H	DMAC	647
DDA1L	DMA destination address register 1L	DMAC	648
DDA2H	DMA destination address register 2H	DMAC	647
DDA2L	DMA destination address register 2L	DMAC	648
DDA3H	DMA destination address register 3H	DMAC	647
DDA3L	DMA destination address register 3L	DMAC	648
DMAIC0	Interrupt control register	INTC	684
DMAIC1	Interrupt control register	INTC	684
DMAIC2	Interrupt control register	INTC	684
DMAIC3	Interrupt control register	INTC	684
DSA0H	DMA source address register 0H	DMAC	645
DSA0L	DMA source address register 0L	DMAC	646
DSA1H	DMA source address register 1H	DMAC	645
DSA1L	DMA source address register 1L	DMAC	646
DSA2H	DMA source address register 2H	DMAC	645
DSA2L	DMA source address register 2L	DMAC	646
DSA3H	DMA source address register 3H	DMAC	645
DSA3L	DMA source address register 3L	DMAC	646
DTFR0	DMA trigger factor register 0	DMAC	654
DTFR1	DMA trigger factor register 1	DMAC	654
DTFR2	DMA trigger factor register 2	DMAC	654
DTFR3	DMA trigger factor register 3	DMAC	654
HZA0CTL0	High-impedance output control register 00	Timer	435
HZA0CTL1	High-impedance output control register 01	Timer	435
HZA1CTL0	High-impedance output control register 10	Timer	435
HZA1CTL1	High-impedance output control register 11	Timer	435
HZA2CTL0	High-impedance output control register 20	Timer	435
HZA2CTL1	High-impedance output control register 21	Timer	435
IMR0	Interrupt mask register 0	INTC	688
IMR0H	Interrupt mask register 0H	INTC	688
IMR0L	Interrupt mask register 0L	INTC	688
IMR1	Interrupt mask register 1	INTC	688
IMR1H	Interrupt mask register 1H	INTC	688
IMR1L	Interrupt mask register 1L	INTC	688
IMR2	Interrupt mask register 2	INTC	688
IMR2H	Interrupt mask register 2H	INTC	688
IMR2L	Interrupt mask register 2L	INTC	688
IMR3	Interrupt mask register 3	INTC	688
IMR3H	Interrupt mask register 3H	INTC	688

(5/9)

Symbol	Name	Unit	Page
IMR3L	Interrupt mask register 3L	INTC	688
IMS	Internal memory size switching register	CPU	63
INTF0	External interrupt falling edge specification register 0	INTC	694
INTPNRC	External interrupt noise elimination control register	INTC	155, 692
INTR0	External interrupt rising edge specification register 0	INTC	694
ISPR	In-service priority register	INTC	690
NRC10	Noise elimination time select register 10	Timer	156, 398
NRC11	Noise elimination time select register 11	Timer	156, 398
OP0CTL0	Operational amplifier 0 control register 0	ADC0	504
OP0CTL1	Operational amplifier 0 control register 1	ADC0	505
OP1CTL0	Operational amplifier 1 control register 0	ADC0	504
OP1CTL1	Operational amplifier 1 control register 1	ADC0	505
OSTS	Oscillation stabilization time select register	CG	166
P0	Port 0 register	Port	91
P1	Port 1 register	Port	97
P2	Port 2 register	Port	106
P3	Port 3 register	Port	112
P4	Port 4 register	Port	122
P5	Port 5 register	Port	129
P7	Port 7 register	Port	135
PCC	Processor clock control register	CG	163
PDL	Port DL register	Port	138
PDLH	Port DL register H	Port	138
PDLL	Port DL register L	Port	138
PFC1	Port 1 function control register	Port	99
PFC3	Port 3 function control register	Port	114
PFC4	Port 4 function control register	Port	124
PFC5	Port 5 function control register	Port	130
PFCE1	Port 1 function control expansion register	Port	99
PIC0	Interrupt control register	INTC	684
PIC1	Interrupt control register	INTC	684
PIC2	Interrupt control register	INTC	684
PIC3	Interrupt control register	INTC	684
PIC4	Interrupt control register	INTC	684
PIC5	Interrupt control register	INTC	684
PIC6	Interrupt control register	INTC	684
PIC7	Interrupt control register	INTC	684
PLLCTL	PLL control register	CG	162
PM0	Port 0 mode register	Port	91
PM1	Port 1 mode register	Port	97
PM2	Port 2 mode register	Port	106
PM3	Port 3 mode register	Port	112
PM4	Port 4 mode register	Port	123

(6/9)

Symbol	Name	Unit	Page
PM5	Port 5 mode register	Port	130
PMC0	Port 0 mode control register	Port	92
PMC1	Port 1 mode control register	Port	98
PMC2	Port 2 mode control register	Port	107
PMC3	Port 3 mode control register	Port	113
PMC4	Port 4 mode control register	Port	123
PMC5	Port 5 mode control register	Port	130
PMC7	Port 7 mode control register	Port	135
PMDL	Port DL mode register	Port	138
PMDLH	Port DL mode register H	Port	138
PMDLL	Port DL mode register L	Port	138
PRCMD	Command register	CPU	78
PRM10	Prescaler mode register 10	Timer	396
PRM11	Prescaler mode register 11	Timer	396
PSC	Power save control register	CPU	164, 708
PSMR	Power save mode register	CPU	165, 709
PU0	Pull-up resistor option register 0	Port	93
PU1	Pull-up resistor option register 1	Port	101
PU2	Pull-up resistor option register 2	Port	108
PU3	Pull-up resistor option register 3	Port	114
PU4	Pull-up resistor option register 4	Port	124
PU5	Pull-up resistor option register 5	Port	131
PUDL	Pull-up resistor option register DL	Port	139
PUDLH	Pull-up resistor option register DLH	Port	139
PUDLL	Pull-up resistor option register DLL	Port	139
RESF	Reset source flag register	Reset	718
SESA10	Valid edge select register 10	Timer	394
SESA11	Valid edge select register 11	Timer	394
STATUS10	Status register 10	Timer	397
STATUS11	Status register 11	Timer	397
SYS	System status register	CPU	79
TM0CMP0	TMM0 compare register 0	Timer	415
TM0CTL0	TMM0 control register 0	Timer	416
TM0EQIC0	Interrupt control register	INTC	684
TMC10	Timer control register 10	Timer	392
TMC11	Timer control register 11	Timer	392
TMENC10	Timer ENC10	Timer	389
TMENC11	Timer ENC11	Timer	389
TP0CCIC0	Interrupt control register	INTC	684
TP0CCIC1	Interrupt control register	INTC	684
TP0CCR0	TMP0 capture/compare register 0	Timer	188
TP0CCR1	TMP0 capture/compare register 1	Timer	190
TP0CNT	TMP0 counter read buffer register	Timer	192
TP0CTL0	TMP0 control register 0	Timer	180

(7/9)

Symbol	Name	Unit	Page
TP0CTL1	TMP0 control register 1	Timer	181
TP0IOC0	TMP0 I/O control register 0	Timer	182
TP0IOC1	TMP0 I/O control register 1	Timer	185
TP0IOC2	TMP0 I/O control register 2	Timer	186
TP0OPT0	TMP0 option register 0	Timer	187
TP0OVIC	Interrupt control register	INTC	684
TP1CCIC0	Interrupt control register	INTC	684
TP1CCIC1	Interrupt control register	INTC	684
TP1CCR0	TMP1 capture/compare register 0	Timer	188
TP1CCR1	TMP1 capture/compare register 1	Timer	190
TP1CNT	TMP1 counter read buffer register	Timer	192
TP1CTL0	TMP1 control register 0	Timer	180
TP1CTL1	TMP1 control register 1	Timer	181
TP1OPT0	TMP1 option register 0	Timer	187
TP1OVIC	Interrupt control register	INTC	684
TP2CCIC0	Interrupt control register	INTC	684
TP2CCIC1	Interrupt control register	INTC	684
TP2CCR0	TMP2 capture/compare register 0	Timer	188
TP2CCR1	TMP2 capture/compare register 1	Timer	190
TP2CNT	TMP2 counter read buffer register	Timer	192
TP2CTL0	TMP2 control register 0	Timer	180
TP2CTL1	TMP2 control register 1	Timer	181
TP2IOC0	TMP2 I/O control register 0	Timer	182
TP2IOC1	TMP2 I/O control register 1	Timer	185
TP2IOC2	TMP2 I/O control register 2	Timer	186
TP2OPT0	TMP2 option register 0	Timer	187
TP2OVIC	Interrupt control register	INTC	684
TP3CCIC0	Interrupt control register	INTC	684
TP3CCIC1	Interrupt control register	INTC	684
TP3CCR0	TMP3 capture/compare register 0	Timer	188
TP3CCR1	TMP3 capture/compare register 1	Timer	190
TP3CNT	TMP3 counter read buffer register	Timer	192
TP3CTL0	TMP3 control register 0	Timer	180
TP3CTL1	TMP3 control register 1	Timer	181
TP3IOC0	TMP3 I/O control register 0	Timer	182
TP3OPT0	TMP3 option register 0	Timer	187
TP3OVIC	Interrupt control register	INTC	684
TQ0CCIC0	Interrupt control register	INTC	684
TQ0CCIC1	Interrupt control register	INTC	684
TQ0CCIC2	Interrupt control register	INTC	684
TQ0CCIC3	Interrupt control register	INTC	684
TQ0CCR0	TMQ0 capture/compare register 0	Timer	286
TQ0CCR1	TMQ0 capture/compare register 1	Timer	288
TQ0CCR2	TMQ0 capture/compare register 2	Timer	290

(8/9)

Symbol	Name	Unit	Page
TQ0CCR3	TMQ0 capture/compare register 3	Timer	292
TQ0CNT	TMQ0 counter read buffer register	Timer	294
TQ0CTL0	TMQ0 control register 0	Timer	279
TQ0CTL1	TMQ0 control register 1	Timer	279
TQ0DTC	TMQ0 dead-time compare register	Timer	426
TQ0IOC0	TMQ0 I/O control register 0	Timer	281
TQ0IOC1	TMQ0 I/O control register 1	Timer	283
TQ0IOC2	TMQ0 I/O control register 2	Timer	284
TQ0OVIC	Interrupt control register	INTC	684
TQ0IOC3	TMQ0 I/O control register 3	Timer	432
TQ0OPT0	TMQ0 option register 0	Timer	285, 427
TQ0OPT1	TMQ0 option register 1	Timer	428
TQ0OPT2	TMQ0 option register 2	Timer	429
TQ0OPT3	TMQ0 option register 3	Timer	431
TQ1CCIC0	Interrupt control register	INTC	684
TQ1CCIC1	Interrupt control register	INTC	684
TQ1CCIC2	Interrupt control register	INTC	684
TQ1CCIC3	Interrupt control register	INTC	684
TQ1CCR0	TMQ1 capture/compare register 0	Timer	286
TQ1CCR1	TMQ1 capture/compare register 1	Timer	288
TQ1CCR2	TMQ1 capture/compare register 2	Timer	290
TQ1CCR3	TMQ1 capture/compare register 3	Timer	292
TQ1CNT	TMQ1 counter read buffer register	Timer	294
TQ1CTL0	TMQ1 control register 0	Timer	279
TQ1CTL1	TMQ1 control register 1	Timer	279
TQ1DTC	TMQ1 dead-time compare register	Timer	426
TQ1IOC0	TMQ1 I/O control register 0	Timer	281
TQ1IOC3	TMQ1 I/O control register 3	Timer	432
TQ1OPT0	TMQ1 option register 0	Timer	285, 427
TQ1OPT1	TMQ1 option register 1	Timer	428
TQ1OPT2	TMQ1 option register 2	Timer	429
TQ1OPT3	TMQ1 option register 3	Timer	431
TQ1OVIC	Interrupt control register	INTC	684
TUM10	Timer unit mode register 10	Timer	391
TUM11	Timer unit mode register 11	Timer	391
UA0CTL0	UARTA0 control register 0	UARTA0	575
UA0CTL1	UARTA0 control register 1	UARTA0	591
UA0CTL2	UARTA0 control register 2	UARTA0	592
UA0OPT0	UARTA0 option control register 0	UARTA0	577
UA0REIC	Interrupt control register	INTC	684
UA0RIC	Interrupt control register	INTC	684
UA0RX	UARTA0 receive data register	UARTA0	579
UA0STR	UARTA0 status register	UARTA0	577
UA0TIC	Interrupt control register	INTC	684

(9/9)

Symbol	Name	Unit	Page
UA0TX	UARTA0 transmit data register	UARTA0	579
UA1CTL0	UARTA1 control register 0	UARTA1	575
UA1CTL1	UARTA1 control register 1	UARTA1	591
UA1CTL2	UARTA1 control register 2	UARTA1	592
UA1OPT0	UARTA1 option control register 0	UARTA1	577
UA1REIC	Interrupt control register	INTC	684
UA1RIC	Interrupt control register	INTC	684
UA1RX	UARTA1 receive data register	UARTA1	579
UA1STR	UARTA1 status register	UARTA1	577
UA1TIC	Interrupt control register	INTC	684
UA1TX	UARTA1 transmit data register	UARTA1	579
VSWC	System wait control register	BCU	80
WDTE	Watchdog timer enable register	WDT	488
WDTM	Watchdog timer mode register	WDT	487

APPENDIX C INSTRUCTION SET LIST

C.1 Conventions

(1) Register symbols used to describe operands

Register Symbol	Explanation
reg1	General-purpose registers: Used as source registers.
reg2	General-purpose registers: Used mainly as destination registers. Also used as source register in some instructions.
reg3	General-purpose registers: Used mainly to store the remainders of division results and the higher order 32 bits of multiplication results.
bit#3	3-bit data for specifying the bit number
immX	X bit immediate data
dispX	X bit displacement data
regID	System register number
vector	5-bit data that specifies the trap vector (00H to 1FH)
cccc	4-bit data that shows the conditions code
sp	Stack pointer (SP)
ep	Element pointer (r30)
listX	X item register list

(2) Register symbols used to describe opcodes

Register Symbol	Explanation
R	1-bit data of a code that specifies reg1 or regID
r	1-bit data of the code that specifies reg2
w	1-bit data of the code that specifies reg3
d	1-bit displacement data
I	1-bit immediate data (indicates the higher bits of immediate data)
i	1-bit immediate data
cccc	4-bit data that shows the condition codes
CCCC	4-bit data that shows the condition codes of Bcond instruction
bbb	3-bit data for specifying the bit number
L	1-bit data that specifies a program register in the register list
S	1-bit data that specifies a system register in the register list

(3) Register symbols used in operations

Register Symbol	Explanation
←	Input for
GR [ ]	General-purpose register
SR [ ]	System register
zero-extend (n)	Expand n with zeros until word length.
sign-extend (n)	Expand n with signs until word length.
load-memory (a, b)	Read size b data from address a.
store-memory (a, b, c)	Write data b into address a in size c.
load-memory-bit (a, b)	Read bit b of address a.
store-memory-bit (a, b, c)	Write c to bit b of address a.
saturated (n)	Execute saturated processing of n (n is a 2's complement). If, as a result of calculations, n ≥ 7FFFFFFFH, let it be 7FFFFFFFH. n ≤ 80000000H, let it be 80000000H.
result	Reflects the results in a flag.
Byte	Byte (8 bits)
Halfword	Half word (16 bits)
Word	Word (32 bits)
+	Addition
–	Subtraction
ll	Bit concatenation
×	Multiplication
÷	Division
%	Remainder from division results
AND	Logical product
OR	Logical sum
XOR	Exclusive OR
NOT	Logical negation
logically shift left by	Logical shift left
logically shift right by	Logical shift right
arithmetically shift right by	Arithmetic shift right

(4) Register symbols used in execution clock

Register Symbol	Explanation
i	If executing another instruction immediately after executing the first instruction (issue).
r	If repeating execution of the same instruction immediately after executing the first instruction (repeat).
l	If using the results of instruction execution in the instruction immediately after the execution (latency).

(5) Register symbols used in flag operations

Identifier	Explanation
(Blank)	No change
0	Clear to 0
X	Set or cleared in accordance with the results.
R	Previously saved values are restored.

(6) Condition codes

Condition Name (cond)	Condition Code (cccc)	Condition Formula	Explanation
V	0 0 0 0	OV = 1	Overflow
NV	1 0 0 0	OV = 0	No overflow
C/L	0 0 0 1	CY = 1	Carry Lower (Less than)
NC/NL	1 0 0 1	CY = 0	No carry Not lower (Greater than or equal)
Z/E	0 0 1 0	Z = 1	Zero Equal
NZ/NE	1 0 1 0	Z = 0	Not zero Not equal
NH	0 0 1 1	(CY or Z) = 1	Not higher (Less than or equal)
H	1 0 1 1	(CY or Z) = 0	Higher (Greater than)
N	0 1 0 0	S = 1	Negative
P	1 1 0 0	S = 0	Positive
T	0 1 0 1	–	Always (Unconditional)
SA	1 1 0 1	SAT = 1	Saturated
LT	0 1 1 0	(S xor OV) = 1	Less than signed
GE	1 1 1 0	(S xor OV) = 0	Greater than or equal signed
LE	0 1 1 1	((S xor OV) or Z) = 1	Less than or equal signed
GT	1 1 1 1	((S xor OV) or Z) = 0	Greater than signed

C.2 Instruction Set (in Alphabetical Order)

(1/6)

Mnemonic	Operand	Opcode	Operation		Execution Clock			Flags				
					i	r	l	CY	OV	S	Z	SAT
ADD	reg1,reg2	rrrrr001110RRRRR	GR[reg2]←GR[reg2]+GR[reg1]		1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010010iiiii	GR[reg2]←GR[reg2]+sign-extend(imm5)		1	1	1	×	×	×	×	
ADDI	imm16,reg1,reg2	rrrrr110000RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]+sign-extend(imm16)		1	1	1	×	×	×	×	
AND	reg1,reg2	rrrrr001010RRRRR	GR[reg2]←GR[reg2]AND GR[reg1]		1	1	1		0	×	×	
ANDI	imm16,reg1,reg2	rrrrr110110RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]AND zero-extend(imm16)		1	1	1		0	0	×	
Bcond	disp9	ddddd1011dddcccc **Note 1**	if conditions are satisfied then PC←PC+sign-extend(disp9)	When conditions are satisfied	3 Note 2	3 Note 2	3 Note 2					
				When conditions are not satisfied	1	1	1					
BSH	reg2,reg3	rrrrr11111100000 wwwww01101000010	GR[reg3]←GR[reg2] (23 : 16) ll GR[reg2] (31 : 24) ll GR[reg2] (7 : 0) ll GR[reg2] (15 : 8)		1	1	1	×	0	×	×	
BSW	reg2,reg3	rrrrr11111100000 wwwww01101000000	GR[reg3]←GR[reg2] (7 : 0) ll GR[reg2] (15 : 8) ll GR [reg2] (23 : 16) ll GR[reg2] (31 : 24)		1	1	1	×	0	×	×	
CALLT	imm6	0000001000iiiiii	CTPC←PC+2(return PC) CTPSW←PSW adr←CTBP+zero-extend(imm6 logically shift left by 1) PC←CTBP+zero-extend(Load-memory(adr,Halfword))		5	5	5					
CLR1	bit#3, disp16[reg1]	10bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,0)		3 Note 3	3 Note 3	3 Note 3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100100	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,0)		3 Note 3	3 Note 3	3 Note 3				×	
CMOV	cccc,imm5,reg2,reg3	rrrrr111111iiiii wwwww011000cccc0	if conditions are satisfied then GR[reg3]←sign-extended(imm5) else GR[reg3]←GR[reg2]		1	1	1					
	cccc,reg1,reg2,reg3	rrrrr111111RRRR wwwww011001cccc0	if conditions are satisfied then GR[reg3]←GR[reg1] else GR[reg3]←GR[reg2]		1	1	1					
CMP	reg1,reg2	rrrrr001111RRRRR	result←GR[reg2]–GR[reg1]		1	1	1	×	×	×	×	
	imm5,reg2	rrrrr010011iiiii	result←GR[reg2]–sign-extend(imm5)		1	1	1	×	×	×	×	
CTRET		0000011111100000 0000000101000100	PC←CTPC PSW←CTPSW		4	4	4	R	R	R	R	R
DBRET		0000011111100000 0000000101000110	PC←DBPC PSW←DBPSW		4	4	4	R	R	R	R	R

(2/6)

Mnemonic	Operand	Opcode	Operation	Execution Clock			Flags				
				i	r	l	CY	OV	S	Z	SAT
DBTRAP		1111110000100000	DBPC←PC+2 (returned PC) DBPSW←PSW PSW.NP←1 PSW.EP←1 PSW.ID←1 PC←00000060H	4	4	4					
DI		0000011111100000 0000000101100000	PSW.ID←1	1	1	1					
DISPOSE	imm5,list12	0000001100 1iiiiiL LLLLLLLLLLL00000	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 repeat 2 steps above until all regs in list12 is loaded	n+1 Note 4	n+1 Note 4	n+1 Note 4					
	imm5,list12,[reg1]	0000001100 1iiiiiL LLLLLLLLLLRRRRR **Note 5**	sp←sp+zero-extend(imm5 logically shift left by 2) GR[reg in list12]←Load-memory(sp,Word) sp←sp+4 repeat 2 steps above until all regs in list12 is loaded PC←GR[reg1]	n+3 Note 4	n+3 Note 4	n+3 Note 4					
DIV	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01011000000	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVH	reg1,reg2	rrrrr000010RRRRR	GR[reg2]←GR[reg2]÷GR[reg1]Note 6	35	35	35		×	×	×	
	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01010000000	GR[reg2]←GR[reg2]÷GR[reg1]Note 6 GR[reg3]←GR[reg2]%GR[reg1]	35	35	35		×	×	×	
DIVHU	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01010000010	GR[reg2]←GR[reg2]÷GR[reg1]Note 6 GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
DIVU	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01011000010	GR[reg2]←GR[reg2]÷GR[reg1] GR[reg3]←GR[reg2]%GR[reg1]	34	34	34		×	×	×	
EI		1000011111100000 0000000101100000	PSW.ID←0	1	1	1					
HALT		0000011111100000 0000000100100000	Stop	1	1	1					
HSW	reg2,reg3	rrrrr11111100000 wwwww01101000100	GR[reg3]←GR[reg2](15 : 0) ll GR[reg2] (31 : 16)	1	1	1	×	0	×	×	
JARL	disp22,reg2	rrrrr11110ddddd ddddddddddddddd0 **Note 7**	GR[reg2]←PC+4 PC←PC+sign-extend(disp22)	3	3	3					
JMP	[reg1]	00000000011RRRRR	PC←GR[reg1]	4	4	4					
JR	disp22	0000011110ddddd ddddddddddddddd0 **Note 7**	PC←PC+sign-extend(disp22)	3	3	3					
LD.B	disp16[reg1],reg2	rrrrr111000RRRRR ddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adr,Byte))	1	1	Note 11					
LD.BU	disp16[reg1],reg2	rrrrr11110bRRRRR ddddddddddddd1 **Notes 8, 10**	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adr,Byte))	1	1	Note 11					

(3/6)

Mnemonic	Operand	Opcode	Operation		Execution Clock			Flags				
					i	r	l	CY	OV	S	Z	SAT
LD.H	disp16[reg1],reg2	rrrrr111001RRRRR ddddddddddddddd0 **Note 8**	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←sign-extend(Load-memory(adr,Halfword))		1	1	Note 11					
LDSR	reg2,regID	rrrrr111111RRRRR 0000000000100000 **Note 12**	SR[regID]←GR[reg2]	Other than regID = PSW	1	1	1					
				regID = PSW	1	1	1	×	×	×	×	×
LD.HU	disp16[reg1],reg2	rrrrr111111RRRRR ddddddddddddddd1 **Note 8**	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←zero-extend(Load-memory(adr,Halfword)		1	1	Note 11					
LD.W	disp16[reg1],reg2	rrrrr111001RRRRR ddddddddddddddd1 **Note 8**	adr←GR[reg1]+sign-extend(disp16) GR[reg2]←Load-memory(adr,Word)		1	1	Note 11					
MOV	reg1,reg2	rrrrr000000RRRRR	GR[reg2]←GR[reg1]		1	1	1					
	imm5,reg2	r r r r r 0 1 0 0 0 0 i i i i i	GR[reg2]←sign-extend(imm5)		1	1	1					
	imm32,reg1	00000110001RRRRR i i i i i i i i i i i i i i i i l l l l l l l l l l l l l l l l	GR[reg1]←imm32		2	2	2					
MOVEA	imm16,reg1,reg2	rrrrr110001RRRRR i i i i i i i i i i i i i i i i	GR[reg2]←GR[reg1]+sign-extend(imm16)		1	1	1					
MOVHI	imm16,reg1,reg2	rrrrr110010RRRRR i i i i i i i i i i i i i i i i	GR[reg2]←GR[reg1]+(imm16 ll 0^{16})		1	1	1					
MUL Note 22	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100000	GR[reg3] ll GR[reg2]←GR[reg2]xGR[reg1]		1	2 Note 14	2					
	imm9,reg2,reg3	r r r r r 1 1 1 1 1 i i i i i wwwww01001IIII00 **Note 13**	GR[reg3] ll GR[reg2]←GR[reg2]xsign-extend(imm9)		1	2 Note 14	2					
MULH	reg1,reg2	rrrrr000111RRRRR	GR[reg2]←GR[reg2] Note 6 xGR[reg1] Note 6		1	1	2					
	imm5,reg2	r r r r r 0 1 0 1 1 1 i i i i i	GR[reg2]←GR[reg2] Note 6 xsign-extend(imm5)		1	1	2					
MULHI	imm16,reg1,reg2	rrrrr110111RRRRR i i i i i i i i i i i i i i i i	GR[reg2]←GR[reg1] Note 6 ximm16		1	1	2					
MULU Note 22	reg1,reg2,reg3	rrrrr111111RRRRR wwwww01000100010	GR[reg3] ll GR[reg2]←GR[reg2]xGR[reg1]		1	2 Note 14	2					
	imm9,reg2,reg3	r r r r r 1 1 1 1 1 i i i i i wwwww01001IIII10 **Note 13**	GR[reg3] ll GR[reg2]←GR[reg2]xzero-extend(imm9)		1	2 Note 14	2					
NOP		0000000000000000	Pass at least one clock cycle doing nothing.		1	1	1					
NOT	reg1,reg2	rrrrr000001RRRRR	GR[reg2]←NOT(GR[reg1])		1	1	1		0	×	×	
NOT1	bit#3,disp16[reg1]	01bbb111110RRRRR ddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not(Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,Z flag)		3 Note 3	3 Note 3	3 Note 3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100010	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,Z flag)		3 Note 3	3 Note 3	3 Note 3				×	

(4/6)

Mnemonic	Operand	Opcode	Operation	Execution Clock			Flags				
				i	r	l	CY	OV	S	Z	SAT
OR	reg1,reg2	rrrrr001000RRRRR	GR[reg2]←GR[reg2]OR GR[reg1]	1	1	1		0	×	×	
ORI	imm16,reg1,reg2	rrrrr110100RRRRR iiiiiiiiiiiiiiii	GR[reg2]←GR[reg1]OR zero-extend(imm16)	1	1	1		0	×	×	
PREPARE	list12,imm5	0000011110iiiiiL LLLLLLLLLLL00001	Store-memory(sp–4,GR[reg in list12],Word) sp←sp–4 repeat 1 step above until all regs in list12 is stored sp←sp-zero-extend(imm5)	n+1 Note 4	n+1 Note 4	n+1 Note 4					
	list12,imm5, sp/imm Note 15	0000011110iiiiiL LLLLLLLLLLLff011 imm16/imm32 **Note 16**	Store-memory(sp–4,GR[reg in list12],Word) GR[reg in list 12]←Load-memory(sp,Word) sp←sp+4 repeat 2 step above until all regs in list12 is loaded PC←GR[reg1]	n+2 Note 4 Note 17	n+2 Note 4 Note 17	n+2 Note 4 Note 17					
RETI		0000011111100000 0000000101000000	if PSW.EP=1 then PC ←EIPC PSW ←EIPSW else if PSW.NP=1 then PC ←FEPC PSW ←FEPSW else PC ←EIPC PSW ←EIPSW	4	4	4	R	R	R	R	R
SAR	reg1,reg2	rrrrr111111RRRRR 0000000010100000	GR[reg2]←GR[reg2]arithmetically shift right by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010101iiiii	GR[reg2]←GR[reg2]arithmetically shift right by zero-extend (imm5)	1	1	1	×	0	×	×	
SASF	cccc,reg2	rrrrr1111110cccc 0000001000000000	if conditions are satisfied then GR[reg2]←(GR[reg2]Logically shift left by 1) OR 00000001H else GR[reg2]←(GR[reg2]Logically shift left by 1) OR 00000000H	1	1	1					
SATADD	reg1,reg2	rrrrr000110RRRRR	GR[reg2]←saturated(GR[reg2]+GR[reg1])	1	1	1	×	×	×	×	×
	imm5,reg2	rrrrr010001iiiii	GR[reg2]←saturated(GR[reg2]+sign-extend(imm5)	1	1	1	×	×	×	×	×
SATSUB	reg1,reg2	rrrrr000101RRRRR	GR[reg2]←saturated(GR[reg2]–GR[reg1])	1	1	1	×	×	×	×	×
SATSUBI	imm16,reg1,reg2	rrrrr110011RRRRR iiiiiiiiiiiiiiii	GR[reg2]←saturated(GR[reg1]–sign-extend(imm16)	1	1	1	×	×	×	×	×
SATSUBR	reg1,reg2	rrrrr000100RRRRR	GR[reg2]←saturated(GR[reg1]–GR[reg2])	1	1	1	×	×	×	×	×
SETF	cccc,reg2	rrrrr1111110cccc 0000000000000000	If conditions are satisfied then GR[reg2]←00000001H else GR[reg2]←00000000H	1	1	1					

(5/6)

Mnemonic	Operand	Opcode	Operation	Execution Clock			Flags				
				i	r	l	CY	OV	S	Z	SAT
SET1	bit#3,disp16[reg1]	00bbb111110RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit(adr,bit#3)) Store-memory-bit(adr,bit#3,1)	3 Note 3	3 Note 3	3 Note 3				×	
	reg2,[reg1]	rrrrr111111RRRRR 0000000011100000	adr←GR[reg1] Z flag←Not(Load-memory-bit(adr,reg2)) Store-memory-bit(adr,reg2,1)	3 Note 3	3 Note 3	3 Note 3				×	
SHL	reg1,reg2	rrrrr111111RRRRR 0000000011000000	GR[reg2]←GR[reg2] logically shift left by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010110iiiii	GR[reg2]←GR[reg2] logically shift left by zero-extend(imm5)	1	1	1	×	0	×	×	
SHR	reg1,reg2	rrrrr111111RRRRR 0000000010000000	GR[reg2]←GR[reg2] logically shift right by GR[reg1]	1	1	1	×	0	×	×	
	imm5,reg2	rrrrr010100iiiii	GR[reg2]←GR[reg2] logically shift right by zero-extend(imm5)	1	1	1	×	0	×	×	
SLD.B	disp7[ep],reg2	rrrrr0110ddddddd	adr←ep+zero-extend(disp7) GR[reg2]←sign-extend(Load-memory(adr,Byte))	1	1	Note 9					
SLD.BU	disp4[ep],reg2	rrrrr0000110dddd **Note 18**	adr←ep+zero-extend(disp4) GR[reg2]←zero-extend(Load-memory(adr,Byte))	1	1	Note 9					
SLD.H	disp8[ep],reg2	rrrrr1000ddddddd **Note 19**	adr←ep+zero-extend(disp8) GR[reg2]←sign-extend(Load-memory(adr,Halfword))	1	1	Note 9					
SLD.HU	disp5[ep],reg2	rrrrr0000111dddd **Notes 18, 20**	adr←ep+zero-extend(disp5) GR[reg2]←zero-extend(Load-memory(adr,Halfword))	1	1	Note 9					
SLD.W	disp8[ep],reg2	rrrrr1010ddddddd0 **Note 21**	adr←ep+zero-extend(disp8) GR[reg2]←Load-memory(adr,Word)	1	1	Note 9					
SST.B	reg2,disp7[ep]	rrrrr0111ddddddd	adr←ep+zero-extend(disp7) Store-memory(adr,GR[reg2],Byte)	1	1	1					
SST.H	reg2,disp8[ep]	rrrrr1001ddddddd **Note 19**	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Halfword)	1	1	1					
SST.W	reg2,disp8[ep]	rrrrr1010ddddddd1 **Note 21**	adr←ep+zero-extend(disp8) Store-memory(adr,GR[reg2],Word)	1	1	1					
ST.B	reg2,disp16[reg1]	rrrrr111010RRRRR dddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Store-memory(adr,GR[reg2],Byte)	1	1	1					
ST.H	reg2,disp16[reg1]	rrrrr111011RRRRR ddddddddddddddd0 **Note 8**	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Halfword)	1	1	1					
ST.W	reg2,disp16[reg1]	rrrrr111011RRRRR ddddddddddddddd1 **Note 8**	adr←GR[reg1]+sign-extend(disp16) Store-memory (adr,GR[reg2], Word)	1	1	1					
STSR	regID,reg2	rrrrr111111RRRRR 0000000001000000	GR[reg2]←SR[regID]	1	1	1					

(6/6)

Mnemonic	Operand	Opcode	Operation	Execution Clock			Flags				
				i	r	l	CY	OV	S	Z	SAT
SUB	reg1,reg2	rrrrr001101RRRRR	GR[reg2]←GR[reg2]–GR[reg1]	1	1	1	×	×	×	×	
SUBR	reg1,reg2	rrrrr001100RRRRR	GR[reg2]←GR[reg1]–GR[reg2]	1	1	1	×	×	×	×	
SWITCH	reg1	00000000010RRRRR	adr←(PC+2) + (GR [reg1] logically shift left by 1) PC←(PC+2) + (sign-extend (Load-memory (adr,Halfword)) logically shift left by 1	5	5	5					
SXB	reg1	00000000101RRRRR	GR[reg1]←sign-extend (GR[reg1] (7 : 0))	1	1	1					
SXH	reg1	00000000111RRRRR	GR[reg1]←sign-extend (GR[reg1] (15 : 0))	1	1	1					
TRAP	vector	0 0 0 0 0 1 1 1 1 1 1 i i i i i 0000000100000000	EIPC ←PC+4 (Return PC) EIPSW ←PSW ECR.EICC ←Exception code (40H to 4FH, 50H to 5FH) PSW.EP ←1 PSW.ID ←1 PC ←00000040H (when vector is 00H to 0FH (exception code: 40H to 4FH)) 00000050H (when vector is 10H to 1FH (exception code: 50H to 5FH))	4	4	4					
TST	reg1,reg2	rrrrr001011RRRRR	result←GR[reg2] AND GR[reg1]	1	1	1		0	×	×	
TST1	bit#3,disp16[reg1]	11bbb111110RRRRR ddddddddddddddd	adr←GR[reg1]+sign-extend(disp16) Z flag←Not (Load-memory-bit (adr,bit#3))	3 Note 3	3 Note 3	3 Note 3				×	
	reg2, [reg1]	rrrrr111111RRRRR 0000000011100110	adr←GR[reg1] Z flag←Not (Load-memory-bit (adr,reg2))	3 Note 3	3 Note 3	3 Note 3				×	
XOR	reg1,reg2	rrrrr001001RRRRR	GR[reg2]←GR[reg2] XOR GR[reg1]	1	1	1		0	×	×	
XORI	imm16,reg1,reg2	rrrrr110101RRRRR i i i i i i i i i i i i i i i i	GR[reg2]←GR[reg1] XOR zero-extend (imm16)	1	1	1		0	×	×	
ZXB	reg1	00000000100RRRRR	GR[reg1]←zero-extend (GR[reg1] (7 : 0))	1	1	1					
ZXH	reg1	00000000110RRRRR	GR[reg1]←zero-extend (GR[reg1] (15 : 0))	1	1	1					

Notes 1. ddddddddd: Higher 8 bits of disp9.

2. 4 if there is an instruction that rewrites the contents of the PSW immediately before.

3. If there is no wait state (3 + the number of read access wait states).

4. n is the total number of list12 load registers. (According to the number of wait states. Also, if there are no wait states, n is the total number of list12 registers. If n = 0, same operation as when n = 1)

5. RRRRR: other than 00000.

6. The lower halfword data only are valid.

7. ddddddddddddddddddddd: The higher 21 bits of disp22.

8. ddddddddddddddd: The higher 15 bits of disp16.

9. According to the number of wait states (1 if there are no wait states).

10. b: bit 0 of disp16.

11. According to the number of wait states (2 if there are no wait states).

Notes 12. In this instruction, for convenience of mnemonic description, the source register is made reg2, but the reg1 field is used in the opcode. Therefore, the meaning of register specification in the mnemonic description and in the opcode differs from other instructions.

rrrrr = regID specification

RRRRR = reg2 specification

13. iiiii: Lower 5 bits of imm9.

IIII: Higher 4 bits of imm9.

14. In the case of reg2 = reg3 (the lower 32 bits of the results are not written in the register) or reg3 = r0 (the higher 32 bits of the results are not written in the register), shortened by 1 clock.

15. sp/imm: specified by bits 19 and 20 of the sub-opcode.

16. ff = 00: Load sp in ep.

01: Load sign expanded 16-bit immediate data (bits 47 to 32) in ep.

10: Load 16-bit logically left shifted 16-bit immediate data (bits 47 to 32) in ep.

11: Load 32-bit immediate data (bits 63 to 32) in ep.

17. If imm = imm32, n + 3 clocks.

18. rrrrr: Other than 00000.

19. ddddddd: Higher 7 bits of disp8.

20. dddd: Higher 4 bits of disp5.

21. dddddd: Higher 6 bits of disp8.

22. Do not make a combination that satisfies all the following conditions when using the “MUL reg1, reg2, reg3” instruction and “MULU reg1, reg2, reg3” instruction. Operation is not guaranteed when an instruction that satisfies the following conditions is executed.

- Reg1 = reg3
- Reg1 ≠ reg2
- Reg1 ≠ r0
- Reg3 ≠ r0

APPENDIX D REVISION HISTORY

D.1 Major Revisions in This Edition

(1/3)

Page	Description
Throughout	Addition of PG-FP5
p.36	Addition of description to **2.1 (1) Port pins**
p.50	Modification of description in **Table 3-2 System Register Numbers**
p.55	Modification of description in **3.2.2 (6) Exception/debug trap status saving registers (DBPC, DBPSW)**
p.74	Modification of description in **3.4.7 On-chip peripheral I/O registers**
p.114	Addition of description to **4.3.4 (1) (e) Pull-up resistor option register 3 (PU3)**
p.153	Modification of description in **Table 4-16 Noise Eliminator**
p.154	Addition of **Figure 4-26 Example of Noise Elimination Timing**
p.160	Modification of description in **Figure 5-1 Clock Generator**
p.161	Modification of description in **5.2 (5) Prescaler 1**
p.169	Modification of description in **Table 5-3 Operation Status of Each Clock**
p.174	Addition of description to **Table 6-1 TMPn Overview**
p.182	Modification of description in **6.4 (2) TMPn control register 1 (TPnCTL1)**
p.183	Modification of description in **6.4 (3) TMPm I/O control register 0 (TPmIOC0)**
p.196	Modification of description in **6.6 (1) (a) Counter start operation**
pp.203, 204	Addition of description to **Figure 6-11 Register Setting for Interval Timer Mode Operation**
p.205	Addition of description to **Figure 6-12 Software Processing Flow in Interval Timer Mode**
p.212	Addition of **6.6.1 (3) Operation by external event count input (TIPk0)**
p.213	Addition of description to **6.6.2 External event count mode (TPkMD2 to TPkMD0 bits = 001)**
p.216	Addition of description to **Figure 6-18 Register Setting for Operation in External Event Count Mode**
p.218	Addition of description to **6.6.2 (2) Operation timing in external event count mode**
p.222	Modification of description in **Figure 6-23 Configuration in External Trigger Pulse Output Mode**
p.223	Modification of **Figure 6-24 Basic Timing in External Trigger Pulse Output Mode**
p.223	Addition of description to **6.6.3 External trigger pulse output mode (TPmMD2 to TPmMD0 bits = 010)**
pp.224, 225	Modification of description in **Figure 6-25 Setting of Registers in External Trigger Pulse Output Mode**
p.231	Modification of figure in **6.6.3 (2) (b) 0%/100% output of PWM waveform**
p.235	Modification of description in **Figure 6-27 Configuration in One-Shot Pulse Output Mode**
pp.237, 238	Modification of description in **Figure 6-29 Setting of Registers in One-Shot Pulse Output Mode**
p.239	Modification of description in **Figure 6-30 Software Processing Flow in One-Shot Pulse Output Mode**
p.240	Modification of figure in **6.6.4 (2) (a) Note on rewriting TPmCCRa register**
p.244	Addition of description to **Figure 6-33 Register Setting in PWM Output Mode**
p.248	Modification of description in **6.6.5 (2) (a) Changing pulse width during operation**
p.249	Modification of description in **6.6.5 (2) (b) 0%/100% output of PWM waveform**
p.268	Modification of description in **Figure 6-41 Configuration in Pulse Width Measurement Mode**
p.269	Modification of description in **6.6.7 Pulse width measurement mode (TPkMD2 to TPkMD0 bits = 110)**
pp.270, 271	Modification of description in **Figure 6-43 Register Setting in Pulse Width Measurement Mode**
p.272	Deletion of description in **Figure 6-44 Software Processing Flow in Pulse Width Measurement Mode**

(2/3)

Page	Description
p.274	Addition of description to **Table 7-1 TMQn Overview**
p.281	Modification of description in **7.4 (3) TMQn I/O control register 0 (TQnIOC0)**
p.297	Modification of description in **7.6 (1) (a) Counter start operation**
pp.305 to 307	Modification of description in **Figure 7-9 Register Setting for Interval Timer Mode Operation**
p.308	Addition of description to **Figure 7-10 Software Processing Flow in Interval Timer Mode**
p.313	Modification of description in **7.6.1 (2) (d) Operation of TQnCCR1 to TQnCCR3 registers**
p.315	Addition of **7.6.1 (3) Operation by external event count input (EVTQ0)**
p.316	Addition of description to **7.6.2 External event count mode (TQ0MD2 to TQ0MD0 bits = 001)**
p.319	Addition of description to **Figure 7-16 Register Setting for Operation in External Event Count Mode**
p.321	Addition of description to **7.6.2 (2) Operation timing in external event count mode**
p.327	Modification of **Figure 7-22 Basic Timing in External Trigger Pulse Output Mode**
p.328	Addition of description to **7.6.3 External trigger pulse output mode (TQ0MD2 to TQ0MD0 bits = 010)**
p.335	Modification of description in **7.6.3 (2) (b) 0%/100% output of PWM waveform**
p.344	Modification of description in **Figure 7-27 Register Setting in One-Shot Pulse Output Mode**
pp.345, 346	Modification of description in **Figure 7-28 Software Processing Flow in One-Shot Pulse Output Mode**
p.356	Modification of description in **7.6.5 (2) (a) Changing pulse width during operation**
p.358	Modification of figure in **7.6.5 (2) (b) 0%/100% output of PWM waveform**
p.381	Modification of description in **7.6.7 Pulse width measurement mode (TQ0MD2 to TQ0MD0 bits = 110)**
p.386	Addition of description to **8.1 Functions**
p.387	Modification of description in **8.2 Features**
p.389	Modification of description in **8.3 (1) (a) General-purpose timer mode**
p.404	Addition of **8.5.1 (5) PWM output operation**
p.426	Addition of description to **10.2 (1) TMQn dead-time compare register (TQnDTC)**
p.428	Addition of description to **10.3 (2) TMQn option register 1 (TQnOPT1)**
p.429	Addition of description to **10.3 (3) TMQn option register 2 (TQnOPT2)**
p.433	Addition of description to **Figure 10-3 TOQnTm and TOQnBm Pin Output Control (Without Dead Time)**
p.441	Modification of **Figure 10-5 Outline of 6-Phase PWM Output Mode**
p.446	Modification of description in **Figure 10-9 0% PWM Output Waveform (With Dead Time)**
p.447	Modification of description in **Figure 10-10 100% PWM Output Waveform (With Dead Time)**
p.448	Modification of description in **Figure 10-11 PWM Output Waveform from 0% to 100% and from 100% to 0% (With Dead Time)**
p.452	Addition of description to **10.4.3 Interrupt culling function**
p.459	Addition of description to **10.4.4 Operation to rewrite register with transfer function**
p.509	Modification of description in **12.4.1 Basic operation**
p.542	Addition of **12.9.6 Variation of A/D conversion results**
p.542	Addition of **12.9.7 A/D conversion result hysteresis characteristics**
p.543	Addition of **12.9.8 Restrictions on setting one-shot mode and software trigger mode**
p.546	Addition of description to **12.10 (6) Differential linearity error**
p.569	Addition of **13.7 Cautions**
p.573	Modification of **Figure 14-2 Block Diagram of UARTAn**
p.574	Modification of description in **14.3 (5) UARTAn status register (UAnSTR)**

(3/3)

Page	Description
p.575	Modification of description in **14.4 (1) UARTAn control register 0 (UAnCTL0)**
p.585	Modification of **Figure 14-6 Continuous Transmission Operation Timing**
p.591	Modification of description in **14.7 (2) UARTAn control register 1 (UAnCTL1)**
p.605	Modification of description in **15.4 (2) CSIBn control register 1 (CBnCTL1)**
p.613	Modification of figure in **15.5.3 (1) Operation flow**
p.622	Modification of figure in **15.5.7 (2) Operation timing**
p.651	Modification of description in **16.3.4 DMA addressing control registers 0 to 3 (DADC0 to DADC3)**
p.652	Addition of description to **16.3.5 DMA channel control registers 0 to 3 (DCHC0 to DCHC3)**
p.654	Modification of description in **16.3.6 DMA trigger factor registers 0 to 3 (DTFR0 to DTFR3)**
p.662	Deletion of description in **16.7 DMA Channel Priorities**
p.666	Modification of description in **16.12 Cautions**
p.684	Addition of description to **17.3.4 Interrupt control registers (xxICn)**
p.699	Addition of description to **17.6.1 (2) Restore**
p.714	Addition of description to **18.5.1 Setting and operation status**
p.727	Modification of **Caution** in **CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)**
p.728	Modification of description in **21.1.2 (8) Mask function**
p.730	Modification of description in **21.1.3 (2) Setting**
p.733	Modification of description in **21.3.1 KEL connector**
p.733	Modification of **Figure 21-1 Connecting On-Chip Debug Emulator (QB-V850MINI)**
p.735	Modification of description in **Table 21-2 Pin Functions of Connector for QB-V850MINI (on Target System Side)**
p.736	Modification of description in **Figure 21-3 Example of Recommended Connection of μPD70F3186 (V850E/IA4) and KEL Connector**
p.741	Modification of description in **Table 22-2 Basic Functions**
p.741	Modification of description in **Table 22-3 Security Functions**
p.742	Addition of **Table 22-4 Security Setting**
p.743	Addition of **22.3 (1) Security setting by PG-FP4 and PG-FP5 (Security flag settings)**
p.748	Modification of description in **Table 22-7 Wiring Correspondence Between Dedicated Flash Programmer and V850E/IA3, V850E/IA4**
p.749	Modification of **Figure 22-4 Procedure for Manipulating Flash Memory**
p.751	Modification of description in **Table 22-8 Flash Memory Control Commands**
p.757	Deletion of figure in **22.4.6 (4) $\overline{\text{RESET}}$ pin**
p.761	Modification of **Figure 22-14 Standard Self Programming Flow**
p.762	Modification of description in **Table 22-11 Flash Function List**
p.772	Modification of description in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3) CSIB Timing**
p.791	Modification of description in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4) CSIB Timing**

D.2 Revision History up to Previous Edition

The following table shows the revision history up to this edition. The “Applied to:” column indicates the chapters of each edition in which the revision was applied.

(1/6)

Edition	Description	Applied to:
2nd edition	Addition of μPD703186	Throughout
	Modification of pin configuration of pin 20 in **1.2.4 Pin configuration (V850E/IA3)**	**CHAPTER 1 INTRODUCTION**
	Modification of description in **2.2 Pin I/O Circuits and Recommended Connection of Unused Pins**	**CHAPTER 2 PIN FUNCTIONS**
	Modification of description in **2.3 Pin I/O Circuits**	
	Addition and modification of **Note 2** in **Table 3-2 System Register Numbers**	**CHAPTER 3 CPU FUNCTION**
	Addition of **3.2.2 (1) Interrupt status saving registers (EIPC, EIPSW), (2) NMI status saving registers (FEPC, FEPSW), (5) CALLT execution status saving registers (CTPC, CTPSW), (6) Exception/debug trap status saving registers (DBPC, DBPSW),** and **(7) CALLT base pointer (CTBP)**	
	Addition of **Cautions** and **Remark** in **3.5.4 (2) (c) Internal memory size switching register (IMS)**	
	Addition of **Note** in **3.4.7 On-chip peripheral I/O registers**	
	Addition of **Caution** in **3.4.9 System wait control register (VSWC)**	
	Modification of description in **3.4.10 Cautions**	
	Addition of **4.3 Port Configuration**	**CHAPTER 4 PORT FUNCTIONS**
	Modification of **Figure 4-22 Block Diagram of P50 Pin**	
	Addition of **4.5 Port Register Settings When Alternate Function Is Used**	
	Modification of **Notes** in **Table 4-16 Noise Eliminator**	
	Modification of description in **4.7 Cautions**	
	Addition and modification of **Cautions** in **5.3 (3) Power save control register (PSC)**	**CHAPTER 5 CLOCK GENERATOR**
	Modification of **CHAPTER 6 16-BIT TIMER/EVENT COUNTER P (TMP)**	**CHAPTER 6 16-BIT TIMER/EVENT COUNTER P (TMP)**
	Modification of **CHAPTER 7 16-BIT TIMER/EVENT COUNTER Q (TMQ)**	**CHAPTER 7 16-BIT TIMER/EVENT COUNTER Q (TMQ)**
	Addition of **Caution** in **8.2 Features**	**CHAPTER 8 16-BIT 2-PHASE ENCODER INPUT UP/DOWN COUNTER/ GENERAL-PURPOSE TIMER (TIMER ENC1n)**
	Modification of description on Mode 3 in **Table 8-4 List of Count Operations in UDC Mode**	
	Addition of description to **9.2 (2) TMM0 compare register 0 (TM0CMP0)**	**CHAPTER 9 16-BIT INTERVAL TIMER M (TMM)**
	Modification of description in **9.3 (1) TMM0 control register 0 (TM0CTL0)**	
	Modification of **9.4 Operation**	

(2/6)

Edition	Description	Applied to:
2nd edition	Addition of description in **10.1 Functional Overview**	**CHAPTER 10 MOTOR CONTROL FUNCTION**
	Addition of description to **Figure 10-1 Block Diagram of Motor Control**	
	Addition of description to **Figure 10-2 TMQn Option**	
	Modification of description in **10.3 (3) TMQn option register 2 (TQnOPT2)**	
	Addition of description to **10.3 (5) TMQn I/O control register 3 (TQnIOC3)**	
	Modification of description in **10.3 (6) High-impedance output control registers 00, 01, 10, 11, 20, 21 (HZAmCTL0, HZAmCTL1)**	
	Addition of description to **Figure 10-5 Outline of 6-Phase PWM Output Mode**	
	Addition of timing and deletion of **Caution** description in **Figure 10-6 Timing Chart of 6-Phase PWM Output Mode**	
	Addition of description in **10.4.1 (3) Rewriting registers during timer operation**	
	Addition of timing to **Figure 10-7 Interrupt and Up/Down Flag**	
	Modification of description in (b) in **Figure 10-12 PWM Output Waveform with Dead Time (2)**	
	Modification of description in (b) in **Figure 10-13 Operation of Dead-Time Counter m (1)**	
	Addition and modification of description in **10.4.3 Interrupt culling function**	
	Addition of description to **Figures 10-15** to **10-20**	
	Addition of description in **10.4.4 Operation to rewrite register with transfer function**	
	Addition of description in **10.4.4 (1) Anytime rewriting mode**	
	Addition of description in **10.4.4 (2) Batch rewrite mode (transfer mode)**	
	Addition of description in **10.4.4 (3) Intermittent batch rewriting mode (transfer culling mode)**	
	Modification of **Figure 10-34 Rewriting TQnCCR1 Register (TQnOPT1.TQnICE bit = 1, TQnOPT1.TQnIOE bit = 0, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 = 00001)**	
	Modification of **Figure 10-35 Rewriting TQnCCR1 Register (TQnOPT1.TQnICE bit = 1, TQnOPT1.TQnIOE bit = 1, TQnOPT1.TQnID4 to TQnOPT1.TQnID0 = 00001)**	
	Addition of description in **10.4.5 (1) (b) Setting of TMQn register**	
	Addition of description in **10.4.5 (3) When not tuning TMPn**	
	Deletion of **Caution** description in **11.3 (1) Watchdog timer mode register (WDTM)**	**CHAPTER 11 WATCHDOG TIMER FUNCTIONS**
	Modification of **Caution** description in **11.3 (2) Watchdog timer enable register (WDTE)**	
	Addition of description in **11.4 Operation**	
	Addition of **11.5 Caution**	
	Addition of **Caution** description in **12.3 (1) A/D converter n mode register 0 (ADAnM0) (n = 0, 1)**	**CHAPTER 12 A/D CONVERTERS 0 AND 1**
	Modification of description in **12.3 (2) A/D converter n mode register 1 (ADAnM1) (n = 0, 1)**	
	Modification of **Note** in **12.3 (4) A/D converter n mode register 2 (ADAnM2) (n = 0, 1)**	
	Addition of **Caution 2** in **12.3 (6) Operational amplifier n control register 1 (OPnCTL1) (n = 0, 1)**	
	Modification of description of (2) in **12.4.1 Basic operation**	
	Addition of **Notes** in **12.4.2 Operation mode and trigger mode**	

(3/6)

Edition	Description	Applied to:
2nd edition	Modification of **Figure 12-8 One-Shot Select 1-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 10, ADA0M2.ADA0BS bit = 0, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 001): V850E/IA4**	**CHAPTER 12 A/D CONVERTERS 0 AND 1**
	Modification of **Figure 12-9 One-Shot Select 4-Buffer Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 10, ADA0M2.ADA0BS bit = 1, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 011): V850E/IA4**	
	Modification of **Figure 12-10 One-Shot Scan Mode Operation Timing (When ADA0M0.ADA0MD1 and ADA0M0.ADA0MD0 bits = 11, ADA0S.ADA0S2 to ADA0S.ADA0S0 bits = 011): V850E/IA4**	
	Modification of description in **Figures 12-11** to **12-16** in **12.5 Operation in Software Trigger Mode**	
	Modification of description in **12.6 Operation in Timer Trigger Modes 0 and 1**	
	Modification of description in **12.7 Operation in External Trigger Mode**	
	Addition of **12.8 Internal Equivalent Circuit**	
	Addition of description to **12.9.1 Stopping conversion operation**	
	Modification of description in **12.9.2 (1) When 0 < trigger occurrence interval < total number of A/D conversion clocks**	
	Addition of **Caution 3** in **13.4.1 Basic operation**	**CHAPTER 13 A/D CONVERTER 2**
	Addition of description to **Figures 13-4** to **13-7**	
	Addition of **13.5 Internal Equivalent Circuit**	
	Modification of **Figure 14-2 Block Diagram of UARTAn**	**CHAPTER 14 ASYNCHRONOUS SERIAL INTERFACE A (UARTA)**
	Modification of description and deletion of **Caution** in **14.4 (1) UARTAn control register 0 (UAnCTL0)**	
	Addition of **Caution** in **14.4 (5) UARTAn status register (UAnSTR)**	
	Modification of description in **14.6.3 Continuous transmission procedure**	
	Addition of **Caution** in **14.6.5 Reception errors**	
	Modification of description in **Figure 14-8 Noise Filter Circuit**	
	Addition of **Figure 14-9 Timing of RXDAn Signal Judged as Noise**	
	Addition of **Caution** in **14.7 (2) UARTAn control register 1 (UAnCTL1)**	
	Addition of **Caution** in **14.7 (3) UARTAn control register 2 (UAnCTL2)**	
	Modification of description in **Table 14-3 Baud Rate Generator Setting Data**	
	Addition of **14.8 Cautions**	
	Modification of **Figure 15-2 Block Diagram of CSIBn**	**CHAPTER 15 CLOCKED SERIAL INTERFACE B (CSIB)**
	Addition of **Remark** in **15.3 (2) CSIBn transmit data register (CBnTX)**	
	Modification of description in **15.4 (1) CSIBn control register 0 (CBnCTL0)**	
	Addition of **Caution** and modification of description in **15.4 (2) CSIBn control register 1 (CBnCTL1)**	
	Modification of description in **15.5 Operation**	
	Modification of description in **15.6 (1) $\overline{\text{SCKBn}}$ pin**	
	Modification of description in **15.7 Operation Flow**	
	Modification of **Caution** description in **16.3.6 DMA trigger factor registers 0 to 3 (DTFR0 to DTFR3)**	**CHAPTER 16 DMA FUNCTIONS (DMA CONTROLLER)**
	Addition of description in **16.8 (2) During DMA transfer (period from the generation of DMA transfer request to completion of DMA transfer)**	

(4/6)

Edition	Description	Applied to:
2nd edition	Addition of description in **16.12 (4) DMA start factors**	**CHAPTER 16 DMA FUNCTIONS (DMA CONTROLLER)**
	Modification of description in **17.9 Periods in Which CPU Does Not Acknowledge Interrupts**	**CHAPTER 17 INTERRUPT/ EXCEPTION PROCESSING FUNCTION**
	Addition of **17.10 Caution**	
	Addition of description to **Table 18-1 Standby Modes**	**CHAPTER 18 STANDBY FUNCTION**
	Modification of **Notes** in **Figure 18-1 Status Transition**	
	Addition and modification of **Cautions** in **18.2 (1) Power save control register (PSC)**	
	Addition of description in **18.3.2 (1) Releasing HALT mode by non-maskable interrupt request signal or unmasked maskable interrupt request signal**	
	Modification of description in **18.4.2 Releasing IDLE mode**	
	Addition of **Caution** and description in **18.4.2 (1) Releasing IDLE mode by unmasked maskable interrupt request signal**	
	Addition of description to **Table 18-5 Operation Status in IDLE Mode**	
	Addition of description in **18.5.2 Releasing STOP mode**	
	Addition of **Caution** and description in **18.5.2 (1) Releasing STOP mode by unmasked maskable interrupt request signal**	
	Addition of description to **Table 18-7 Operation Status in STOP Mode**	
	Addition of **Caution 5** in **20.3 ROM Correction Operation and Program Flow**	**CHAPTER 20 ROM CORRECTION FUNCTION**
	Modification of description in **21.1.3 ROM Security Function**	**CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)**
	Modification of description in **Figure 21-3 Example of Recommended Connection of μPD70F3186 (V850E/IA4) and KEL Connector**	
	Addition of description of (7) in **21.4 Cautions**	
	Modification of **CHAPTER 22 FLASH MEMORY**	**CHAPTER 22 FLASH MEMORY**
	Addition of **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3) (TARGET)**	**CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3) (TARGET)**
	Addition of **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4) (TARGET)**	**CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4) (TARGET)**
	Addition of **CHAPTER 25 PACKAGE DRAWINGS**	**CHAPTER 25 PACKAGE DRAWINGS**
	Addition of **APPENDIX A CAUTIONS**	**APPENDIX A CAUTIONS**
	Addition of **APPENDIX D REVISION HISTORY**	**APPENDIX D REVISION HISTORY**

(4/6)

Edition	Description	Applied to:
3rd edition	• Modification of part numbers of all products to lead-free products • All products under development → Development completed	Throughout
	Addition of **Note** in **2.1 (2) Non-port pins**	**CHAPTER 2 PIN FUNCTIONS**
	Addition of **Caution** in **3.4.8 (1) Setting data to special registers**	**CHAPTER 3 CPU FUNCTION**
	Addition of description to **3.4.9 System wait control register (VSWC)**	
	Addition of **Caution** in **4.6 (2) Noise elimination time select register 1n (NRC1n) (V850E/IA3: n = 0, V850E/IA4: n = 0, 1)**	**CHAPTER 4 PORT FUNCTIONS**
	Addition of **Caution** in **5.3 (6) Clock monitor mode register (CLM)**	**CHAPTER 5 CLOCK GENERATOR**
	Addition of **Note** to **Figure 6-4 TMP3 Block Diagram**	**CHAPTER 6 16-BIT TIMER/EVENT COUNTER P (TMP)**
	Modification of **Note** in **6.4 (1) TMPn control register 0 (TPnCTL0)**	
	Addition of description to **6.4 (6) TMPn option register 0 (TPnOPT0)**	
	Addition of description to **6.4 (7) (a) Function as compare register** and **(b) Function as capture register (TP0CCR0 and TP2CCR0 registers only)**	
	Addition of description to **6.4 (8) (a) Function as compare register** and **(b) Function as capture register (TP0CCR1 and TP2CCR1 registers only)**	
	Addition of **Note** to **Figure 6-5 Flowchart of Basic Operation for Anytime Write**	
	Addition of **Note** to **Figure 6-7 Flowchart of Basic Operation for Batch Write**	
	Modification of description in **6.6.3 (2) (b) 0%/100% output of PWM waveform**	
	Modification of **Note** in **7.4 (1) TMQn control register 0 (TQnCTL0)**	**CHAPTER 7 16-BIT TIMER/EVENT COUNTER Q (TMQ)**
	Modification of **Note** in **7.4 (3) TMQn I/O control register 0 (TQnIOC0)**	
	Addition of description in **7.4 (6) TMQn option register 0 (TQnOPT0)**	
	Addition of description in **7.4 (7) (a) Function as compare register** and **(b) Function as capture register (TQ0CCR0 register only)**	
	Addition of description in **7.4 (8) (a) Function as compare register** and **(b) Function as capture register (TQ0CCR1 register only)**	
	Addition of description in **7.4 (9) (a) Function as compare register** and **(b) Function as capture register (TQ0CCR2 register only)**	
	Addition of description in **7.4 (10) (a) Function as compare register** and **(b) Function as capture register (TQ0CCR3 register only)**	
	Addition of **Note** to **Figure 7-3 Flowchart of Basic Operation for Anytime Write**	
	Addition of **Note** to **Figure 7-5 Flowchart of Basic Operation for Batch Write**	
	Modification of description in **7.6.5 (2) (b) 0%/100% output of PWM waveform**	
	Modification of table description in **8.4 (5) (b) UDC mode (TUM1n.CMDn bit = 1)**	**CHAPTER 8 16-BIT 2-PHASE ENCODER INPUT UP/DOWN COUNTER/ GENERAL-PURPOSE TIMER (TIMER ENC1n)**
	Addition of **Caution** in **8.4 (8) Noise elimination time select register 1n (NRC1n)**	
	Addition of **Caution** in **10.3 (3) TMQn option register 2 (TQnOPT2)**	**CHAPTER 10 MOTOR CONTROL FUNCTION**
	Addition of **Note** in **10.3 (6) High-impedance output control registers 00, 01, 10, 11, 20, 21 (HZAmCTL0, HZAmCTL1)**	
	Modification of **Caution** in **Figure 10-6 Timing Chart of 6-Phase PWM Output Mode**	

(5/6)

Edition	Description	Applied to:
3rd edition	Addition of **Caution** in **10.4.3 Interrupt culling function**	**CHAPTER 10 MOTOR CONTROL FUNCTION**
	Addition of description in **10.4.5 (1) (b) Setting of TMQn register**	
	Modification of description in **11.4 Operation**	**CHAPTER 11 WATCHDOG TIMER FUNCTIONS**
	Addition of **Remark** to **Figure 12-1 Block Diagram of A/D Converters 0 and 1**	**CHAPTER 12 A/D CONVERTERS 0 AND 1**
	Modification of description in **Figure 12-23 ANInm Pin Internal Equivalent Circuit**	
	Addition of **Figure 12-24 Example of Calculating Overall Error of A/D Converters 0 and 1**	
	Addition of **12.9.5 Re-conversion start trigger input during stabilization time**	
	Modification of figure and addition of **Remark** to **Figure 13-1 Block Diagram of A/D Converter 2**	**CHAPTER 13 A/D CONVERTER 2**
	Modification of description in **14.4 (1) UARTAn control register 0 (UAnCTL0)**	**CHAPTER 14 ASYNCHRONOUS SERIAL INTERFACE A (UARTA)**
	Addition of description in **14.4 (4) UARTAn option control register 0 (UAnOPT0)**	
	Modification of description in **14.6.4 UART reception**	
	Modification of **Caution** in **14.6.5 Reception errors**	
	Modification of description in **15.4 (1) CSIBn control register 0 (CBnCTL0)**	**CHAPTER 15 CLOCKED SERIAL INTERFACE B (CSIB)**
	Addition of **15.4 (1) (a) How to use CBnSCE bit**	
	Addition of **Caution** in **15.4 (4) CSIBn status register (CBnSTR)**	
	Addition of **15.5.1 Single transfer mode (master mode, transmission mode)**	
	Addition of **15.5.2 Single transfer mode (master mode, reception mode)**	
	Addition of **15.5.3 Single transfer mode (master mode, transmission/reception mode)**	
	Addition of **15.5.4 Single transfer mode (slave mode, transmission mode)**	
	Addition of **15.5.5 Single transfer mode (slave mode, reception mode)**	
	Addition of **15.5.6 Single transfer mode (slave mode, transmission/reception mode)**	
	Addition of **15.5.7 Continuous transfer mode (master mode, transmission mode)**	
	Addition of **15.5.8 Continuous transfer mode (master mode, reception mode)**	
	Addition of **15.5.9 Continuous transfer mode (master mode, transmission/reception mode)**	
	Addition of **15.5.10 Continuous transfer mode (slave mode, transmission mode)**	
	Addition of **15.5.11 Continuous transfer mode (slave mode, reception mode)**	
	Addition of **15.5.12 Continuous transfer mode (slave mode, transmission/reception mode)**	
	Addition of **15.5.13 Reception error**	
	Addition of **Caution** in **15.5.14 Clock timing**	
	Deletion of **Caution** in **16.3.3 DMA transfer count registers 0 to 3 (DBC0 to DBC3)**	**CHAPTER 16 DMA FUNCTIONS (DMA CONTROLLER)**
	Partial deletion of description in **16.4.3 Block transfer mode**	
	Modification of description in **19.3 (1) Reset source flag register (RESF)**	**CHAPTER 19 RESET FUNCTIONS**
	Addition of **Note** in **19.4 (1) Reset operation by $\overline{\text{RESET}}$ pin input**	

(6/6)

Edition	Description	Applied to:
3rd edition	Modification of ID850NWC to ID850QB in **CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)**	**CHAPTER 21 ON-CHIP DEBUG FUNCTION (ON-CHIP DEBUG UNIT)**
	Modification of description in **21.1.3 (2) Setting**	
	Modification of specification of **High-Impedance Control Timing** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	**CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**
	Deletion of gain error item of **Characteristics of A/D Converter 2** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	
	Modification of specification of delay time from EV_{DD}, AV_{DD} rise to V_{DD}, CV_{DD} rise (t_{DVDD}) and addition of specification of delay time from EV_{DD}, AV_{DD} rise to $\overline{RESET}\uparrow$ (t_{DRES2}) of **Supply Voltage Application/Cutoff Timing** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	
	Addition of **(b) Power supply sequence recommended condition 2** of **Supply Voltage Application Timing** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	
	Modification of description of **Supply Voltage Cutoff Timing** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	
	Modification of specification of write current and erase current of **Flash Memory Programming Characteristics (μPD70F3184 only)** in **CHAPTER 23 ELECTRICAL SPECIFICATIONS (V850E/IA3)**	
	Modification of specification of **High-Impedance Control Timing** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	**CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**
	Deletion of gain error item of **Characteristics of A/D Converter 2** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	
	Modification of specification of delay time from EV_{DD}, AV_{DD} rise to V_{DD}, CV_{DD} rise (t_{DVDD}) and addition of specification of delay time from EV_{DD}, AV_{DD} rise to $\overline{RESET}\uparrow$ (t_{DRES2}) of **Supply Voltage Application/Cutoff Timing** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	
	Addition of **(b) Power supply sequence recommended condition 2** of **Supply Voltage Application Timing** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	
	Addition of **Caution** of **Supply Voltage Cutoff Timing** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	
	Modification of specification of write current and erase current of **Flash Memory Programming Characteristics (μPD70F3186 only)** in **CHAPTER 24 ELECTRICAL SPECIFICATIONS (V850E/IA4)**	
	Addition of **CHAPTER 26 RECOMMENDED SOLDERING CONDITIONS**	**CHAPTER 26 RECOMMENDED SOLDERING CONDITIONS**
	Addition of **D.2 Revision History up to Previous Edition**	**APPENDIX D REVISION HISTORY**

For further information,
please contact:

NEC Electronics Corporation
1753, Shimonumabe, Nakahara-ku,
Kawasaki, Kanagawa 211-8668,
Japan
Tel: 044-435-5111
http://www.necel.com/

[America]

NEC Electronics America, Inc.
2880 Scott Blvd.
Santa Clara, CA 95050-2554, U.S.A.
Tel: 408-588-6000
800-366-9782
http://www.am.necel.com/

[Europe]

NEC Electronics (Europe) GmbH
Arcadiastrasse 10
40472 Düsseldorf, Germany
Tel: 0211-65030
http://www.eu.necel.com/

Hanover Office
Podbielskistrasse 166 B
30177 Hannover
Tel: 0 511 33 40 2-0

Munich Office
Werner-Eckert-Strasse 9
81829 München
Tel: 0 89 92 10 03-0

Stuttgart Office
Industriestrasse 3
70565 Stuttgart
Tel: 0 711 99 01 0-0

United Kingdom Branch
Cygnus House, Sunrise Parkway
Linford Wood, Milton Keynes
MK14 6NP, U.K.
Tel: 01908-691-133

Succursale Française
9, rue Paul Dautier, B.P. 52
78142 Velizy-Villacoublay Cédex
France
Tel: 01-3067-5800

Sucursal en España
Juan Esplandiu, 15
28007 Madrid, Spain
Tel: 091-504-2787

Tyskland Filial
Täby Centrum
Entrance S (7th floor)
18322 Täby, Sweden
Tel: 08 638 72 00

Filiale Italiana
Via Fabio Filzi, 25/A
20124 Milano, Italy
Tel: 02-667541

Branch The Netherlands
Steijgerweg 6
5616 HS Eindhoven
The Netherlands
Tel: 040 265 40 10

[Asia & Oceania]

NEC Electronics (China) Co., Ltd
7th Floor, Quantum Plaza, No. 27 ZhiChunLu Haidian
District, Beijing 100083, P.R.China
Tel: 010-8235-1155
http://www.cn.necel.com/

Shanghai Branch
Room 2509-2510, Bank of China Tower,
200 Yincheng Road Central,
Pudong New Area, Shanghai, P.R.China P.C:200120
Tel:021-5888-5400
http://www.cn.necel.com/

Shenzhen Branch
Unit 01, 39/F, Excellence Times Square Building,
No. 4068 Yi Tian Road, Futian District, Shenzhen,
P.R.China P.C:518048
Tel:0755-8282-9800
http://www.cn.necel.com/

NEC Electronics Hong Kong Ltd.
Unit 1601-1613, 16/F., Tower 2, Grand Century Place,
193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: 2886-9318
http://www.hk.necel.com/

NEC Electronics Taiwan Ltd.
7F, No. 363 Fu Shing North Road
Taipei, Taiwan, R. O. C.
Tel: 02-8175-9600
http://www.tw.necel.com/

NEC Electronics Singapore Pte. Ltd.
238A Thomson Road,
#12-08 Novena Square,
Singapore 307684
Tel: 6253-8311
http://www.sg.necel.com/

NEC Electronics Korea Ltd.
11F., Samik Lavied'or Bldg., 720-2,
Yeoksam-Dong, Kangnam-Ku,
Seoul, 135-080, Korea
Tel: 02-558-3737
http://www.kr.necel.com/

G0706